Naming

G54ACC

Lecture 13

richard.mortier@nottingham.ac.uk

Contents

- Naming
- DNS outline
- DNS protocol
- DNS details
- Issues

Contents

- Naming
  - HOSTS.TXT
  - DNS
- DNS outline
- DNS protocol
- DNS details
- Issues

Naming

- IP addresses are all very well, but...
- Not particularly human-readable (esp. IPv6)
    - `fe80:0000:0000:0000:0202:b3ff:fe1e:8329`
    - `fe80:0:0:0:202:b3ff:fe1e:8329`
    - `fe80::202:b3ff:fe1e:8329`
- Not always the appropriate granularity
    - The address names an interface
    - We might want to name a server, a service, a site
    - We might have dynamic address allocation

HOSTS.TXT

- A file mapping names to numbers
  - Distributed from NIC using FTP – `/etc/hosts`
  - Simple, but neither automatic or scalable
  - See also `/etc/services`, mapping ports to names
- Led to the **Domain Name Service, DNS**
  - Initially RFC882 & RFC883, 1983
  - Later RFC1034 & RFC1035, 1987
  - ...and many many more

The DNS

- A consistent namespace
  - No reference to addresses, routes, etc.
- Key characteristics
  - Hierarchical, distributed, cached
    - For scale [but does this still apply?]
  - Federated – sources control trade-offs
  - Flexible – many record types
  - Simple client-server name resolution protocol

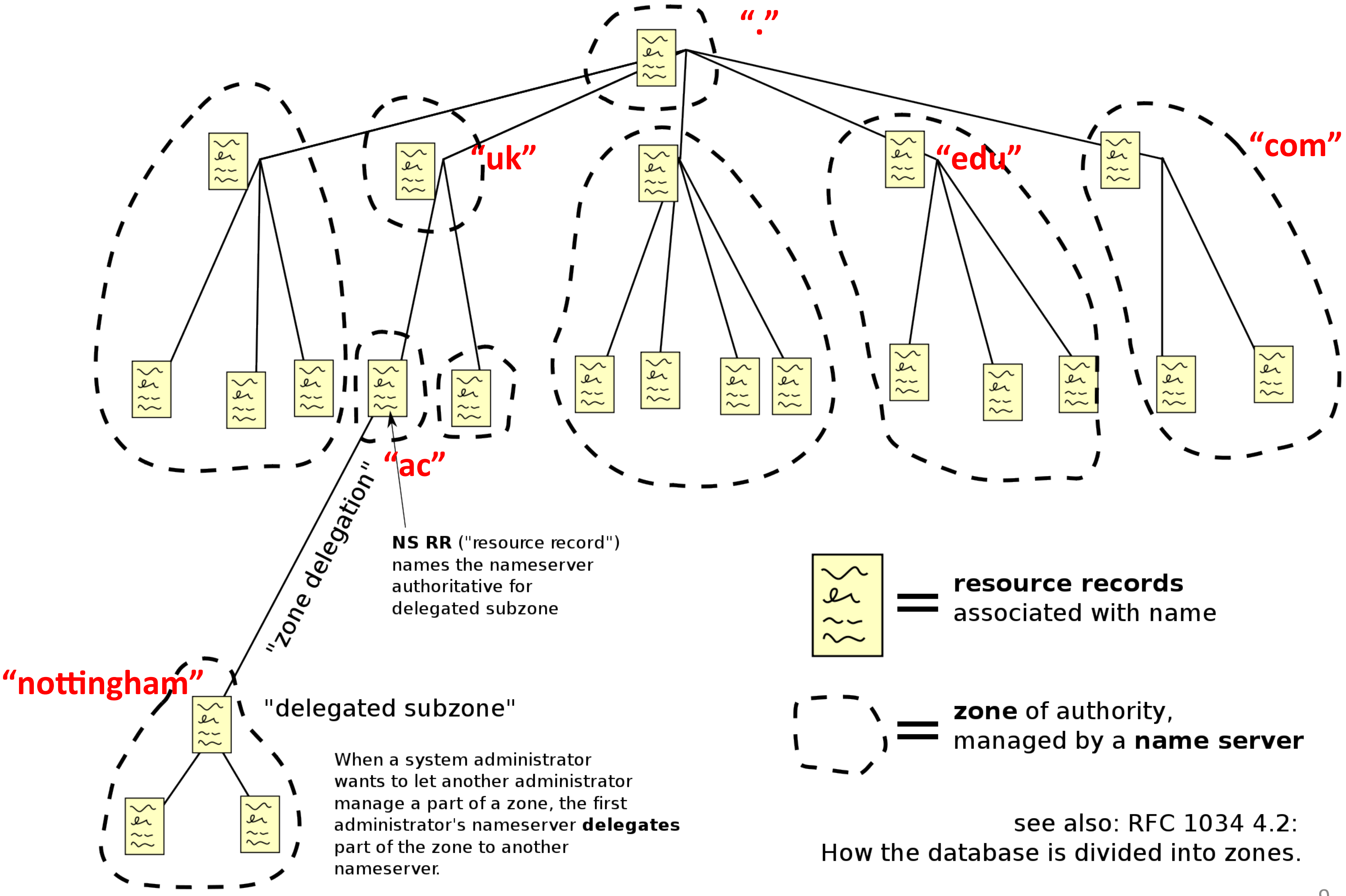
Contents

- Naming
- DNS outline
  - Components
  - Hierarchy
- DNS protocol
- DNS details
- Issues

Components

- *Domain Name Space* and *Resource Records*
  - Tree structured name space
  - Data associated with names
- *Name Server*
  - Contains records for a subtree
  - May cache information about any part of the tree
- *Resolver*
  - Extract information from tree upon client requests
  - `gethostbyname()`

http://en.wikipedia.org/wiki/Domain_Name_System



NS RR ("resource record") names the nameserver authoritative for delegated subzone

"delegated subzone"

When a system administrator wants to let another administrator manage a part of a zone, the first administrator's nameserver **delegates** part of the zone to another nameserver.

= **resource records** associated with name

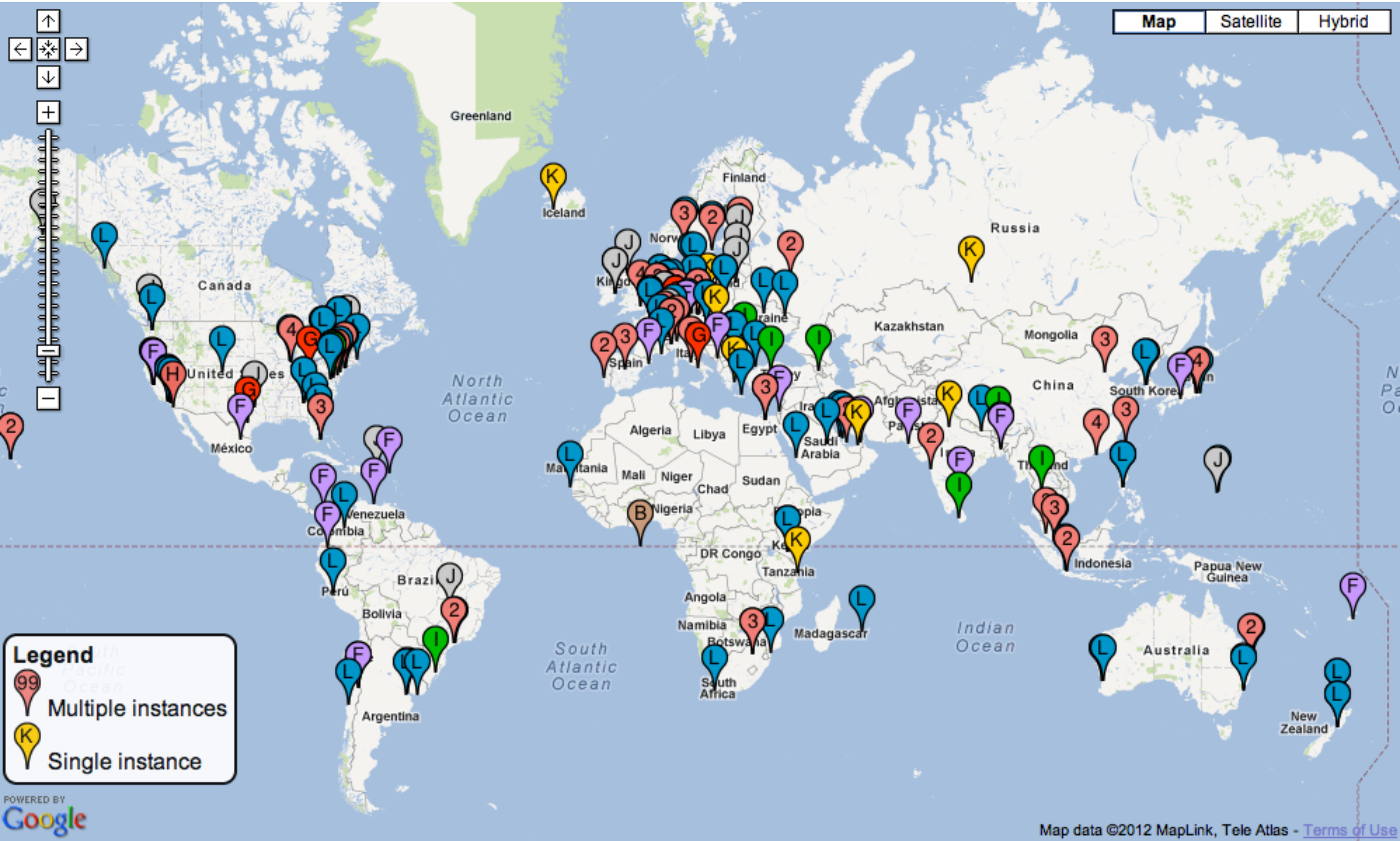
= **zone** of authority, managed by a **name server**

see also: RFC 1034 4.2:
How the database is divided into zones.

DNS Hierarchy

- Root
  - Ultimate authority with the US Dept. of Commerce NTIA
  - Managed by IANA, operated by ICANN, maintained by Verisign
  - Thirteen root server clusters
    - `a.root-servers.net` .. `m.root-servers.net`

Map of Root Servers



DNS Hierarchy

- Root
  - Ultimate authority with the US Dept. of Commerce NTIA
  - Managed by IANA, operated by ICANN, maintained by Verisign
  - Thirteen root server clusters
    - `a.root-servers.net` .. `m.root-servers.net`
- What's the obvious problem with this?

DNS Hierarchy

- Root, http://root-servers.org/
- Top Level Domains, TLDs
  - Operated by registrars, delegated from ICANN
- Delegate zones to other registrars
  - ...and on down the hierarchy
- Eventually customer rents a name – their **zone**
  - Registrar installs appropriate *resource records*
  - Associated with names within the zone

Contents

- Naming
- DNS outline
- DNS protocol
  - Queries
  - Responses
  - Resource Records
- DNS details
- Issues

Query

- Query generated by resolver
  - E.g., a call to `gethostbyname()`, `gethostbyaddr()`
- Carried in single UDP/53 packet
  - Or more rarely, TCP/53, in case of truncation
- Header followed by Question
  - Id, Q/R, opcode, AA/TC/RD/RA, response code, counts
  - Query Type, Query Class, Query Name

Response

- Response consists of three *RRsets* following the header and question
  - **Answers**:
    RRs that the server had for the QNAME
  - **Authoritatives**:
    RRs pointing to an authority for the name
  - **Additionals**:
    RRs related to the question but don’t answer it

HEADER

```
                                1  1  1  1  1  1
  0  1  2  3  4  5  6  7  8  9  0  1  2  3  4  5
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                      ID                       |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|QR|   Opcode  |AA|TC|RD|RA|   Z    |   RCODE   |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                    QDCOUNT                    |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                    ANCOUNT                    |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                    NSCOUNT                    |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                    ARCOUNT                    |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

QUESTION

```
                                1  1  1  1  1  1
  0  1  2  3  4  5  6  7  8  9  0  1  2  3  4  5
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                                               |
/                     QNAME                     /
/                                               /
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                     QTYPE                     |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                     QCLASS                    |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

RESOURCE RECORD

```
                                1  1  1  1  1  1
  0  1  2  3  4  5  6  7  8  9  0  1  2  3  4  5
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                                               |
/                                               /
/                      NAME                     /
|                                               |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                      TYPE                     |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                     CLASS                     |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                      TTL                      |
|                                               |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                   RDLENGTH                    |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--|
/                     RDATA                     /
/                                               /
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

Common Resource Records

- A/CNAME/PTR

```
www.cs.nott.ac.uk.             61272    IN   CNAME    pat.cs.nott.ac.uk.
pat.cs.nott.ac.uk.             68622    IN   A        128.243.20.9
pat.cs.nott.ac.uk.             68622    IN   A        128.243.21.19
9.20.243.128.in-addr.arpa.     39617    IN   PTR      pat.cs.nott.ac.uk.
```

- NS

```
cs.nott.ac.uk.   10585    IN   NS   ns1.nottingham.ac.uk.
cs.nott.ac.uk.   10585    IN   NS   ns2.nottingham.ac.uk.
cs.nott.ac.uk.   10585    IN   NS   marian.cs.nott.ac.uk.
cs.nott.ac.uk.   10585    IN   NS   extdns1.warwick.ac.uk.
cs.nott.ac.uk.   10585    IN   NS   extdns2.warwick.ac.uk.
```

- MX

```
nott.ac.uk.   3600 IN   MX   1 mx191.emailfiltering.com.
nott.ac.uk.   3600 IN   MX   2 mx192.emailfiltering.com.
nott.ac.uk    3600 IN   MX   3 mx193.emailfiltering.com.
```

Start of Authority, SOA

```
: mort@greyjay:~$; dig -t SOA .

; <<>> DiG 9.6-ESV-R4-P3 <<>> -t SOA .
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15862
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;.                IN   SOA

;; ANSWER SECTION:
.             78314    IN   SOA  a.root-servers.net. nstld.verisign-grs.com.
2012032800 1800 900 604800 86400

;; Query time: 161 msec
;; SERVER: 194.168.4.100#53(194.168.4.100)
;; WHEN: Wed Mar 28 19:41:47 2012
;; MSG SIZE  rcvd: 92
```

Contents

- Naming
- DNS outline
- DNS protocol
- DNS details
  - – Recursive vs. iterative resolution
  - – Names, labels and compression
  - – Load balancing
- Issues

Recursive *vs*. Iterative

- What happens when the resolver queries a server that doesn't know the answer?

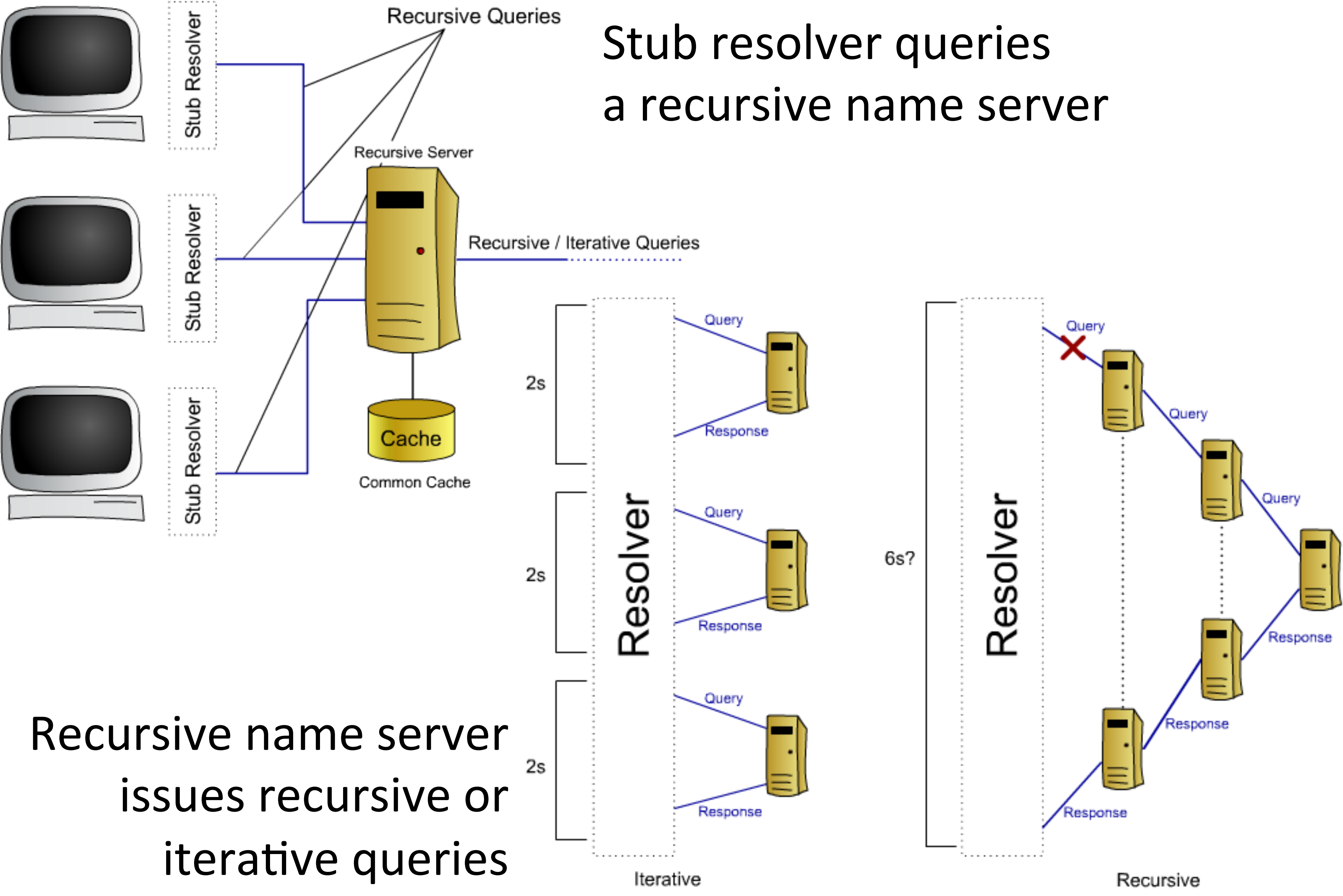
Iterative (*required*)

  – Server responds with a hint who to ask next

Recursive (*optional*)

  – Server generates a new query to the next server

Stub resolver queries
a recursive name server



Recursive name server
issues recursive or
iterative queries

http://services.eng.uts.edu.au/~kumbes/ra/dns/DNS.html

Names & Labels

- A **domain name** is a sequence of **labels**
  - Each label is a sequence of **characters** preceded by the label's **length** in 1 byte
  - Each name ends in a null label (0x00)
  - Names, either complete or suffixes, will appear many times in a response
- DNS UDP packets are limited to 512 bytes
  - Longer messages are truncated and the TC bit set
  - Messages are getting longer thanks to new RR types
  - (Use TCP for *Zone Transfer*, AXFR, though)

Label Compression

- Thus, some form of compression is required
- Labels are limited to 63 characters long
  - What does this imply for the length byte?
  - We have the top two bits unused
- So, DNS introduces pointers
  - Point back in the packet to a previous occurrence of the name or label suffix
  - Pointer is therefore

```
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
| 1  1|                OFFSET                   |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

F.ISI.ARPA,
FOO.F.ISI.ARPA,
ARPA, and the root
in a datagram

```
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
20 |           1           |           F           |
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
22 |           3           |           I           |
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
24 |           S           |           I           |
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
26 |           4           |           A           |
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
28 |           R           |           P           |
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
30 |           A           |           0           |
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
40 |           3           |           F           |
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
42 |           O           |           O           |
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
44 | 1  1|                20                       |
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
64 | 1  1|                26                       |
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
92 |           0           |                       |
   +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

Load Balancing

```
;; QUESTION SECTION:
;www.google.com.        IN  A

;; ANSWER SECTION:
www.google.com.     509678 IN  CNAME  www.l.google.com.
www.l.google.com.   55  IN  A   173.194.78.104
www.l.google.com.   55  IN  A   173.194.78.99
www.l.google.com.   55  IN  A   173.194.78.106
www.l.google.com.   55  IN  A   173.194.78.103
www.l.google.com.   55  IN  A   173.194.78.105
www.l.google.com.   55  IN  A   173.194.78.147
```

Authority & Caching

- SOA record indicates this server is authoritative for the zone
  - Ultimate authority resides with the root
- But! Servers can cache RRs
  - Helps to distribute load
  - Requires TTL (seconds) to indicate when caching server should remove RR from cache

Contents

- Naming
- DNS outline
- DNS protocol
- DNS details
- Issues
  - – DNSSEC outline

Operational & Security Issues

- Usually need primary and secondary servers
  - Separate IP netblocks, physical networks, etc.
  - DNS is a *very* common single-point-of-failure
- Cache poisoning
  - Caching and soft-state mean bad data propagates and can persist for some time
  - Even if through a simple mistake
- Man-in-the-middle attacks
  - Iterative/Recursive queries almost demand this
- Bad fonts, etc

DNSSEC (outline)

- How do you know your answer is correct?
  - Authority signs the relevant RRs with private key
  - You verify with public key
- Chain of trust up to the (signed) root
  - Root points (signed) DS record at (signed) DNSKEY records of children indicating
  - Querier checks signatures (RRSIG records) from root downwards, ensuring authenticity at each stage

Contents

- Naming
  - HOSTS.TXT
  - DNS
- DNS outline
  - Components
  - Hierarchy
- DNS protocol
  - Queries
  - Responses
  - Resource Records
- DNS details
  - Recursive vs. iterative resolution
  - Names, labels and compression
  - Load balancing
- Issues
  - DNSSEC outline

Summary

- DNS is a distributed hierarchical database
- Supports resolution of names to attributes represented in resource records
- A range of technical details/tricks
  - Recursive/iterative resolution
  - Label compression
  - Load balancing
- More recent support for authenticated names

Quiz

1. What was the scalability problem posed by the original use of HOSTS.TXT?
2. Why is it important that DNS is consistent?
3. Describe the four key components of the DNS and what they do.
4. If I own the zone *g54acc.net* and I delegate to you the zone *quiz.g54acc.net*, what does that mean in the DNS?
5. Discuss the distinction between a DNS resolver and a DNS server?
6. Draw a diagram depicting a recursive resolution, and another depicting an iterative resolution. The diagram should include the client, the resolver, and at least two DNS servers.
7. The names g54acc.net, quiz.g54acc.net, exam.g54acc.net and question.exam.g54acc.net all appear in a DNS response packet. Assuming they appear consecutively starting at offset 0x20, show how they would be represented with and without label compression.