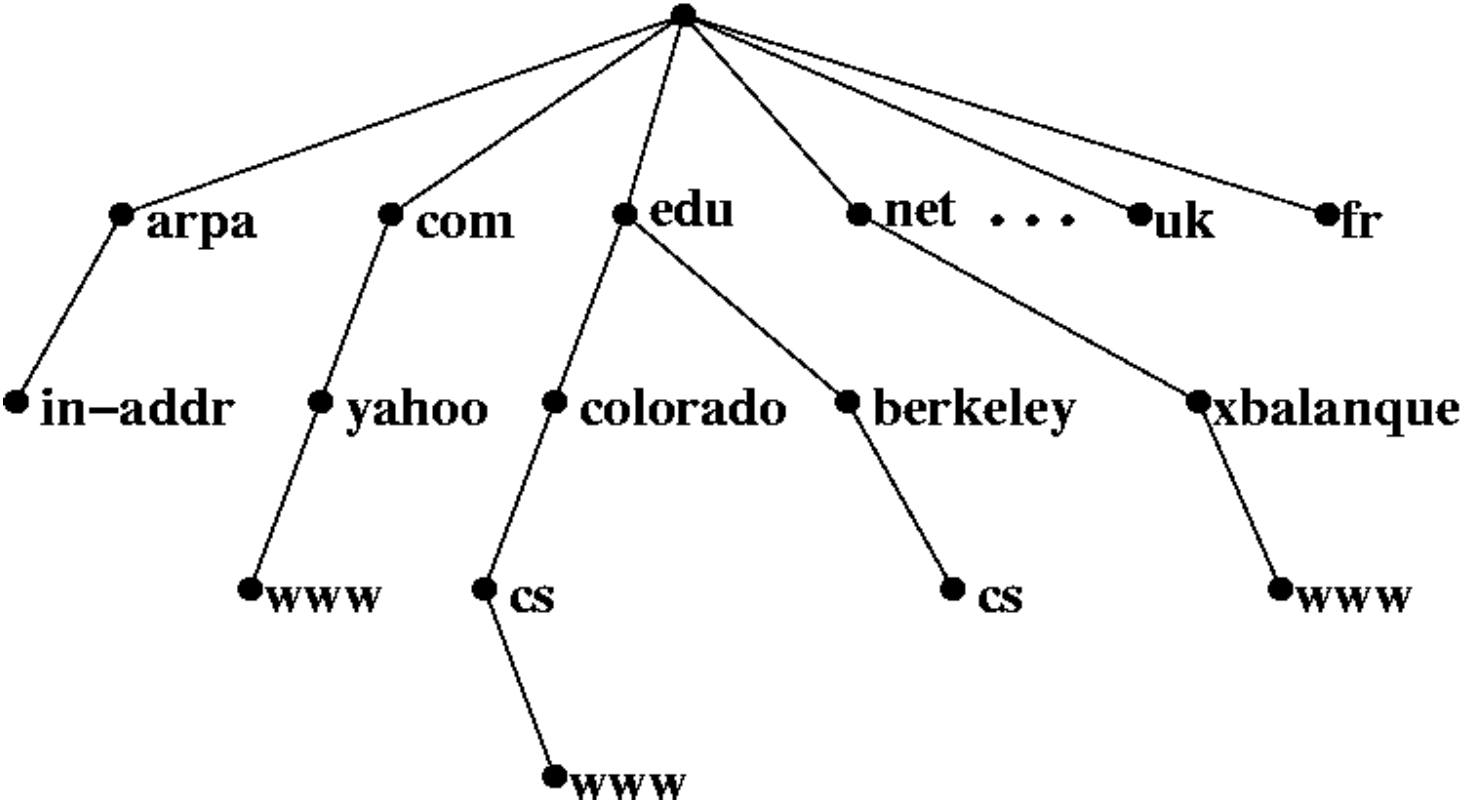
DNS and CDNs


arpa
com
edu
net
. . .
uk
fr
in–addr
yahoo
colorado
berkeley
xbalanque
www
cs
cs
www
www

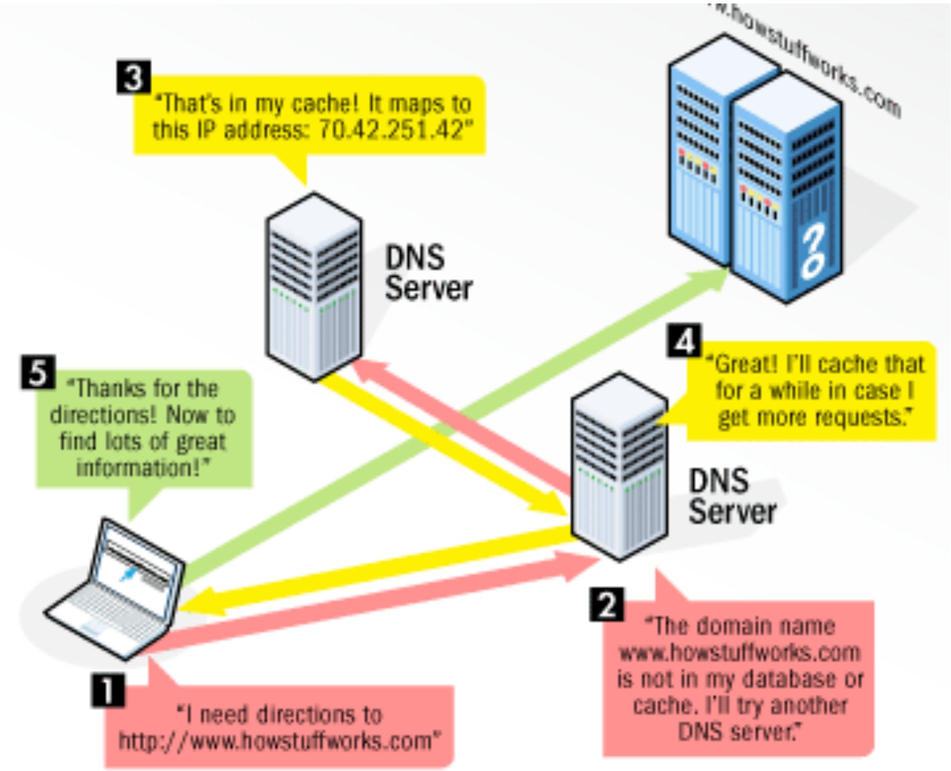
Locating Things Online

- **Addresses** describe a location
  - A map is nice, but not for lookup
  - IP addresses, street addresses
- **Names** are mapped to addresses
  - Domain names; PO Boxes; email addresses
- **Content-based names** use the actual content to find the destination
  - Basis of publish/subscribe and peer-to-peer

Domain Name System

- The Internet's routers know only IP addresses
  - Fine for routers, not great for humans
- DNS translates domain names to IP addresses
  - `google.com => 74.125.95.99`
  - (or something similar)
  - The "phone book" of the internet

DNS at work


www.howstuffworks.com
3
"That's in my cache! It maps to this IP address: 70.42.251.42"
DNS Server
5
"Thanks for the directions! Now to find lots of great information!"
4
"Great! I'll cache that for a while in case I get more requests."
DNS Server
1
"I need directions to http://www.howstuffworks.com"
2
"The domain name www.howstuffworks.com is not in my database or cache. I'll try another DNS server."

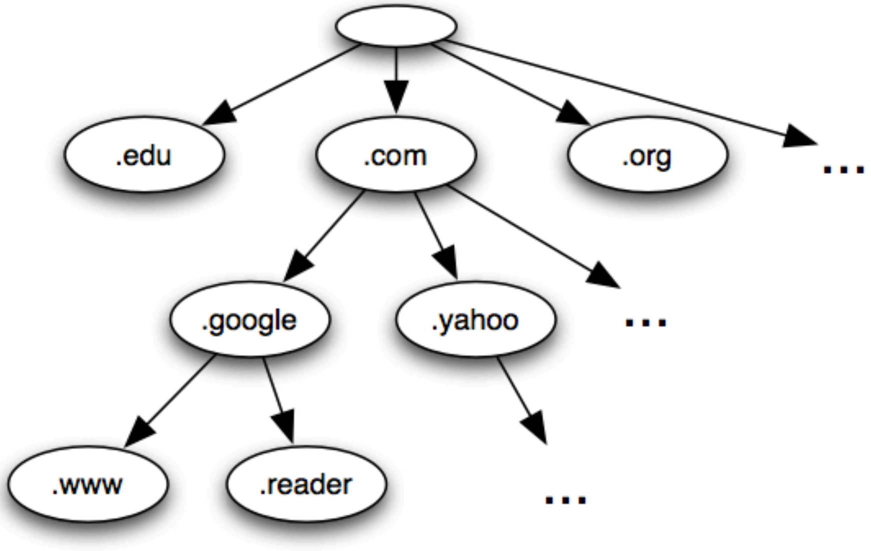
howstuffworks.com

DNS Design

- DNS is a globally-accessible database of name/IP pairs. Some requirements:
  - Needs to be up all the time
  - Continuously updated by many parties
  - Must be accurate; errors prevent connections
  - Serves massive query load
  - Needs distributed administration
  - Source of political & commercial disputes
  - Potential terrorism target
- Overall: utterly shocking that it works

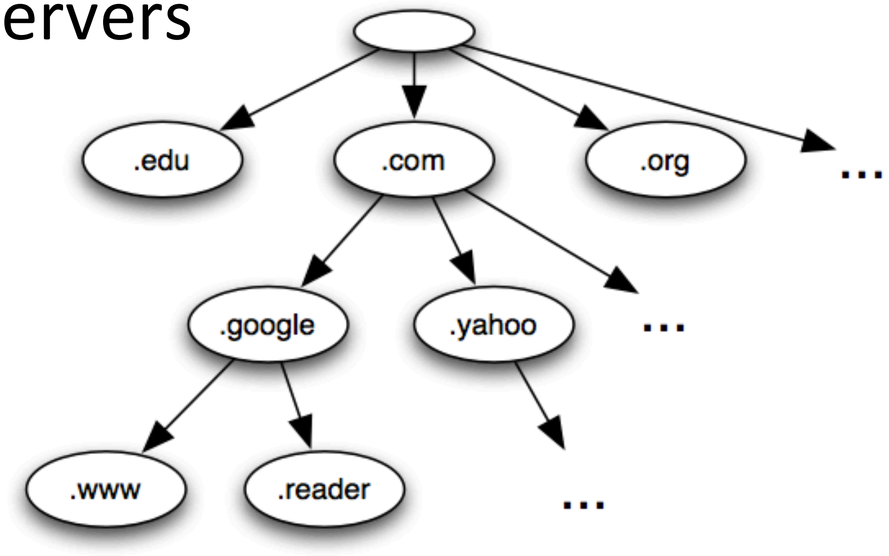
The Namespace

- Tree structure, 1-63 chars per node
- *Fully-qualified domain name* is leaf-to-root name
- Only *DNS root* has no parent



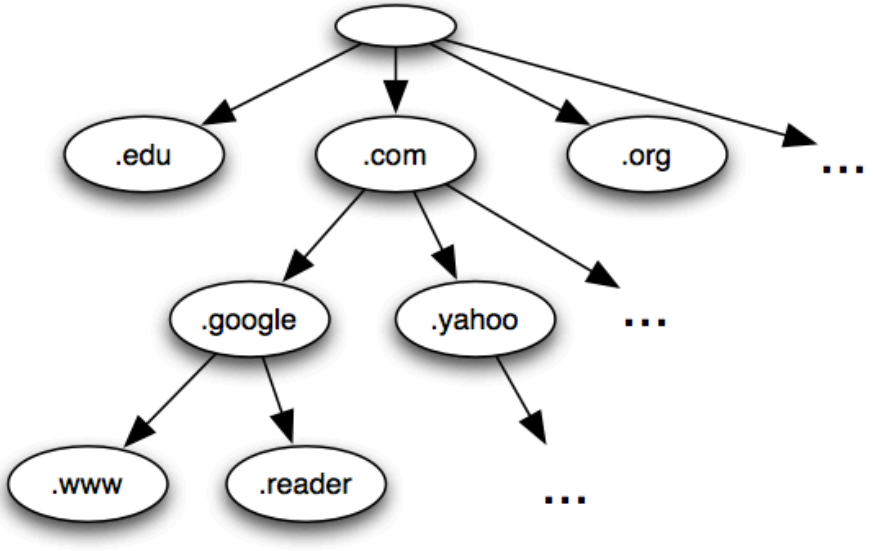
The Namespace

- Nodes grouped into administrative zones
  - E.g., root, com, google.com
- Each zone served by authority servers
  - AKA authoritative name servers
- Authority server can delegate subdomains to other authority servers



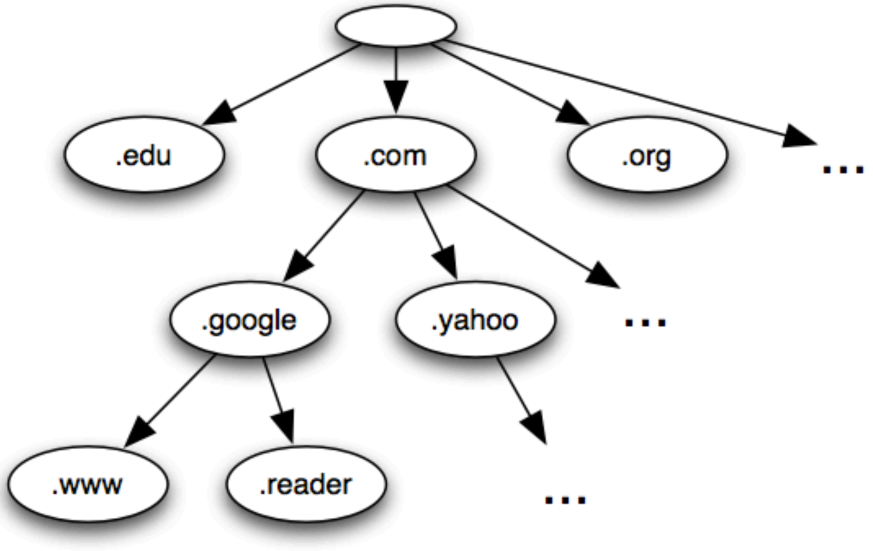
The Namespace

- Servers are primary or secondary
  - Primary authority servers are given content by admins
  - Secondary authority servers grab from primaries
- A domain registrar inserts your name into the primary authority server for .com (or .net, .org, etc)



The Namespace

- Purchased domain names inserted into 1 primary, 1 secondary (in case of primary failure)



Try it yourself

- Check if a domain name is registered

```
$ whois umich.edu
...
Registrant:
   University of Michigan -- ITD
   ITCS, Arbor Lakes
   4251 Plymouth Road
   Ann Arbor, MI 48105-2785
   UNITED STATES
...
Domain record activated:    07-Oct-1985
Domain record last updated: 14-Feb-2014
Domain expires:             31-Jul-2016
```

The Data

- Content of DNS consists of *resource records*
  - Host/IP and IP/Host are most popular type
  - Also: zone info, email servers, other info
- Most DNS activity is this:
  - DNS client sends request to authority servers, receives resource record in response
- DNS clients built into network libraries
  - Clients use UDP (not TCP) to grab data
  - If your connection is taking a long time to establish, possibly waiting for DNS

Resolution

- Single authority server may not be enough for a frequently queried domain name

Try it yourself

- Look up IP address for web.eecs.umich.edu

```
$ nslookup web.eecs.umich.edu
Server:         192.168.1.1
Address:        192.168.1.1#53

Non-authoritative answer:
Name:    web.eecs.umich.edu
Address: 141.212.113.110
```

- Non-authoritative means it’s returning cached results

Try it yourself

- Look up IP address for eecs.umich.edu

```
$ nslookup -type=ns eecs.umich.edu
Server:          192.168.1.1
Address:   192.168.1.1#53

Non-authoritative answer:
eecs.umich.edu   nameserver = csedns.eecs.umich.edu.
eecs.umich.edu   nameserver = dns.eecs.umich.edu.
eecs.umich.edu   nameserver = eecsdns.eecs.umich.edu.

Authoritative answers can be found from:
eecsdns.eecs.umich.eduinternet address = 141.213.4.4
csedns.eecs.umich.edu internet address = 141.212.113.4
```

- Return authoritative name servers (complete zone information)

Try it yourself

- **Alternative:** `dig`

```
$ dig eecs.umich.edu
; <<>> DiG 9.8.3-P1 <<>> eecs.umich.edu
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 29174
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;eecs.umich.edu.                IN      A

;; ANSWER SECTION:
eecs.umich.edu.         900     IN      A       141.212.113.199

;; Query time: 35 msec
;; SERVER: 192.168.1.1#53(192.168.1.1)
;; WHEN: Mon Nov 23 09:14:48 2015
;; MSG SIZE  rcvd: 48
```

Reverse DNS lookup

- Reverse DNS lookup
- Start with IP, find name

```
$ dig -x 141.212.113.199
```

OR

```
$ dig +short -x 141.212.113.199
www.eecs.umich.edu
```

Exercise

- What domain hosts this website?
  - http://andrewdeorio.com
- HINT: use both a DNS lookup and a reverse lookup
- Extra credit: Linux one-liner

Exercise

- Where is this website hosted?
- andrewdeorio.com
- DNS lookup

```
$ dig +short andrewdeorio.com
141.212.113.110
```

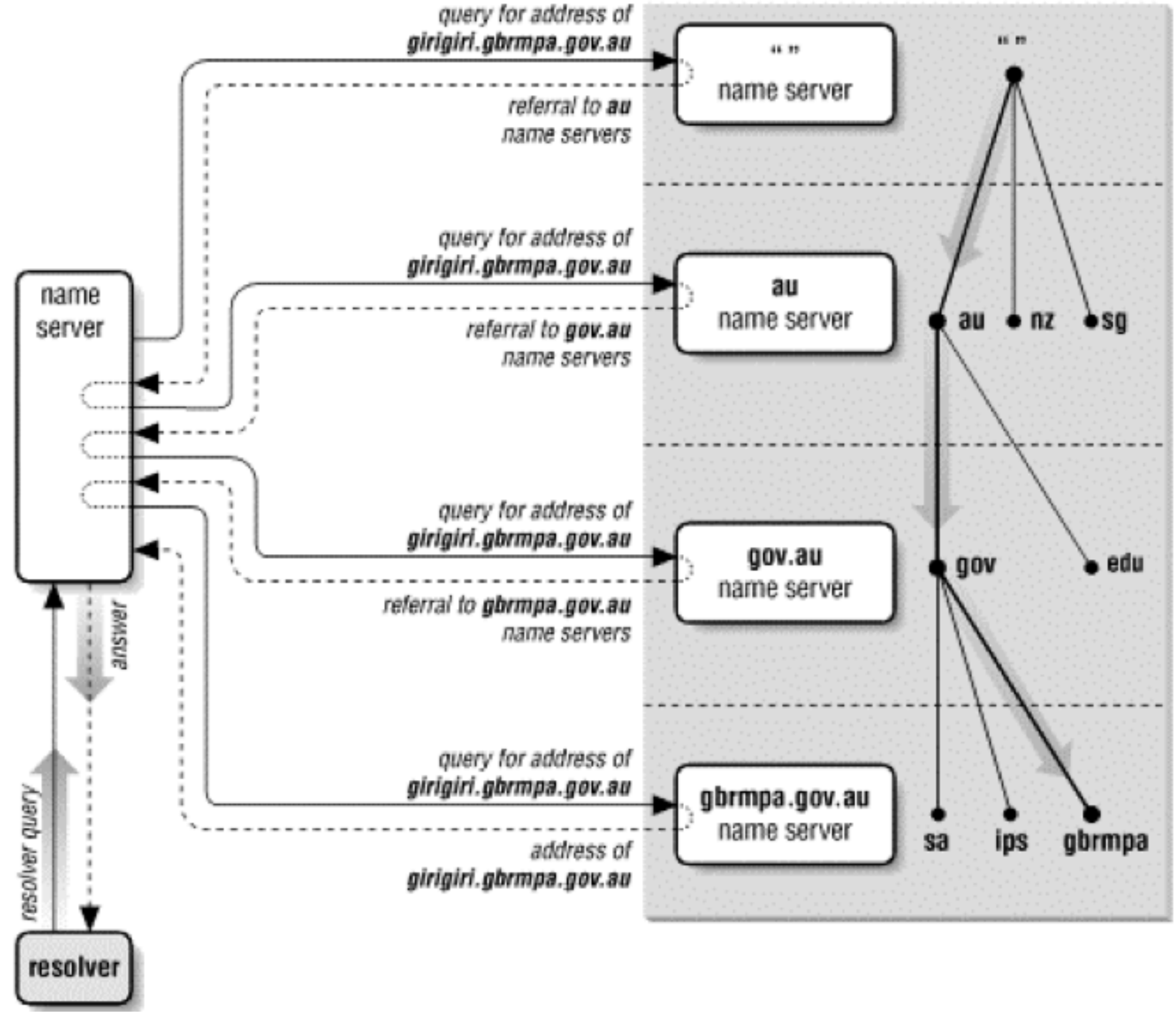
- Reverse DNS lookup

```
$ dig +short -x 141.212.113.110
web.eecs.umich.edu.
```

- One-liner

```
$ dig +short andrewdeorio.com | xargs dig +short -x
web.eecs.umich.edu.
```

Resolution


query for address of
girigiri.gbrmpa.gov.au
" "
name server
referral to au
name servers
query for address of
girigiri.gbrmpa.gov.au
au
name server
referral to gov.au
name servers
query for address of
girigiri.gbrmpa.gov.au
gov.au
name server
referral to gbrmpa.gov.au
name servers
query for address of
girigiri.gbrmpa.gov.au
gbrmpa.gov.au
name server
address of
girigiri.gbrmpa.gov.au
name
server
answer
resolver query
resolver
" "
au
nz
sg
gov
edu
sa
ips
gbrmpa

Caching

- There may be a chain of caching DNS servers between client and authority server
  - There's one for CSE
  - Caches DNS requests; answers new requests from cache whenever possible
- Great! But when data changes?
  - Each resource record has time-to-live (TTL) in seconds
  - Counts down from moment authority server emits resource record (RR)
  - Caches and clients must throw out resource records with expired time-to-live

Root Nameservers

- Responsible for locating the top level domain name servers (.com, .net, .org, …)
- Thirteen root name servers in world
  - Their locations are hard-coded in resolving DNS servers
  - Due to caching, involved in few queries

Administration

- Until 1999, all US top level domains run by IANA
  - (At the time, IANA = Jon Postel)
  - IANA: Internet Assigned Numbers Authority
- Now, non-profit ICANN administers set of for-profit registrars (under contract with government)
  - ICANN: Internet Corporation for Assigned Names and Numbers

Administration

- For a time, looked like ICANN might be transferred to United Nations. Probably won't be
- Postel's "Law": Be liberal in what you accept, and conservative in what you send

DNS Challenges

- Shocking scale
  - Every device, service, etc
  - Spam-filtering email requires 10+ DNS lookups per message
- Extreme security vulnerability
  - What if someone can remap **`google.com`**?
  - What about just denial-of-service?

DNS Cache Poisoning

- Bad info is inserted into a DNS server and cached
  - Both inadvertent and malicious
  - How could one attack CSE's DNS server?
- Remedies
  - DNSSEC requires that DNS entries be cryptographically signed
    - Signed by whom?
      - Chain of Trust
  - If you use HTTPS/SSL/TSL, problem mitigated

DNS Cache Poisoning

- DNS cache poisoning is one of the ways that China's Great Firewall works
- Just direct requests to twitter.com to the wrong IP

ZAPIRO
SUNDAY TIMES
Johannesburg
SOUTH AFRICA
CWS / CARTOONARTS INTERNATIONAL www.cartoonweb.com
humanrights.org Page cannot be displayed
tibet.com Page cannot be displayed
..THE GREAT FIREWALL OF CHINA!
Beijing 2008

Invalid Domain Name

- DNS returns `NXDOMAIN`
- Browser shows user an error page
- Wasted “eyeball” opportunity
- Sell advertising instead
  - By inserting default entries into DNS

- Can use same trick to steal cookies
  - Insert DNS entry for non-existent sub-domain and get user to request it somehow

Content Distribution

- Now, on to Content Distribution Networks (CDNs)

Proxy

- Also called "proxy server"
- Looks like a server to the client, looks like client to the server
- Can have any number of proxies in the middle between client and server
- Proxy can
  - Change request or not
  - Announce itself or not
  - Change response or not
- Used for a variety of functions

Proxy Uses

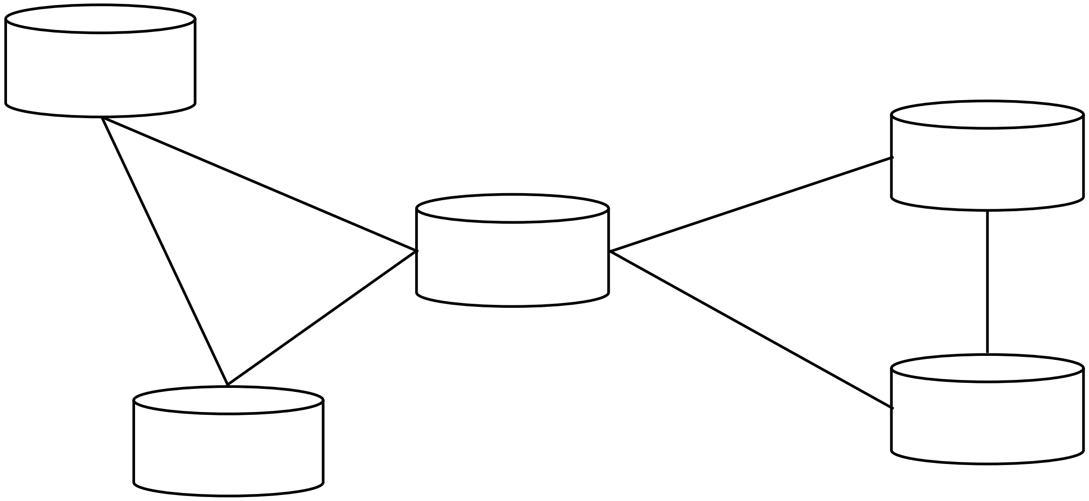
- Anonymity
- Content Filtering
- Cache Sharing

Akamai

- Stores images, videos, at many locations throughout the world
  - Large read-only data stored close to client
  - Reduce latency to faraway datacenter
  - Reduce bandwidth costs by sending data only a short distance
- The 1st CDN (content-delivery network)
  - You might think the image is from Yahoo, but Akamai is serving it
  - Yahoo pays Akamai to do so
- Built on top of giant DNS hack!

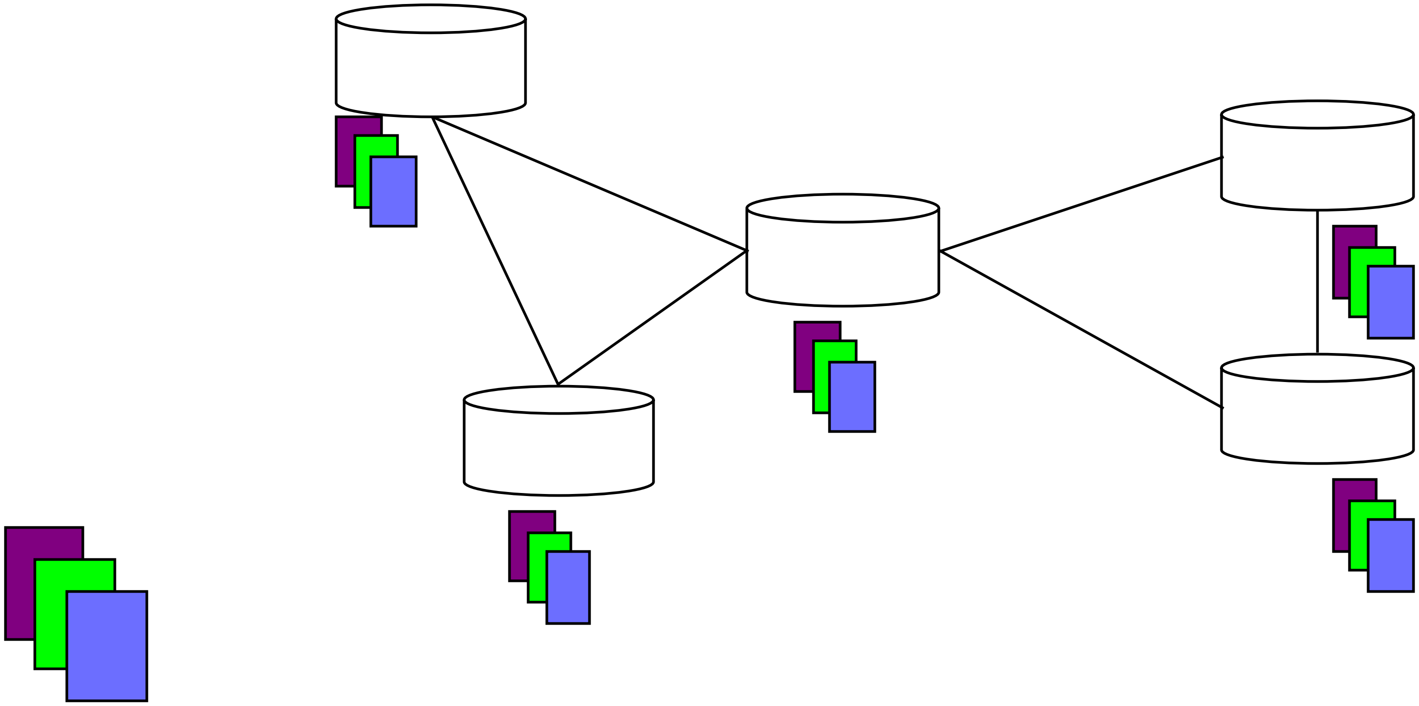
How it works: step 1

- Akamai places servers across country



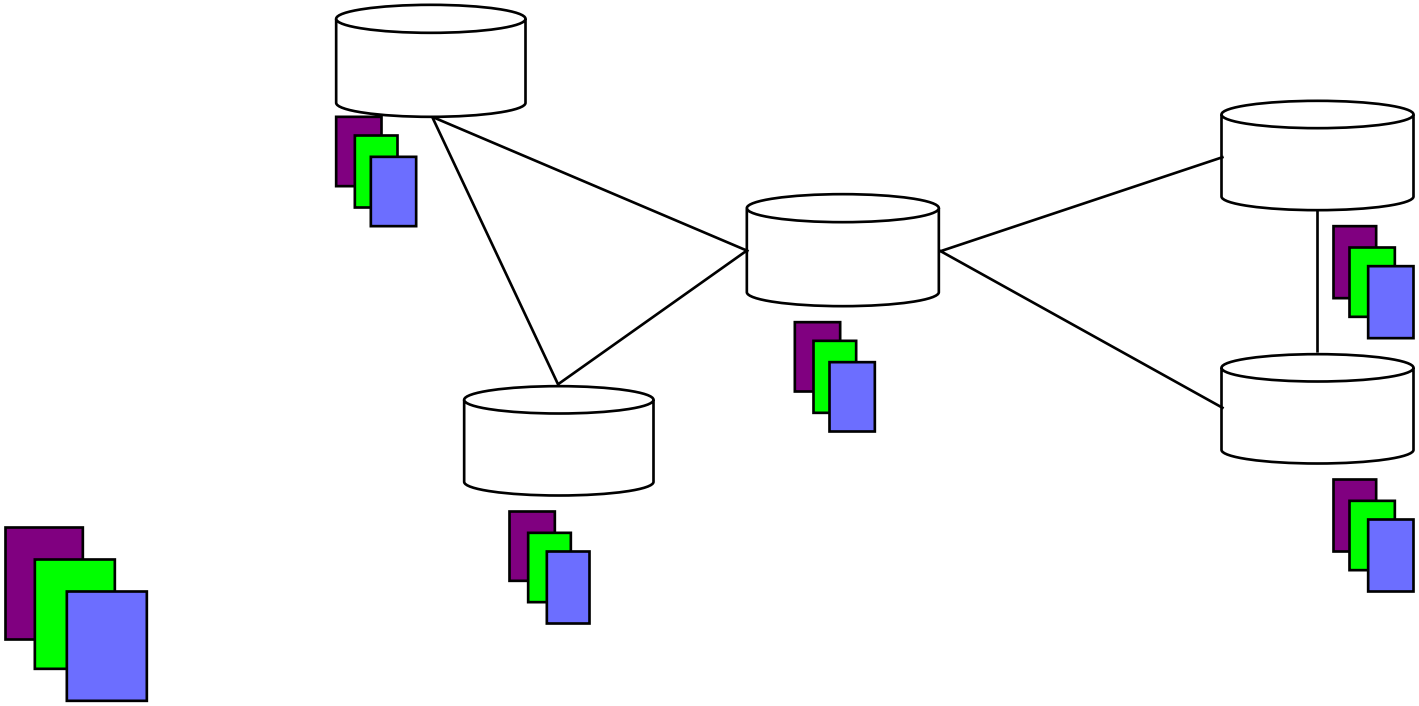
How it works: step 2

- Akamai gets client data (videos, imgs), copies to all nodes in network



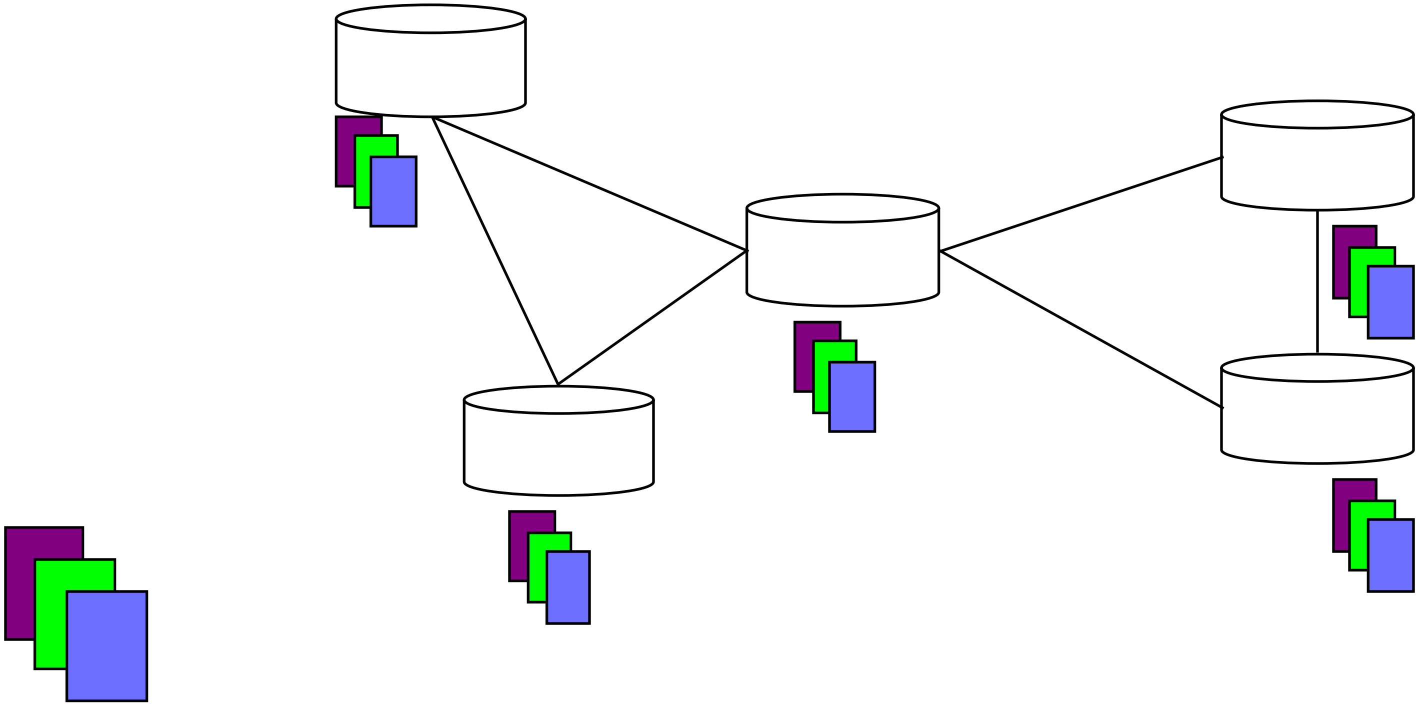
How it works: step 3

- How does client find nearby data?
  - Network conditions can change



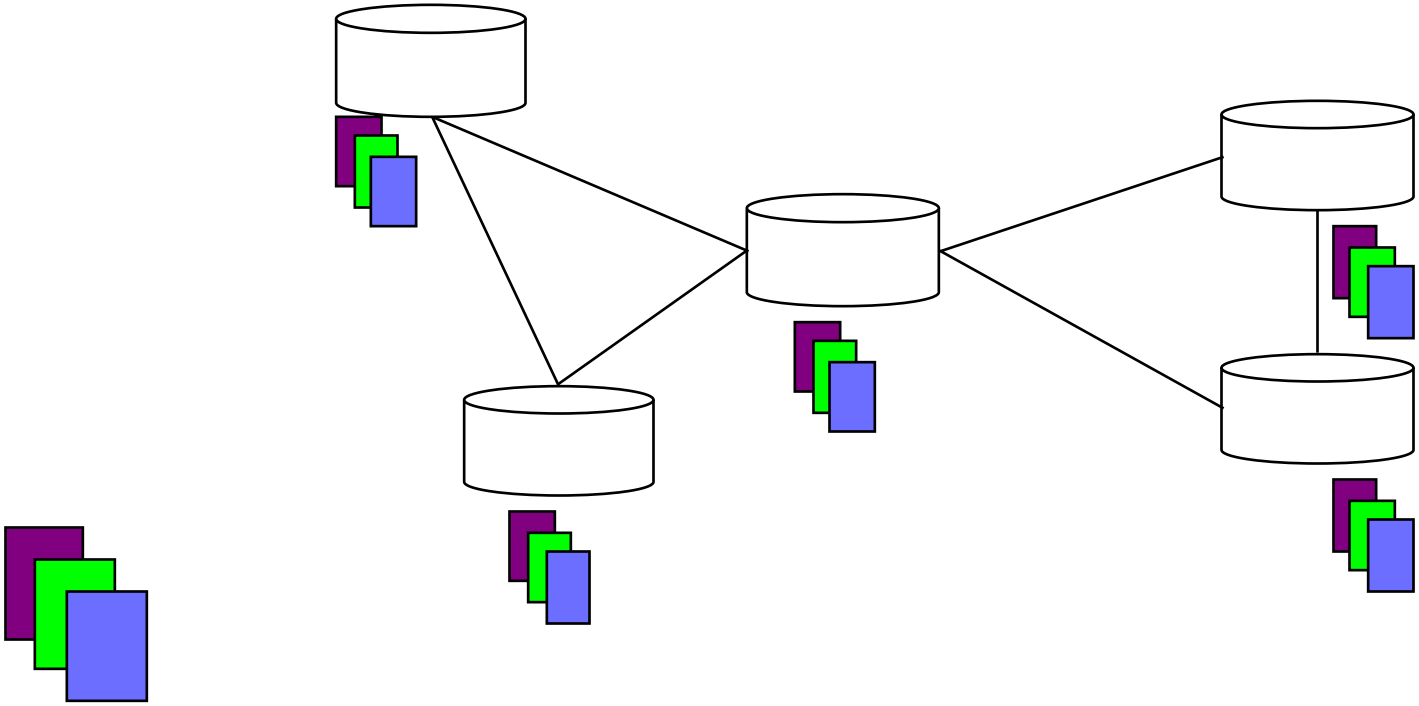
How it works: step 3

- Client rewrites URLs, uses new URLs
  - http://yahoo.com/… => http://akadns.akamai.net/….



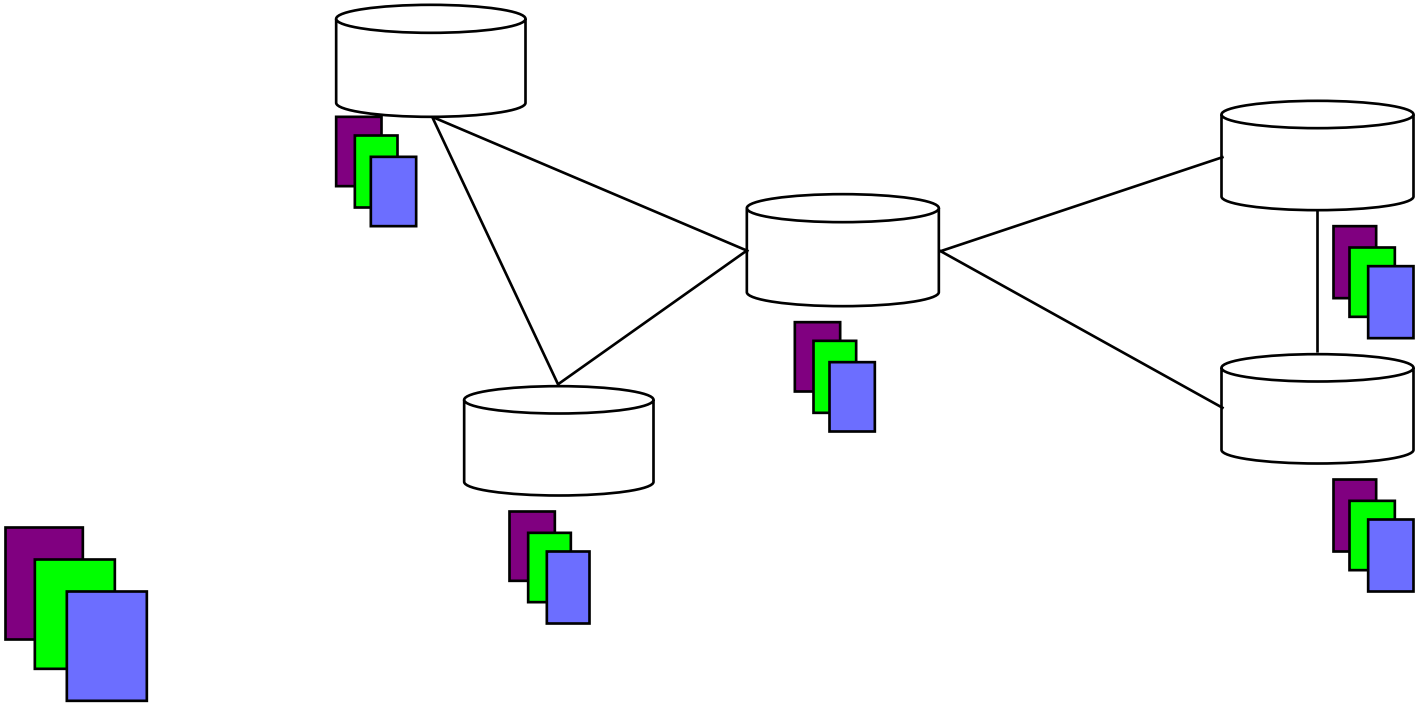
How it works: step 3

- URL contains Akamai-controlled name
  - Specialized Akamai DNS server resolves name to nearby server; tiny TTL



How it works: step 3

- Resolution strongly dependent on
  - Location of client; network conditions; load on Akamai servers; traffic estimation errors



Akamai Recap

- Uses custom DNS servers to dynamically direct clients to right caching servers

Push Technologies

- Email is "push"
  - Sender proactively sends information to the recipient
  - Good (commercial) senders require explicit subscriptions to send
- Web is "pull" only
  - HTTP is asymmetric request/response
  - Web Server cannot push content to user with an explicit user request
  - Address by having the user subscribe

RSS

- RSS = Rich Site Summary, or Really Simple Syndication
- Pack updates to your site into an XML file using RSS tags
- Consider this a broadcast "channel"
- RSS-reader periodically checks specified sites for updates, and automatically downloads them

Atom

- Do exactly the same steps as described for RSS, except with a slightly different specification
    - Differences are minor and in the details
    - Atom is newer and "better"
- Use Atom Syndication Format to define a "feed" as a set of "entries" in XML
- Use Atom Publishing Protocol to GET (and also to POST) the feed XML file

Website Syndication

- Content creator places content in files created according to Atom or RSS
- Content duplicator accesses syndicated content (headlines) and displays on their own website.
- Very popular model for Blogs
- Podcasts are similar

Writing to the Server

- Primary focus has been reading from the website, and writing back very limited information, e.g. in forms
- How about uploading data sets, videos, large formatted blog entries, …
- SWORD: A Simple Web-Service Offering Repository Deposit
    - Based on Atom
    - Supported by many repositories

Publish Subscribe

- In email, or other ordinary messaging, the sender identifies the recipient(s)
- In broadcast, the sender just publishes, and whoever wants to can tune in
- In subscription, the recipient specified precisely what messages it is interested in receiving
- Publish/subscribe extends the traditional messaging model in that the sender may not know who the recipients will be when creating message

Implementation Models

- The basic concept is the same, but there are several implementations:
- List-based
  - Subscriber adds themselves to a list. Publisher sends to whoever is on the list at the time of message sending
- Broadcast
  - Publisher sends to all; local filters select
- Content-based
- Distribution Networks