



Introduction to Computer Networking

Dr. Mahmoud M. Elkhoully

www.elkhoully.net

Structured Naming

Internet Naming Service: DNS

- DNS History
- Introduction to the DNS
- DNS Components
- DNS Structure and Hierarchy
- The DNS in Context

DNS History (1)

- ARPANET utilized a central file HOSTS.TXT
 - Contains names to addresses mapping
 - Maintained by SRI's NIC (*Stanford-Research-Institute: Network-Information-Center*)
- Administrators email changes to NIC
 - NIC updates HOSTS.TXT periodically
- Administrators FTP (download) HOSTS.TXT

DNS History (2)

- As the system grew, HOSTS.TXT had problems with:
 - Scalability (traffic and load)
 - Name collisions
 - Consistency
- In 1984, Paul Mockapetris released the first version (RFCs 882 and 883, superseded by 1034 and 1035 ...)

The DNS is...

- The “Domain Name System”
- What Internet users use to reference anything by name on the Internet
- The mechanism by which Internet software translates names to attributes such as addresses

The DNS is also...

- A globally distributed, scalable, reliable database
- Comprised of three components
 - A “name space”
 - Servers making that name space available
 - Resolvers (clients) which query the servers about the name space

DNS as a Lookup Mechanism

- Users generally prefer names to numbers
- Computers prefer numbers to names
- DNS provides the mapping between the two
 - I have “x”, give me “y”

Global Distribution

- Data is maintained locally, but retrievable globally
 - No single computer has all DNS data
- DNS lookups can be performed by any device
- Remote DNS data is locally cachable to improve performance

Scalability

- No limit to the size of the database
- No limit to the number of queries
 - Tens of thousands of queries handled easily every second
- Queries distributed among masters, slaves, and caches

Reliability

- Data is replicated
 - Data from master is copied to multiple slaves
- Clients can query
 - Master server
 - Any of the copies at slave servers
- Clients will typically query local caches
- DNS protocols can use either UDP or TCP
 - If UDP, DNS protocol handles retransmission, sequencing, etc.

Dynamicity

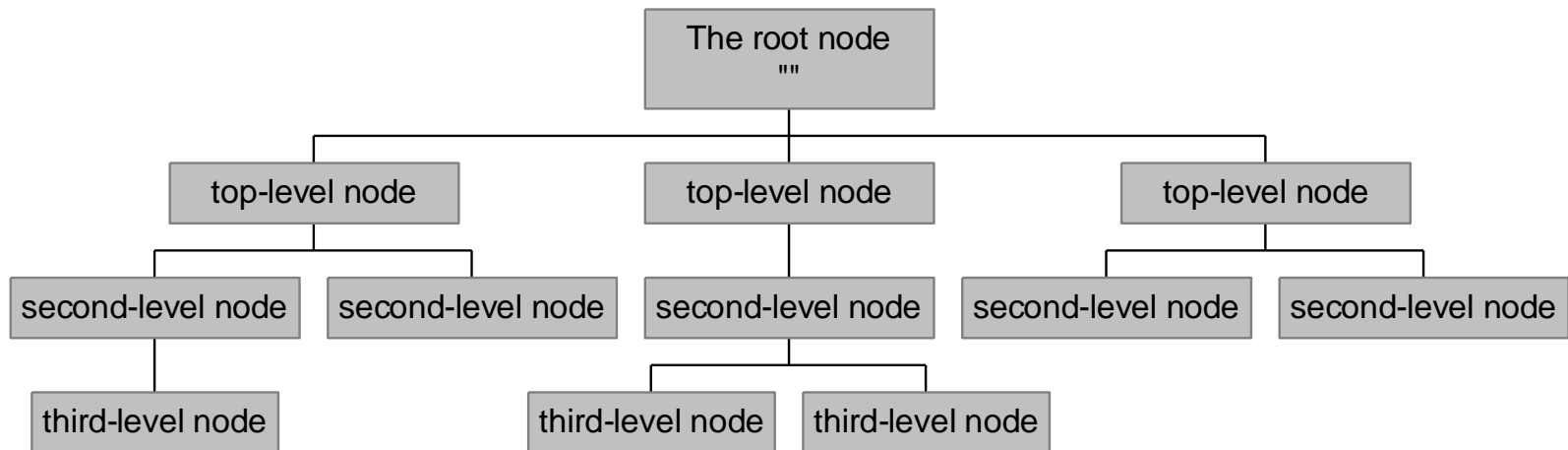
- Database can be updated dynamically
 - Add/delete/modify of any record
 - Only master can be dynamically updated
- Modification of the master database triggers replication

Overview

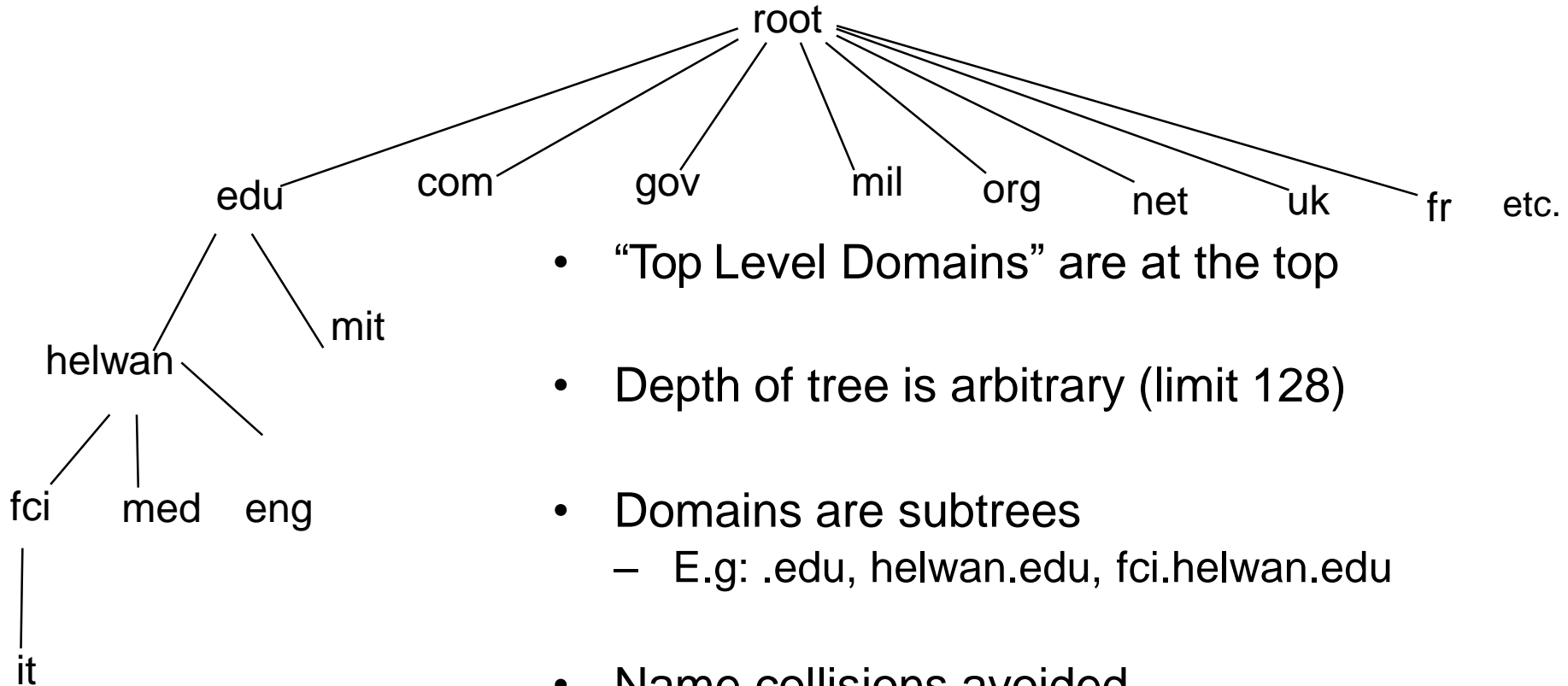
- Introduction to the DNS
- DNS Components
 - The name space
 - The servers
 - The resolvers
- DNS Structure and Hierarchy
- The DNS in Context

The Name Space

- The *name space* is the structure of the DNS database
 - An inverted tree with the root node at the top
- Each node has a label
 - The root node has a null label, written as “”

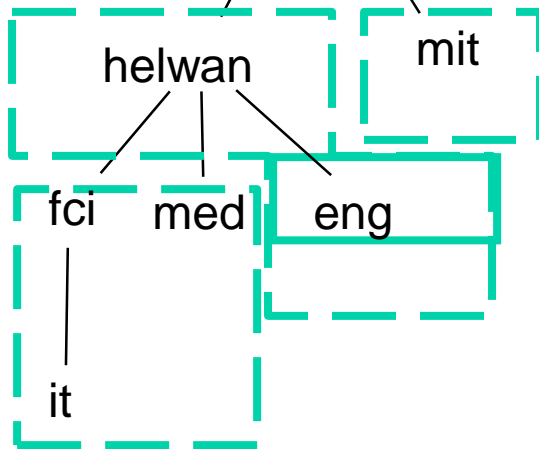
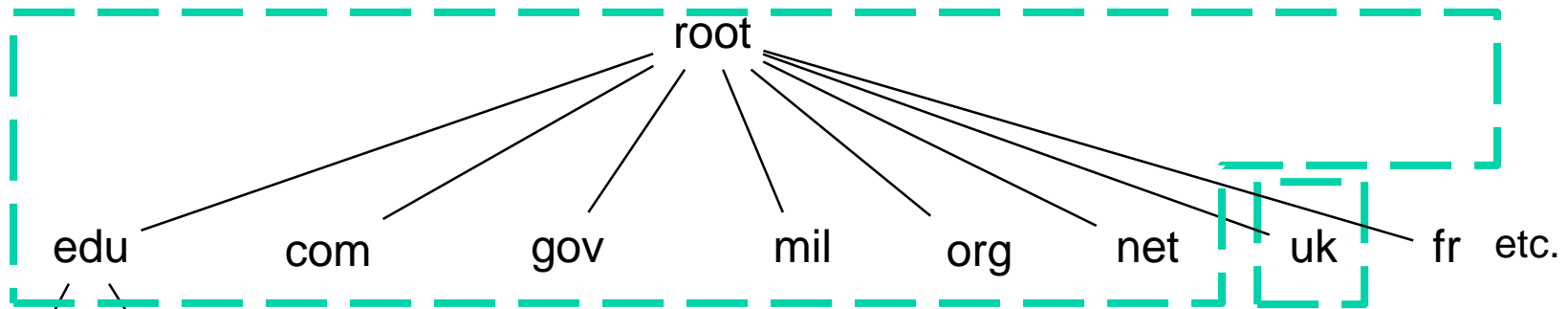


Naming Hierarchy

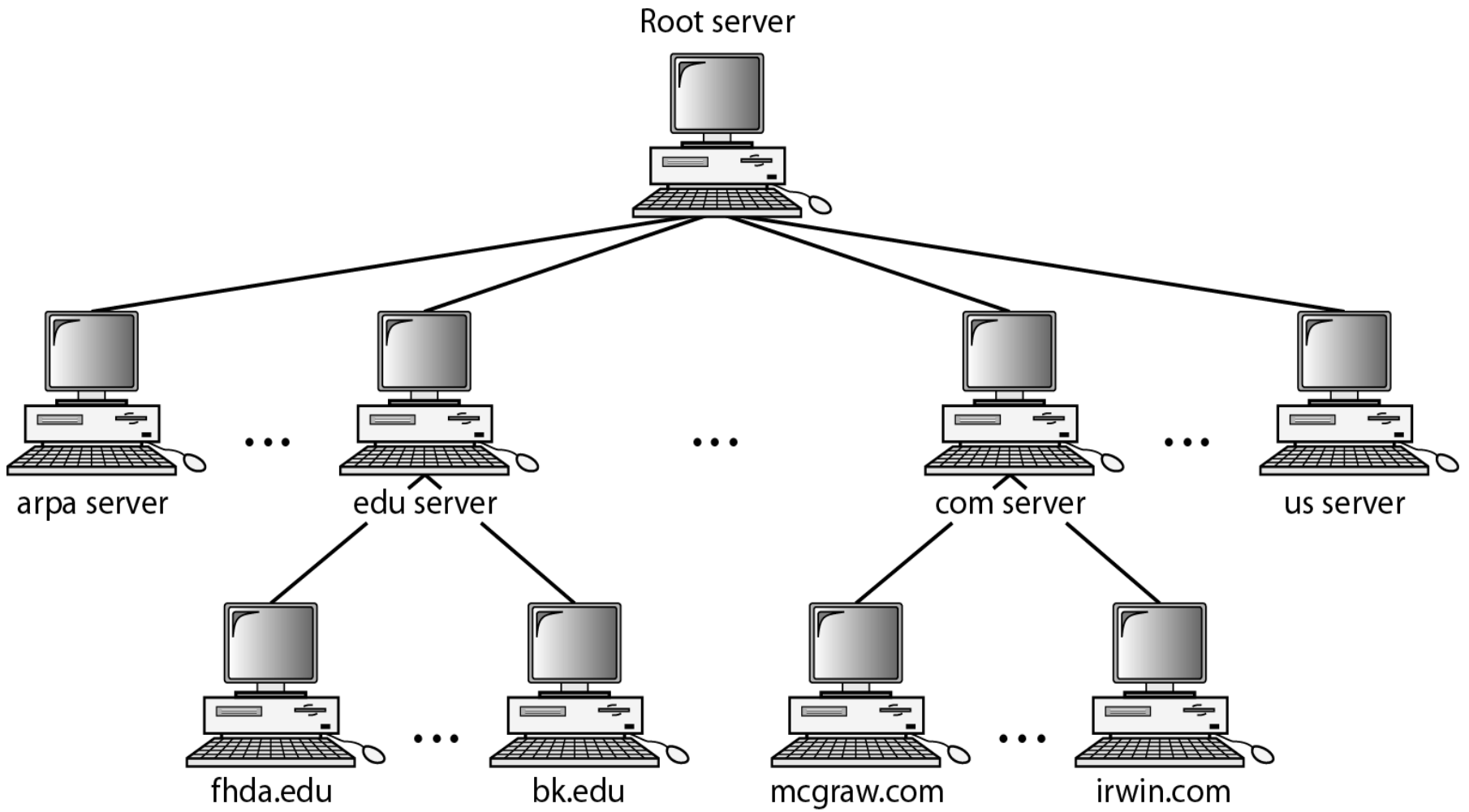


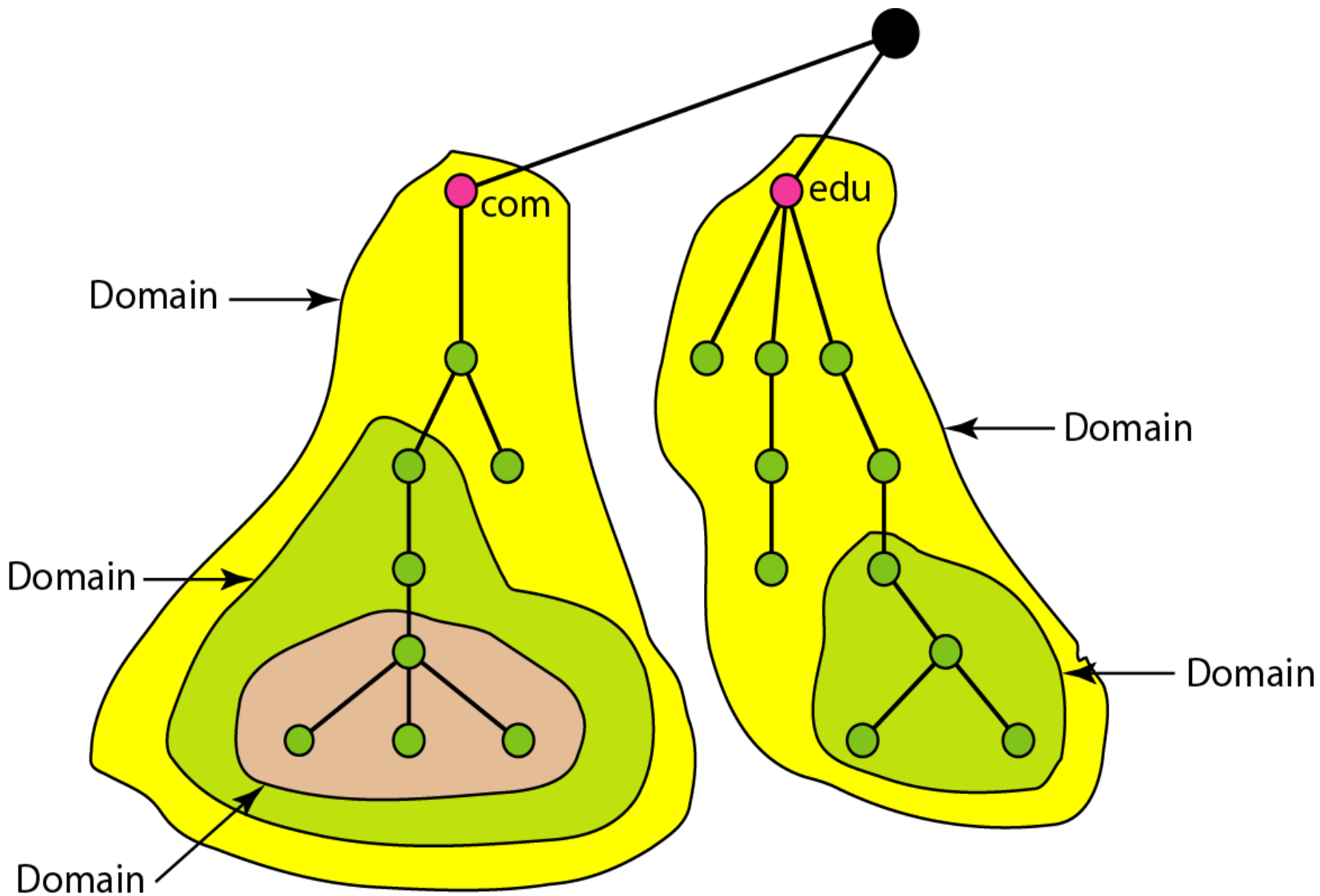
- “Top Level Domains” are at the top
- Depth of tree is arbitrary (limit 128)
- Domains are subtrees
 - E.g: .edu, helwan.edu, fci.helwan.edu
- Name collisions avoided
 - E.g. helwan.edu and helwan.com can coexist, but uniqueness is job of domain

Host names are administered hierarchically



- A **zone** corresponds to an administrative authority that is responsible for that portion of the hierarchy
- E.g. CIO controls names:
x.fci.helwan.edu and y.fci.helwan.edu
- E.g. The President controls names:
x.helwan.edu and y.eng.helwan.edu





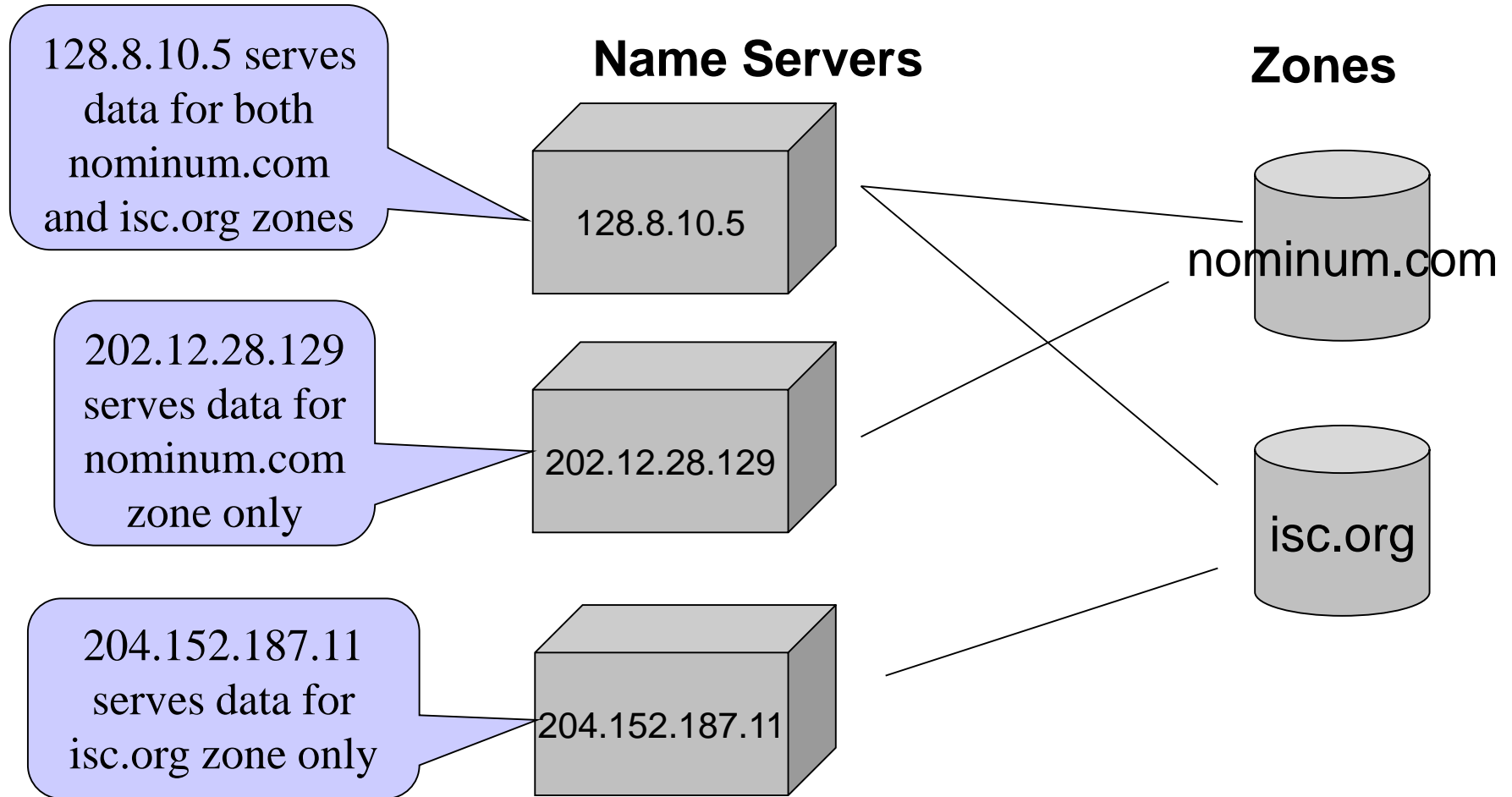
Overview

- Introduction to the DNS
- DNS Components
 - The name space
 - The servers
 - The resolvers
- DNS Structure and Hierarchy
- The DNS in Context

Name Servers

- Name servers store information about the name space in units called “zones”
 - The name servers that load a complete zone are said to “have authority for” or “be authoritative for” the zone
- Usually, more than one name server are authoritative for the same zone
 - This ensures redundancy and spreads the load
- Also, a single name server may be authoritative for many zones

Name Servers and Zones



Types of Name Servers

- Two main types of servers
 - Authoritative – maintains the data
 - Master – where the data is edited
 - Slave – where data is replicated to
 - Caching – stores data obtained from an authoritative server
- No special hardware necessary

Name Server Architecture

- You can think of a name server as part of:
 - *database server*, answering queries about the parts of the name space it knows about (i.e., is authoritative for),
 - *cache*, temporarily storing data it learns from other name servers, and
 - *agent*, helping resolvers and other name servers find data

Name Server Architecture

Name Server Process

Authoritative Data

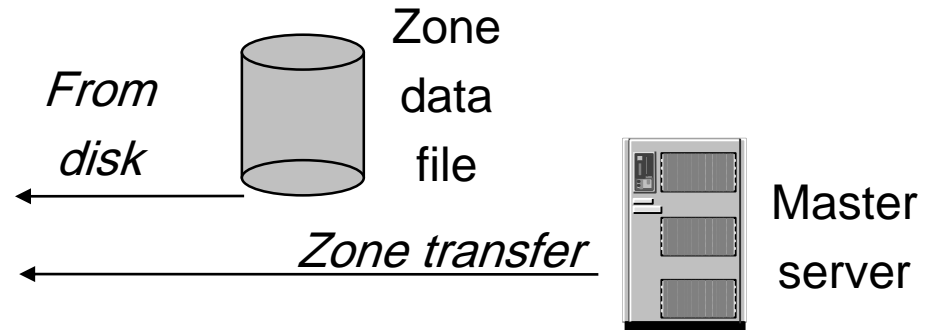
(primary master and slave zones)

Cache Data

(responses from other name servers)

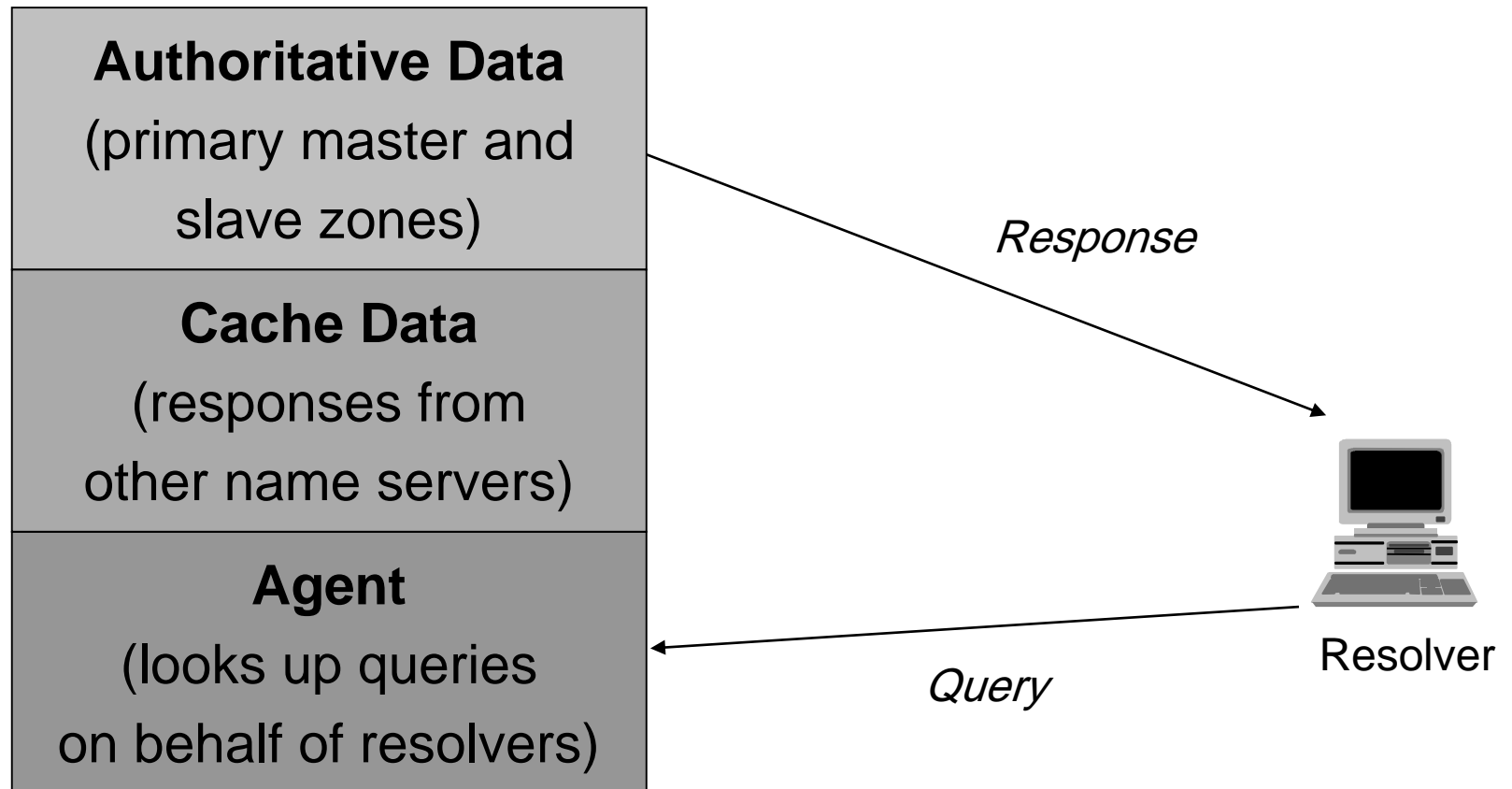
Agent

(looks up queries on behalf of resolvers)



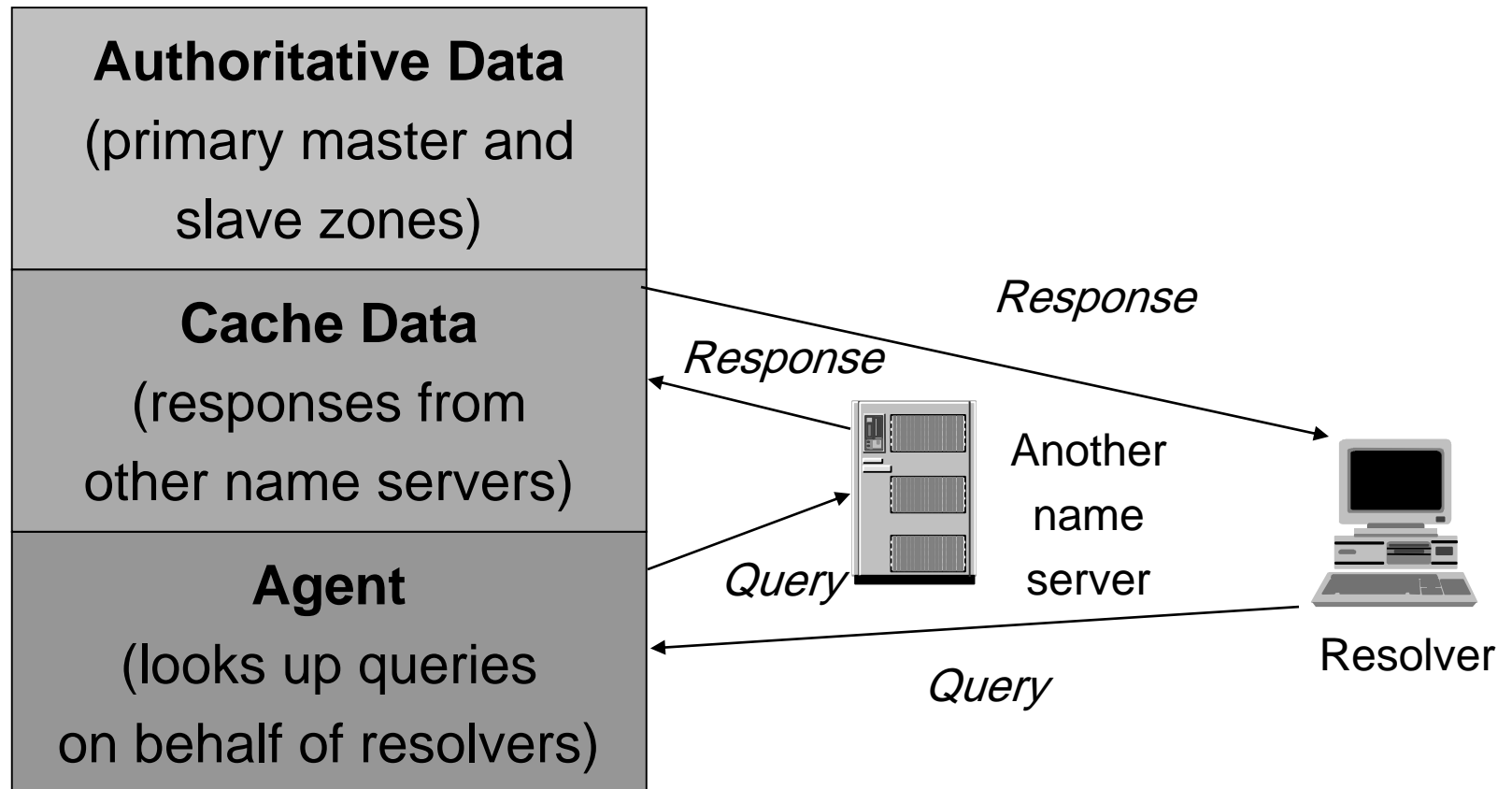
Authoritative Data

Name Server Process



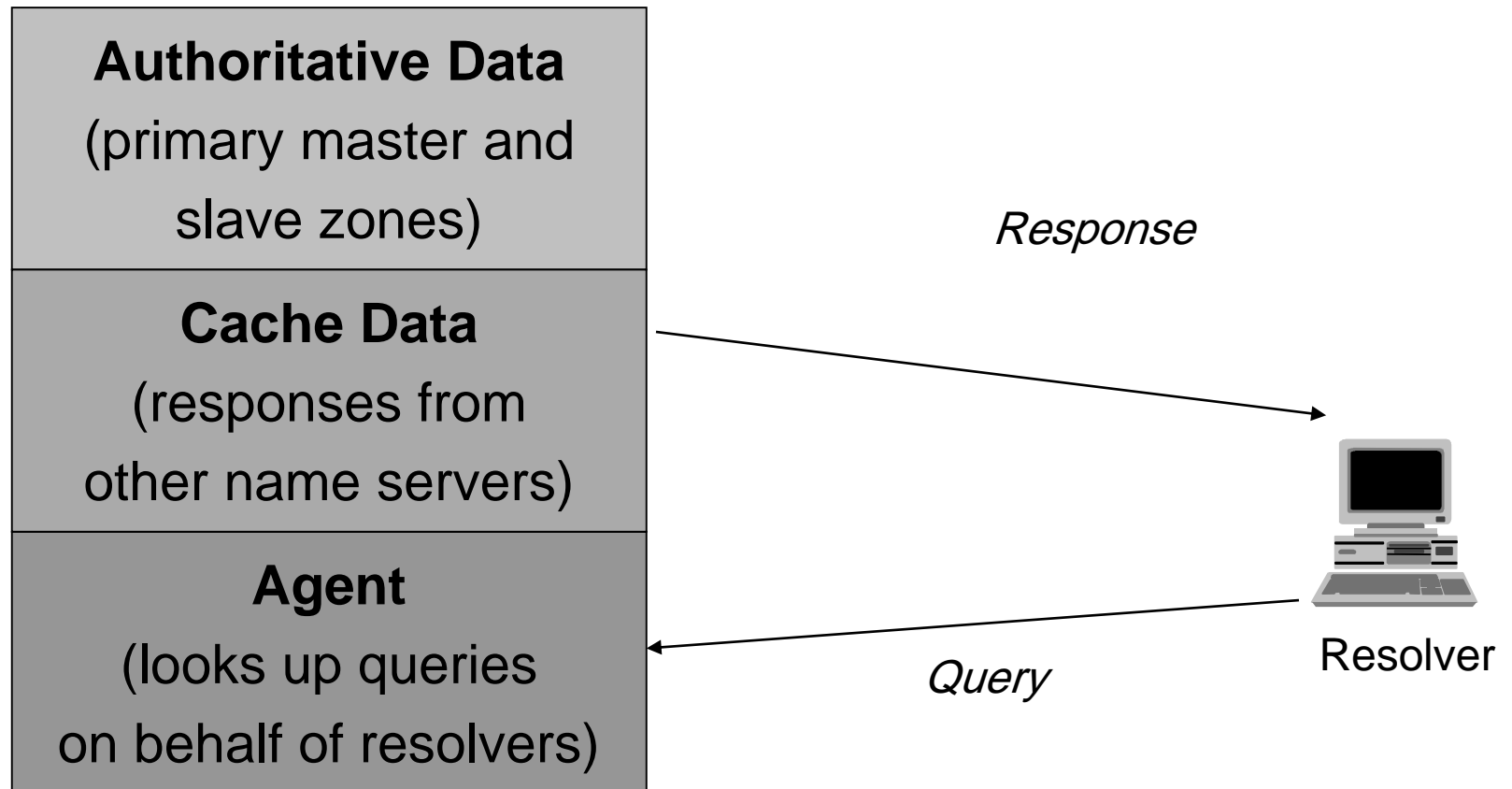
Using Other Name Servers

Name Server Process



Cached Data

Name Server Process



Overview

- Introduction to the DNS
- DNS Components
 - The name space
 - The servers
 - The resolvers
- DNS Structure and Hierarchy
- The DNS in Context

Name Resolution

- *Name resolution* is the process by which resolvers and name servers cooperate to find data in the name space
- Closure mechanism for DNS?
 - Starting point: the names and IP addresses of the name servers for the root zone (the “root name servers”)
 - The root name servers know about the top-level zones and can tell name servers whom to contact for all TLDs

Name Resolution

- A DNS query has three parameters:
 - A domain name (e.g., *www.nominum.com*),
 - Remember, every node has a domain name!
 - A class (e.g., *IN*), and
 - A type (e.g., *A*)
 - <http://network-tools.com/nslook/>
- Upon receiving a query from a resolver, a name server
 - 1) looks for the answer in its authoritative data and its cache
 - 2) If step 1 fails, the answer must be looked up

The Resolution Process

- Let's look at the resolution process step-by-step:

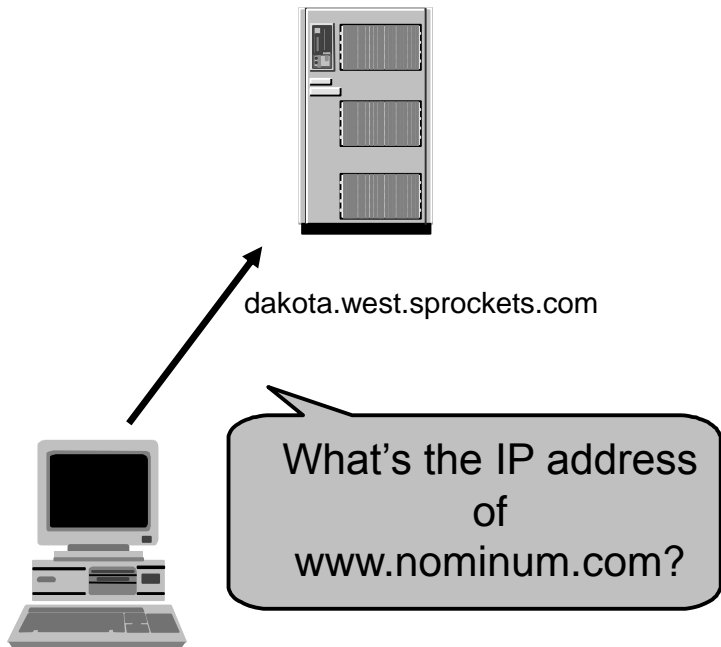


`annie.west.sprockets.com`

`ping www.nominum.com.`

The Resolution Process

- The workstation *annie* asks its configured name server, *dakota*, for *www.nominum.com*'s address

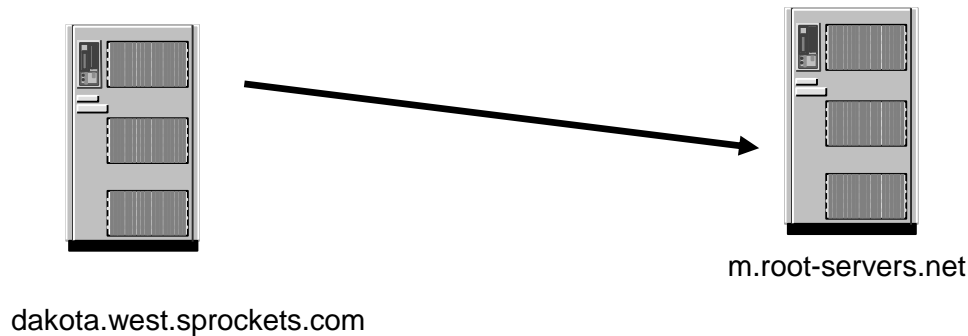


annie.west.sprockets.com

`ping www.nominum.com.`

The Resolution Process

- The name server *dakota* asks a root name server, *m*, for *www.nominum.com*'s address



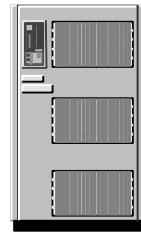
`annie.west.sprockets.com`

What's the IP address
of
`www.nominum.com`?

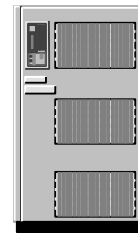
`ping www.nominum.com.`

The Resolution Process

- The root server *m* refers *dakota* to the *com* name servers
- This type of response is called a “referral”



dakota.west.sprockets.com



m.root-servers.net

Here's a list of the
com name servers.
Ask one of them.

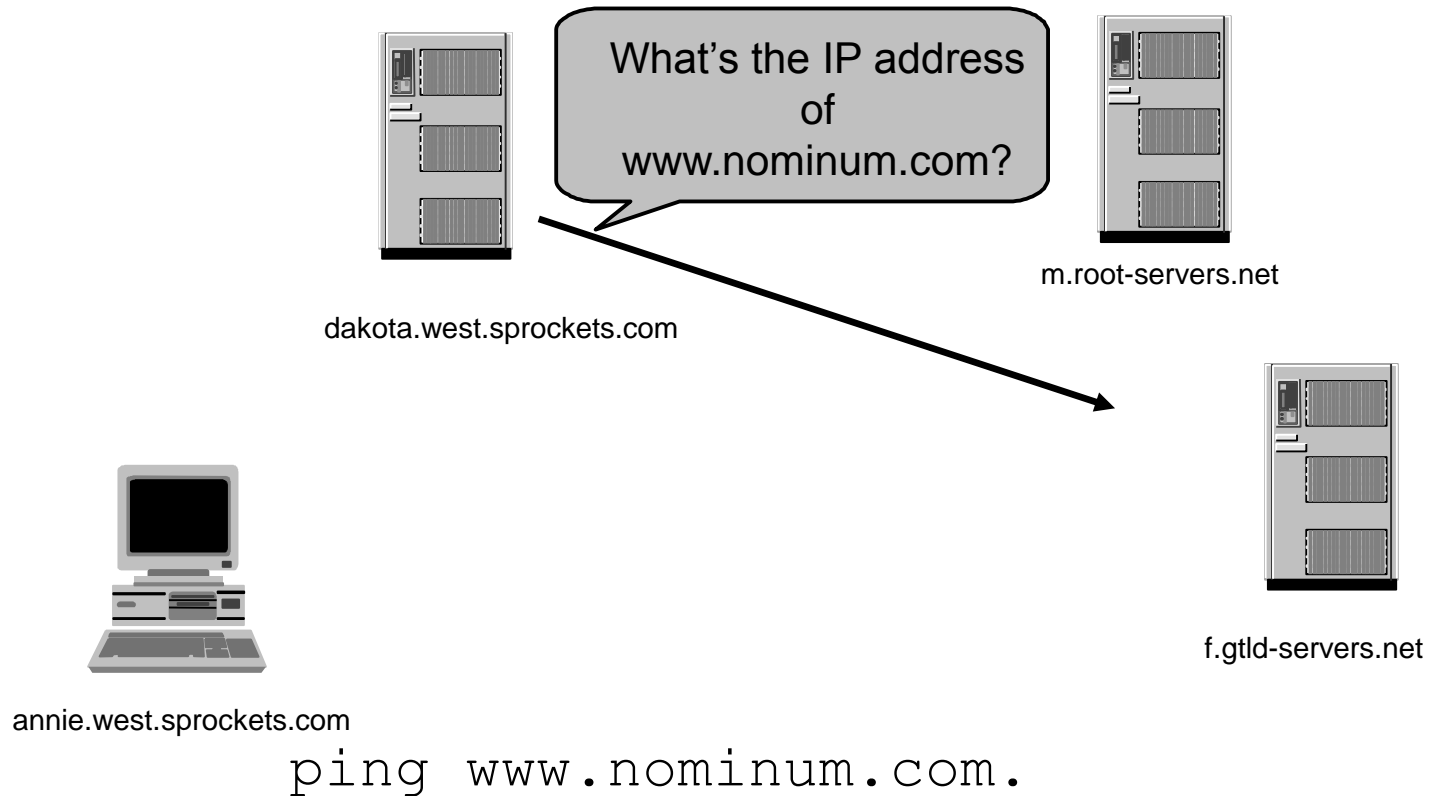


annie.west.sprockets.com

ping www.nominum.com.

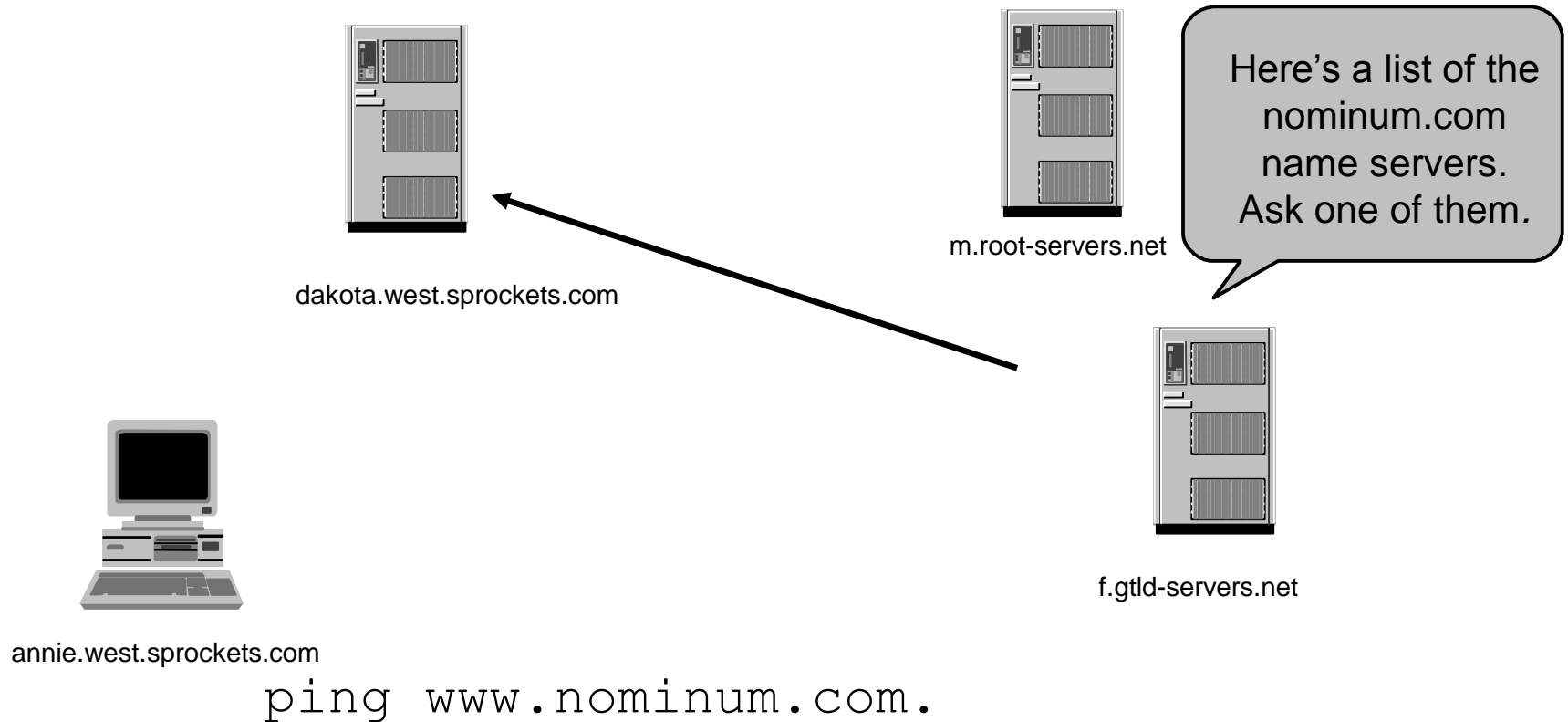
The Resolution Process

- The name server *dakota* asks a *com* name server, *f*, for *www.nominum.com*'s address



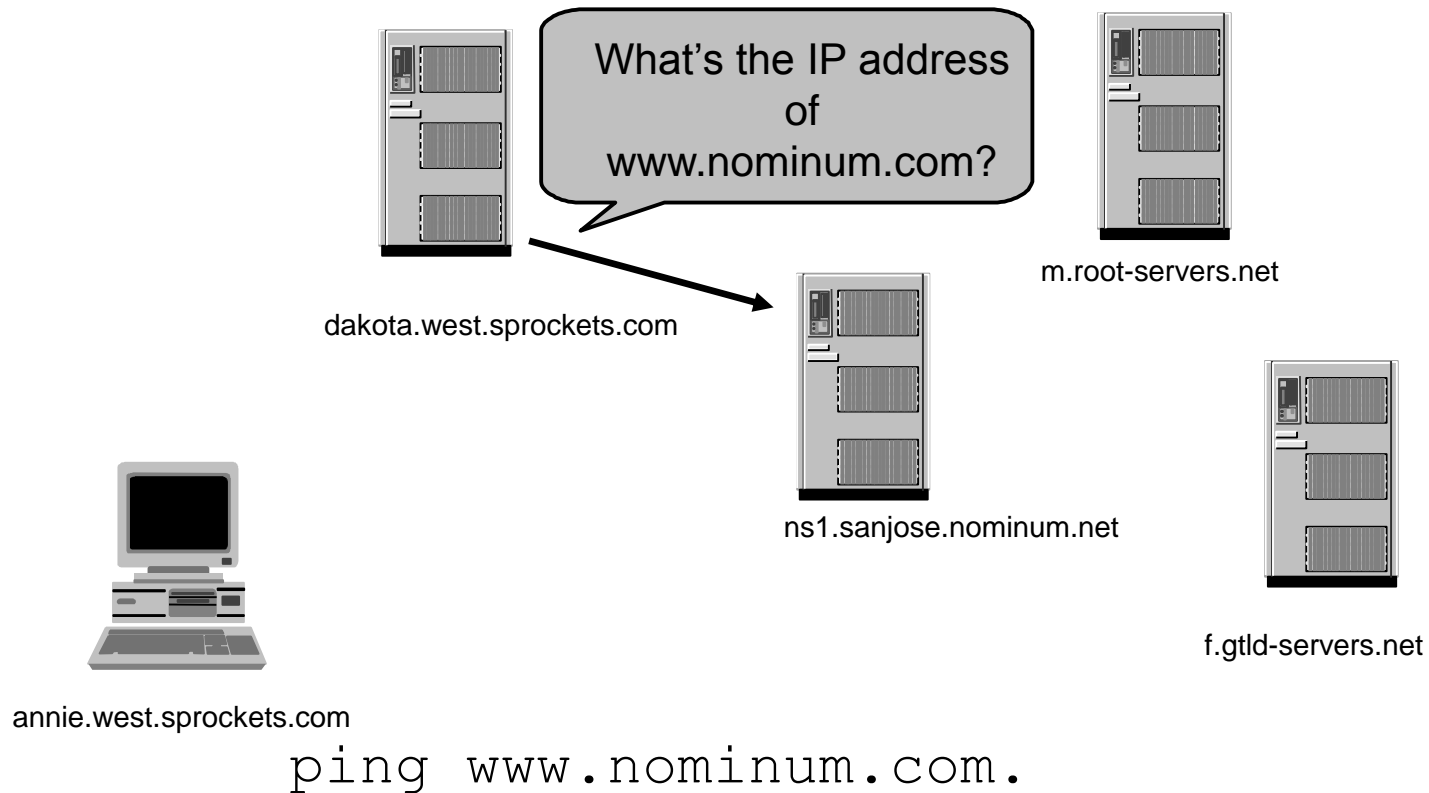
The Resolution Process

- The *com* name server *prefers* *dakota* to the *nominum.com* name servers



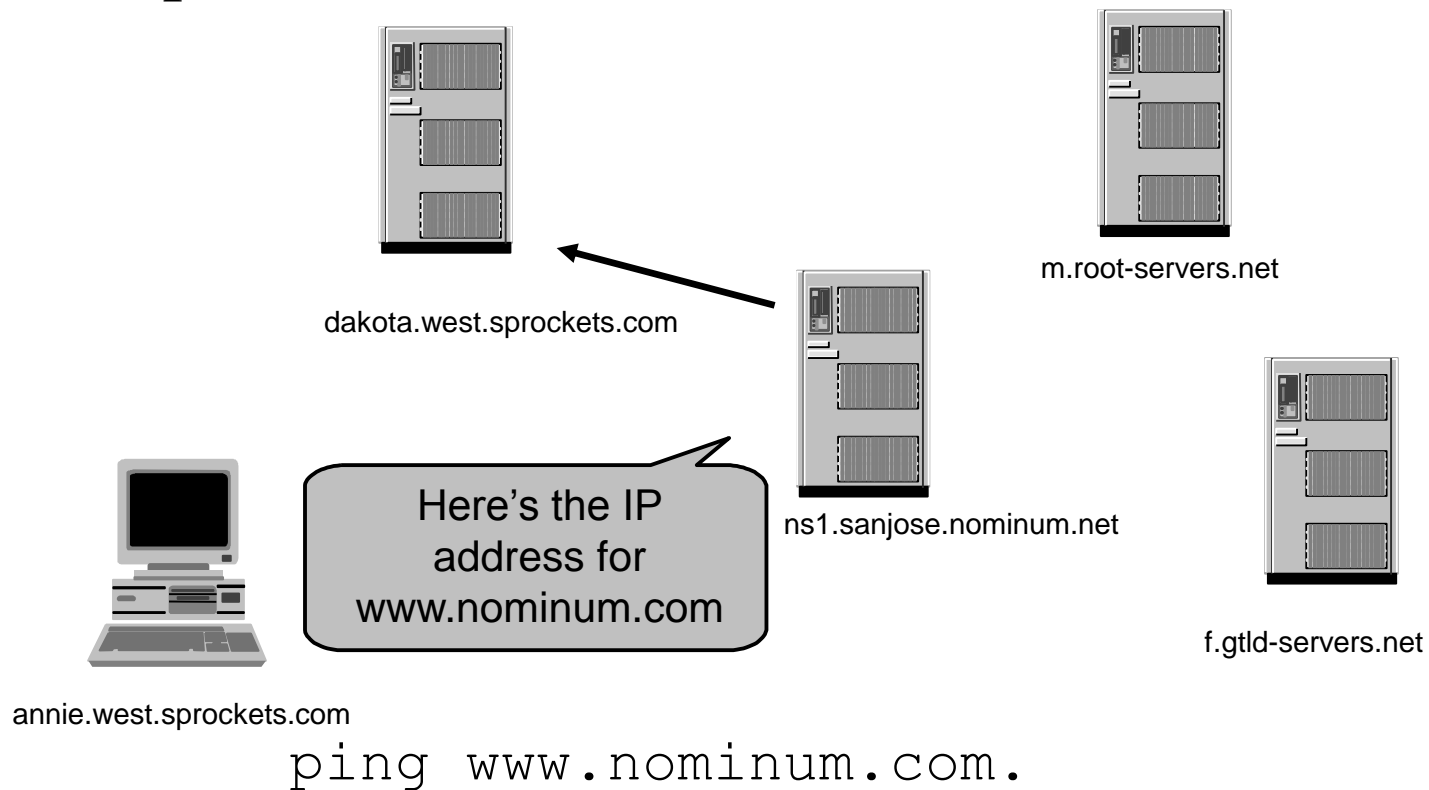
The Resolution Process

- The name server *dakota* asks a *nominum.com* name server, *ns1.sanjose*, for *www.nominum.com*'s address



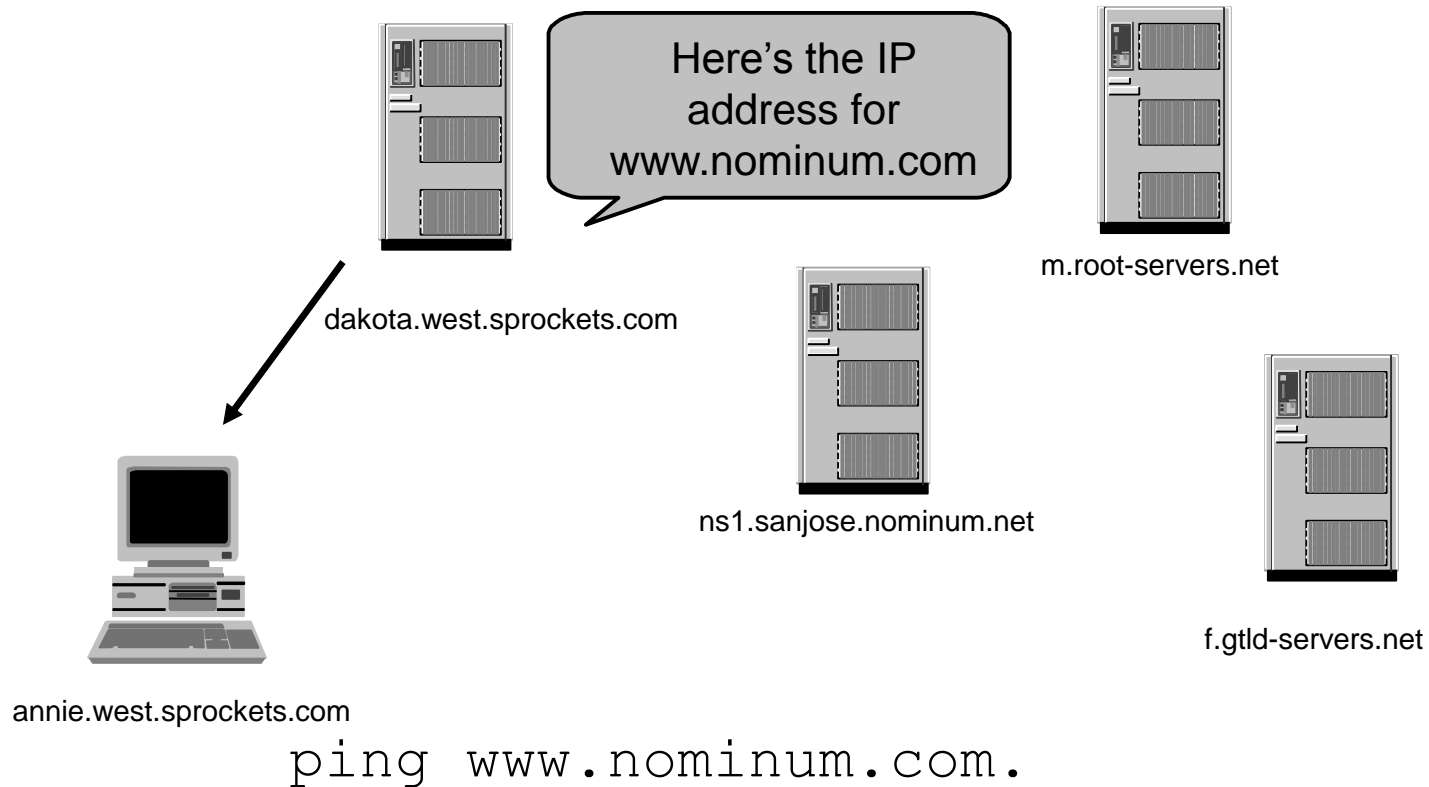
The Resolution Process

- The *nominum.com* name server *ns1.sanjose* responds with *www.nominum.com*'s address



The Resolution Process

- The name server *dakota* responds to *annie* with *www.nominum.com*'s address



Resolution Process (Caching)

- After the previous query, the name server *dakota* now knows:
 - The names and IP addresses of the *com* name servers
 - The names and IP addresses of the *nominum.com* name servers
 - The IP address of *www.nominum.com*
- Let's look at the resolution process again

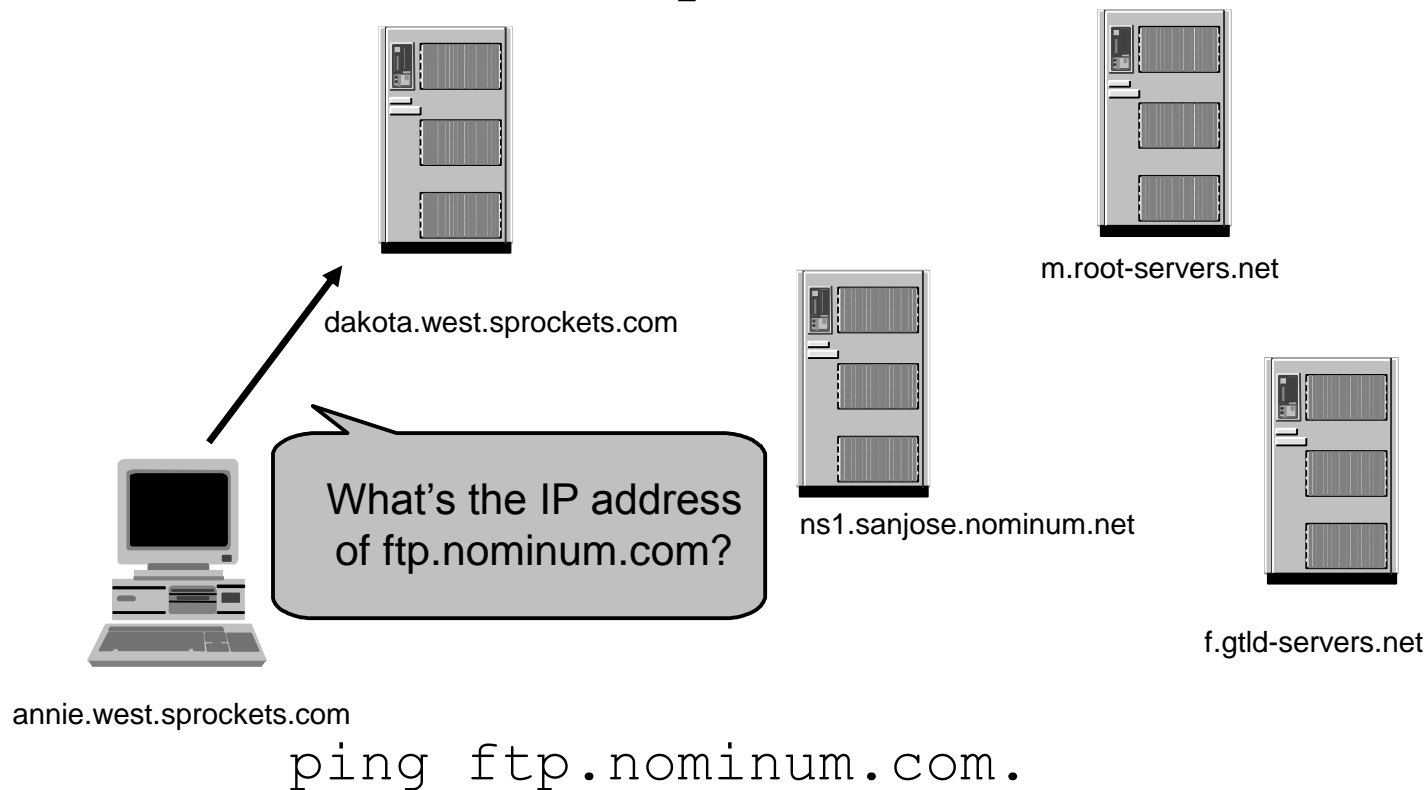


annie.west.sprockets.com

ping **ftp**.nominum.com.

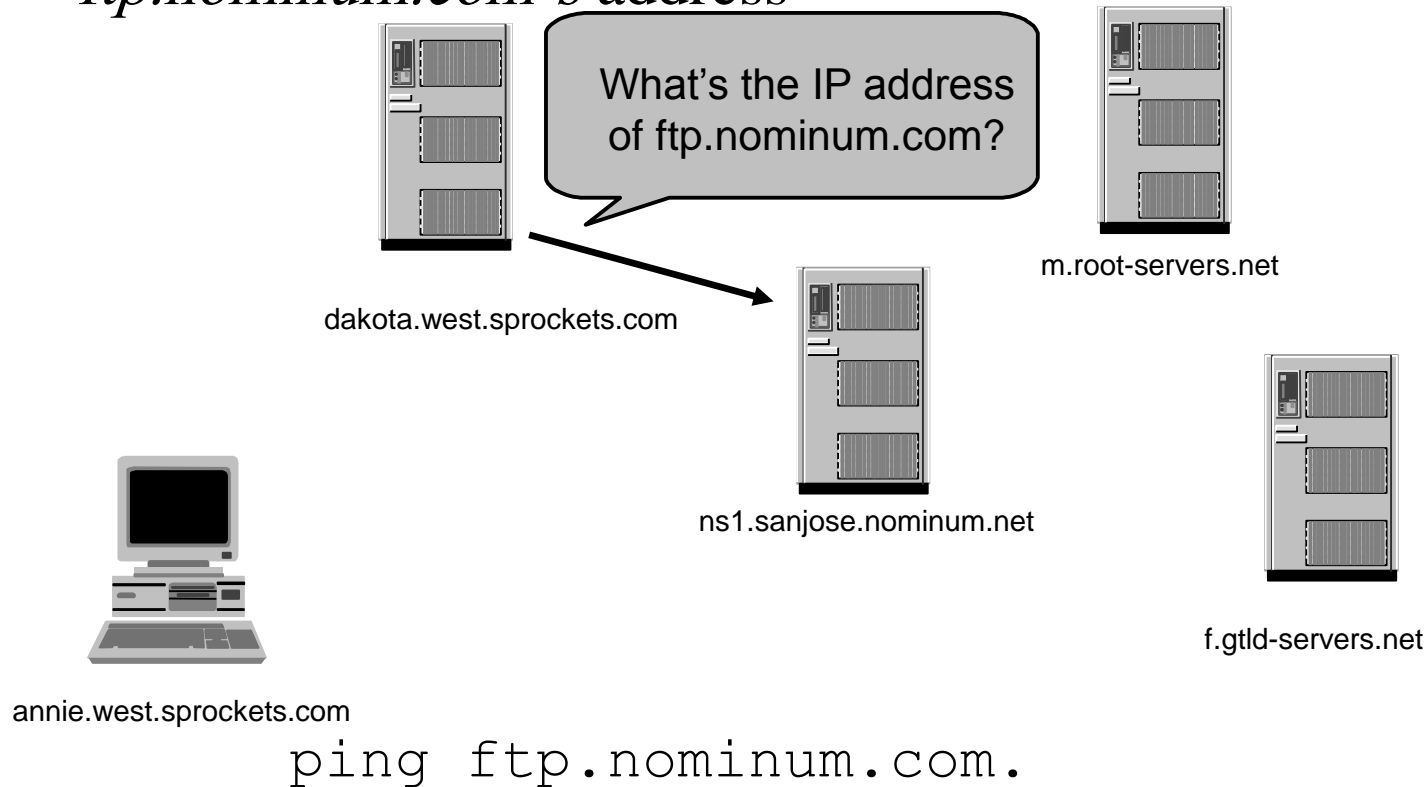
Resolution Process (Caching)

- The workstation *annie* asks its configured name server, *dakota*, for *ftp.nominum.com*'s address



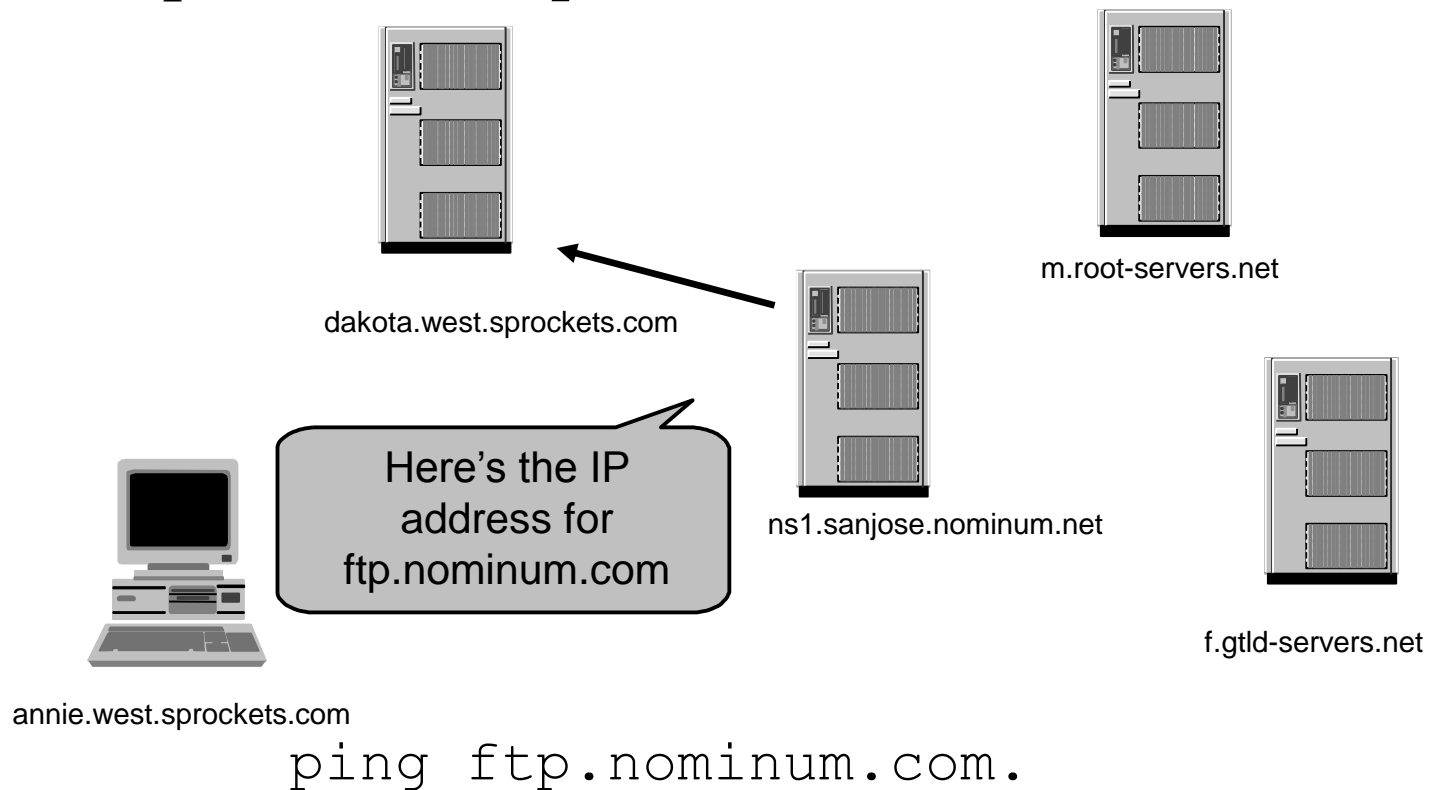
Resolution Process (Caching)

- dakota* has cached a NS record indicating *ns1.sanjose* is an *nominum.com* name server, so it asks it for *ftp.nominum.com*'s address



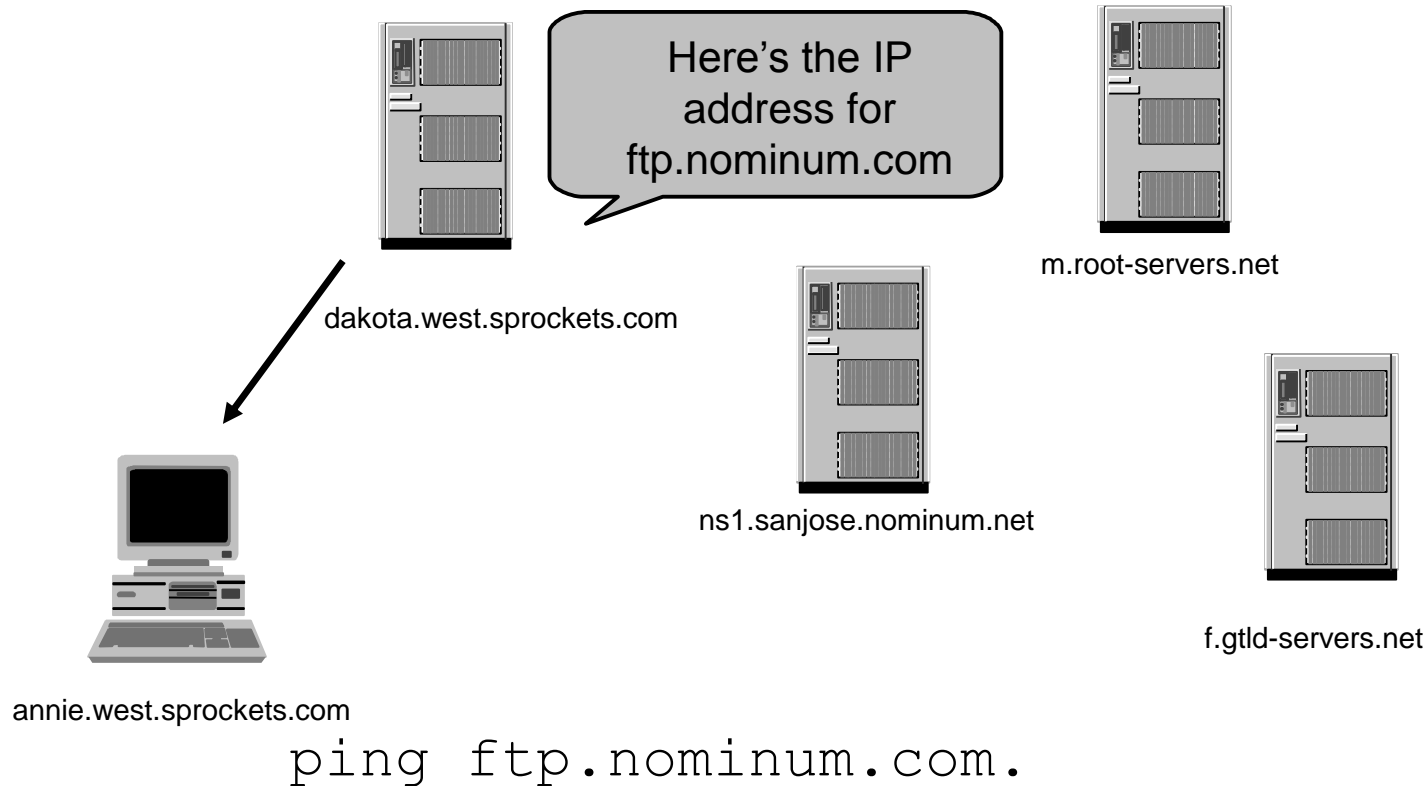
Resolution Process (Caching)

- The *nominum.com* name server *ns1.sanjose* responds with *ftp.nominum.com*'s address



Resolution Process (Caching)

- The name server *dakota* responds to *annie* with *ftp.nominum.com*'s address



Overview

- Introduction to the DNS
- DNS Components
- DNS Structure and Hierarchy
- The DNS in Context

DNS Structure and Hierarchy

- The DNS imposes no constraints on how the DNS hierarchy is implemented except:
 - A single root
 - The label restrictions
 - So, can we create a host with a name *a.wonderful.world*?
- If a site is not connected to the Internet, it can use any domain hierarchy it chooses
 - Can make up whatever TLDs (top level domains) you want
- Connecting to the Internet implies use of the existing DNS hierarchy

Internet Corporation for Assigned Names and Numbers (ICANN)

- ICANN's role: to oversee the management of Internet resources including
 - Addresses
 - Delegating blocks of addresses to the regional registries
 - Protocol identifiers and parameters
 - Allocating port numbers, etc.
 - Names
 - Administration of the root zone file
 - Oversee the operation of the root name servers

The Root Nameservers

- The root zone file lists the names and IP addresses of the authoritative DNS servers for all top-level domains (TLDs)
- The root zone file is published on 13 servers, “A” through “M”, around the Internet
- Root name server operations currently provided by volunteer efforts by a very diverse set of organizations

Root Name Server Operators

Nameserver	Operated by:
A	Verisign (US East Coast)
B	University of S. California –Information Sciences Institute (US West Coast)
C	Cogent Communications (US East Coast)
D	University of Maryland (US East Coast)
E	NASA (Ames) (US West Coast)
F	Internet Software Consortium (US West Coast)
G	U. S. Dept. of Defense (ARL) (US East Coast)
H	U. S. Dept. of Defense (DISA) (US East Coast)
I	Autonomica (SE)
J	Verisign (US East Coast)
K	RIPE-NCC (UK)
L	ICANN (US West Coast)
M	WIDE (JP)

Registries, Registrars, and Registrants

- A classification of roles in the operation of a domain name space
- Registry
 - the name space's database
 - the organization which has edit control of that database
 - the organization which runs the authoritative name servers for that name space
- Registrar
 - the agent which submits change requests to the registry on behalf of the registrant
- Registrant
 - the entity which makes use of the domain name

Registries, Registrars, and Registrants

