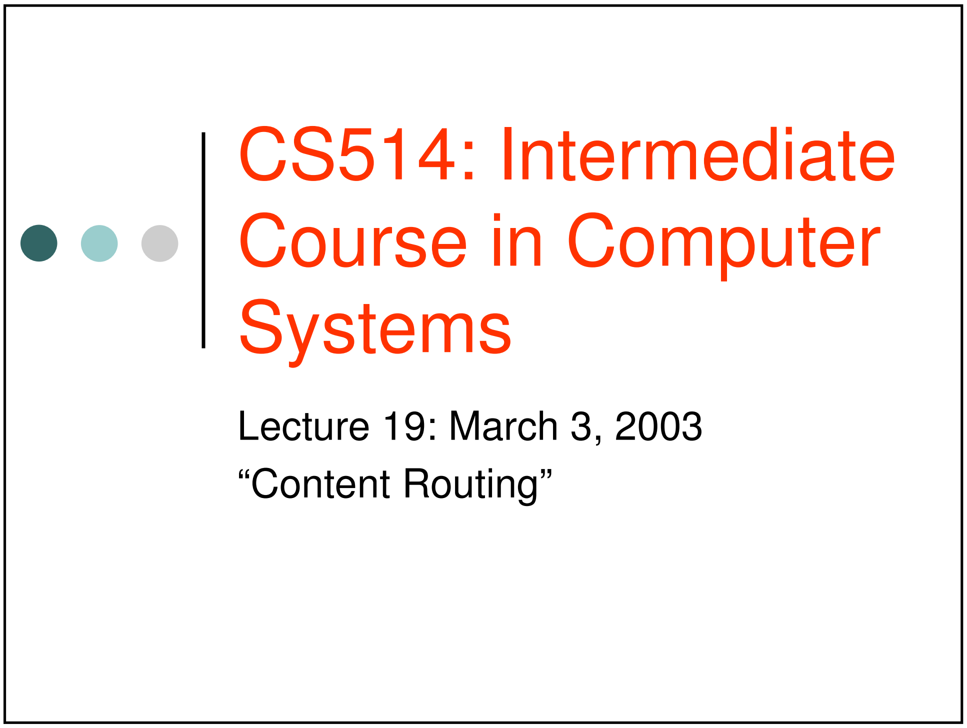
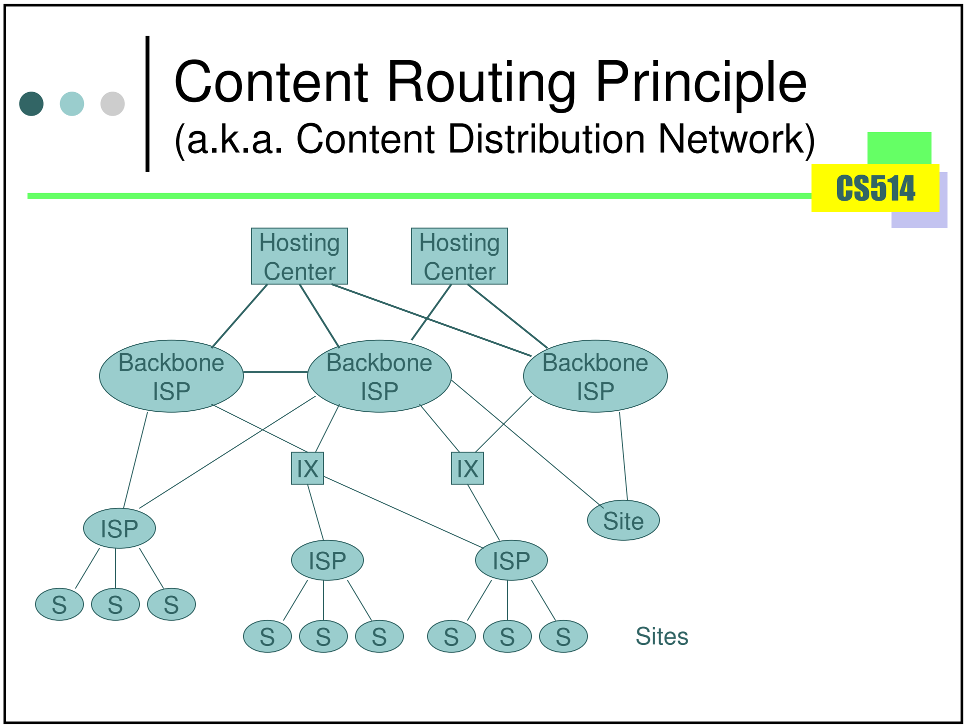


CS514: Intermediate Course in Computer Systems

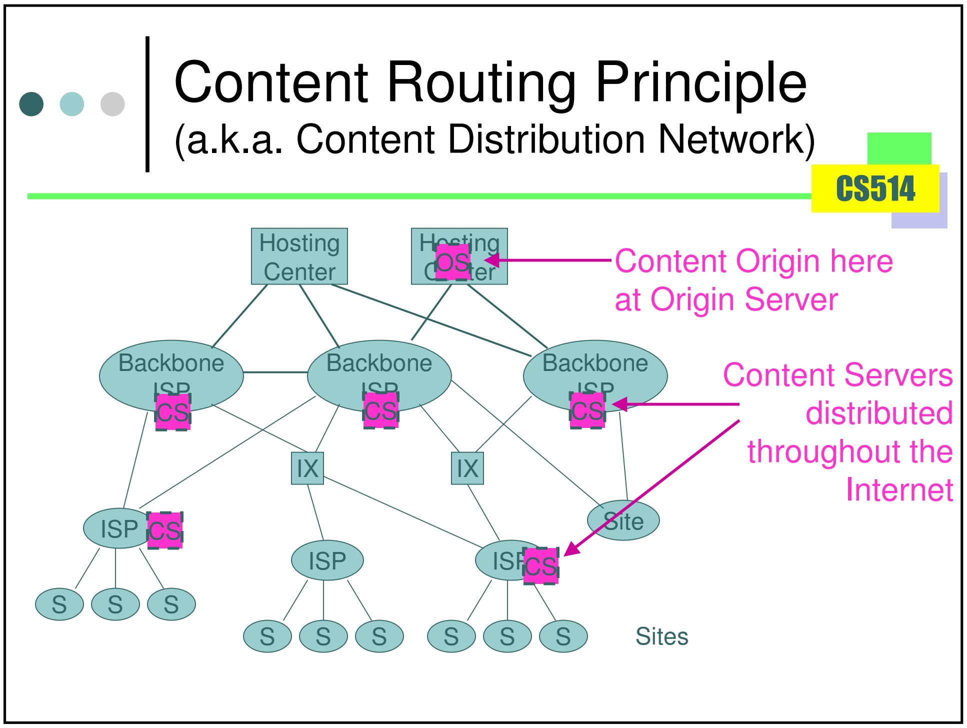
Lecture 19: March 3, 2003

“Content Routing”



Content Routing Principle

(a.k.a. Content Distribution Network)



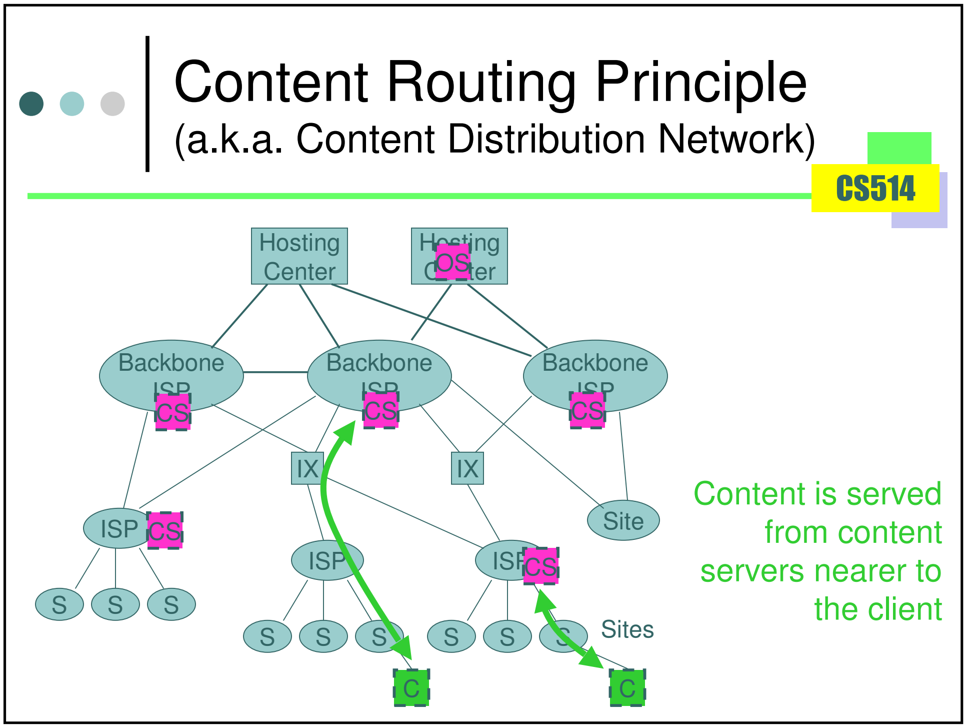
Content Routing Principle

(a.k.a. Content Distribution Network)

CS514

Content Origin here at Origin Server

Content Servers distributed throughout the Internet

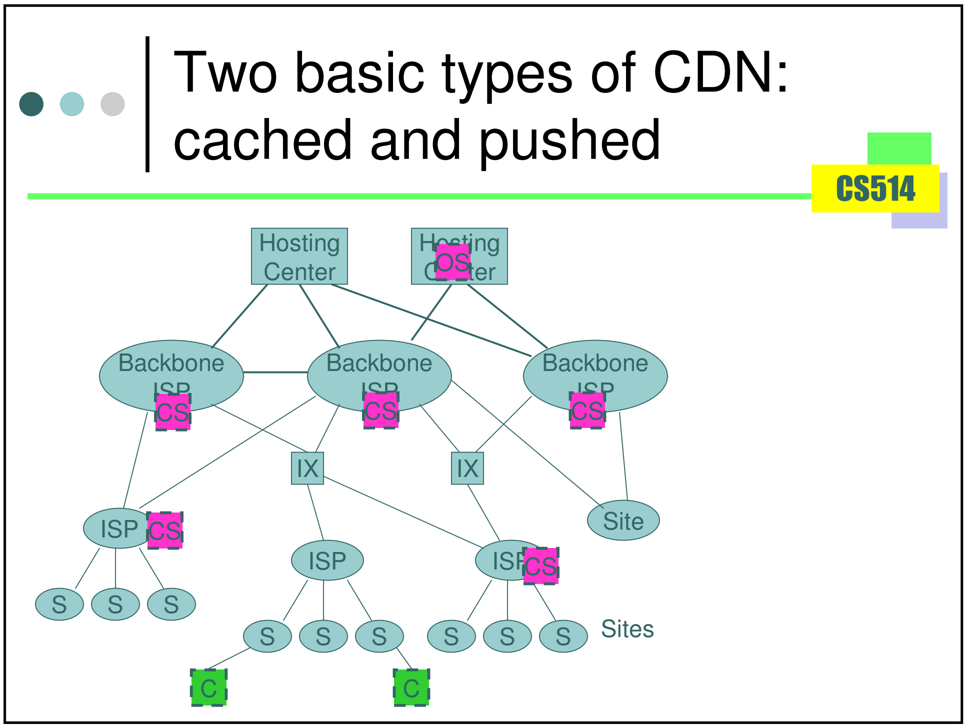


Content Routing Principle

(a.k.a. Content Distribution Network)

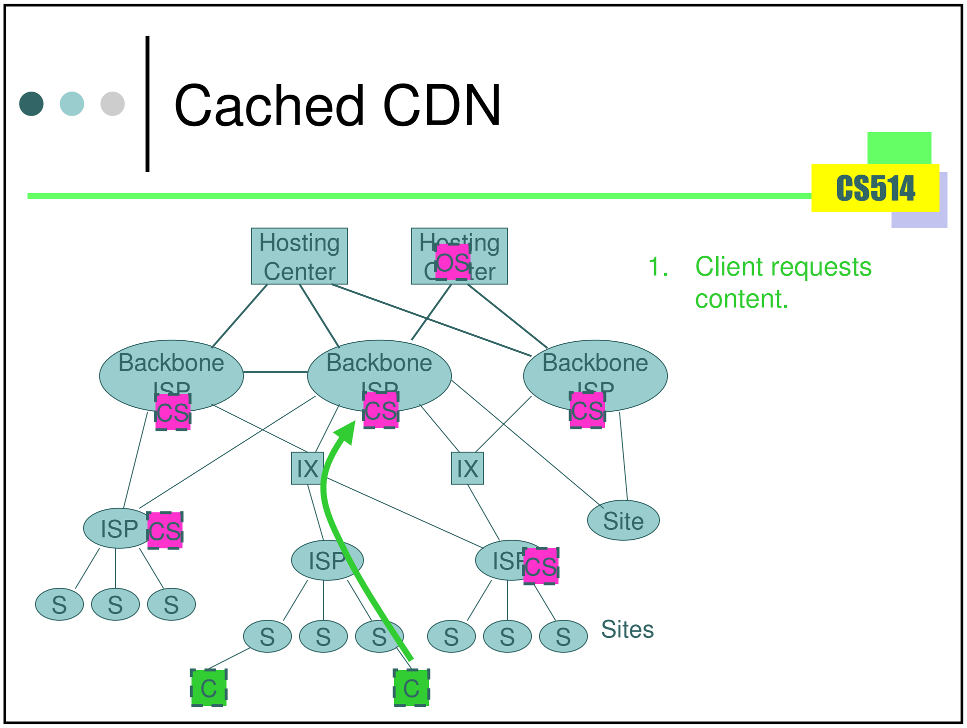
CS514

Content is served from content servers nearer to the client



Two basic types of CDN: cached and pushed

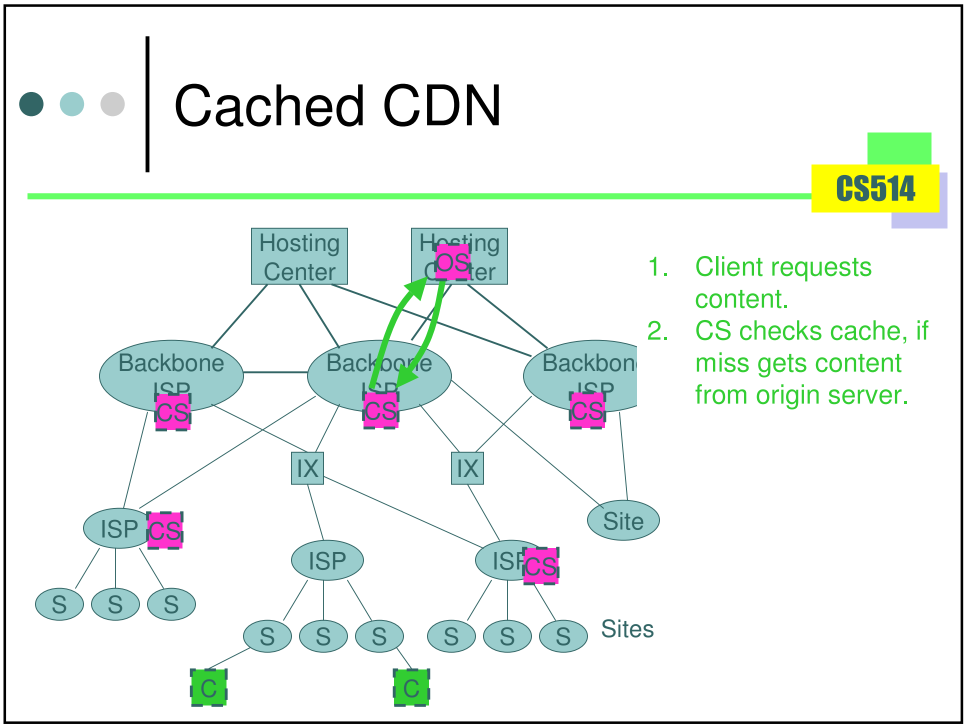
CS514



Cached CDN

CS514

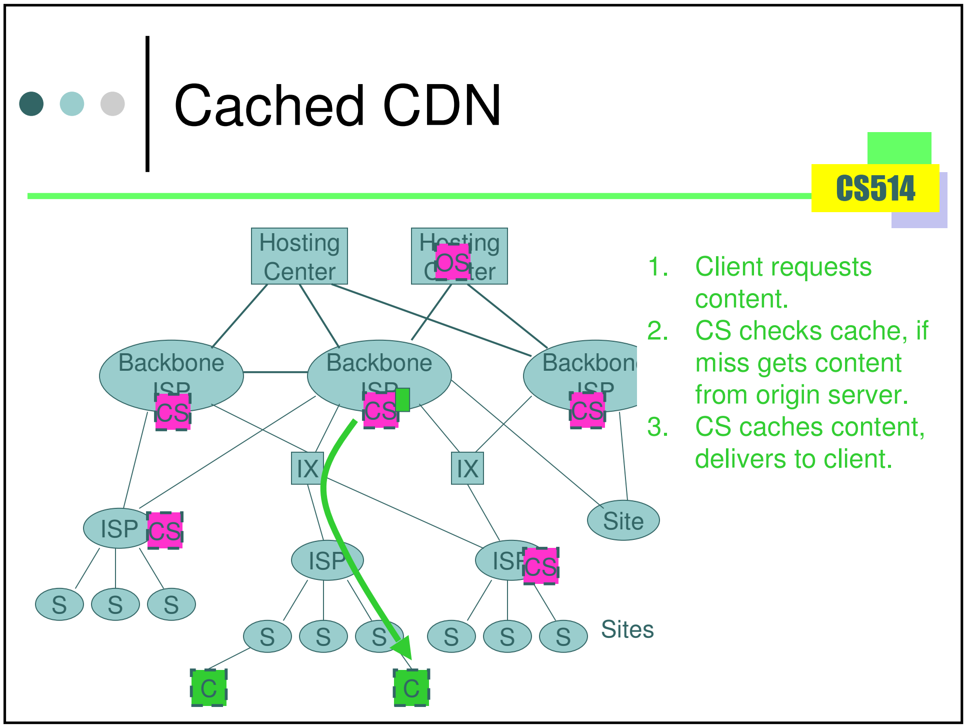
1. Client requests content.



Cached CDN

CS514

1. Client requests content.
2. CS checks cache, if miss gets content from origin server.



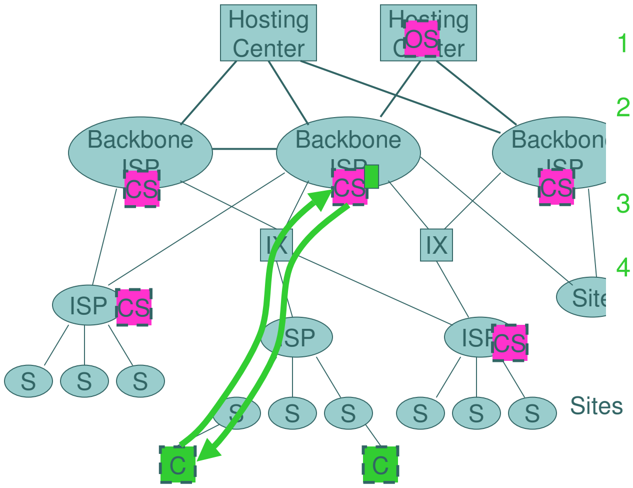
Cached CDN

CS514

1. Client requests content.
2. CS checks cache, if miss gets content from origin server.
3. CS caches content, delivers to client.

Cached CDN

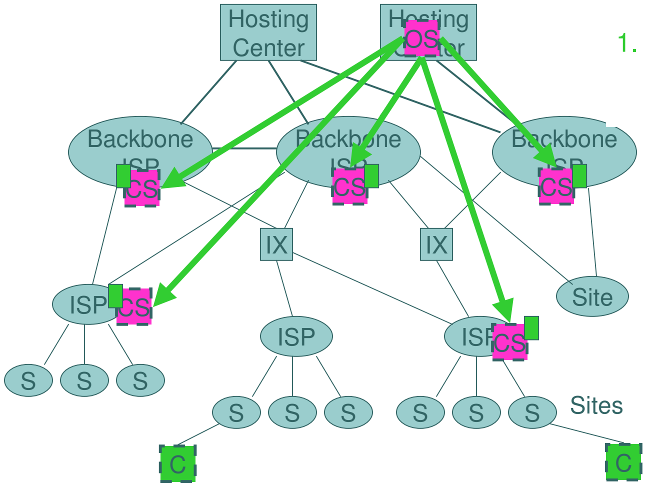
CS514



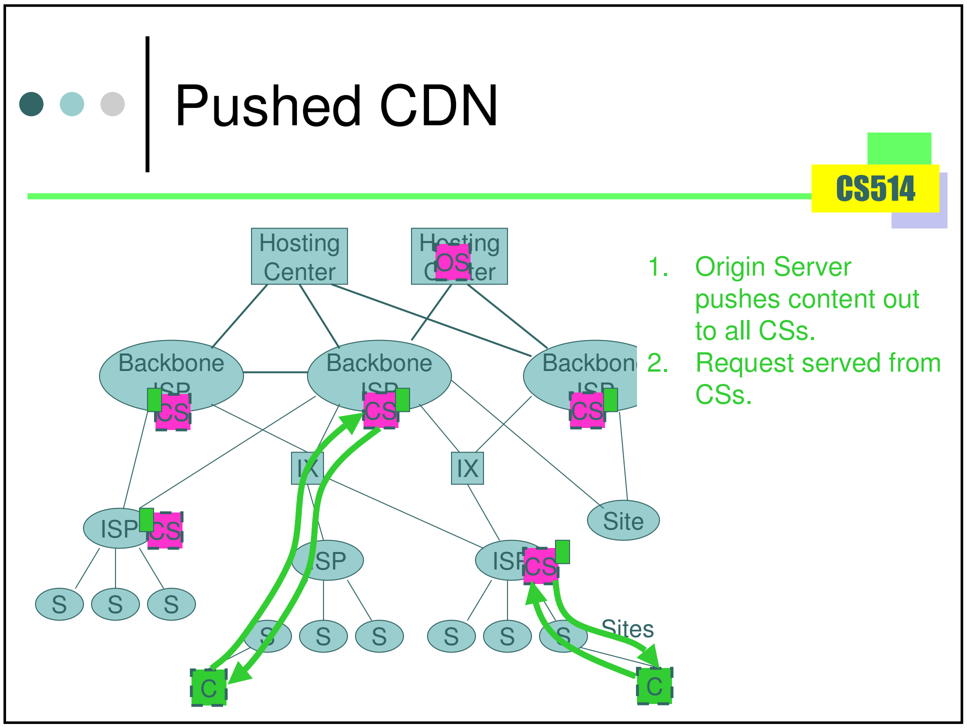
1. Client requests content.
2. CS checks cache, if miss gets content from origin server.
3. CS caches content, delivers to client.
4. Delivers content out of cache on subsequent requests.

Pushed CDN

CS514



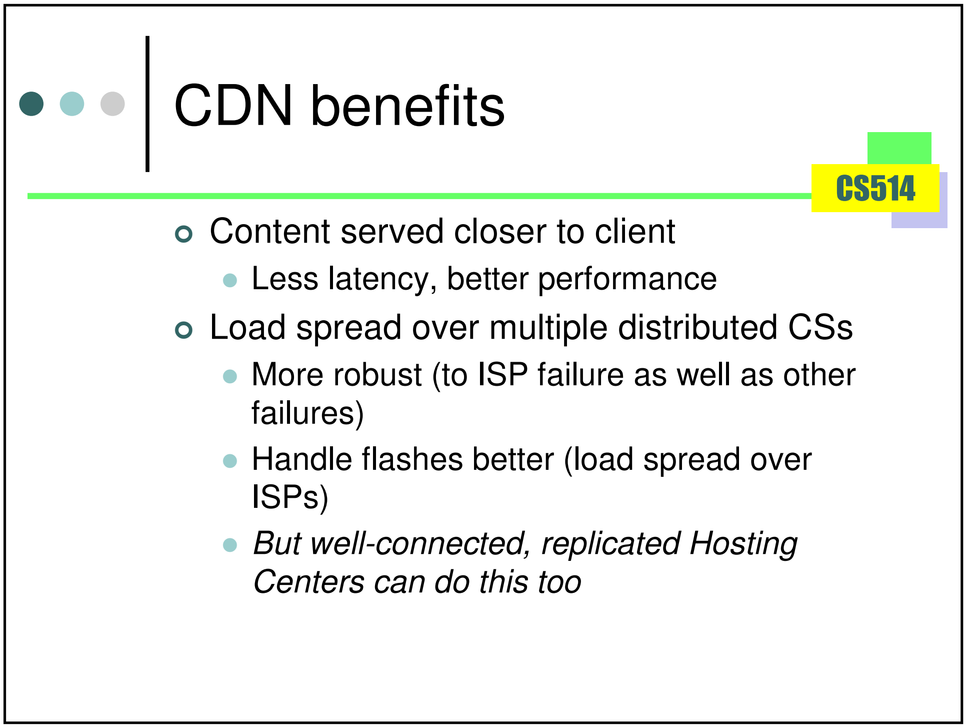
1. Origin Server pushes content out to all CSs.



Pushed CDN

CS514

1. Origin Server pushes content out to all CSs.
2. Request served from CSs.



CDN benefits

CS514

- Content served closer to client
  - Less latency, better performance
- Load spread over multiple distributed CSs
  - More robust (to ISP failure as well as other failures)
  - Handle flashes better (load spread over ISPs)
  - *But well-connected, replicated Hosting Centers can do this too*

CDN costs and limitations

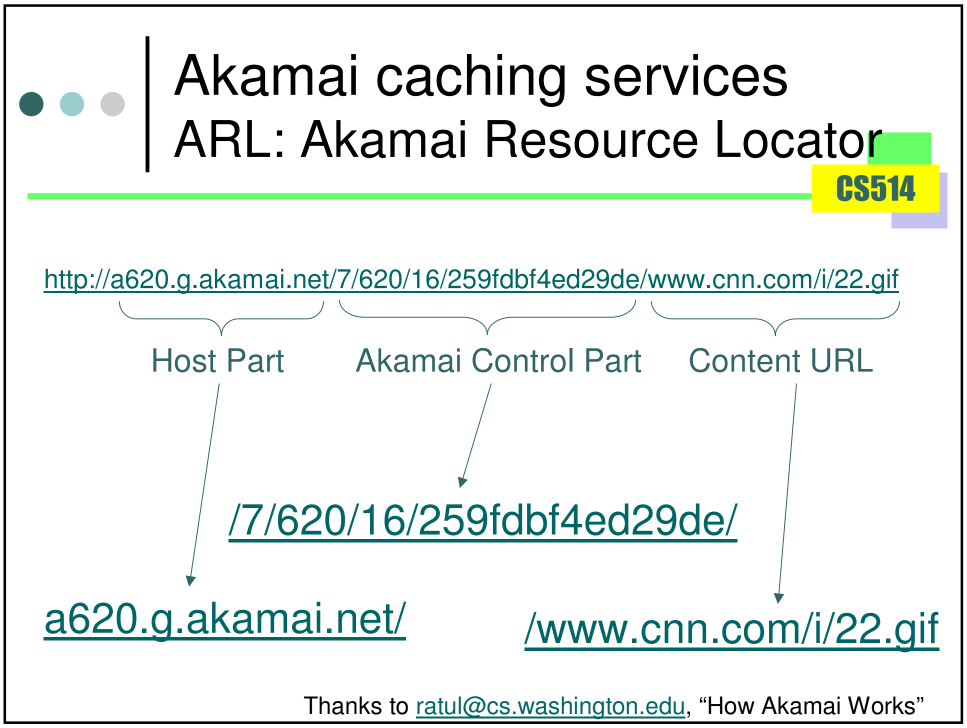
CS514

- Cached CDNs can’t deal with dynamic/personalized content
  - More and more content is dynamic
  - “Classic” CDNs limited to images
- Managing content distribution is non-trivial
  - Tension between content lifetimes and cache performance
  - Dynamic cache invalidation
  - Keeping pushed content synchronized and current

CDN example: Akamai

CS514

- Won huge market share of CDN business late 90’s
- Cached approach
- Now offers full web hosting services in addition to caching services
  - Called edgesuite



Akamai caching services ARL: Akamai Resource Locator

CS514

http://a620.g.akamai.net/7/620/16/259fdbf4ed29de/www.cnn.com/i/22.gif

Host Part

Akamai Control Part

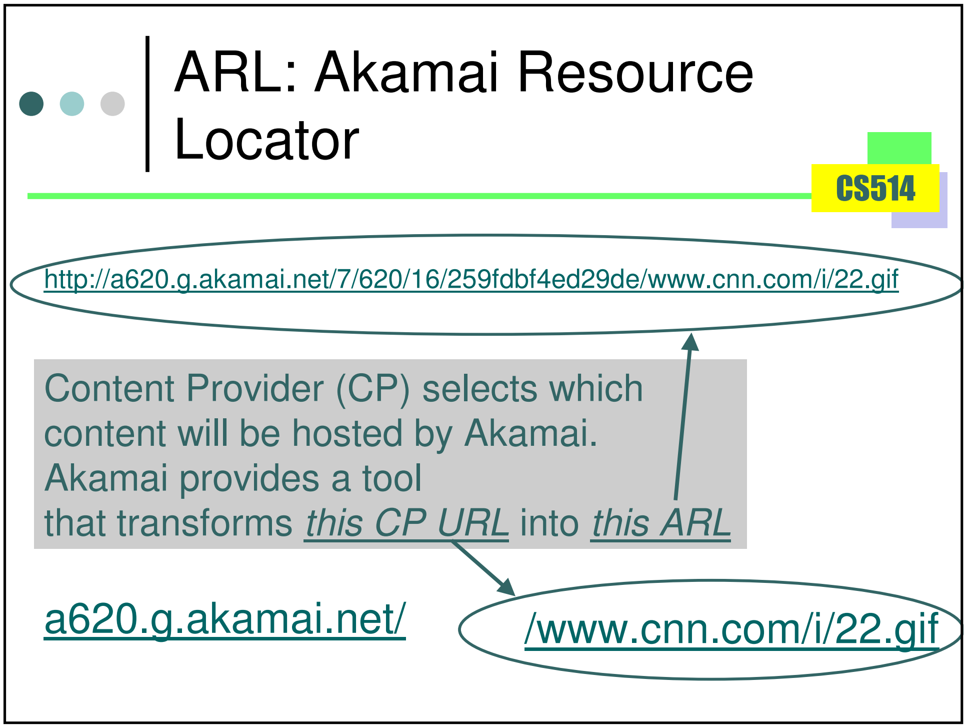
Content URL

/7/620/16/259fdbf4ed29de/

a620.g.akamai.net/

/www.cnn.com/i/22.gif

Thanks to ratul@cs.washington.edu, “How Akamai Works”



ARL: Akamai Resource Locator

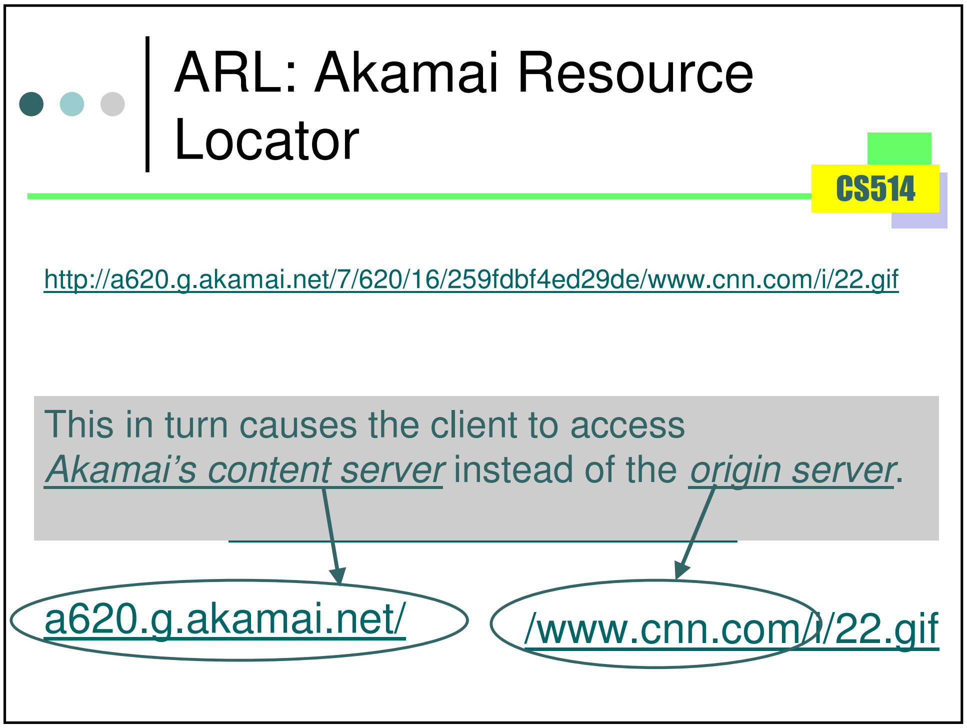
CS514

http://a620.g.akamai.net/7/620/16/259fdbf4ed29de/www.cnn.com/i/22.gif

Content Provider (CP) selects which content will be hosted by Akamai. Akamai provides a tool that transforms *this CP URL* into *this ARL*

a620.g.akamai.net/

/www.cnn.com/i/22.gif



ARL: Akamai Resource Locator

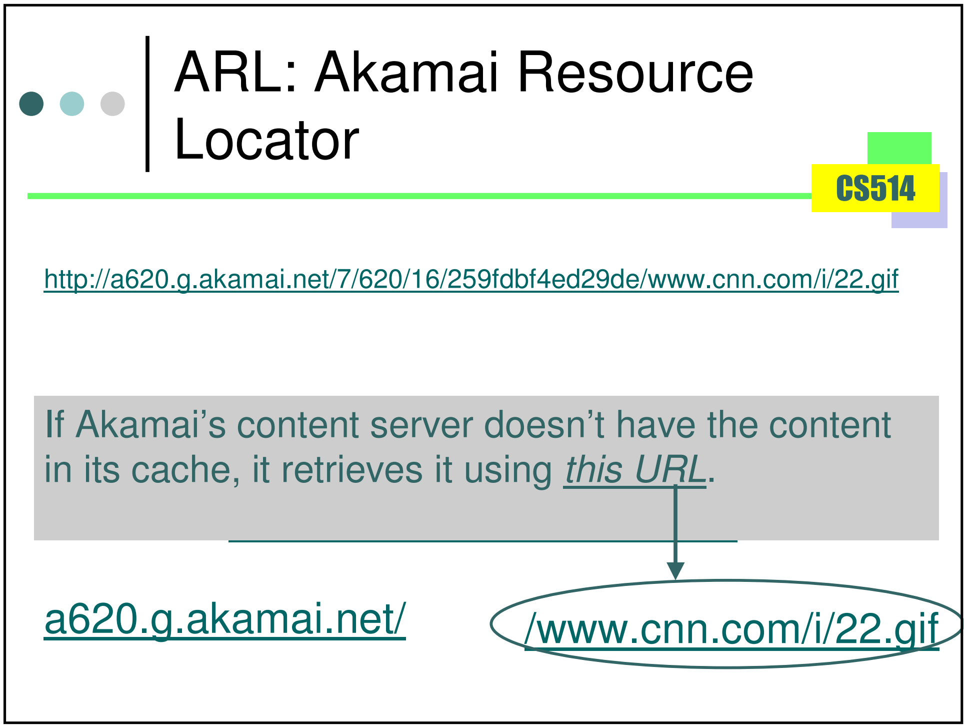
CS514

http://a620.g.akamai.net/7/620/16/259fdbf4ed29de/www.cnn.com/i/22.gif

This in turn causes the client to access *Akamai's content server* instead of the *origin server*.

a620.g.akamai.net/

/www.cnn.com/i/22.gif



ARL: Akamai Resource Locator

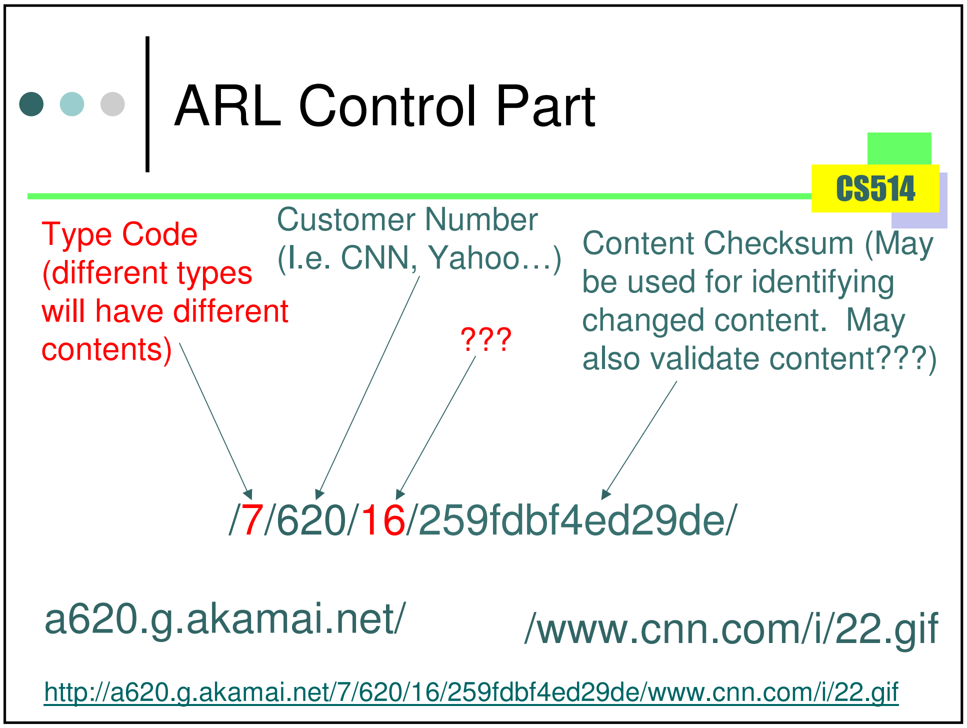
CS514

http://a620.g.akamai.net/7/620/16/259fdbf4ed29de/www.cnn.com/i/22.gif

If Akamai's content server doesn't have the content in its cache, it retrieves it using *this URL*.

a620.g.akamai.net/

/www.cnn.com/i/22.gif



ARL Control Part

CS514

Type Code (different types will have different contents)

Customer Number (I.e. CNN, Yahoo…)

???

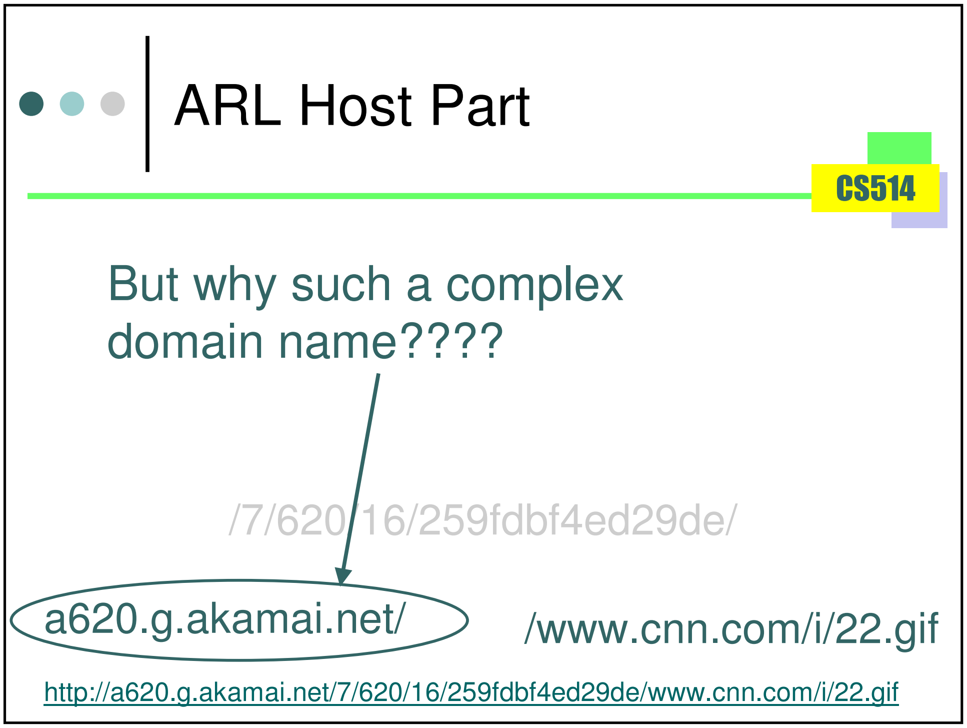
Content Checksum (May be used for identifying changed content. May also validate content???)

/7/620/16/259fdbf4ed29de/

a620.g.akamai.net/

/www.cnn.com/i/22.gif

http://a620.g.akamai.net/7/620/16/259fdbf4ed29de/www.cnn.com/i/22.gif



ARL Host Part

CS514

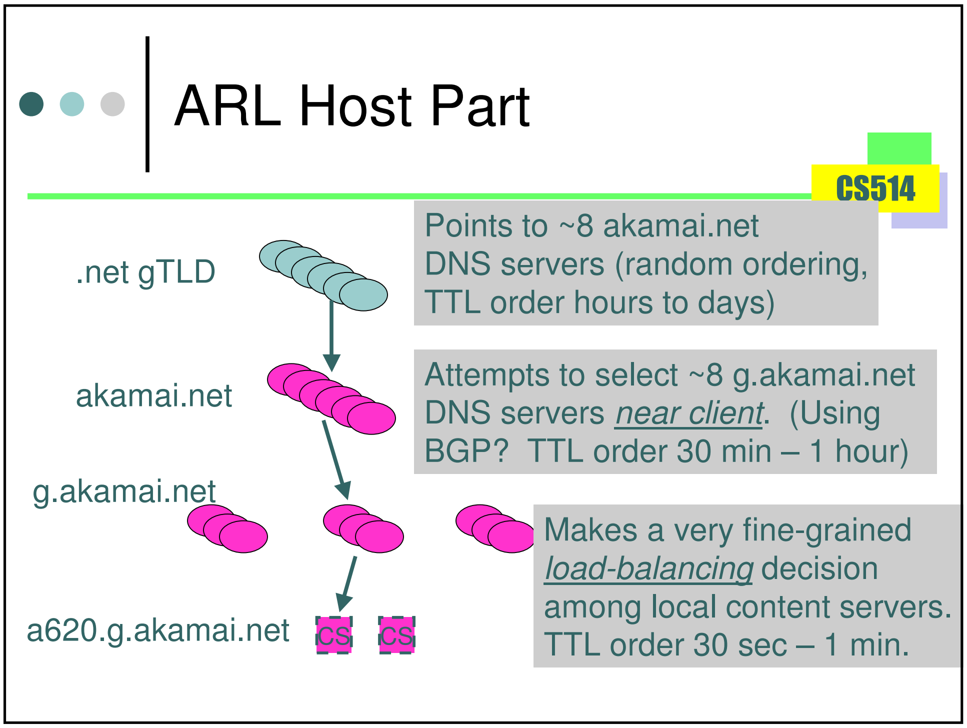
But why such a complex domain name????

/7/620/16/259fdbf4ed29de/

a620.g.akamai.net/

/www.cnn.com/i/22.gif

http://a620.g.akamai.net/7/620/16/259fdbf4ed29de/www.cnn.com/i/22.gif



ARL Host Part

CS514

.net gTLD

Points to ~8 akamai.net DNS servers (random ordering, TTL order hours to days)

akamai.net

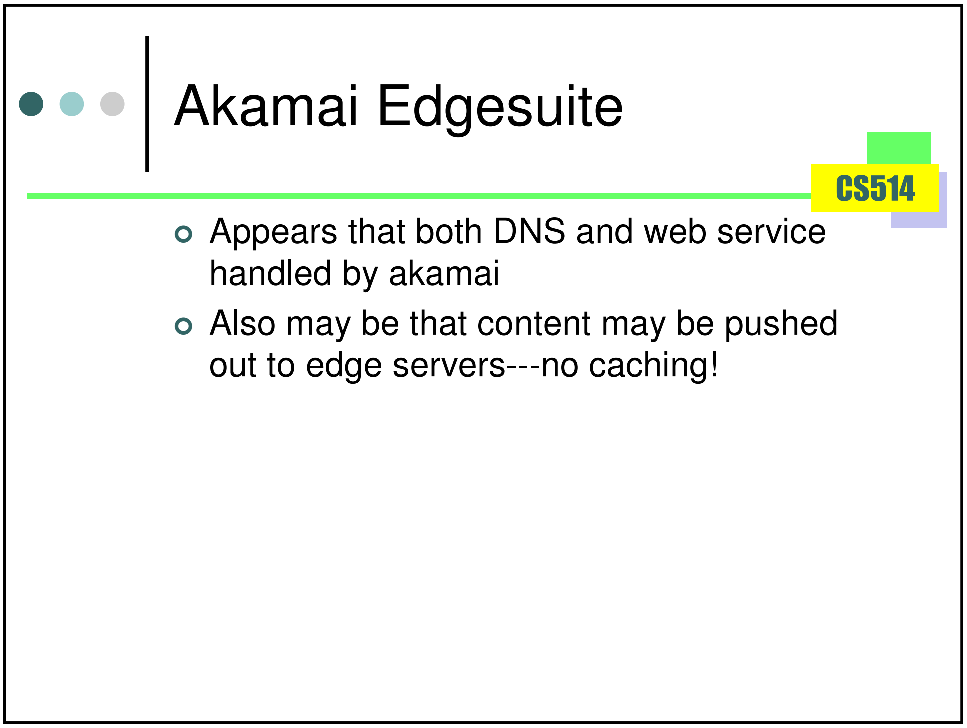
Attempts to select ~8 g.akamai.net DNS servers *near client*. (Using BGP? TTL order 30 min – 1 hour)

g.akamai.net

Makes a very fine-grained *load-balancing* decision among local content servers. TTL order 30 sec – 1 min.

a620.g.akamai.net

CS CS

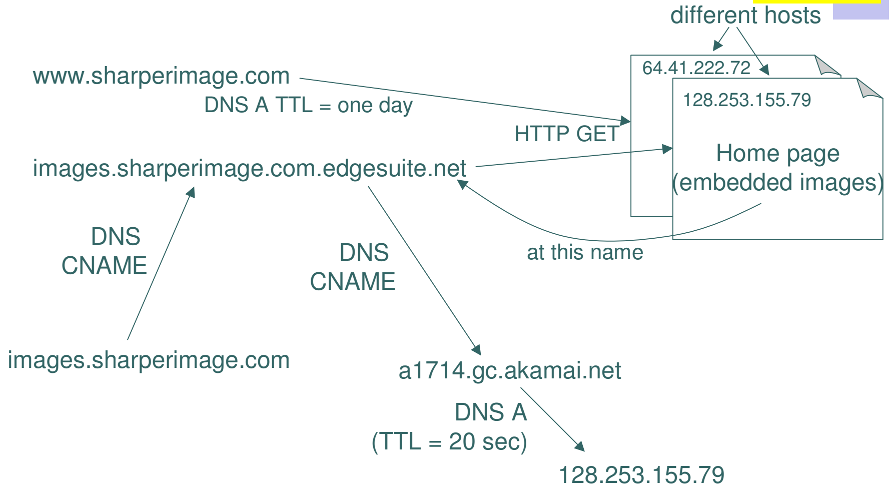


Akamai Edgesuite

CS514

- Appears that both DNS and web service handled by akamai
- Also may be that content may be pushed out to edge servers---no caching!

Sharper Image and Edgesuite



different hosts

64.41.222.72

128.253.155.79

www.sharperimage.com

DNS A TTL = one day

HTTP GET

images.sharperimage.com.edgesuite.net

Home page (embedded images)

at this name

DNS CNAME

DNS CNAME

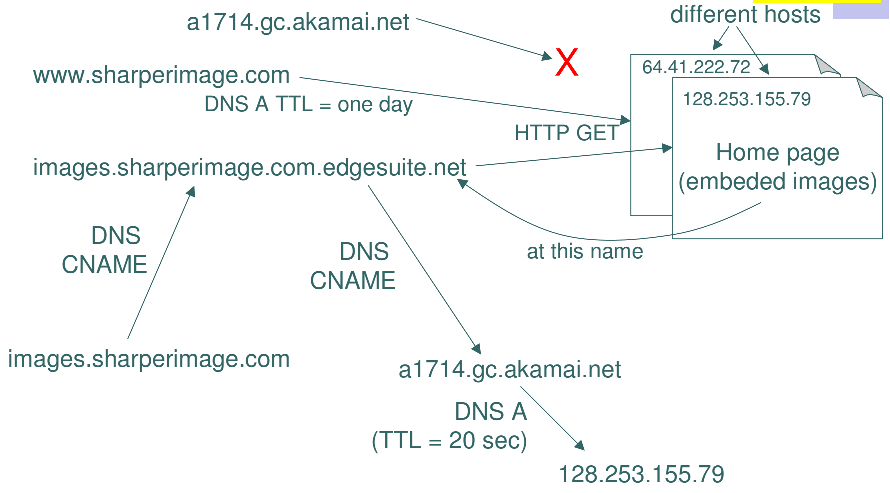
images.sharperimage.com

a1714.gc.akamai.net

DNS A (TTL = 20 sec)

128.253.155.79

Sharper Image and Edgesuite



a1714.gc.akamai.net

X

different hosts

64.41.222.72

128.253.155.79

www.sharperimage.com

DNS A TTL = one day

HTTP GET

images.sharperimage.com.edgesuite.net

Home page (embeded images)

at this name

DNS CNAME

DNS CNAME

images.sharperimage.com

a1714.gc.akamai.net

DNS A (TTL = 20 sec)

128.253.155.79

What may be happening…

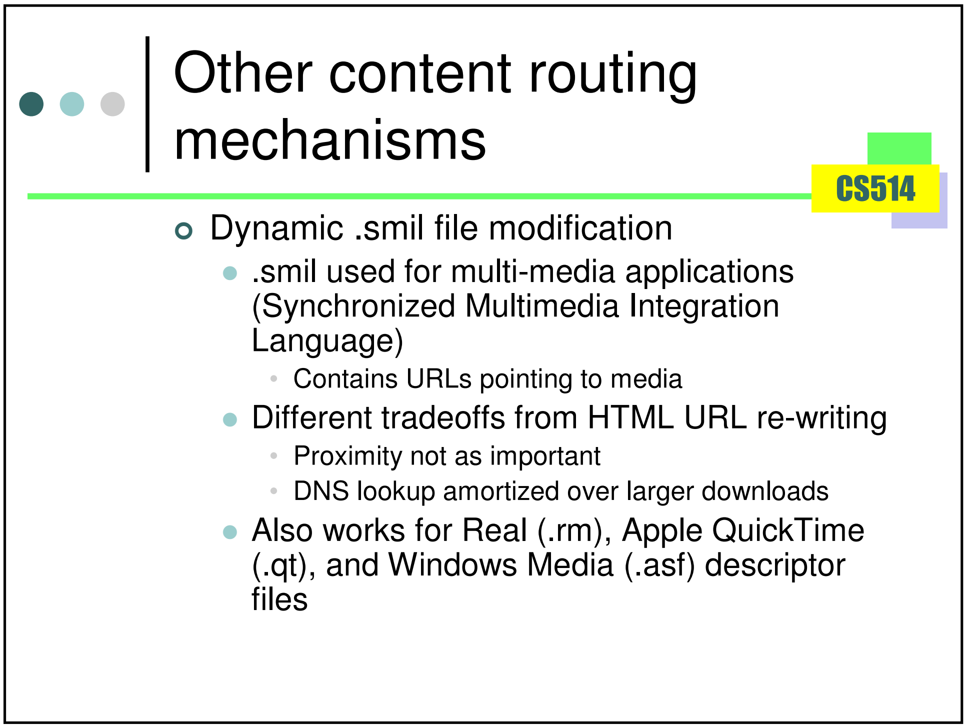
CS514

- images.sharperimage.com.edgesuite.net returns same pages as www.sharperimage.com
  - But the shopping basket doesn't work!!
- Perhaps akamai cache blindly maps foo.bar.com.edgesuite.net into bar.com to retrieve web page
  - No more sophisticated akamaization
  - Easier to maintain origin web server??
  - Simpler akamai web caches??

Other content routing mechanisms

CS514

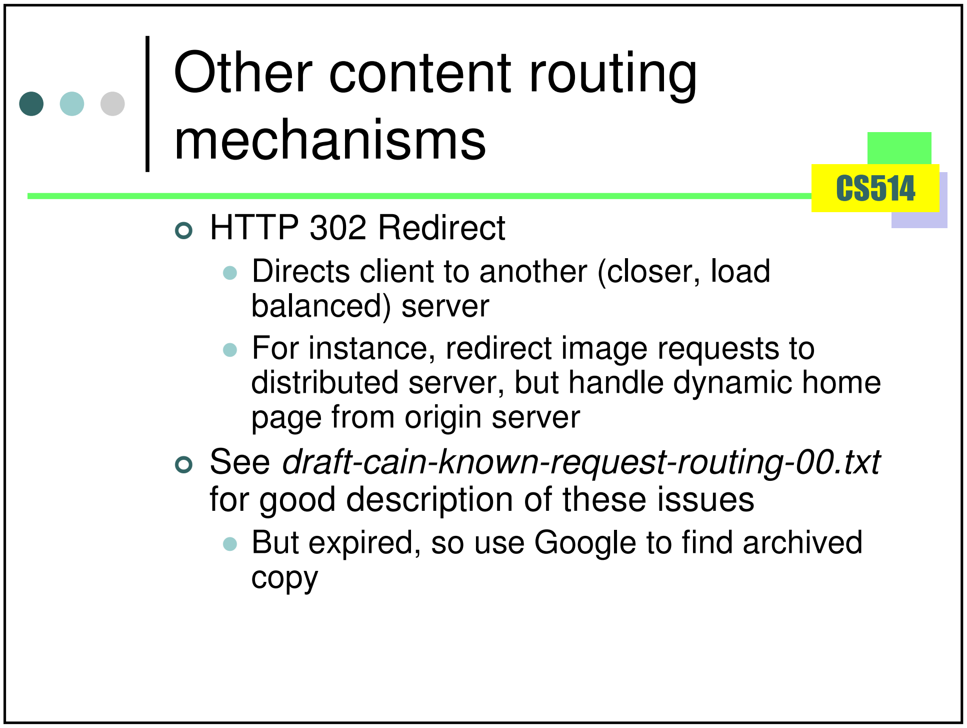
- Dynamic HTML URL re-writing
  - URLs in HTML pages re-written to point at nearby and non-overloaded content server
  - In theory, finer-grained proximity decision
    - Because know true client, not clients DNS resolver
    - In practice very hard to be fine-grained
  - Clearway and Fasttide did this
  - Could in theory put IP address in re-written URL, save a DNS lookup
    - But problem if user bookmarks page



Other content routing mechanisms

CS514

- Dynamic .smil file modification
  - .smil used for multi-media applications (Synchronized Multimedia Integration Language)
    - Contains URLs pointing to media
  - Different tradeoffs from HTML URL re-writing
    - Proximity not as important
    - DNS lookup amortized over larger downloads
  - Also works for Real (.rm), Apple QuickTime (.qt), and Windows Media (.asf) descriptor files



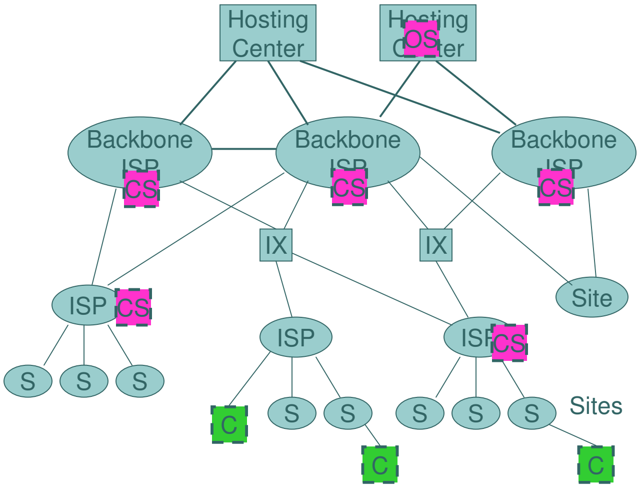
Other content routing mechanisms

CS514

- HTTP 302 Redirect
  - Directs client to another (closer, load balanced) server
  - For instance, redirect image requests to distributed server, but handle dynamic home page from origin server
- See *draft-cain-known-request-routing-00.txt* for good description of these issues
  - But expired, so use Google to find archived copy

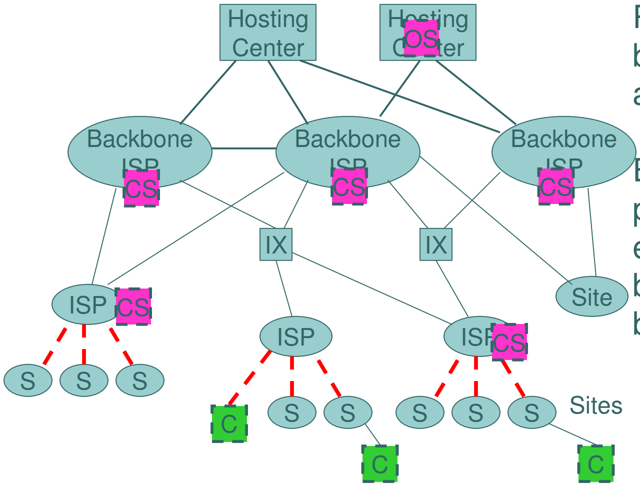
How well do CDNs work?

CS514



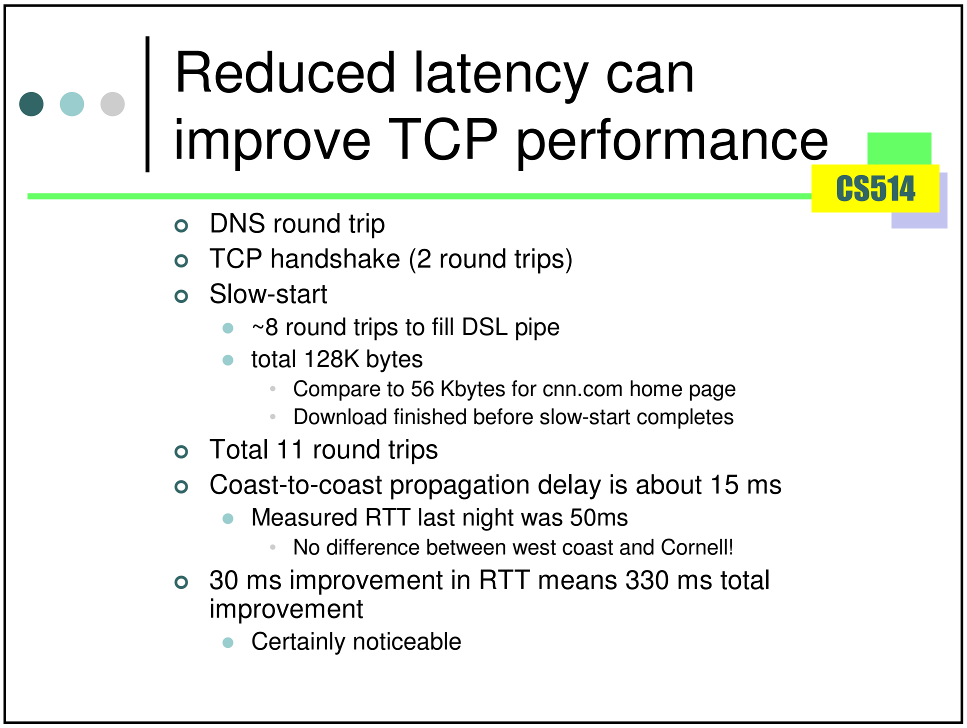
How well do CDNs work?

CS514



Recall that the bottleneck links are at the edges.

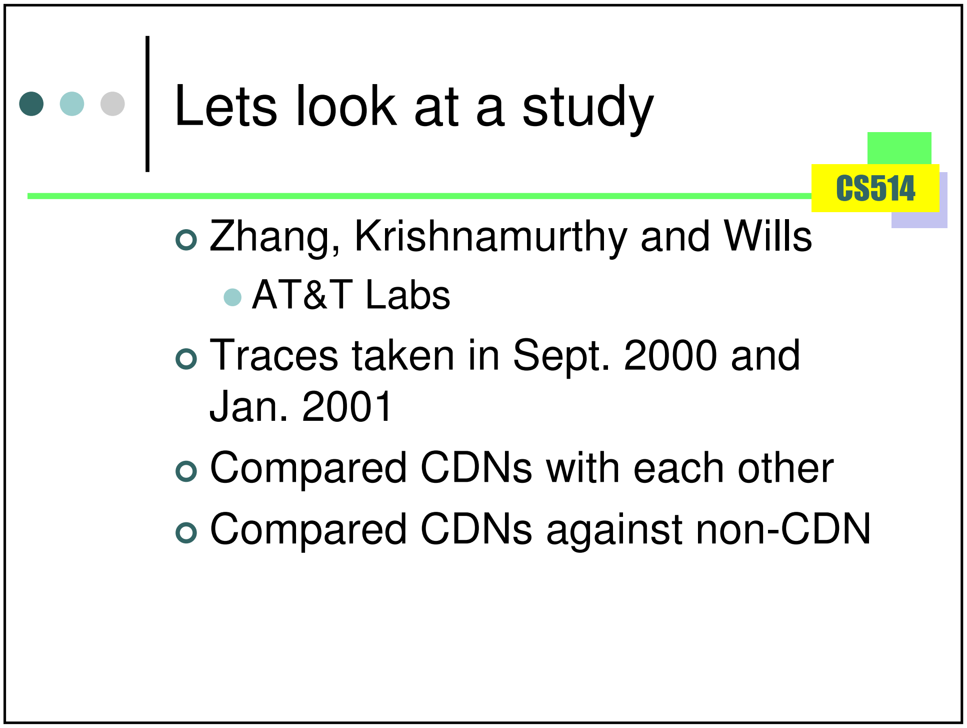
Even if CSs are pushed towards the edge, they are still behind the bottleneck link!



Reduced latency can improve TCP performance

CS514

- DNS round trip
- TCP handshake (2 round trips)
- Slow-start
  - ~8 round trips to fill DSL pipe
  - total 128K bytes
    - Compare to 56 Kbytes for cnn.com home page
    - Download finished before slow-start completes
- Total 11 round trips
- Coast-to-coast propagation delay is about 15 ms
  - Measured RTT last night was 50ms
    - No difference between west coast and Cornell!
- 30 ms improvement in RTT means 330 ms total improvement
  - Certainly noticeable



Lets look at a study

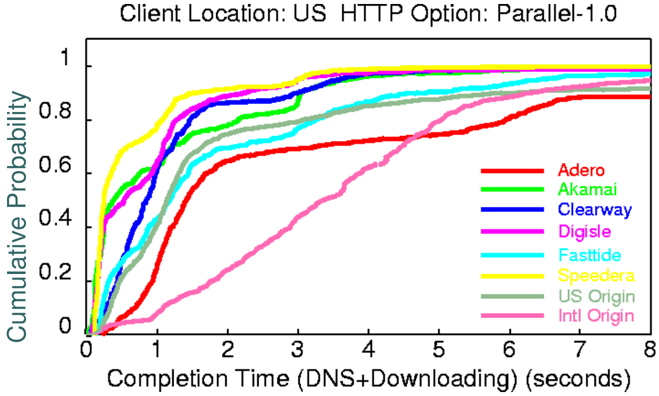
CS514

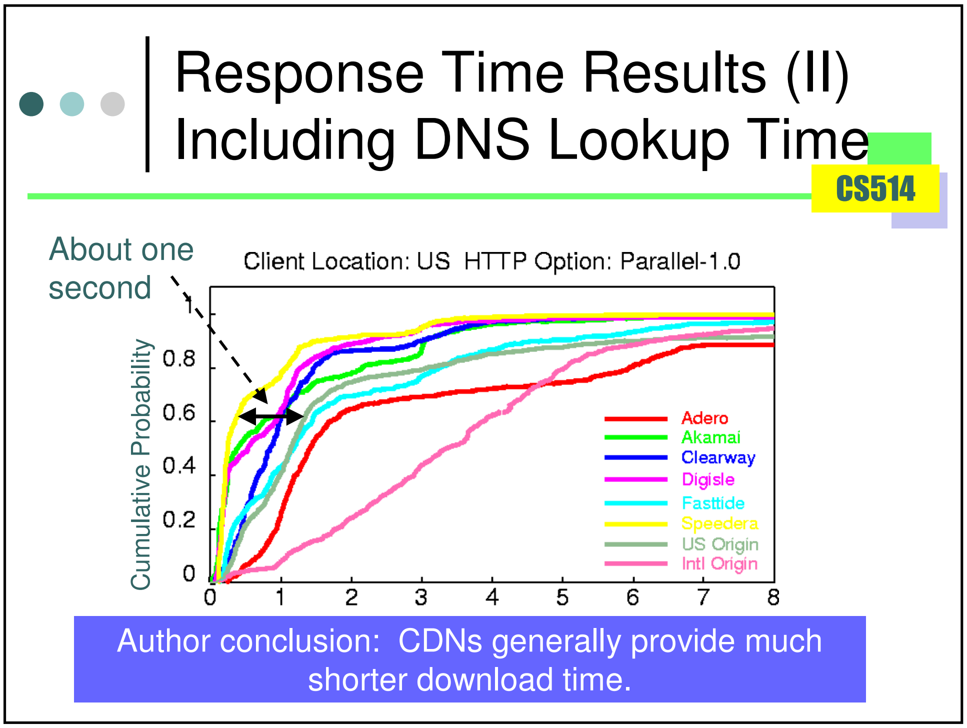
- Zhang, Krishnamurthy and Wills
  - AT&T Labs
- Traces taken in Sept. 2000 and Jan. 2001
- Compared CDNs with each other
- Compared CDNs against non-CDN

Methodology

- Selected a bunch of CDNs
  - Akamai, Speedera, Digital Island
    - Note, most of these gone now!
- Selected a number of non-CDN sites for which good performance could be expected
  - U.S. and international origin
  - U.S.: Amazon, Bloomberg, CNN, ESPN, MTV, NASA, Playboy, Sony, Yahoo
- Selected a set of images of comparable size for each CDN and non-CDN site
  - Compare apples to apples
- Downloaded images from 24 NIMI machines

Response Time Results (II) Including DNS Lookup Time





Response Time Results (II) Including DNS Lookup Time

CS514

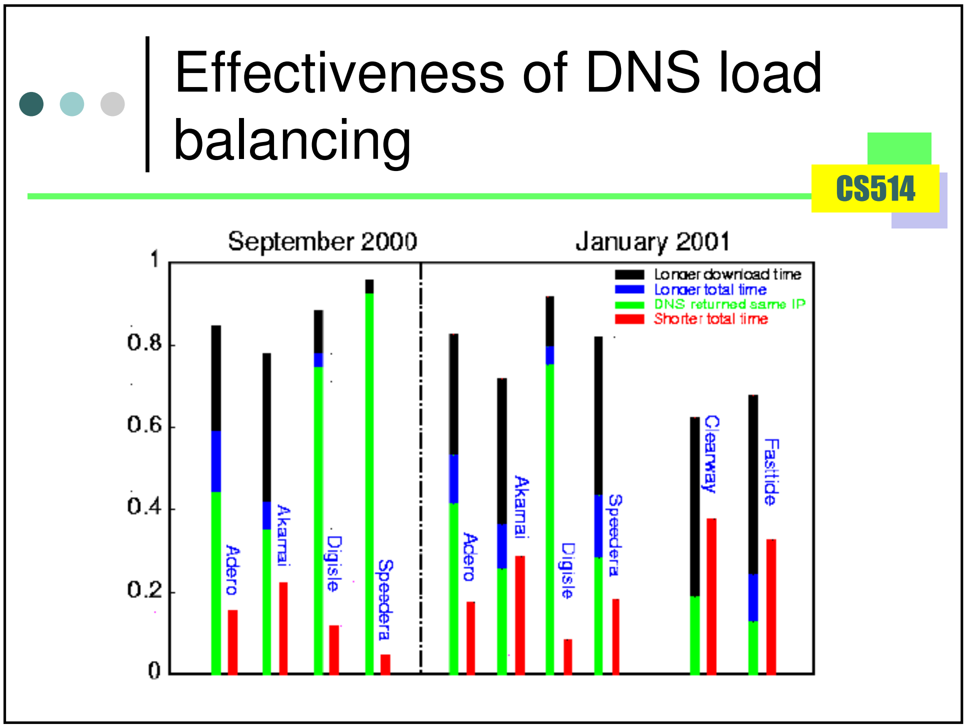
About one second

Author conclusion: CDNs generally provide much shorter download time.

CDNs out-performed non-CDNs

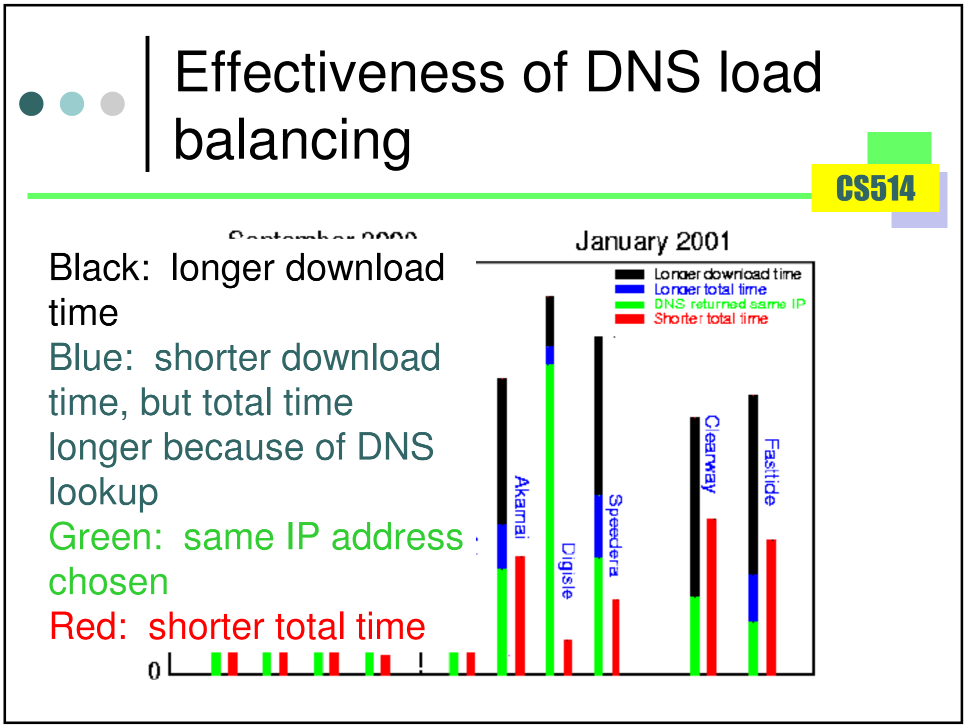
CS514

- Why is this?
- Lets consider ability to pick good content servers…
- They compared time to download with a fixed IP address versus the IP address dynamically selected by the CDN for each download
  - Recall: short DNS TTLs



Effectiveness of DNS load balancing

CS514



Effectiveness of DNS load balancing

CS514

Black: longer download time

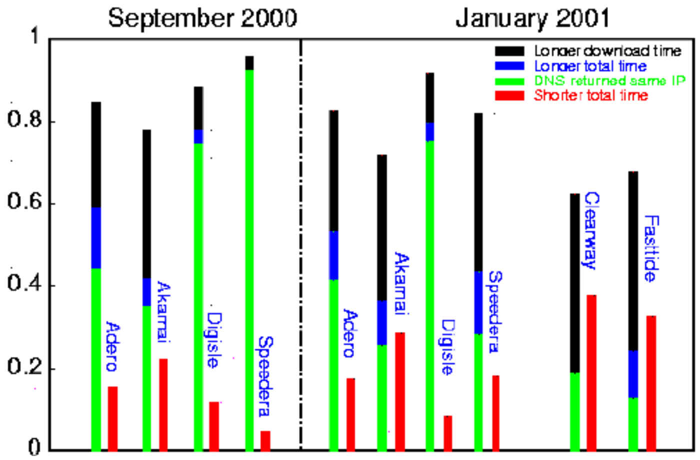
Blue: shorter download time, but total time longer because of DNS lookup

Green: same IP address chosen

Red: shorter total time

DNS load balancing not very effective

CS514



Other findings of study

CS514

- Each CDN performed best for at least one (NIMI) client
  - Why? Because of proximity?
- The best origin sites were better than the worst CDNs
- CDNs with more servers don’t necessarily perform better
  - Note that they don’t know load on servers…
- HTTP 1.1 improvements (parallel download, pipelined download) help a lot
  - Even more so for origin (non-CDN) cases
  - Note not all origin sites implement pipelining

Ultimately a frustrating study

CS514

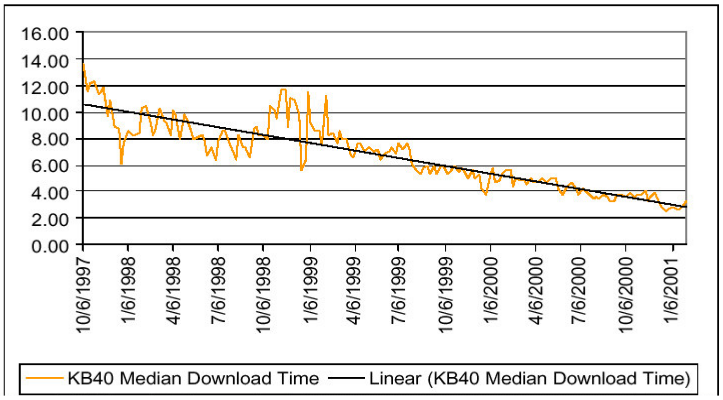
- Never actually says *why* CDNs perform better, only that they do
- For all we know, maybe it is because CDNs threw more money at the problem
  - More server capacity and bandwidth relative to load

Another study

CS514

- Keynote Systems
  - “A Performance Analysis of 40 e-Business Web Sites”
- Doing measurements since 1997
  - (All from one location, near as I can tell)
- Latest measurement January 2001

Historical trend: Clear improvement



Performance breakdown

Basically says that smaller content leads to shorter download times (duh!)

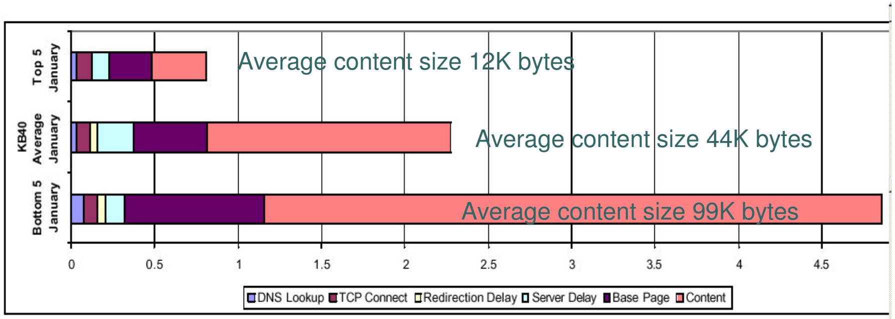
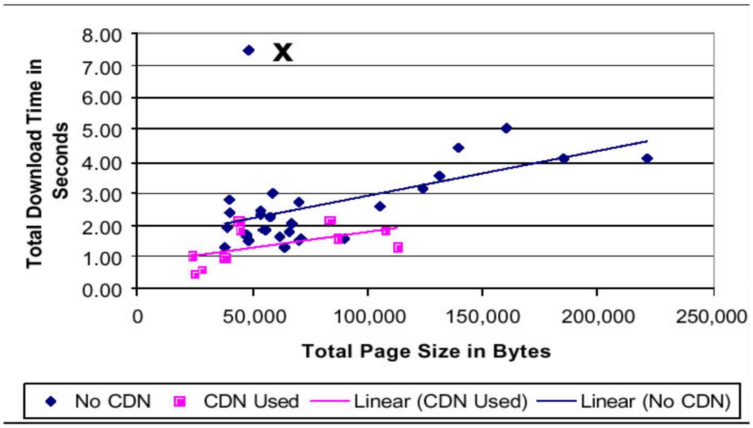
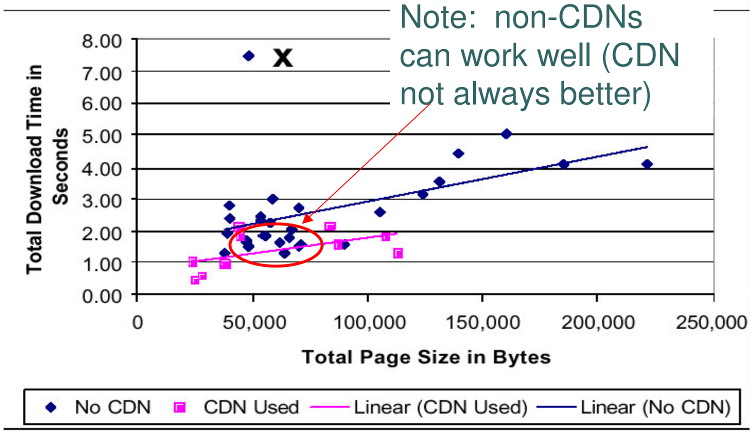


Figure 3. Download Time Components for Top 5, Average Site, and Bottom 5 in January 2001

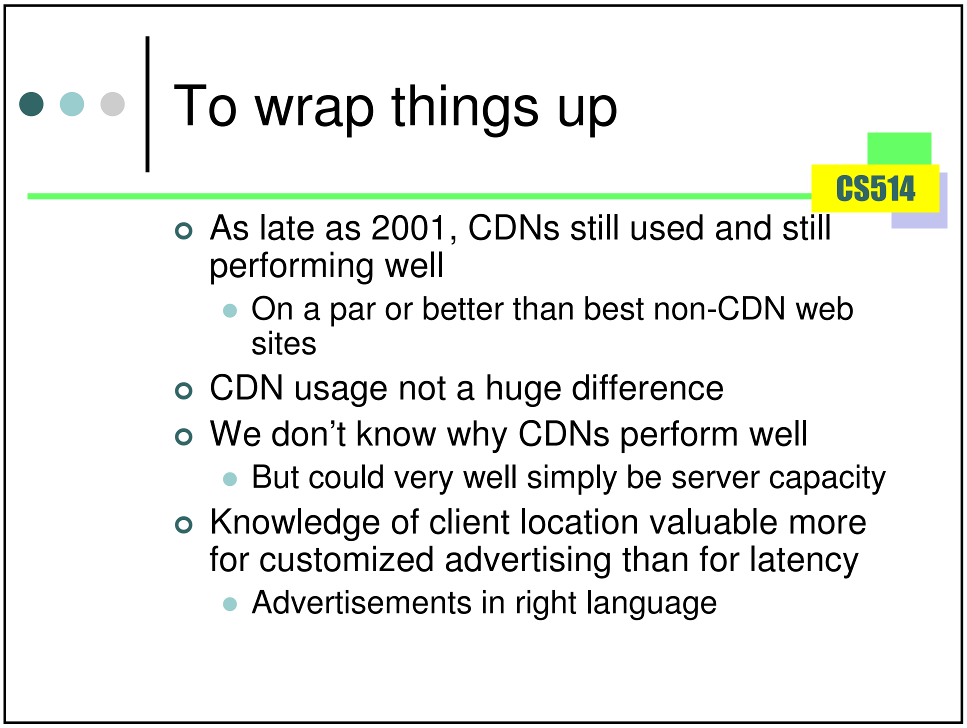
Effect of CDN: Positive
(but again, we don’t know why)



Most web sites not using CDN (4-1)



Note: non-CDNs can work well (CDN not always better)



To wrap things up

- As late as 2001, CDNs still used and still performing well
  - On a par or better than best non-CDN web sites
- CDN usage not a huge difference
- We don't know why CDNs perform well
  - But could very well simply be server capacity
- Knowledge of client location valuable more for customized advertising than for latency
  - Advertisements in right language