DNS Introduction

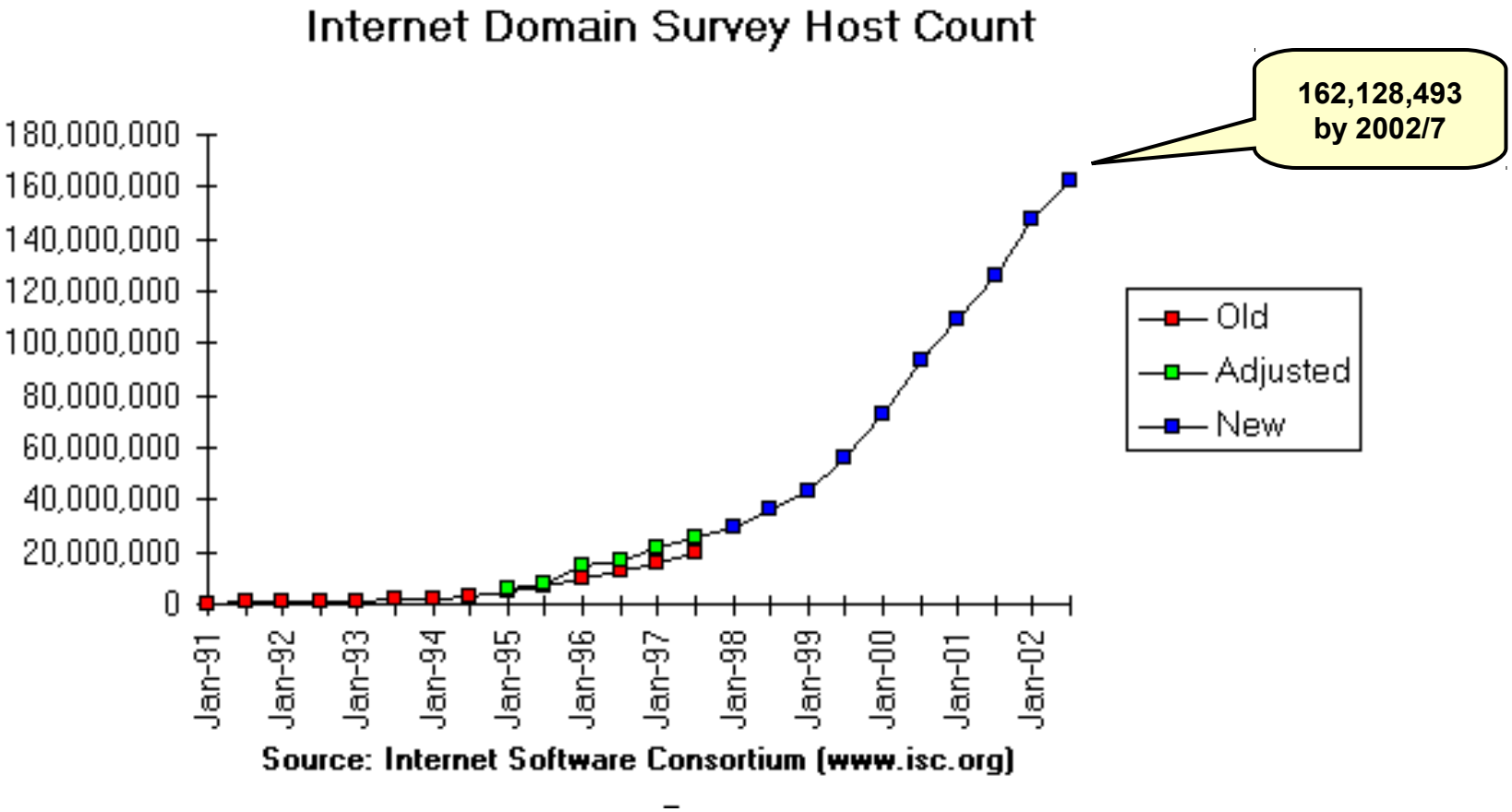
www.what-is-my-ip-address.com

"Except for Great Britain. According to ISO 3166 and Internet tradition, Great Britain's top-level domain name should be gb. Instead, most organizations in Great Britain and Northern Ireland (i.e., the United Kingdom) use the top-level domain name uk. They drive on the wrong side of the road, too."

DNS and BIND book

Footnote to the ISO 3166 two-letter country code TLDs

DNS Tree Growth



Top Host Names – Worldwide

Top Host Names July 2002

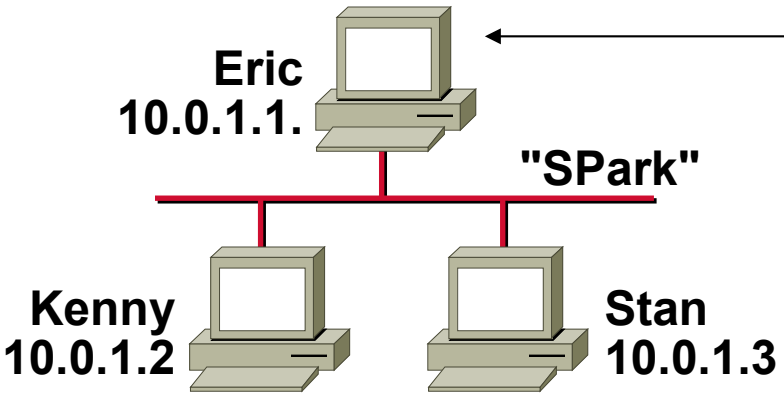
956841	www	3883	venus
336393	mail	3867	dev
56958	cpe	3795	zeus
36107	router	3765	jupiter
35004	ftp	3720	mars
33720	ns2	3656	l0
33128	gw	3647	t3
27548	ns1	3567	www3
23019	pc1	3511	
21775	pc2	loopback0	
16432	smtp	3470	pop
15265	pc3	3452	mercury
15177	pc4	3438	intranet
14979	broadcast	3404	demo
14891	pc5	3397	alpha
14877	gateway	3388	pc13
14138	server	3330	pluto
...big gap...		3308	exchange
3884	cisco	3253	linux

Top Host Names Jan 1992

384	venus	204	mac4	172	mac9
356	pluto	201	hobbes	172	mac11
323	mars	201	hermes	170	mac8
288	jupiter	198	thor	169	phoenix
286	saturn	198	sirius	169	mac12
285	pc1	196	gw	169	hal
282	zeus	195	calvin	168	snoopy
262	iris	194	mac5	168	mac13
260	mercury	191	mac10	167	mac15
259	mac1	190	fred	167	mac14
258	orion	189	titan	167	grumpy
254	mac2	189	pc3	163	gandalf
240	newton	186	opus	162	pc4
234	neptune	186	mac6	160	uranus
233	pc2	185	charon	159	mac16
224	gauss	185	apollo	158	sleepy
222	eagle	179	mac7	158	io
213	mac3	179	athena	157	earth
209	merlin	177	alpha	156	europa
207	cisco	172	mozart	155	rigel

History

- Even in the early Arpanet hosts have been identified by names
  - For People, not machines!
- Name/Address bindings in HOSTS.TXT files



/etc/hosts

127.0.0.1	eric	localhost
10.0.1.1	eric.spark	eric
10.0.1.2	kenny.spark	kenny
10.0.1.3	stan.spark	stan

(Kenny and Stan have similar hostfiles)

Hostfile Problems

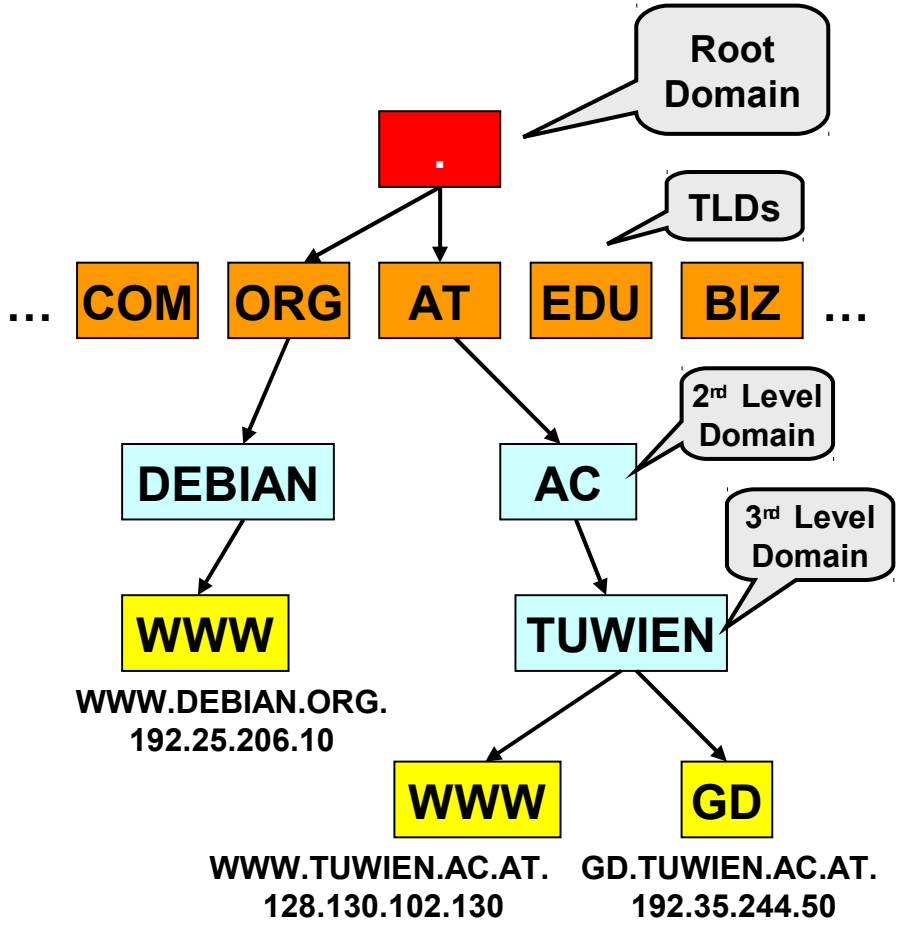
- Centrally maintained by Network Information Center (NIC)
- Copied by all hosts
- Scalability problem
- Consistency problem
- Maintenance problem

1984: DNS

- Paul Mockapetris (IAB) created DNS
- Distributed database
  - World-wide and redundant
  - Maintained by Name Servers
  - Simulates hierarchical tree of mnemonic names
  - Each domain name is a node in a database
  - Goal: Simple "Hostname resolution"
  - But also stores other information

Logical Tree of Names

- IP net-IDs are "flat"
  - Arbitrary assignment without semantical or logical considerations
  - Hard to remember
- DNS maps addresses to names
- DNS allows hierarchical tree of names
  - No name collisions anymore!
  - Max 127 levels
  - Concatenation results in Fully Qualified Domain Name (FQDN)

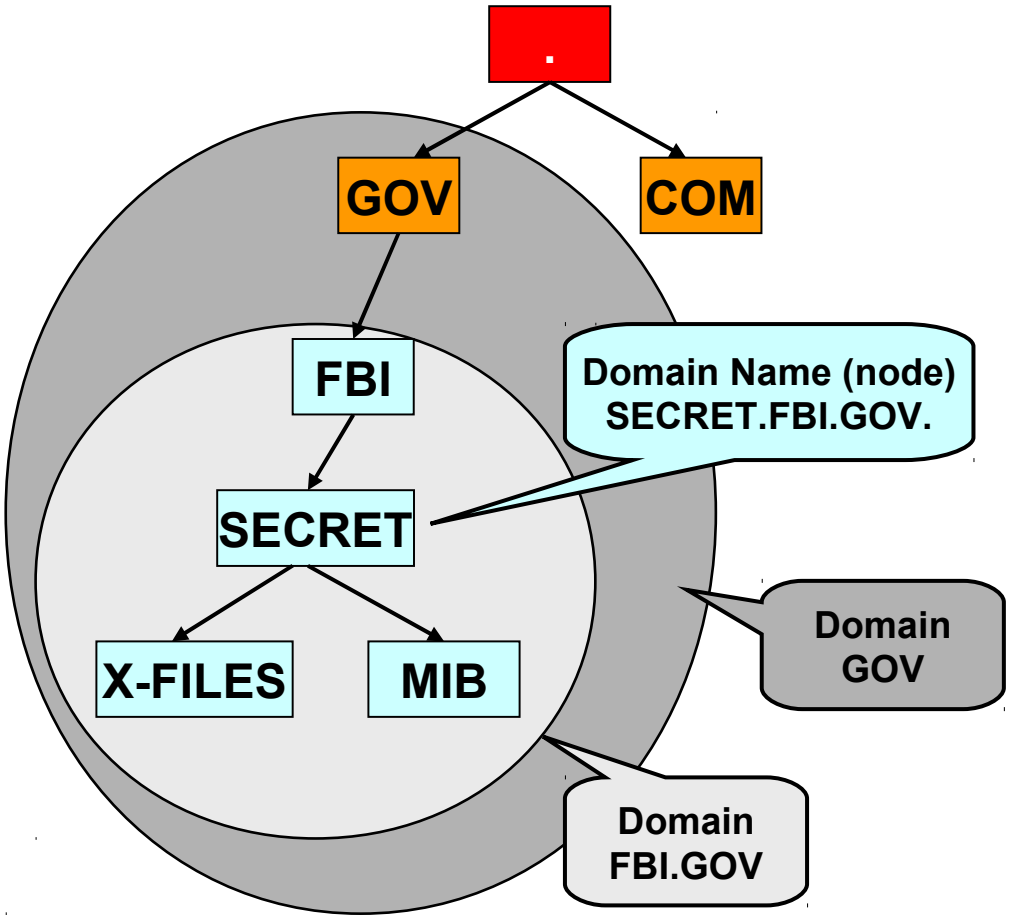


Name Servers

- The DNS tree is realized by Name Servers
- The Domain Name Tree does NOT reflect the physical network structure!
- Each NS cares for a subset of the DNS tree: zones
- Flexible mappings
  - 1:n (Routers or servers with several network interfaces)
  - n:1 (Multiple services behind a single IP address)

Terminology

- A "Domain" is a subtree of the domain name space
- A "Domain Name" is the name of a node in the tree
  - Concatenated labels from the root to the current domain
  - Listed from right to left
  - Separated by dots
  - Max 255 characters
- A "Label" is a component of the domain name
  - Max 63 characters



The Root Domain

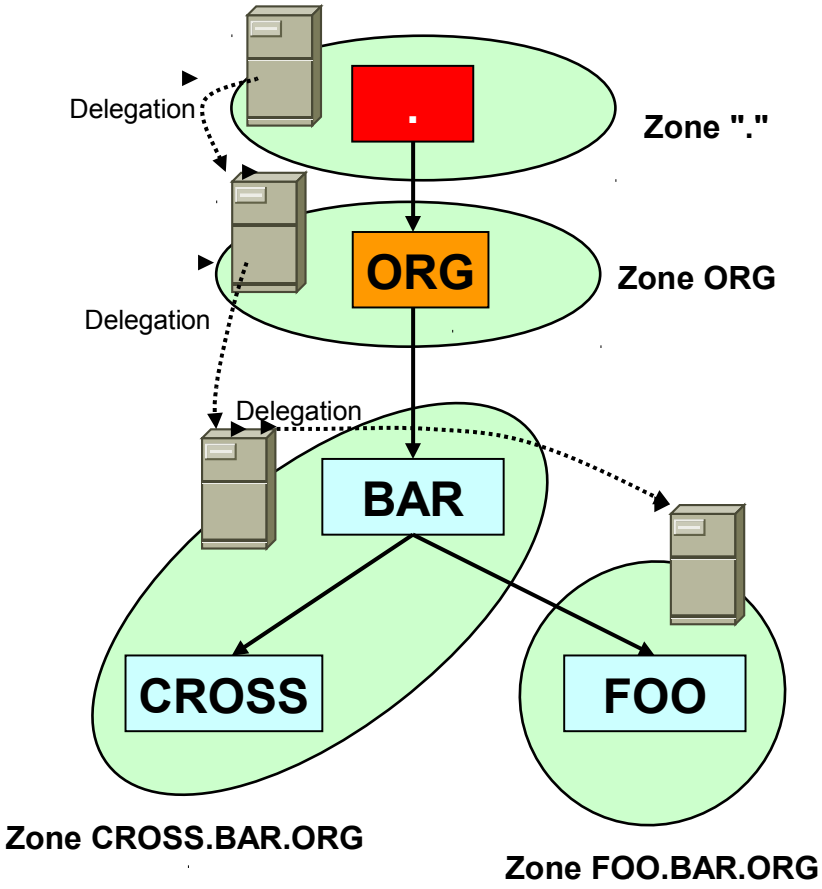
- The root of the DNS tree is represented as a dot "."
  - A true FQDN includes the dot
  - Otherwise "relative" domain name
  - Most people/applications don't care
  - However, DNS does care!
- The root is implemented by several root-servers (currently 13)
- Below the root, a domain may be called top-level, second-level, third-level etc...

Top Level Domains

- Seven "generic domains" (gTLDs)
  - COM, EDU, GOV, INT, ORG, MIL, NET
  - Initially inside USA, now globally used
- 244 Two-letter country codes
  - E.g. AT, DE, UK, ES, RU, CH, IT, AQ, …
  - Initially outside USA only, now also "US"
  - Country code does not necessarily reflect real location!
- Seven new TLDs
  - BIZ, INFO, NAME, MUSEUM, COOP, AERO, PRO

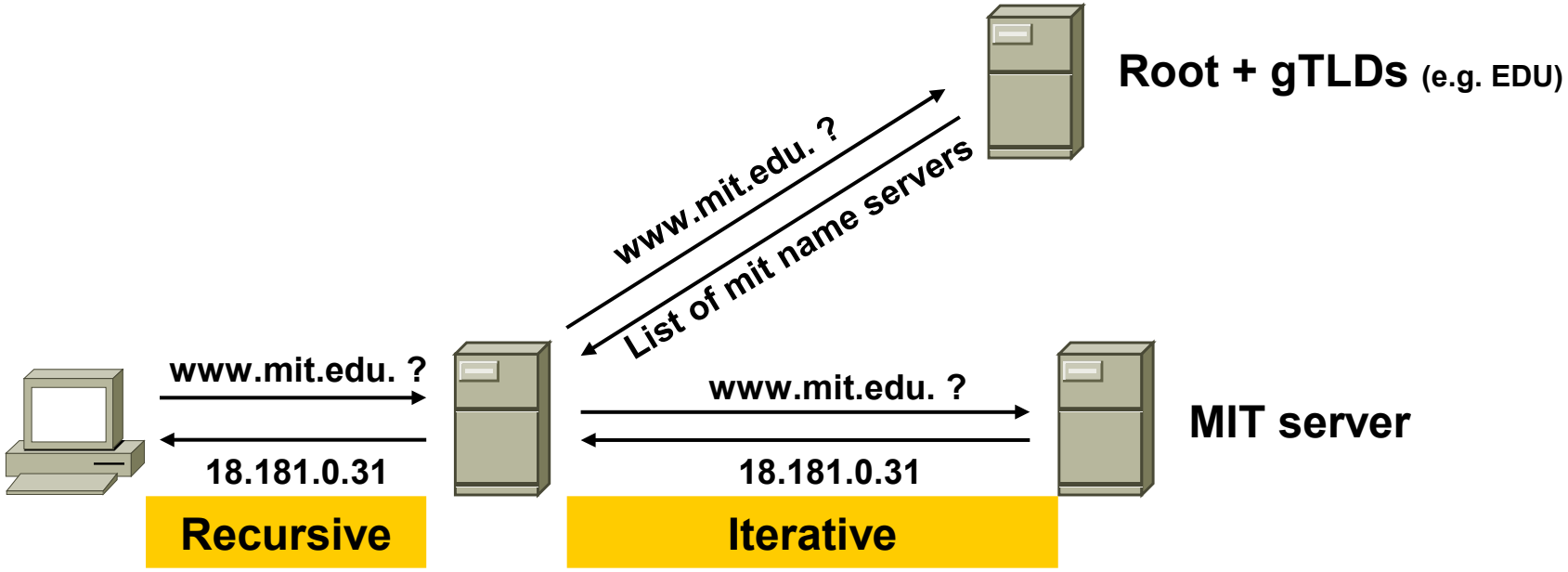
Delegation and Zones

- To ease administration, the authority over subdomains is delegated to other nameservers
- A zone is a point of delegation or "Start of Authority" (SOA)
- Zones relate to the way the database is partitioned and distributed

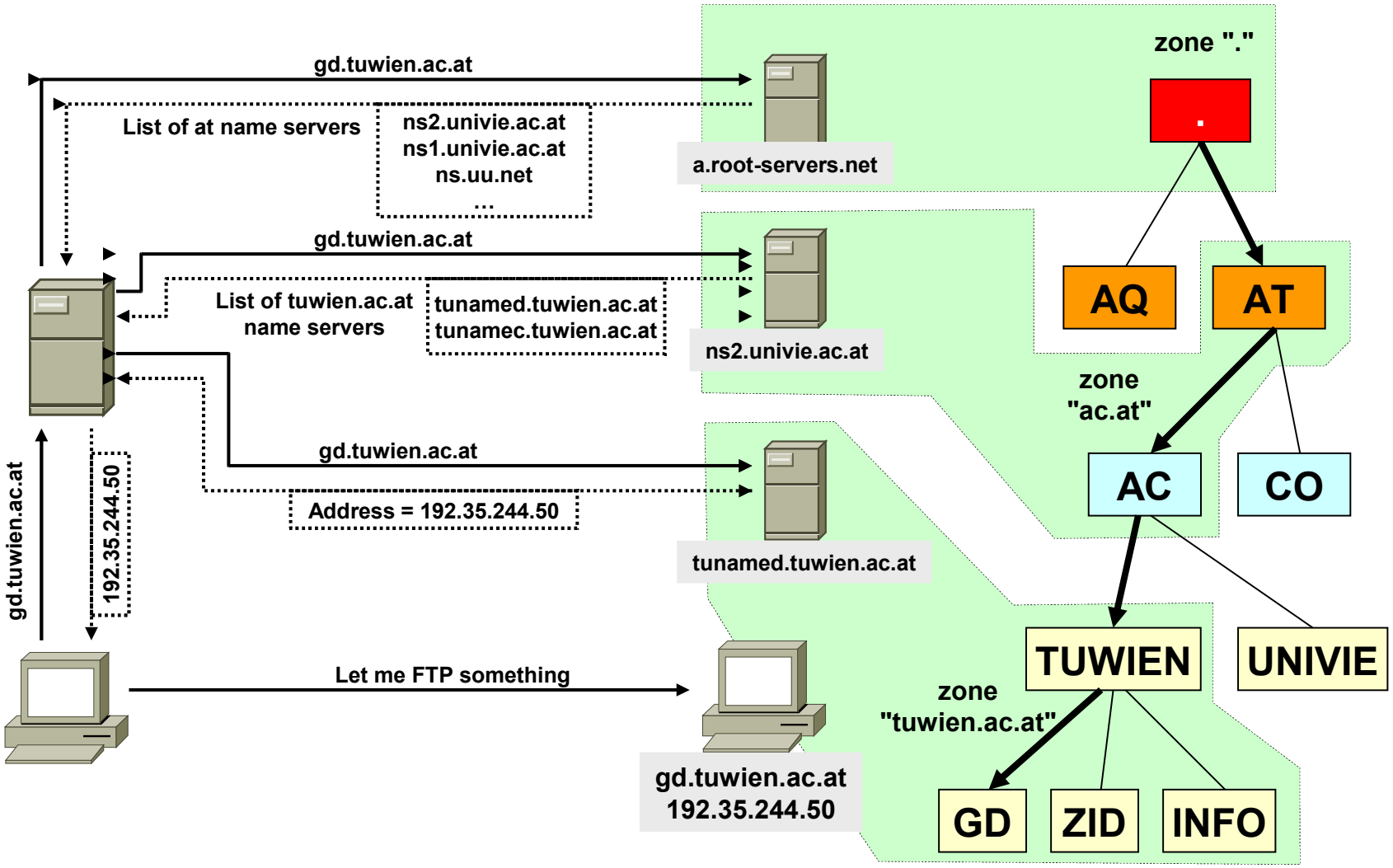


Hostname Resolution

- **Recursive** queries = the job is forwarded
  - The response must be exact (or error message)
  - Most burden on next name server
- **Iterative** queries = All NS are queried top-down
  - The response contains best answer already known
  - Requested name server makes no further queries

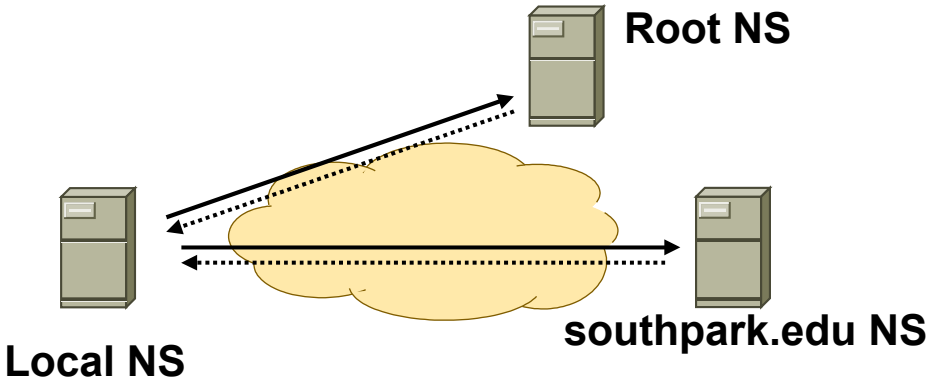


A Detailed Real-World Example

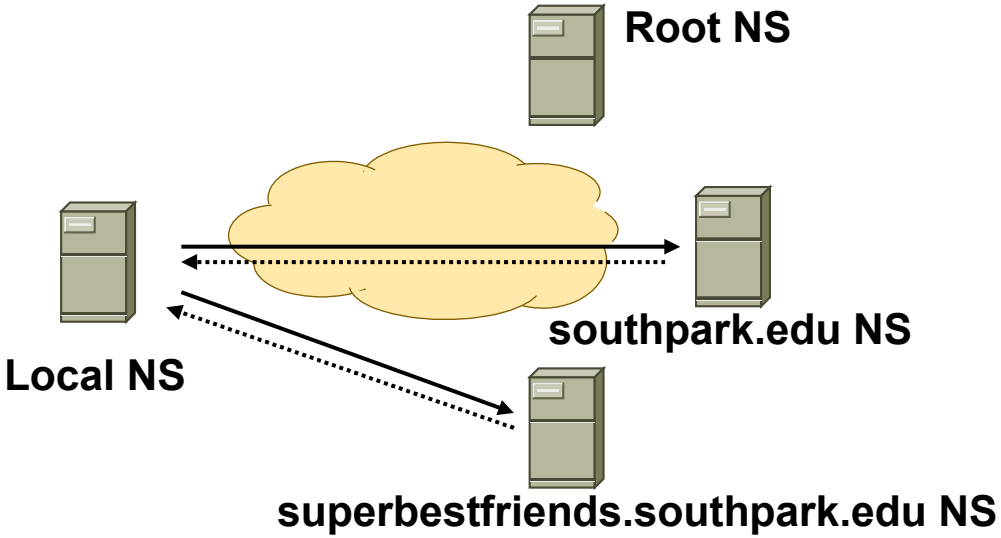


- Each questioned name server replies with more detailed information…or the desired information itself!
- A reference to another NS gives precious information about new zone authority – cached!

Caching



- First, the local NS resolves the name kenny.southpark.edu
- Hereby it learns also the addresses of the southpark.edu NS
- All this information is cached!



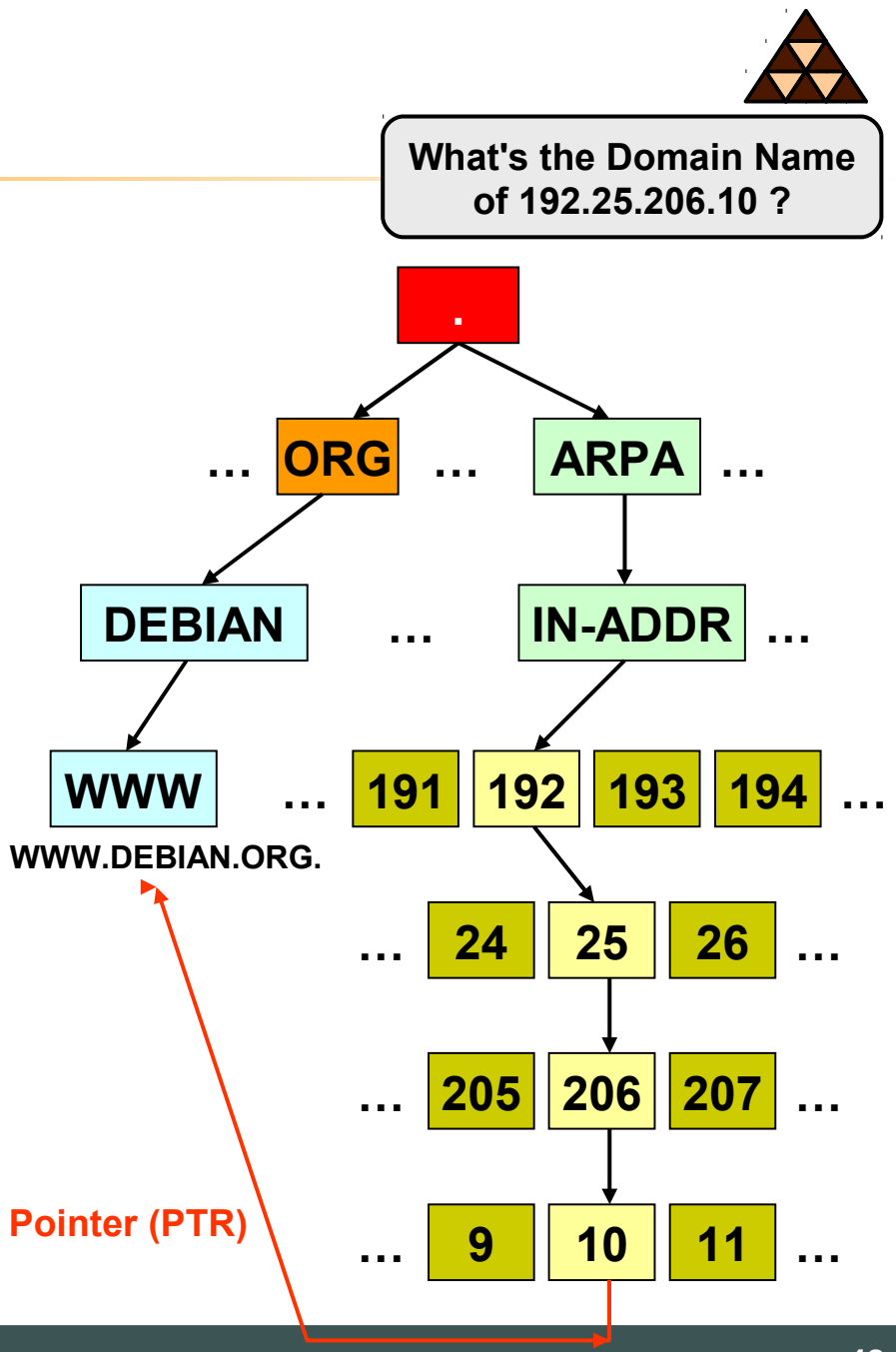
- When resolving the name seamen.superbestfriends.southpark.edu the local NS notices that this name is member of southpark.edu
- Address of southpark.edu NS is cached
- No need to start at root NS!

Reverse Lookups

- Very often reverse lookups are necessary
  - "Have address but want name"
  - For logging purposes or service restriction
- Therefore the in-addr.arpa domain was created
  - Given an IP-address the associated hostname can be found
  - Otherwise an exhaustive search in the domain space would be necessary to find any desired hostname

In-Addr.Arpa

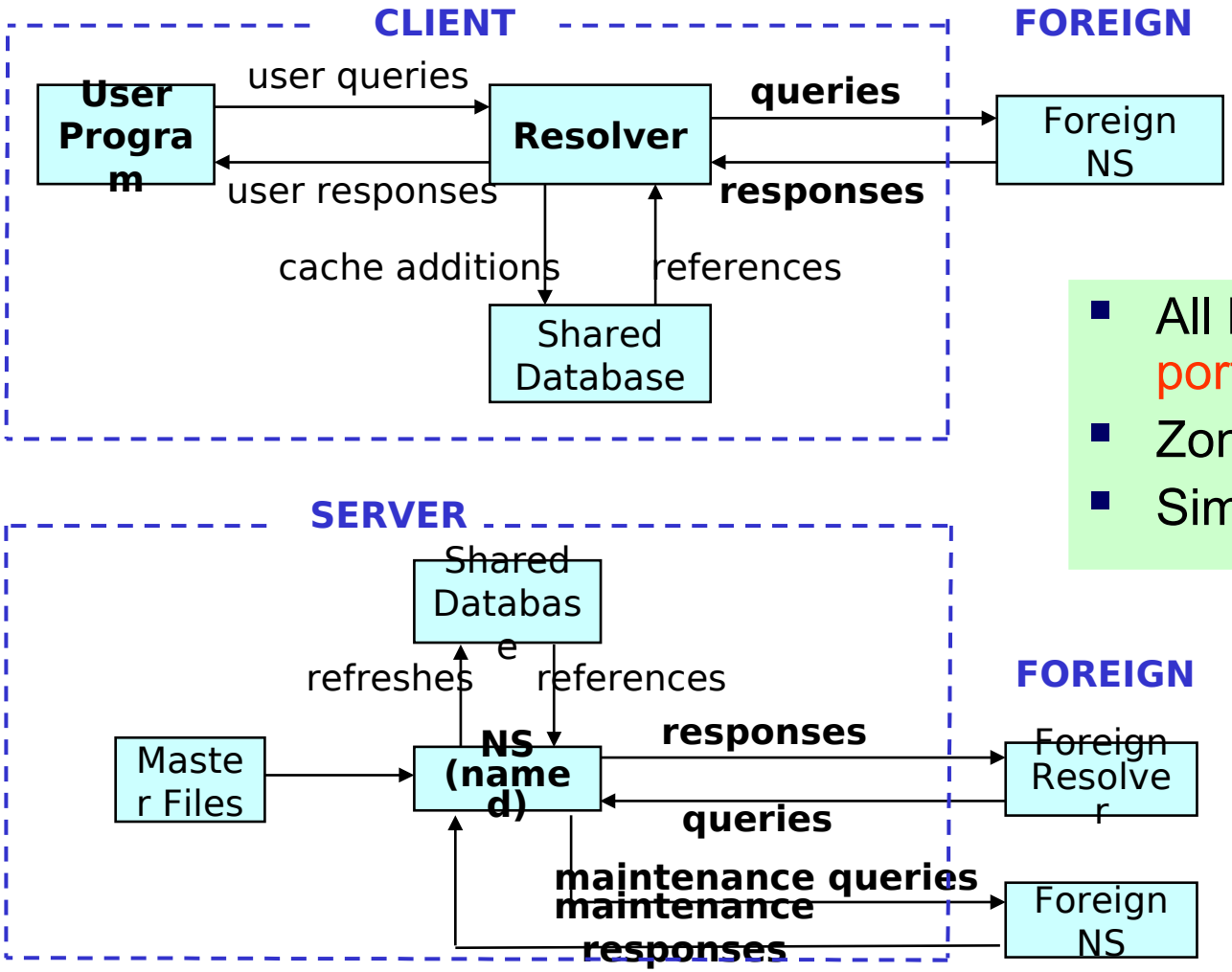
- Each byte of an IP address is treated as label and attached under the in-addr.arpa TLD
  - Expressed as character string for its decimal value ("0" - "255")
- Labels are concatenated in reverse order
  - "10.206.25.192.in-addr.arpa"



BIND

- Berkeley Internet Name Domain (BIND)
  - Implemented by Paul Vixie as an Internet name server for BSD-derived systems
  - Most widely used name server on the Internet
  - Version numbers: 4 (old but still used), 8, 9
- BIND consists of
  - A name server program "named"
  - A resolver library for client applications
- BIND deals with zones!

Resolver and Name Server



- All DNS messages use port 53
- Zone transfers use TCP
- Simple queries use UDP

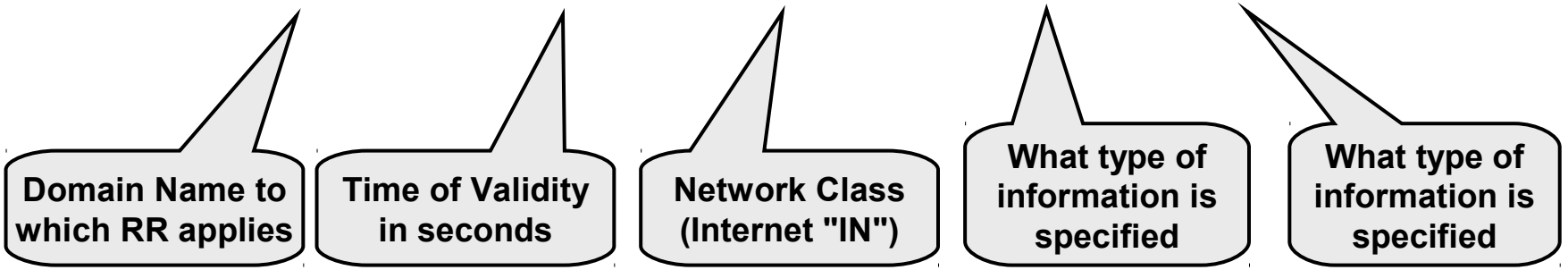
Types of Name Servers

- Primary Masters (or "Master")
  - Has data about a zone in a local file
  - Therefore is authoritative about a zone
  - Each zone has exactly one Primary
- Secondary Masters (or "Slave")
  - Copies zonefiles from a Master Server (P or S)
  - This is called "zone transfer" (TCP)
  - Therefore also authoritative
  - Each zone must have at least one Secondary

Resource Records

- All database information is stored in resource records (RR)
- Different classes: IN, HS, CH
  - Only IN (Internet) is important today
- RR Format:

```
[DOMAIN] [TTL] [CLASS] TYPE RDATA
```



Some Important RR Types

Type	Value	Meaning
A	1	Host address
NS	2	Authoritative name server
CNAME	5	Canonical name for an alias
SOA	6	Marks the start of a zone of authority
WKS	11	Well known service description
PTR	12	Domain name pointer
HINFO	13	Host information
MINFO	14	Mailbox or mail list information
MX	15	Mail exchange
TX	16	Text strings

Root Servers

- 13 root servers implement the "."
  - Maintained by ICANN
  - Each of them knows all TLD name servers
  - Most are even authoritative for the generic top-level domains
- Name Servers must maintain a list of root servers
  - Stored in "root.hints" file (BIND)
  - Queried one after the other until positive reply
  - This list is also updated by requesting single root servers

Root Hints Example

```
.       604800  IN      NS      G.ROOT-SERVERS.NET.
.       604800  IN      NS      K.ROOT-SERVERS.NET.
.       604800  IN      NS      H.ROOT-SERVERS.NET.
.       604800  IN      NS      A.ROOT-SERVERS.NET.
.       604800  IN      NS      B.ROOT-SERVERS.NET.
```

root

Internet

Name servers

```
G.ROOT.SERVERS.NET.     604800  IN  A  192.112.36.4
K.ROOT.SERVERS.NET.     604800  IN  A  193.0.14.129
H.ROOT.SERVERS.NET.     604800  IN  A  128.63.2.53
A.ROOT.SERVERS.NET.     604800  IN  A  198.41.0.4
B.ROOT.SERVERS.NET.     604800  IN  A  128.9.0.107
```

TTL [s]

Addres
s

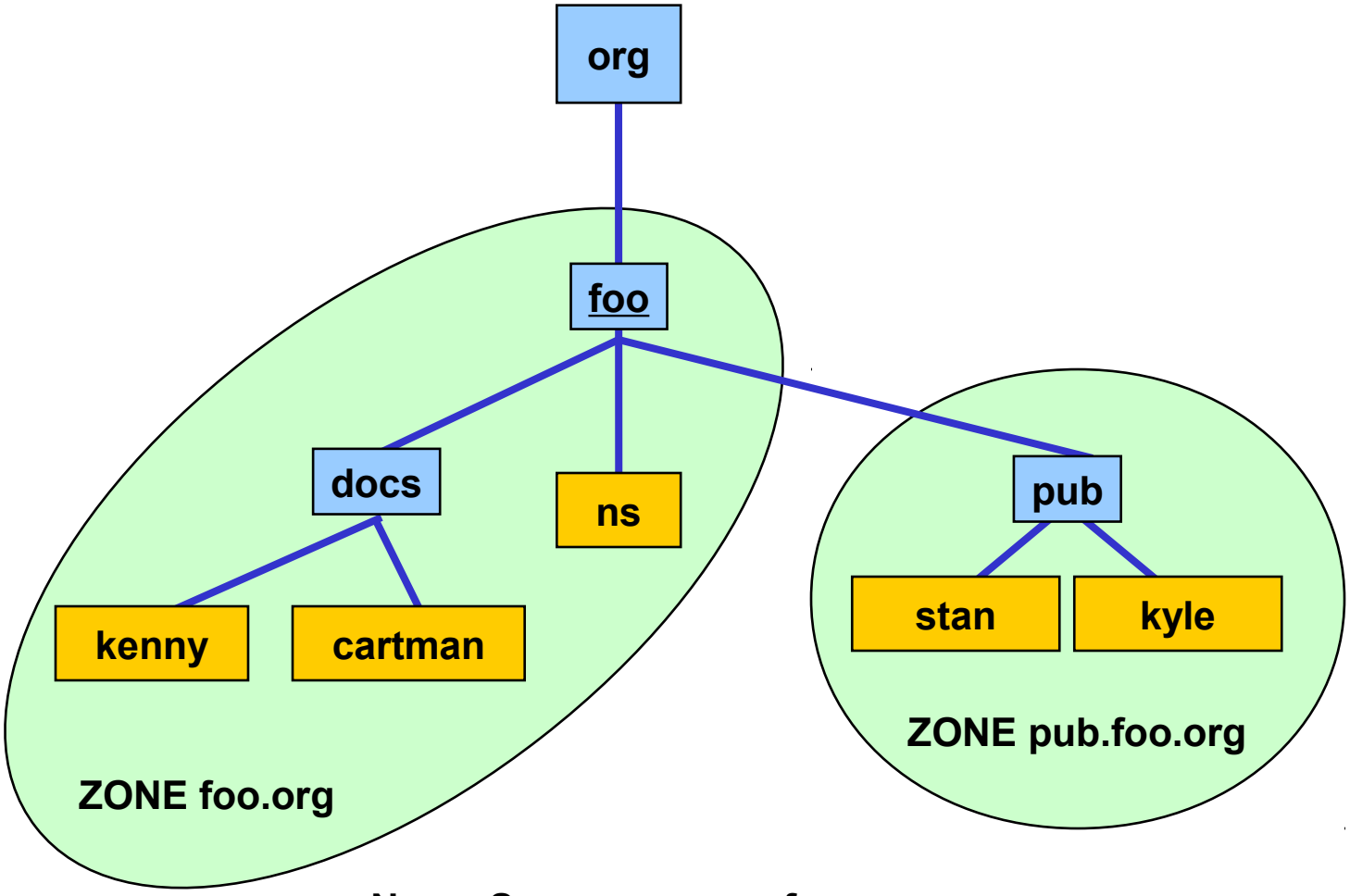
Behind the Scenes

- Frequently private root servers are used within organizations
  - Isolated from official DNS
- Recently several unofficial "roots" were available in the Internet
  - Overlaps official DNS and introduces new unofficial TLDs
- Now ICANN is responsible for managing and coordinating the DNS to ensure universal resolvability
  - ICANN: Global, NPO, public interest
  - Cares for distribution of unique IP addresses and domain names

Caching

- Caching is critical for DNS performance
  - Offload root NS (only 13 root servers!)
  - Offload other authoritative NS
- Cached information
  - Is non-authoritative
  - Is valid as specified in TTL

Example Config (1)



Name Servers: ns.foo.org
stan.pub.foo.org

Example Config (2)

```
; zone file for the foo.org. zone
@           IN    SOA  ns.foo.org.        admin.kenny.docs.foo.org (
                       199912245          ;serial number
                       360000             ;refresh time
                       3600               ;retry time
                       3600000            ;expire time
                       3600               ;default TTL )
            IN    NS   ns.foo.org.
            IN    NS   ns.xyz.com.        ;secondary nameserver for @
            IN    MX   mail.foo.org.      ;mailserver for @
Pub         IN    NS   stan.pub.foo.org.
; glue records
ns          IN    A    216.32.78.1
stan.pub    IN    A    216.32.78.99
; hosts in the zone foo.org
Mail        IN    A    216.32.78.10
Linus       IN    A    216.32.78.20
kenny.docs  IN    A    216.32.78.100
cartman.docs IN   A    216.32.78.150
```

Records describing zone .foo.org. = @

Delegation for the zone pub.foo.org.

Timers in the SOA RR

- Refresh time
  - Tells slave at which time intervals it should check for zone changes
  - Some hours (3-12 typically)
- Retry time
  - If master could not be reached
  - Typically shorter than refresh time
- Expire time
  - Time after which unrefreshed zone data is definitely outdated (removed)
  - Typically one week (also months)
- TTL
  - BIND pre 8.2: Specifies how long any cached entry is valid
  - BIND 8.2 and later: Only valid for negative caching!
  - Performance versus consistency!

Example Config (3)

```
; zone file for the 78.32.216.in-addr.arpa domain
@       IN SOA  ns.foo.org    admin.kenny.docs.foo.org.
                              (
                              1034
                              3600
                              600
                              3600000
                              86400
                              )
        IN NS       ns.foo.org.
1       IN PTR      ns.foo.org.
10      IN PTR      mail.foo.org.
20      IN PTR      linus.foo.org.
99      IN  PTR     stan.pub.foo.org.
100     IN PTR      kenny.docs.foo.org.
150     IN PTR      cartman.docs.foo.org.
```

Example Config (4)

```
; zone file for pub.foo.org
@           IN  SOA  stan.pub.foo.org   hostmaster.stan.pub.foo.org.
                             ( 1034
                             3600
                             600
                             3600000
                             86400 )
; Name Servers
            IN  NS           stan
            IN  NS           ns.foo.org.   ; secondary NS
; glue records
   stan     IN  A            216.32.78.99
```

```
nameserver  IN    CNAME              stan
; other hosts:
kyle        IN    A                  216.32.22.50
            IN    MX                 1  mail.foo.com
            IN    MX                 2  picasso.art.net
            IN    MX                 5  mail.ct.oberon.tuwien.ac.at
butters     IN    A                  216.32.22.51
garison     IN    A                  216.32.22.52
            IN    HINFO              VAX-11/780  UNIX
            IN    WKS                216.32.22.52  TCP
                                  (telnet ftp netstat finger pop)
wendy        IN   A                  216.32.34.2
            IN    HINFO              SUN  UNIX
; etc.....
```

Delegations

- Delegations are made when a zone has a parent domain
- A parent name server acting as delegation point keeps a Name Server record (NS) that specifies responsible name servers for that subzone
- A-records that correspond with associated NS records are called glue records
- Glue records are only necessary if the specified nameserver (NS record) is inside the subzone it serves!
  - AND the parent is no secondary server for that zone

Registration Terms

- Registry
  - Responsible of TLD zone maintenance
  - One unique registry per TLD
- Registrar
  - Intermediate agent between customer and registry (ISP)
- Registration
  - Customer tells registrar which NS should be used for delegation to reach a subdomain
  - Plus contact information

Domain Registrations

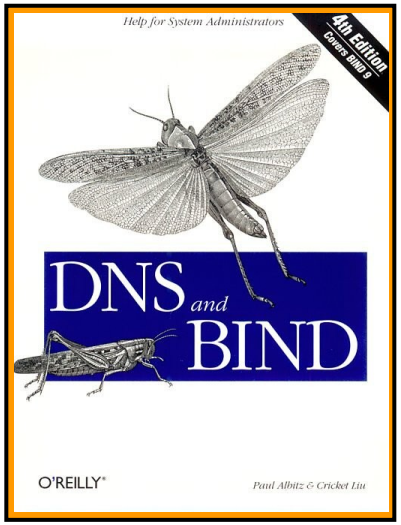
- Many providers act as "registrars"
- ICANN controls continental registrars
  - USA: InterNIC (www.internic.net)
  - Europe: RIPE (www.ripe.net)
  - Asia: APNIC (www.apnic.net)

ICANN

Diagnostic Tools

- DIG - Domain Information Groper
  - Send domain name query packets to name servers
  - Results are printed in a human-readable format
- NSLOOKUP
  - Query Internet name servers interactively

Recommended Resources



- DNS and BIND (4th Edition)
  - by Paul Albitz, Cricket Liu
  - The "Bible"
- The Internet Software Consortium
  - http://www.isc.org/
  - Where BIND comes from
- The Linux Documentation Project
  - http://www.tldp.org/
  - HOWTOs, FAQs, BOOKS, ...free!

INTERNET SOFTWARE CONSORTIUM

The Linux Documentation Project

Selected RFCs (1)

- RFC 1034
  - Domain Name Concept And Facilities
- RFC 1035
  - Domain Name Implementation and Specification
- RFC 1101
  - DNS Encoding Network Names And Other Types
- RFC 1183
  - New DNS RR Definitions

Selected RFCs (2)

- RFC 1591
  - Domain Name System Structure And Delegation
- RFC 1664
  - Using The Internet DNS To Distribute RFC1327 Mail Address Mapping Tables
- RFC 1712
  - DNS Encoding Of Geographical Location
- RFC 1788
  - ICMP Domain Name Messages
- RFC 1794
  - DNS Support For Load Balancing

Selected RFCs (3)

- RFC 1876
  - A Means For Expressing Location Information In The Domain Name System
- RFC 1886
  - DNS Extensions To Support IP Version 6
- RFC 1918
  - Address Allocation for Private Internets
- RFC 1982
  - Serial Number Arithmetic
- RFC 1995
  - Incremental Zone Transfers In DNS
- RFC 1996
  - A Mechanism For Prompt Notification Of Zone Changes (DNS Notify)
- RFC 2052
  - A DNS RR For Specifying The Location Of Services (DNS SRV)
- RFC 2065
  - Domain Name System Security Extensions
- RFC 2136
  - Dynamic Updates In The Domain Name System (DNS Update)

Selected RFCs (4)

- RFC 2308
  - Negative Caching Of DNS Queries (DNS Ncache)
- RFC 2535
  - Domain Name System Security Extensions
- RFC 2541
  - DNS Security Operational Considerations
- RFC 2606
  - Reserved Top Level DNS Names

Selected RFCs (5)

- RFC 2672
  - Non-Terminal DNS Name Redirection
- RFC 2673
  - Binary Labels In The Domain Name System
- RFC 2845
  - Secret Key Transaction Authentication For DNS (TSIG)
- RFC 2870
  - Root Name Server Operational Requirements
- RFC 2874
  - DNS Extensions To Support IPv6 Address Aggregation And Renumbering
- RFC 3007
  - Secure Domain Name System Dynamic Update

Selected RFCs (6)

- RFC 3090
  - DNS Security Extension Clarification On Zone Status
- RFC 3152
  - Delegation Of IP6.ARPA
- RFC 3172
  - Management Guidelines & Operational Requirements For the Address And Routing Parameter Area Domain (ARPA)
- RFC 3363
  - Representing Internet Protocol Version 6 Addresses In The Domain Name System
- RFC 3364
  - Tradeoffs In Domain Name System Support For Internet Protocol Version 6

Summary

- DNS initially only created for humans
- Hierarchical tree of names
- Addresses and other database information
- Inverse resolution using in-addr.arpa TLD
- Primary vs Secondary nameservers
- Port 53, TCP and UDP

Any Questions?