

# *Domain Name Service*

## *An Introduction*



## **PSINet** *Domain Name Service*

---

- Distributed database of Internet host information.
- Distributes the workload and responsibility for assigning and maintaining host names.
- Provides a way for users to use host names (instead of IP addresses) to reference machines.
- Used by most Internet applications.
- Scalable solution for organizing and finding all of the hosts on the Internet.

Copyright © 1997 PSINet Inc.

Confidential and Proprietary Information

T09.2

### **Domain Name Service (DNS)**

DNS is often termed the Internet's phone book because one of its most important functions is to map domain names to IP addresses. This is useful when navigating the Internet's vast virtual space. DNS is a distributed database of several types of host information such as IP addresses, host names, mail exchangers and other information necessary for the name service overhead. The responsibility for maintaining the DNS is also distributed and falls to zone administrators. Each zone administrator is responsible for maintaining correct and current information for all domains within their zone.



## *The "Name Space"*

- The Issues
  - Centralized vs. decentralized authority
  - Access to naming information
  - Reliable distribution of naming information
- Flat name space
  - Simple to implement
  - Does not scale
- Hierarchical name space (DNS)
  - Complex to implement and maintain
  - Designed to scale

Copyright © 1997 PSINet Inc.

Confidential and Proprietary Information

T09.3

The names of all the hosts on the Internet make up the **name space**. Name service has evolved with the growth of the Internet and is now organized as a hierarchical tree structure. This structure is relatively complex and certain conventions need to be followed when adding a domain name.

The changes in the organization of domain names occurred mostly out of the need for a solution that would scale as the Internet grew. The main issues that needed to be addressed were zone authority, access to correct information and distribution of name changes, deletions and additions.



## *Flat Name Space*

- Authority is central.
- A complete list of names: a “host table”
- Used in store-and-forward networks:  
UUCP, BITNET
- Does not scale.
  - Information must be completely replicated and stored locally.
  - Impossible to keep the list up-to-date.

Copyright © 1997 PSINet Inc.

Confidential and Proprietary Information

T09.4

The **flat name space** featured central authority over all domains as well as a central distribution point of information. Information for all the hosts on the Internet were stored in one host table, replicated in full and distributed to all Internet sites. It was relatively easy to implement, but could not scale to the potential size of the Internet.

When changes were made to the host table, the new data needed to be added and distributed to all the Internet sites. This became increasingly difficult and less consistent as the number of hosts on the Internet grew.

The **Internet Network Information Center (NIC)** administrated the name space and was directly responsible for registering new names and distributing the name list. By 1990, the flat list of Internet hosts was obsolete containing only 6400+ names.



## *Flat Name Space: Host Table & Structure*

### Internet Host Table

IP Address	Host
<b>10.0.0.51</b>	<b>sri-nic</b>
<b>26.0.0.73</b>	<b>sri-nic</b>
<b>10.0.0.21</b>	<b>mars</b>
<b>10.1.0.21</b>	<b>venus</b>
<b>10.2.0.21</b>	<b>saturn</b>
<b>10.3.0.21</b>	<b>pluto</b>

Here is an example of a portion of a flat host table. Note that names were only one level deep (flat). The NIC was responsible to insure that there were no repetitive names.

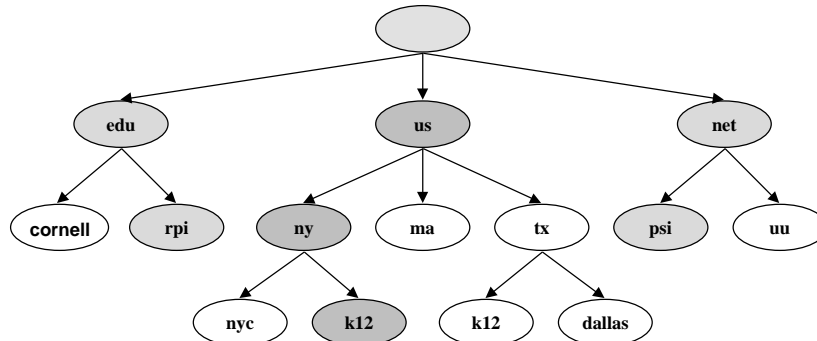


## *Domain Name Space*

- Authority is decentralized.
- Organized as a hierarchical tree.
- Scales well.
  - Information is easily kept up-to-date.
  - Information is shared on an as needed basis.

In the DNS, authority over information content and propagation is decentralized. It is organized as a hierarchy and authority over subhierarchies is delegated to name servers and their administrators across the Internet. The most important thing to remember about DNS is that it is a distributed database of host information. Information about a certain domain is stored in full on a certain set of name servers and shared on an as needed basis. Since information about a domain is directly controlled by the zone administrator, host information is easily kept up-to-date, and generally consistent (albeit not always correct). As new sites join the net, authority and responsibility for their domain is delegated to them or their ISP. This allows the service to scale easily as the Internet grows.

- ❑ Uniquely identify every node.
- ❑ FQDN - Fully Qualified Domain Name
- ❑ Siblings must have unique names.



Copyright © 1997 PSINet Inc.

Confidential and Proprietary Information

T09.7

By now many of you have your domain name, so you are somewhat familiar with what it is and how it is used. Each node in your network should be named something.domain.com. When naming machines, you should be sure that two siblings do not have the same name. This will insure that the domain name of the machine is unique to the Internet.

Each node in the tree represents some domain name. The domain name of a node is derived by choosing a node on the tree and reading up the tree to the root node. For example, the **FQDN (Fully Qualified Domain Name)** of the Dallas node is dallas.tx.us. Note that the “.” at the end of the name is required and very important when configuring your name server or making a DNS query.

The administrator of the com. domain is responsible to insure that all child nodes of com are unique. You (or whoever administers your name service) are responsible to insure that you use unique domain names at each level of your subtree “domain.com.”



## *Domains*

---

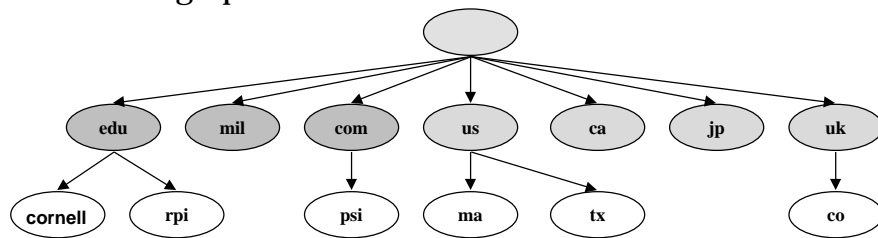
- Define a place in the name space.
- Encompass an entire subtree of the name space (subdomains).
- Can be delegated away.
- Organize information of the domain name space.

The **domains** define a certain area of the name space. In general, it is an entire subtree of the DNS structure.

Domains can be delegated away. For example, the person who was in charge of the net domain delegated the psi.net domain to administrators at PSINet. This means that the NIC knows that to find host information about machines in the psi.net domain, it must ask a psi.net name server.

Also, psi.net has subdomains of psi.net, news.psi.net and dns.psi.net. Since we are authoritative for the domain, we can create as many subdomains of the psi.net subtree as we need. These subdomains can be delegated away to another zone or they can remain part of the psi.net zone.

- Direct descendants of the root
- In the US
  - Organizational & geographic
- Outside the US
  - Geographic



Copyright © 1997 PSINet Inc.

Confidential and Proprietary Information

T09.9

The direct descendants of the root domain are called **top-level domains**. They define the overall structure of the DNS tree. In the United States, the domains are mostly organizational, but the us domain is used for geographic naming.

Outside the the U.S., countries use ISO country codes as top level domains. Many of these countries further subdivide their top level domain into organizational designations. The organizational designations do not necessarily follow the naming conventions set forth in the U.S. In the U.K., for example, co is used for corporations, not com.

<b>com</b>	for-profit businesses (corporate or individuals)
<b>net</b>	network providers
<b>org</b>	not-for-profit organizations
<b>edu</b>	4-year degree granting colleges and universities not public or private secondary schools not community colleges
<b>gov</b>	non-military government facilities
<b>mil</b>	military facilities (not administered by InterNIC)
<b>us</b>	geographic designation of sites in the United States
<b>uk</b>	geographic designation of sites in the United Kingdom

## *Top Level arpa. Domain*

- All pre-“domain name” hosts were first put into arpa.
- Created as a transition from flat to hierarchical name space.
- in-addr.arpa
  - Established to map IP addresses to hostnames.
  - IP addresses must be reversed to be added.

Copyright © 1997 PSINet Inc.

Confidential and Proprietary Information

T09.10

The **arpa** domain has some interesting history. It was created to transition from the flat name space to the DNS. All the pre-DNS hosts were put into the arpa domain and then transitioned to the appropriate subtree.

Today this domain is largely unused except for the **in-addr.arpa** domain which is used for all address to name mappings. IP addresses must be reversed when converted into a domain name. The reason for this is that domain names read from most specific (node name) on the left to most general (top level domain) on the right. IP addresses have most specific (host) on the right and most general (network) on the left.

The DNS tree is organized with the most general information at the top to most specific at the leaves. In order to place network numbers in the DNS, they must be reversed so that they are organized in the same fashion as the rest of the tree. They read with their most general (network) information at the top of the tree to their most specific at the leaves.

Examples:

192.33.4.10 -----> 10.4.33.192.in-addr.arpa.  
192.35.82.2 -----> 2.82.35.192.in-addr.arpa.  
136.161.128.2 -----> 2.128.161.136.in-addr.arpa.



## *Zones of Authority: Delegation*

- Represent administrative control over some unique set of domains.
- Zones begin with a node and include all nodes below it until explicitly redelegated.
- Every node has properties associated with it.
- The IP address of a node is a kind of property.
- Properties are called “resource records” (RR).

Copyright © 1997 PSINet Inc.

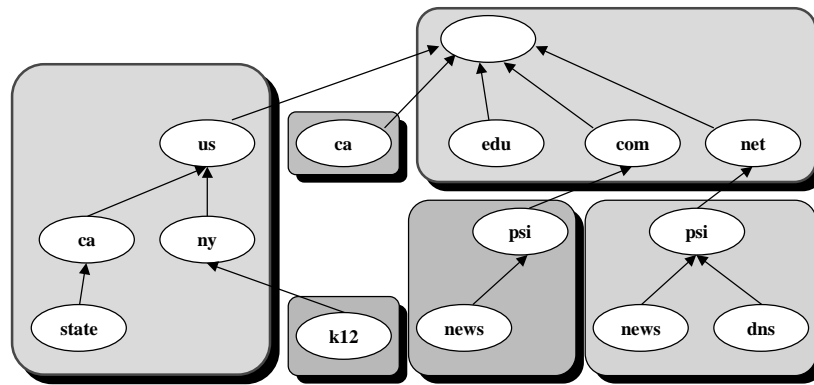
Confidential and Proprietary Information

T09.11

**Zone delegation** facilitates the distribution of the workload of the administration and storage of zone information. **Zone administrators** control the domains in their zone until they explicitly delegate the information away.

The most important rule for zone delegation is that a domain can never be in more than one zone but a zone can contain multiple domains. Specifically, zones start at a single node and include all nodes below it until they are explicitly delegated away.

Every node has properties associated with it. These properties are called **resource records** and include such information as IP to name mappings, mail exchanger priorities, etc.



The above zones are defined by the delegation on the name servers that have authoritative information about the zone. The PSINet name servers know that they are authoritative for the psi.net domain and for the psi.com domain. But these domains are part of different zones.

The servers that have authoritative information for the com domain know that the PSINet name servers are authoritative for the psi.com domain. The servers that are authoritative for the net domain know that the PSINet name servers are authoritative for the psi.net zone. Therefore, psi.net and psi.com are in separate zones because psi.com was once (potentially) part of the com zone and psi.net was once part of the net zone.

Keep in mind that psi.net is still part of the net domain even though it is not in the same zone as net.



## **PSINet** *Two sides of the same coin*

---

- Resolver
  - Client side
  - Routines used by applications to access DNS information from name servers.
- Name Server
  - Server side
  - Provides all pertinent information it knows when queried.
  - Stores the zone information.

There are terabytes of DNS information stored in name servers across the Internet. There are several different types of name servers, but they all work together to provide Internet users with up-to-date DNS information. The resolvers are the client side of the DNS client-server equation.



## *The Resolver*

- System calls or library code
  - Authenticate connections
  - Find IP addresses
  - Find host names
  - Find mail exchanger information
- Completely dependent on name servers.
- Must “point” to some name server.

**Resolvers** are generally not a stand alone application, but are most often built into some other application that needs name information to authenticate connections or find an IP address.

For efficiency, many resolver applications use a cache to store the information that they receive so that they do not have to make subsequent queries for the same information. Resolving software must be configured to point to a name server by IP address.



## *The Name Server*

- Several configuration options, including:
  - Primary name server
    - THE authoritative information source for the zone.
    - Resource records are directly loaded into zone files by the administrator.
  - Secondary name server
    - Also AN authoritative information source for the zone.
    - Obtains zone data from the primary or some other secondary name server of the zone.
  - Caching-only name server
    - NOT authoritative for zone.

Copyright © 1997 PSINet Inc.

Confidential and Proprietary Information

T09.15

The **name server** provides the answers to resolver queries. It can be configured to provide authoritative answers for one or more DNS zones or as a caching-only name server. When properly configured it can provide information on any zone in the Domain Name Service although it doesn't actually have all the information stored locally.

A **primary name server** and a **secondary name server** for a certain zone each contain a copy of the same zone information. The only difference is how the information gets on the server. The information is loaded into a primary name server by a zone administrator. Secondary name servers connect to a primary name server for the zone and transfer the data when it has changed. Both primary and secondary name servers return authoritative answers to queries about information within their zone.

The name server can be configured to be a primary name server for a certain zone, and a secondary for another or just primary or just secondary.

A **caching-only name server** can be configured to find information from authoritative name servers and return it to resolvers. They do not store whole zone files but they will cache any records that they find during a query for the amount of time indicated in the zone file.

- Two types

- Recursive

- The response to a recursive query will be:

- The answer to the query, or

- An error

- Iterative

- The response to an iterative query will be:

- The answer to the query, or

- A list of name servers that may have the answer to the query, or

- An error

When a DNS query is made, the resolver makes a recursive query to some name server. This is part of the resolver configuration. If that name server has the answer, either because it is authoritative or has it cached, it will return it to the resolver. If the name server does not have the answer, it will try to find the answer from other servers in the DNS. When it finds the answer, it returns it to the resolver. If it does not find the answer, then it returns an error. So a response to a recursive query is either the answer or an error.

Name servers are usually configured to make iterative queries. This type of query returns either the answer, a list of other name servers to ask, or an error. When a name server is queried, it will return the answer if it knows it, or it will query a root name server to find out where to look for the answer.

A root name server is always the starting point for each iterative query made by a name server. The root name servers have the delegation information to tell the querying name server where to find the name information. The querying name server then queries the name server that has the requested information. This name server returns any information it has: either another list, the answer or an error. The name server continues to query a name server from each list until it finds the answer or receives an error.



## *A Typical Query*

**mac1.abc.com.**  
**(resolver)**



**What is the IP address of  
www.la.company.com.?**

Copyright © 1997 PSINet Inc.

Confidential and Proprietary Information

T09.17

This is an example of a DNS query by a resolver on mac1.abc.com. for the IP address of www.la.company.com.

In most cases a resolver will query some local host information like a cache or NIS or other local hosts file before querying a name server. Let us assume that the IP address was not found locally so a recursive query to some name server must be made.

res1.dns.psi.net.  
(a caching name  
server)

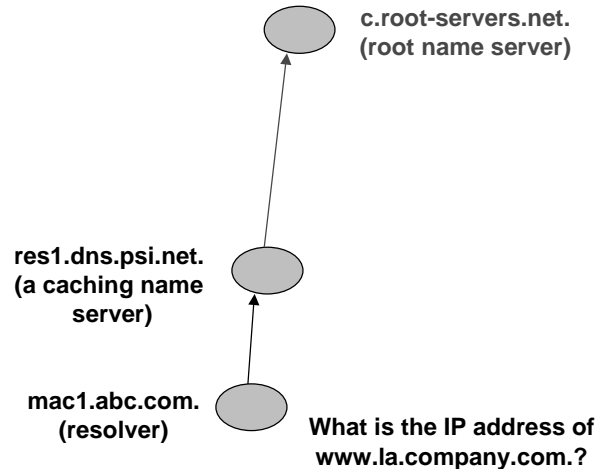


mac1.abc.com.  
(resolver)

What is the IP address of  
www.la.company.com.?

The resolver on mac1 is configured to point to 38.8.81.2, a caching only name server operated by PSINet called res.dns.psi.net. Res1 checks the information it has cached to see if it already knows the IP address for www.la.company.com.

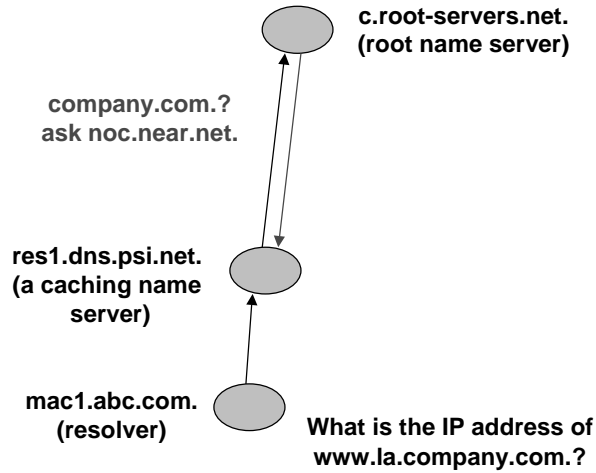
It does not, so it immediately queries a root name server to find out which name server on the Internet has the IP address it is looking for or information on where it will be found.



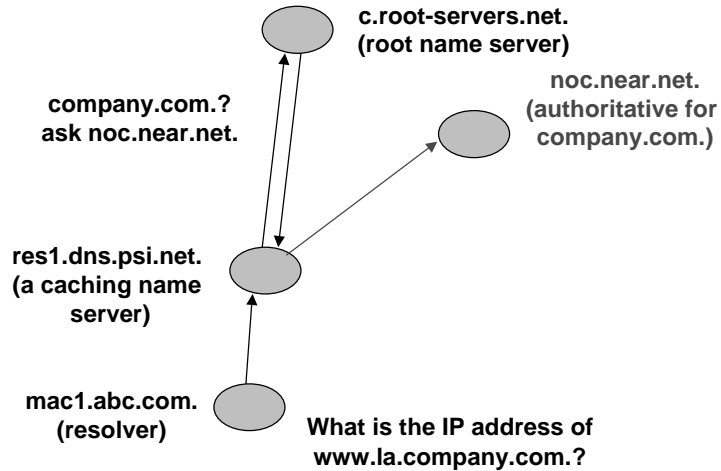
Res1 makes an iterative query to c.root-servers.net, one of the root name servers. All root name servers have the same information, so it does not matter which root name server is queried.

Res1 makes an iterative query to c.root-servers.net and c checks to see what name servers were delegated the responsibility of the company.com domain.

Note that c.root-servers.net only tells res1 where to look for the answer and does not attempt to look for the final answer. Res1 must iteratively query other name servers until it finds the final answer or receives an error.

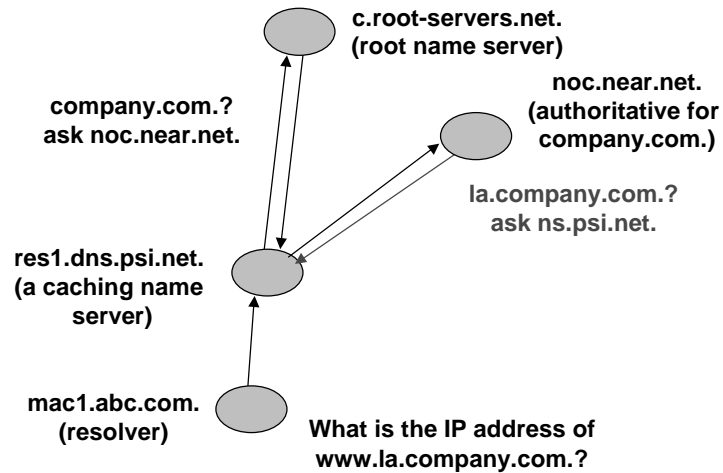


The root name server determines that the name server noc.near.net. is one of the authoritative name servers for the domain company.com and reports this to res1 along with the IP address of noc.near.net.



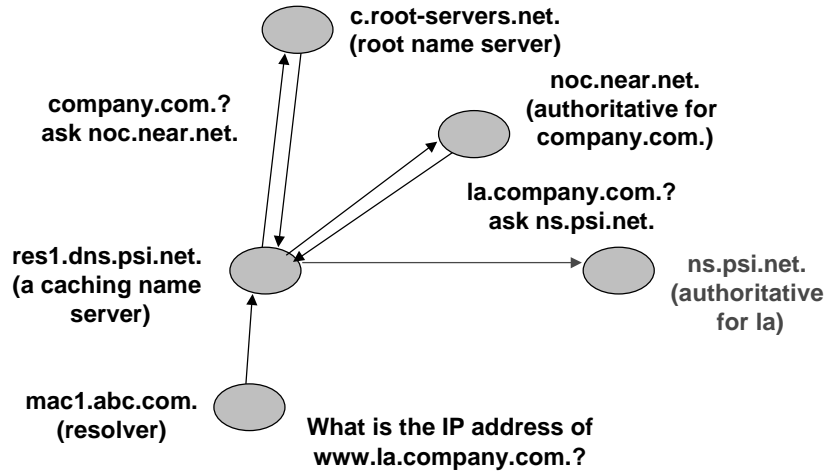
Res1 queries noc.near.net. for the IP of www.la.company.com.

In this case, noc.near.net. is not authoritative for the la.company.com. zone. It has delegated authority of the la.company.com. zone to some other name server.



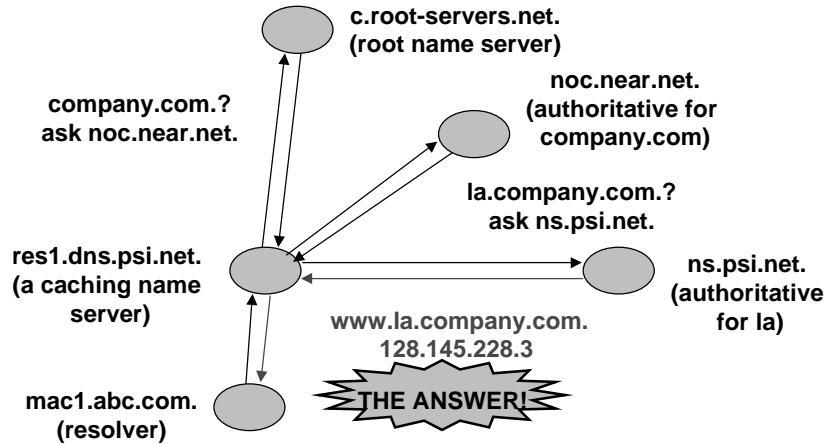
Noc.near.net. returns the name ns.psi.net. as an authoritative name server for la.company.com.

In reality noc would have returned a list of authoritative name servers so that res1 would be able to query any name server from that list and get the same answer.



Res1 finally queries ns.psi.net. for the IP address of www.la.company.com. Since ns is authoritative for the domain la.company.com., it is able to return the IP of www.

If the host www did not exist, ns would have returned an error and res1 would have stopped looking.



In this final picture, ns returns the IP address of www and res1 stores the IP address in its cache and reports the answer to mac1.

The application on mac1 that made the query uses the IP address as appropriate for the application.



*Questions?*



Copyright © 1997 PSINet Inc.

Confidential and Proprietary Information

T09.25