

Content Distribution Networks

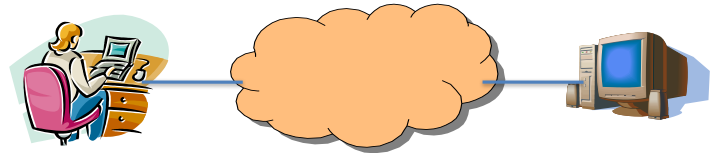


COS 518: Advanced Computer Systems

Lecture 16

Mike Freedman

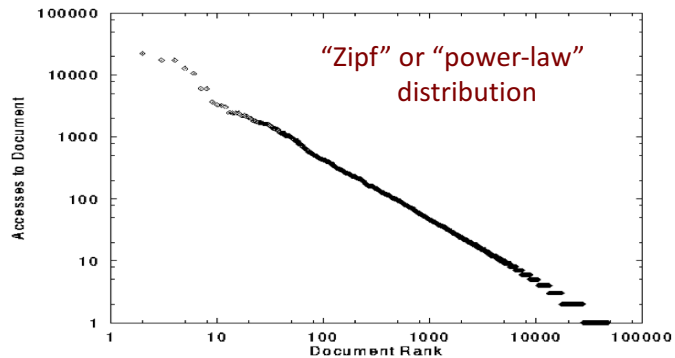
Single Server, Poor Performance



- **Single server**
 - Single point of failure
 - Easily overloaded
 - Far from most clients
- **Popular content**
 - Popular site
 - “Flash crowd”
 - Denial of Service attack

2

Skewed Popularity of Web Traffic



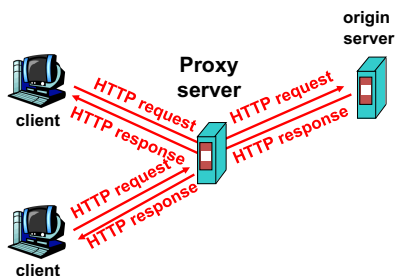
Characteristics of WWW Client-based Traces
Carlos R. Cunha, Azer Bestavros, Mark E. Crovella, BU-CS-95-01

3

Web Caching

4

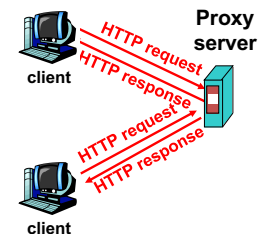
Proxy Caches



5

Forward Proxy

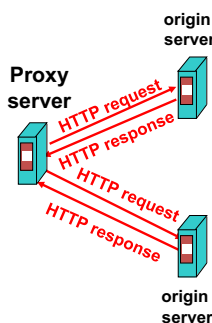
- Cache “close” to the client
 - Under administrative control of client-side AS
- Explicit proxy
 - Requires configuring browser
- Implicit proxy
 - Service provider deploys an “on path” proxy
 - ... that intercepts and handles Web requests



6

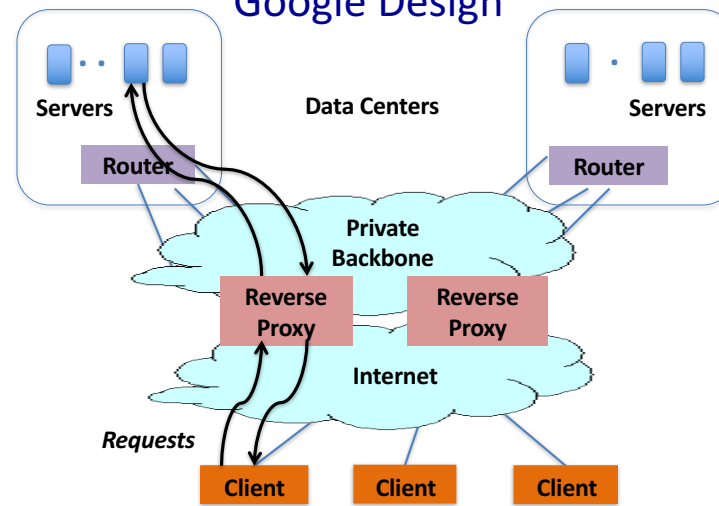
Reverse Proxy

- Cache “close” to server
 - Either by proxy run by server or in third-party CDNs
- Directing clients to the proxy
 - Map the site name to the IP address of the proxy



7

Google Design



8

Proxy Caches

(A) Forward (B) Reverse (C) Both (D) Neither

- Reactively replicates popular content
- Reduces origin server costs
- Reduces client ISP costs
- Intelligent load balancing between origin servers
- Offload form submissions (POSTs) and user auth
- Content reassembly or transcoding on behalf of origin
- Smaller round-trip times to clients
- Maintain persistent connections to avoid TCP setup delay (handshake, slow start)

9

Proxy Caches

(A) Forward (B) Reverse (C) Both (D) Neither

- Reactively replicates popular content (C)
- Reduces origin server costs (C)
- Reduces client ISP costs (A)
- Intelligent load balancing between origin servers (B)
- Offload form submissions (POSTs) and user auth (D)
- Content reassembly, transcoding on behalf of origin (C)
- Smaller round-trip times to clients (C)
- Maintain persistent connections to avoid TCP setup delay (handshake, slow start) (C)

10

Limitations of Web Caching

- **Much content is not cacheable**
 - Dynamic data: stock prices, scores, web cams
 - CGI scripts: results depend on parameters
 - Cookies: results may depend on passed data
 - SSL: encrypted data is not cacheable
 - Analytics: owner wants to measure hits
- **Stale data**
 - Or, overhead of refreshing the cached data

11

Modern HTTP Video-on-Demand

- **Download “content manifest” from origin server**
- **List of video segments belonging to video**
 - Each segment 1-2 seconds in length
 - Client can know time offset associated with each
 - Standard naming for different video resolutions and formats: e.g., 320dpi, 720dpi, 1040dpi, ...
- **Client downloads video segment (at certain resolution) using standard HTTP request.**
 - HTTP request can be satisfied by cache: it’s a static object
- **Client observes download time vs. segment duration, increases/decreases resolution if appropriate**

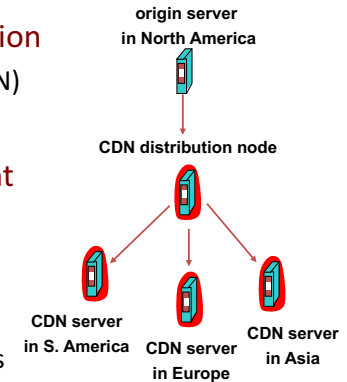
12

Content Distribution Networks

13

Content Distribution Network

- **Proactive content replication**
 - Content provider (e.g., CNN) contracts with a CDN
- **CDN replicates the content**
 - On many servers spread throughout the Internet
- **Updating the replicas**
 - Updates pushed to replicas when the content changes



14

Server Selection Policy

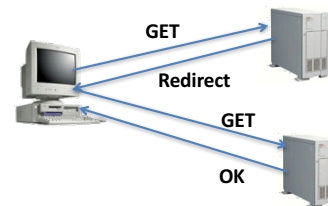
- **Live server**
 - For availability
- **Lowest load**
 - To balance load across the servers
- **Closest**
 - Nearest geographically, or in round-trip time
- **Best performance**
 - Throughput, latency, ...
- **Cheapest bandwidth, electricity, ...**

Requires continuous monitoring of liveness, load, and performance

15

Server Selection Mechanism

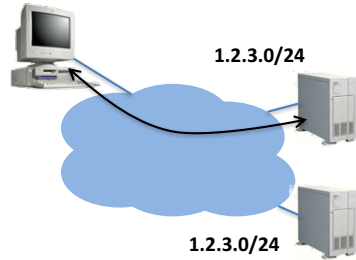
- **Application**
 - HTTP redirection
- **Advantages**
 - Fine-grain control
 - Selection based on client IP address
- **Disadvantages**
 - Extra round-trips for TCP connection to server
 - Overhead on the server



16

Server Selection Mechanism

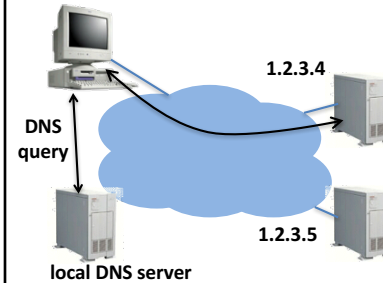
- **Routing**
 - Anycast routing
- **Advantages**
 - No extra round trips
 - Route to nearby server
- **Disadvantages**
 - Does not consider network or server load
 - Different packets may go to different servers
 - Used only for simple request-response apps



17

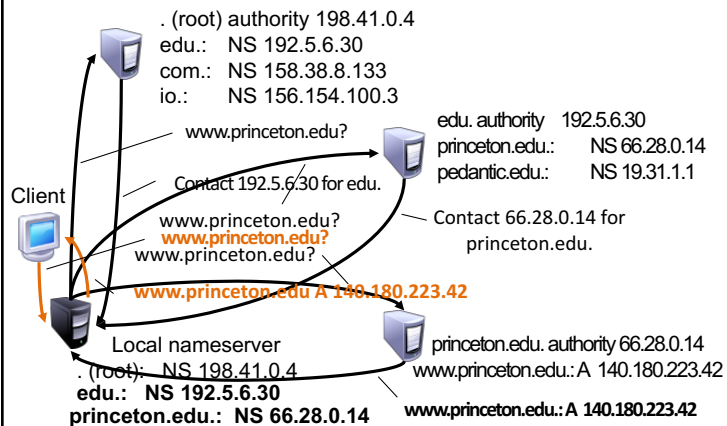
Server Selection Mechanism

- **Naming**
 - DNS-based server selection



18

A DNS lookup traverses DNS hierarchy



19

DNS caching

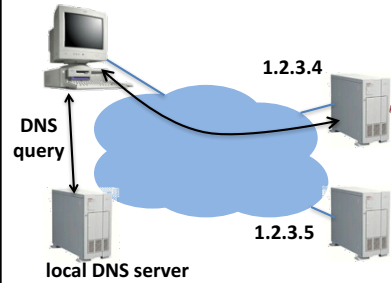
- **Performing all these queries takes time**
 - And all this before actual communication takes place
- **Caching can greatly reduce overhead**
 - Top-level servers very rarely change, popular sites visited often
 - Local DNS server often has information cached
- **How DNS caching works**
 - All DNS servers **cache responses to queries**
 - Responses include a time-to-live (TTL) field, akin to cache expiry

20

Server Selection Mechanism

- **Naming**

- DNS-based server selection



- **Advantages**

- Avoid TCP set-up delay
- DNS caching reduces overhead
- Relatively fine control

- **Disadvantage**

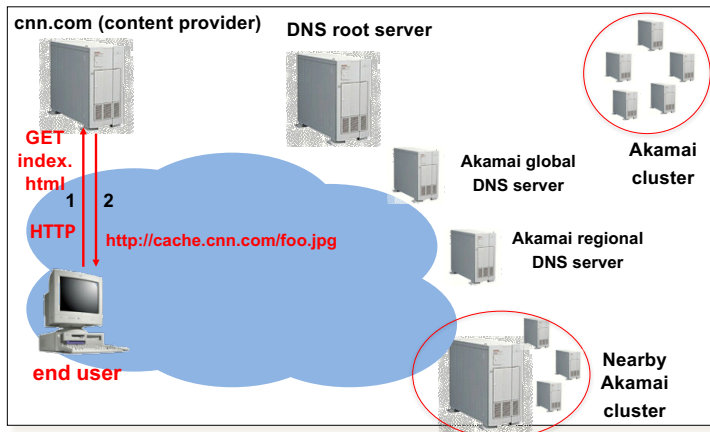
- Based on IP address of local DNS server
- “Hidden load” effect
- DNS TTL limits adaptation

21

How Akamai Works

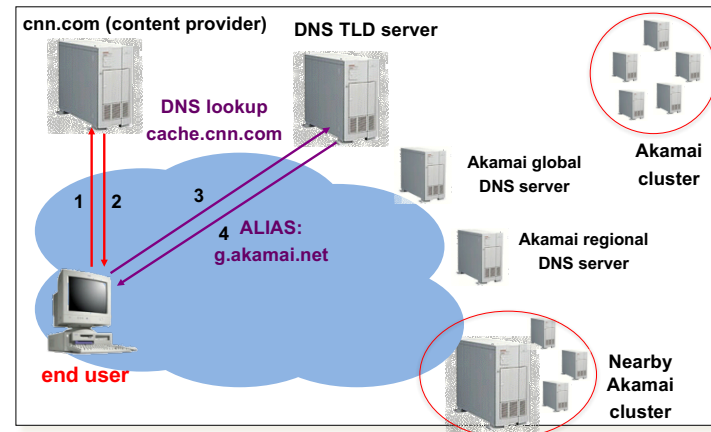
22

How Akamai Uses DNS

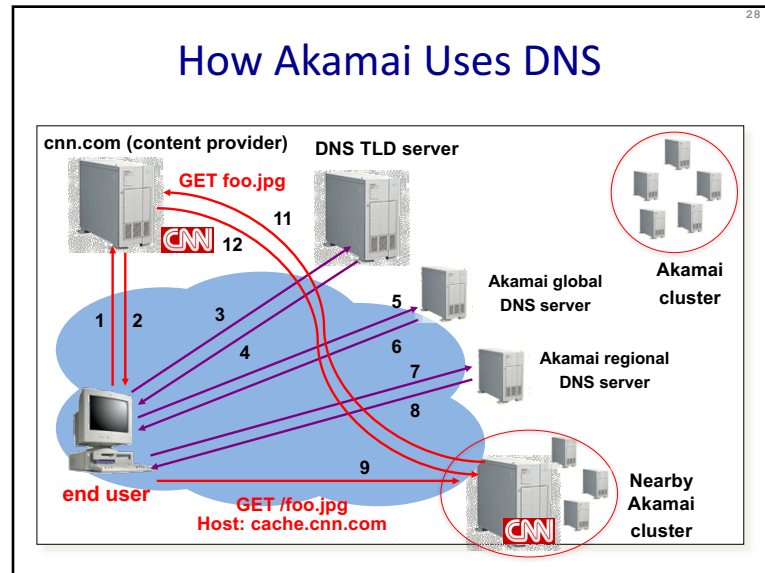
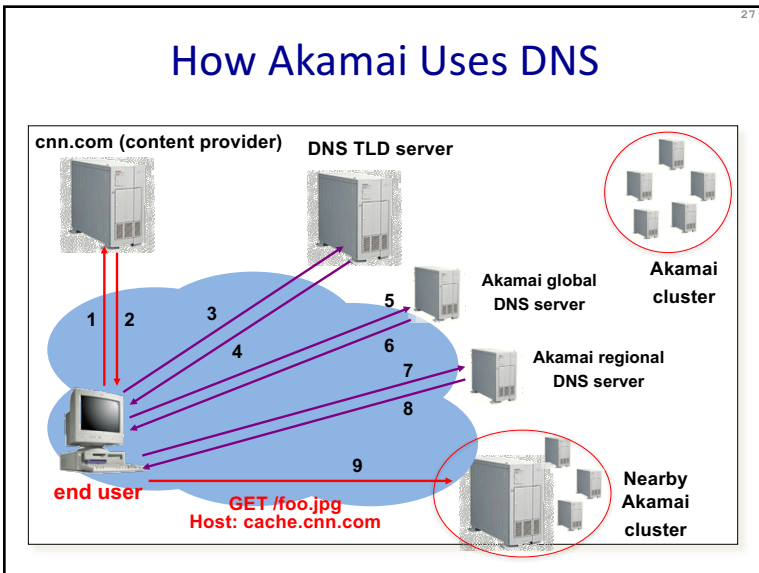
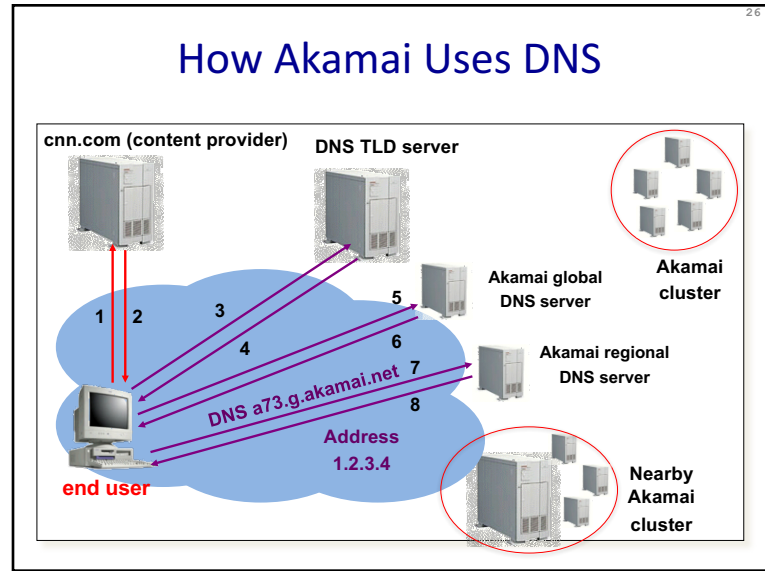
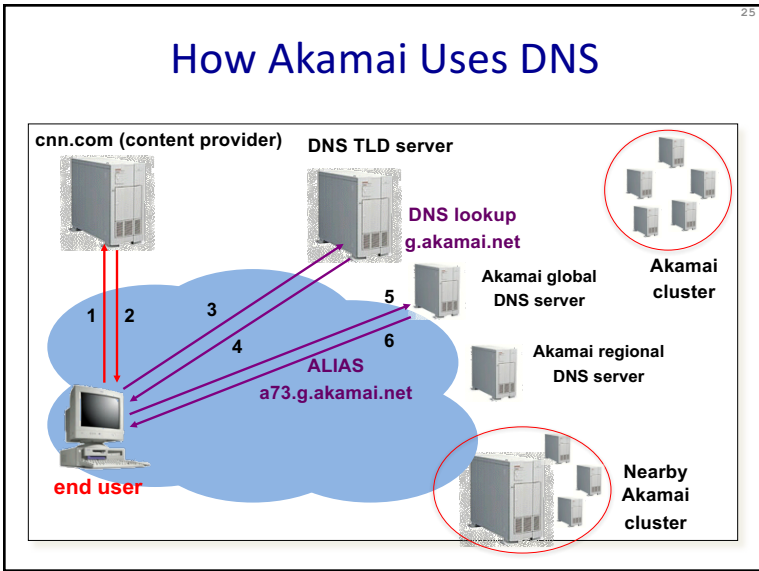


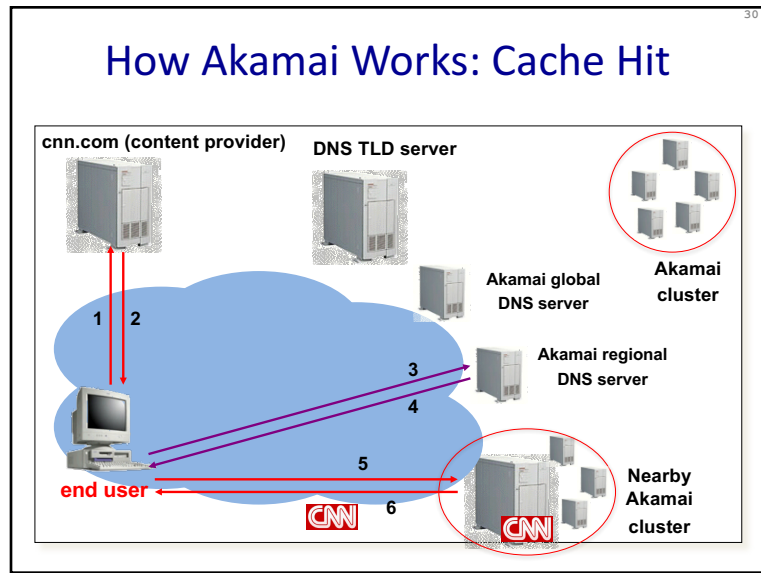
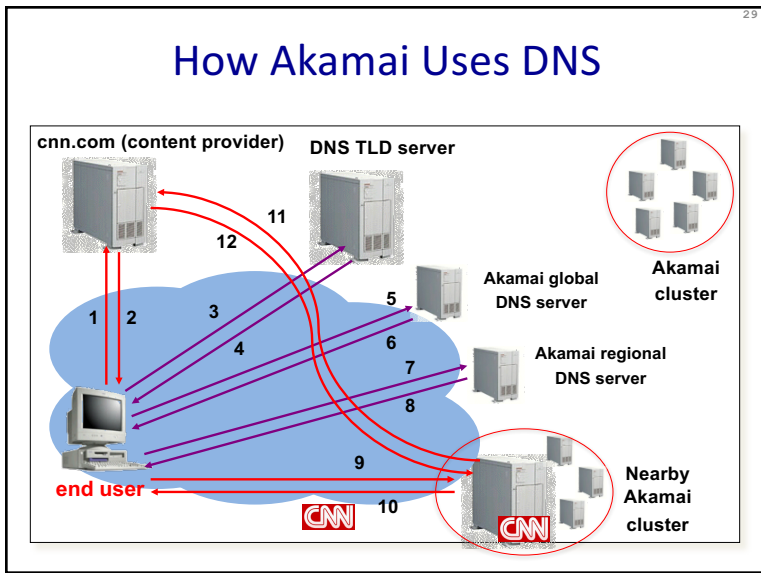
23

How Akamai Uses DNS



24





- ### Mapping System
- **Equivalence classes of IP addresses**
 - IP addresses experiencing similar performance
 - Quantify how well they connect to each other
 - **Collect and combine measurements**
 - Ping, traceroute, BGP routes, server logs
 - E.g., over 100 TB of logs per days
 - Network latency, loss, and connectivity

- ### Mapping System
- **Map each IP class to a preferred server cluster**
 - Based on performance, cluster health, etc.
 - Updated roughly every minute
 - **Map client request to a server in the cluster**
 - Load balancer selects a specific server
 - E.g., to maximize the cache hit rate

Adapting to Failures

- **Failing hard drive on a server**
 - Suspends after finishing “in progress” requests
- **Failed server**
 - Another server takes over for the IP address
 - Low-level map updated quickly
- **Failed cluster**
 - High-level map updated quickly
- **Failed path to customer’s origin server**
 - Route packets through an intermediate node

33

Conclusion

- **Content distribution is hard**
 - Many, diverse, changing objects
 - Clients distributed all over the world
 - Reducing latency is king
- **Contribution distribution solutions**
 - Reactive caching
 - Proactive content distribution networks

34