Transparencias

de

Redes de Ordenadores

Tema 9

Nivel de Red: IP
5ª Parte – DNS

Uploaded by

IngTeleco

DNS

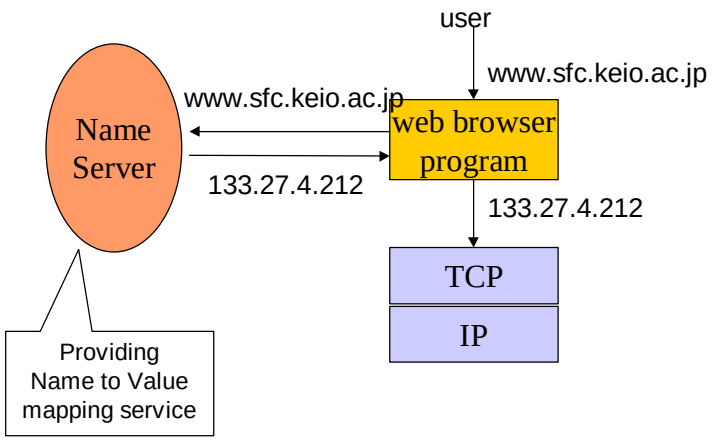
Nombres y Direcciones

- Un nodo debe estar identificado antes de poder comunicarse
  - Un nodo se identifica mediante una dirección IP única
  - Un nodo se identifica por un Nombre único (por conveniencia del usuario)
- Nombres
  - Longitud variable y nemónicos
  - Fácil de recordar para los usuarios
  - No contiene información sobre la ubicación del nodo
- Direcciones
  - Longitud fija
  - Fácil de procesar para los ordenadores
  - Ligado al routing

Name Space and Resolution

- Name Space
  - defines set of possible names
  - flat versus hierarchical
- Naming system
  - maintains collections of a set of name to value bindings
- Resolution mechanism
  - mechanism to get a corresponding value (IP address) from a name (Hostname)
- Name server
  - one implementation of a resolution mechanism
  - widely used in the Internet

A simple example



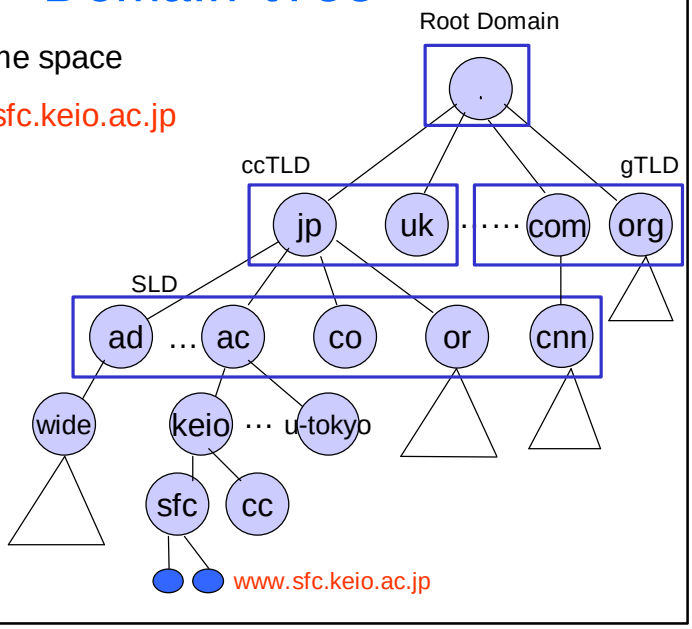
Name Service in Internet

- How to manage Name Space ?
  - flat vs hierarchical
  - Who names hosts ?
  - What kind of mapping data to maintain ?
  - How to maintain the mapping database ?
- How to resolve the names of all over the Internet ?
  - How to look up ?
  - Who will answer ?
- RFC (Standard 13)
  - RFC 1034 "Domain names – concepts and facilities"
  - RFC 1035 "Domain Names - Implementation and Specification"

Domain tree

Hierarchical name space

Example: www.sfc.keio.ac.jp



TLD: Top Level Domain
gTLD: generic TLD
ccTLD: Country Code TLD
SLD: Second Level Domain

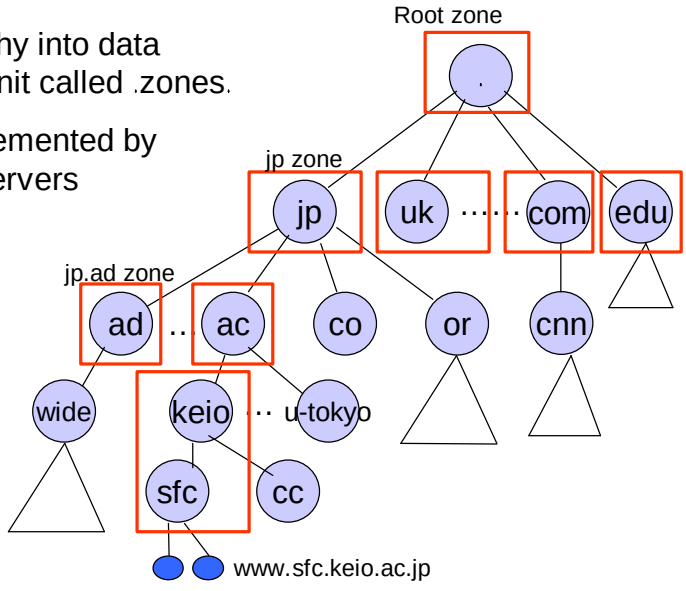
Delegations

- Names are unique in domain
- Each domain is maintained by the domain registry
  - The root domain is maintained by ICANN/IANA
  - Top level domains are maintained by TLD registries
    - jp ccTLD domain : JPNIC
    - .COM gTLD domain : NSI
    - Registry for a domain delegates its sub domains to lower registries
  - JPNIC delegates "keio.ac.jp" domain to Keio University

Domains and Zones

Partition hierarchy into data administration unit called .zones.

Each zone implemented by a set of name servers



Zones and name servers

- Name-value mapping information are maintained by each zone
  - "domain" is name administrative boundary
  - "zone" is mapping data administrative boundary
- A set of responsible name servers are running for each zone
- A server can be responsible for multiple zones

Name Servers

- Each name server maintains a name-value mapping information called "resource records" of a zone it is responsible for.
- Each name server resolves names and answers to queries based on the resource records it maintains.
- Name servers have a mechanism to synchronize and update the resource records for a zone among primary and secondary servers.

Resource Records

- A resource record contains;
  <Name, Value, Type, Class, TTL>
- Name/Value
  - not necessarily hostnames to IP addresses
  - depend on "TYPE"
- Class = IN
  - allows other entities to define a new class
- TTL (Time To Live)
  - How long the RR is valid
  - optional

Types or RR

- A (Address)
  - Name=hostname / Value = IP address
- CNAME (Canonical NAME)
  - Name = hostname / Value = canonical name
  - Used for aliasing
- NS (Name Server)
  - Name=domain name / Value = hostname of Name server for that domain
- MX (Mail eXchange)
  - Name=domain name / Value = hostname of Mail server for that domain, preference=dd
- SOA (Start Of Authority)
  - Name = domain name / Value = several information of name servers and data it maintains
- PTR (domain name PoinTeR)
  - Name = IP address / Value = hostname
  - Used for "reverse" lookup

RR Examples (1)

- RR in "Root zone" name servers

<jp, ns1.nic.ad.jp, NS, IN>
< ns1.nic.ad.jp, 202.12.30.33, A, IN>

- RR in "JP zone" name servers

<ad.jp, ns0.nic.ad.jp, NS, IN>
<ns0.nic.ad.jp, 202.12.30.131, A, IN>
<ac.jp, ns0.nic.ad.jp, NS, IN>
<ns0.nic.ad.jp, 202.12.30.131, A, IN>

<keio.ac.jp, ns0.keio.ac.jp, NS, IN>
<ns0.keio.ac.jp, 133.27.4.121, A, IN>
<wide.ad.jp, ns.wide.ad.jp, NS, IN>
<ns.wide.ad.jp, 203.178.136.63, A, IN>

RR Examples (2)

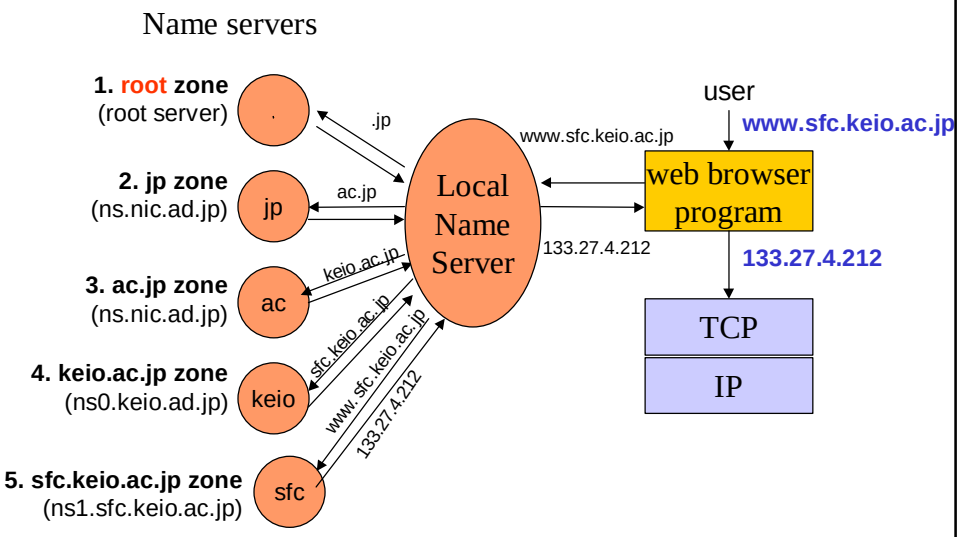
- RR in "keio.ac.jp zone" name server

<sfc.keio.ac.jp, ns1.sfc.keio.ac.jp, NS, IN>
<ns1.sfc.keio.ac.jp, 133.27.4.2, A, IN>
<cc.keio.ac.jp, kogwy.cc.keio.ac.jp, NS, IN>
<kogwy.cc.keio.ac.jp, 131.113.1.1, A, IN>

- RR in "sfc.keio.ac.jp" name server

<ccz02.sfc.keio.ac.jp, 133.27.4.212, A, IN>
<www.sfc.keio.ac.jp, ccz02.sfc.keio.ac.jp, CNAME, IN>
<sfc.keio.ac.jp, mail.sfc.keio.ac.jp, MX, IN>

Name resolution



bootstrap ?

- How the “web browser” program find a local server ?
- How the local server find the root server which it sends a query first ?

Local and Root

- Each host somehow should know the local name server(s) which answers all queries.
  - /etc/resolv.conf (in one unix implementation)
- Each local name server somehow should know the Root name servers to send query first.
  - bootstrap information called "Root cache"
  - currently manually distributed from InterNIC

Root name servers

- RR in Root cache

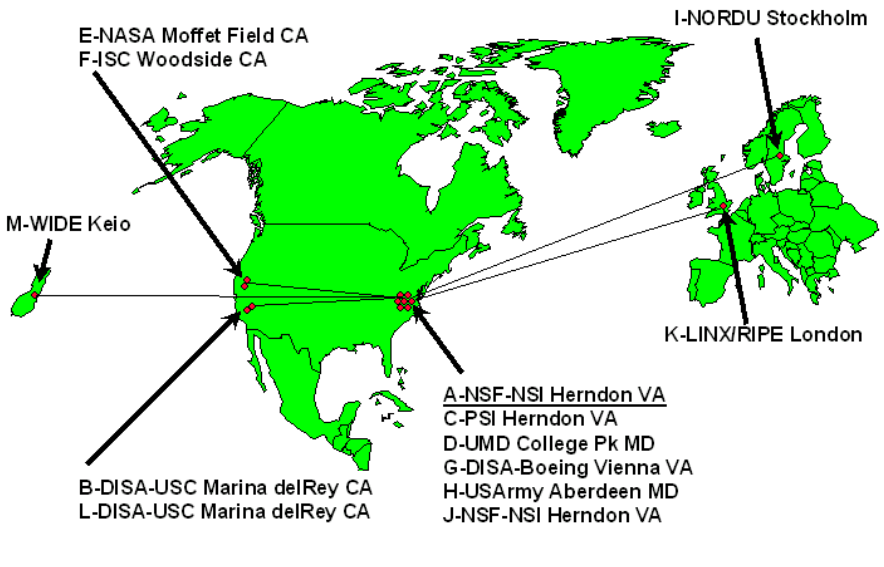
  <. , A.ROOT-SERVERS.NET, NS, IN, 3600000>
  <A.ROOT-SERVERS.NET, 198.41.0.4, A, IN, 3600000>
  <. , B.ROOT-SERVERS.NET, NS, IN, 3600000>
  <B.ROOT-SERVERS.NET, 128.9.0.107, A, IN, 3600000>

- Root name servers
  - Theoretically all queries come to the root server first
  - Connectivity from a local server to one of the root server is mandatory for name lookup
  - caching helps for better performance

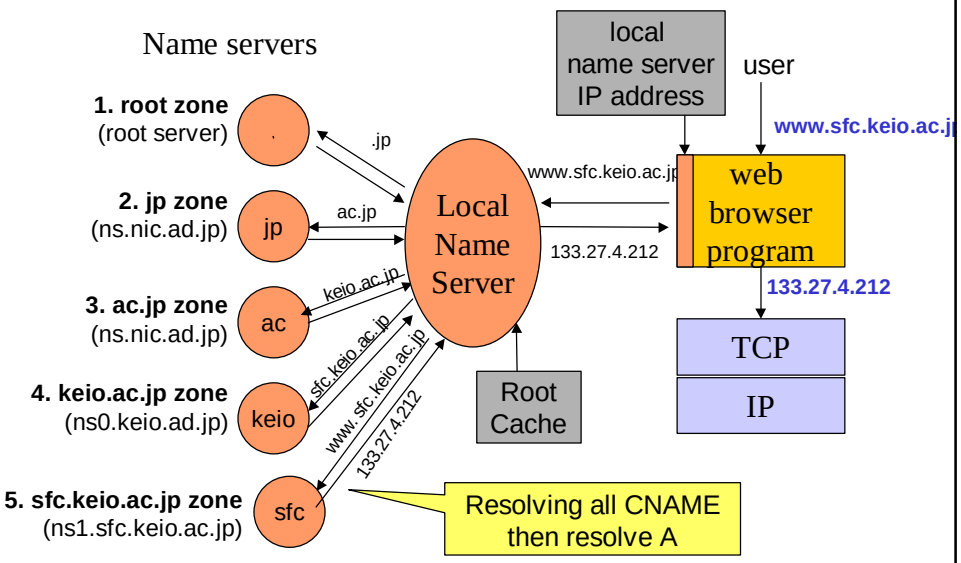
List of the root name servers

- 13 name servers on the Internet

name	org	city	type	url
a	InterNIC	Herndon,VA, US	com	http://www.internic.org
b	ISI	Marina del Rev.CA, U	edu	http://www.isi.edu/
c	PSInet	Herndon,VA, US	com	http://www.psi.net/
d	UMD	College Park,MD, US	edu	http://www.umd.edu/
e	NASA	Mt View, CA, US	usg	http://www.nasa.gov/
f	ISC	Palo Alto, CA, US	com	http://www.isc.org/
g	DISA	Vienna, VA, US	usg	http://nic.mil/
h	ARL	Aberdeen, MD, US	usg	http://www.arl.mil/
i	NORDUnet	Stockholm, SE	int	http://www.nordu.net/
j	(TBD)	(colo w/ A)	()	http://www.iana.org/
k	RIPE	London, UK	int	http://www.ripe.net/
l	(TBD)	(colo w/ B)	()	http://www.iana.org/
m	WIDE	Tokyo, JP	int	http://www.wide.ad.jp/



Name resolution with root cache



MX RR

- MX RR has "preference" in numbers
  - **<sfc.keio.ac.jp, mail.sfc.keio.ac.jp, MX, IN, 0>**
  - **<sfc.keio.ac.jp, mail2.sfc.keio.ac.jp, MX, IN, 10>**
  - **– lower digits indicate higher priority**
- Wild card in Name
  - **<*.keio.ac.jp, mail.sfc.keio.ac.jp, MX, IN, 10>**

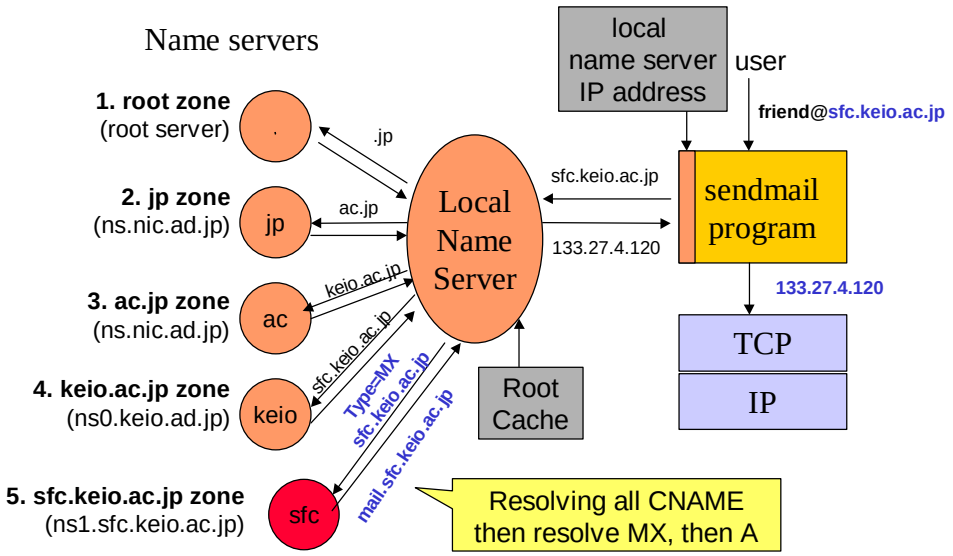
Mail example(1)

- Mail to *jun@wide.ad.jp*
  - Resolve all CNAME for *"wide.ad.jp"*
    answer: *wide.ad.jp*
  - Resolve MX for *"wide.ad.jp"*
    answer: *sh.wide.ad.jp, 10 (sorted by preferences)*
    additional information:
    - *sh.wide.ad.jp = 203.178.137.73*
  - smtp connection to 203.178.137.73

Mail example (2)

- Mail to *jun@xxx.wide.ad.jp*
  - Resolve all CNAME for *"xxx.wide.ad.jp"*
    answer: *xxx.wide.ad.jp*
  - Resolve MX for *"xxx.wide.ad.jp"*
    answer: *N/A*
  - Resolve A for *"xxx.wide.ad.jp"*
    answer: *203.178.139.33*
  - smtp connection to 203.178.139.33

Mail example (summary)

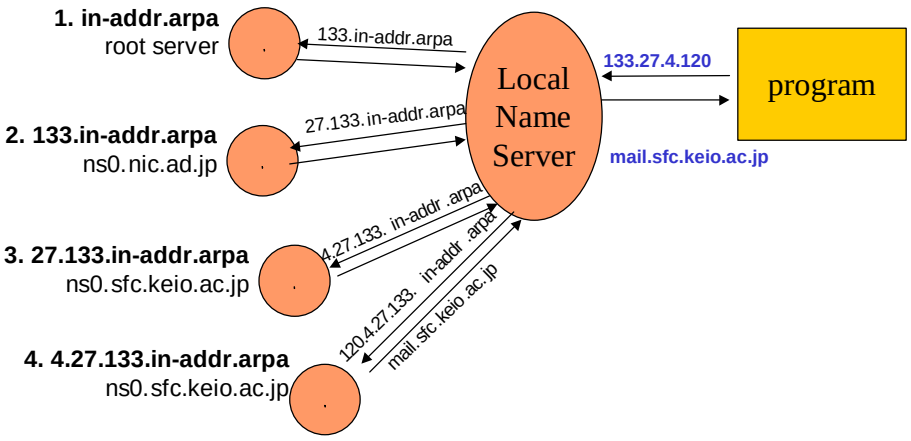


PTR RR

- used for reverse name mapping – *IP address* to a *host name*
- Special domain name "in-addr.arpa"
- Example
  < 7.140.178.203.in-addr.arpa, shonan.sfc.wide.ad.jp, PTR, IN>

- If a translation IP to host name fail, which may cause troubles when using some Internet services and accessing some public sites. (Some IRC server denies access from such hosts)

Reverse mapping using PTR



PTR and CIDR

- PTR is designed for Class A,B,C,D address space model.
  - 192.0.2.* (2.0.192.IN-ADDR.APRA) to organization X,
  - 192.0.3.* (3.0.192.IN-ADDR.APRA) to organization Y
  - 192.0.4.* (4.0.192.IN-ADDR.APRA) to organization Z
- How to handle address space with CIDR ?
  - 192.0.2.0/25 to organization X
  - 192.0.2.128/26 to organization Y
  - 192.0.2.192/26 to organization Z
- RFC2317 "Classless IN-ADDR.ARPA delegation "

Classless IN-ADDR.ARPA delegation

- Using CNAME on upper zone

  <0/25.2.0.192, ns.A.domain, NS,IN>

  < 1.2.0.192.in-addr.arpa, 1.0/25.2.0.192.in-addr.arpa, CNAME, IN>

  < 2.2.0.192.in-addr.arpa, 2.0/25.2.0.192.in-addr.arpa, CNAME, IN>

- At name server for A domain

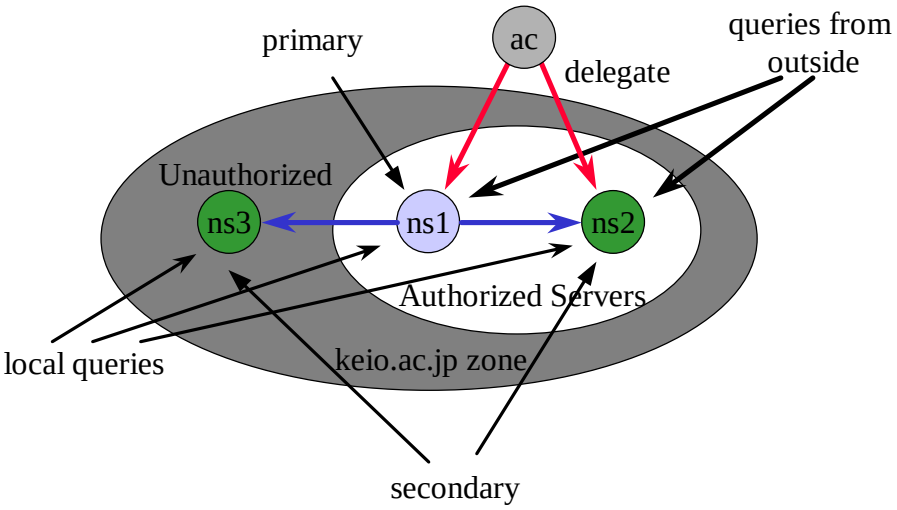
  <1.0/25.2.0.192, host1.ns.A.domain, A,IN>

  <2.0/25.2.0.192, host2.ns.A.domain, A,IN>

Redundancy and performance

- Each zone has one primary name server and several secondary name servers
  - primary name server maintains a master zone data
  - secondary name server maintains a copy of zone data
  - copies are synchronized periodical by zone transfer
  - Servers are better to be distributed topologically
- Servers can be authorized or unauthorized
  - authorized servers are registered in the upper zone
  - unauthorized servers are not in the upper zone, usually used internally only
- Servers keeps cache for a while for quicker

Primary and secondary servers



SOA RR example

```
sfc.wide.ad.jp   IN    SOA    shonan.sfc.wide.ad.jp. root.sfc.wide.ad.jp. (
                             1999112901     ; Serial
                             1800           ; Refresh
                             900            ; Retry
                             3600000        ; Expire
                             10800          ; TTL
                             )
```

- primary server forsfc.wide.ad.jp is shonan.sfc.wide.ad.jp
- administrator isroot@sfc.wide.ad.jp
- Secondary servers refresh its copy every.refresh. seconds (only if .serial. is updated)
- Secondary keep trying refresh every.retry. seconds if failed.
- zone becomes out of service if after.expire. seconds of no answer.
- cached RR of this zone is valid for.TTL. seconds in default

DNS implementations

- BIND (Berkeley Internet Name Domain)
  - reference implementation of the DNS (Domain Name System) for many vervions of Unix.
  - Started from UCB graduate school project, now developed and distributed by Internet Software Consortium (source distribution)
  - Widely used on the Internet
  - Contains
    - Server (named)
    - Resolver liberary
    - Operational tools (such as nslookup etc)
  - 8.2.2 patch level5 is the latest (11/12/1999)
  - http://www.isc.org/products/BIND/

DNS implementations based on BIND

- BIND 4.9.7 for NT
- BindNT, from Software.com.
- NetID 4.1 from Nortel Networks.
- Meta IP from Check Point Software Technologies.
- Shadow IP from Network TeleSystems.
- QIP Enterprise 5.0 from Lucent Technologies.
- JOIN DHCP/DDNS from JOIN Systems.
- F-Secure NameSurfer from Data Fellows.
- DNS Pro from FBL Inc.

DNS enhancements

- Dynamic update
  - RFC 2136 *(Proposed Standard),* "Dynamic Updates in the Domain Name System (DNS UPDATE) "
- Incremental Zone Transfer
  - RFC 1995 *(Proposed Standard),* "Incremental Zone Transfer in DNS "
- Security extensions
  - RFC 2535 *(Proposed Standard),* "DNS security extensions "
- IPv6 support
  - RFC 1886 *(Proposed Standard),* "DNS Extensions to support IP version6 "

Dynamic update

- DNS was originally designed
  - to support queries of statically configured database
  - assuming data update frequency is fairly low and all updates were made as external edits to a zone's Master File
- Requirements for automatic update (ie,DHCP)
- Adding UPDATE operation for the server to accept RR(s) update from other process.
- Only accept from statically specified hosts for security reason.
- Need Secure DNS update

Security extensions

- Use digital signatures for data integrity and authentication in the DNS
- RFC specifies
  - key distribution
  - data origin authentication
  - transaction and request authentication

IPv6 support

- New Type for IPv6 – AAAA
- New Domain - IP6.INT

```
% nslookup
➢set type=A
➢pc2.fujisawa.wide.ad.jp
---- omitt
Name:    pc2.fujisawa.wide.ad.jp
Addresses:  203.178.137.79, 203.178.141.12

➢set type=AAAA
➢pc2.fujisawa.wide.ad.jp
Non-authoritative answer:
pc2.fujisawa.wide.ad.jp IPv6 address = 3ffe:501:0:1000:2e0:18ff:fe98:7824
pc2.fujisawa.wide.ad.jp IPv6 address = 2001:200:0:1000:2e0:18ff:fe98:7824

Authoritative answers can be found from:
wide.ad.jp      nameserver= ns.wide.ad.jp
---- omitt
```

Incremental zone transfer

- A mechanism for use with NOTIFY which allows transferring only that part of the zone that changed
- Less traffic for zone transfer
- faster update

RFC2010

Operational Criteria for Root Name Servers

- NS software should be BIND
- Use UDP checksums
- Dedicated host
- Clock synchronization
- The advertised address should be that of the "best" interface
- NS host must be located in a secure space
- network security
- A name server must be able to answer 1,200 UDP transactions per second with less than 5 milliseconds of average latency
- respond to e-mail trouble reports within 24 hours
- The name server shall be configured so that outbound zone transfers are permitted only to destinations on the server's local networks
- DNS AXFR shall be used as zone transfer protocol
- Recursion shall be disabled
- Outage shall be reported
- Inverse name lookup

Internet Draft
Root name server operational requirement

- Host requirements
- Physical security requirements
- Network security requirements
- Protocol authentication requirements
- Communication requirements

Extracts from ICANN Bylaws

- To advise the Board about the operation of the root name servers of the domain name system.
- To advise the Board on the operational requirements of root name servers, including host hardware capacities, operating systems and name server software versions, network connectivity and physical environment.
- To examine and to advise on the security aspects of the root name server system.
- To review the number, location, and distribution of root name servers considering the total system performance, robustness, and reliability.

Semantics of TLDs
Which TLD should be added/deleted?
Who own that specific TLD?

ICANN/IANA

Who and Where are the (new) root servers?

- Update the database
- Share the database among the distributed root servers
- Make it available to everyone

IANA/Root Server Operators

Major updates after the last meeting

- Technical specs/procedure to operate and change the root servers/zones.
  - Document work done at IETF dnsop Working group
  - 'Root Name Server Operational Requirements'
    - draft-ietf-dnsop-root-opreq-00.txt
- Measurement and Analysis for extend/change the root name servers.
  - Work initiated by Evi Nemeth, Univ. of Cololado
  - Jointly working with CAIDA
- Y2K statement
  - Published on the ICANN WEB on July 15, 1999
- Formal 'procedure for the operational roles
  - **Cooperative Research & Development Agreement(CRADA)**
  - Further detailed being discussed

Important IETF efforts

- RFC2010
  - "Operational Criteria for Root Name Servers" by Bill Manning and Paul Vixie
- IETF DNSOP Working group
  - Since March 1999
  - Root Server Operation
  - co-chaired by Lars-Johan Liman and Ray Plzak

Important URLs

- ICANN RSSAC
  - http://www.icann.org/dnsroot-com.html
- RSSAC Y2K Statement
  - http://www.icann.org/committees/dns-root/y2k-statement.htm
- IETF DNSOP
  - http://www.ietf.org/html.charters/dnsop-charter.html
- CRADA
  - http://www.icann.org/committees/dns-root/crada.htm
- CAIDA
  - http://www.caida.org/