

CSE 486/586 Distributed Systems

Case Study: Facebook Photo Stores

Steve Ko
Computer Sciences and Engineering
University at Buffalo

Engineering a System

- Generally, when you engineer a system, you need to understand your workload.
 - And design your system according to the workload
 - (Perhaps not in the beginning since there's no workload)
- Engineering principle
 - Make the common case fast, and rare cases correct
 - (From Patterson & Hennessy books)
 - This principle cuts through generations of systems.
- Example?
 - Caching
- Knowing common cases == understanding your workload
 - E.g., read dominated? Write dominated? Mixed?
- We'll look at Facebook's example.

Facebook Workload

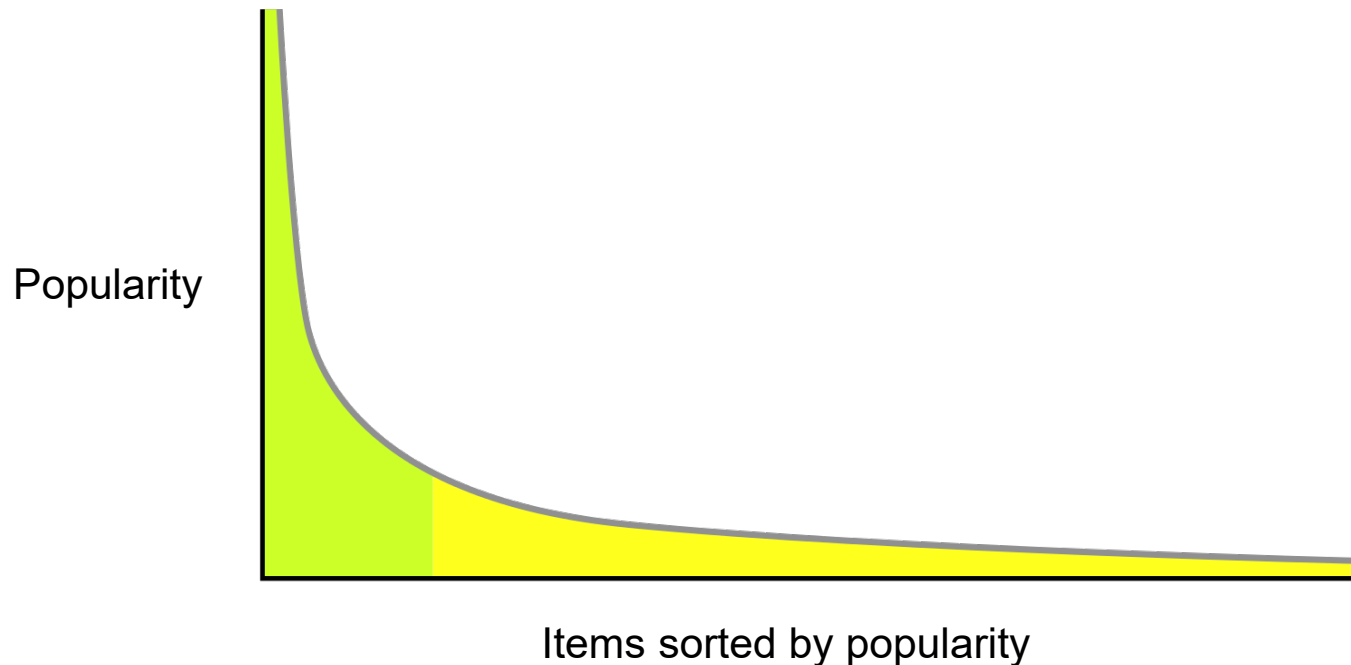
- What are the most frequent things you do on Facebook?
 - Read/write wall posts/comments/likes
 - View/upload photos
 - Very different in their characteristics
- Read/write wall posts/comments/likes
 - Mix of reads and writes so more care is necessary in terms of consistency
 - But small in size so probably less performance sensitive
- Photos
 - Write-once, read-many so less care is necessary in terms of consistency
 - But large in size so more performance sensitive

Facebook Photo Workload

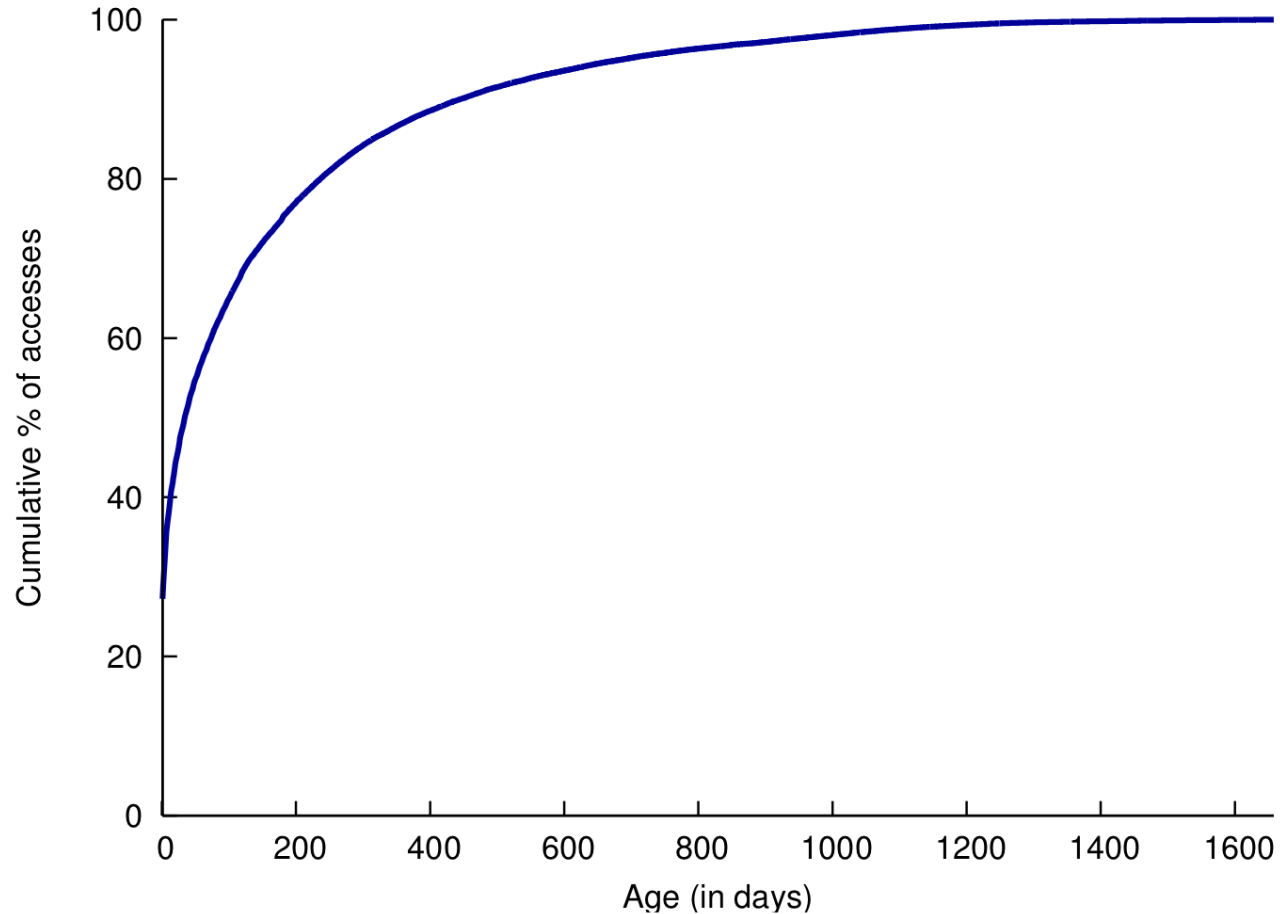
- (This is from 2010.)
- 260 billion images (~20 PB)
- 1 billion new photos per week (~60 TB)
- One million image views per second at peak
- **Two characteristics:** Facebook has analyzed their photo workload and discovered two characteristics.
 - The popularity distribution follows Zipf.
 - Popularity changes over time as photos “age.”

Zipf distribution

- Based on the power law
- Models a lot of natural phenomena
- Social graphs, media popularity, wealth distribution, etc.
- A lot of Web contents too.

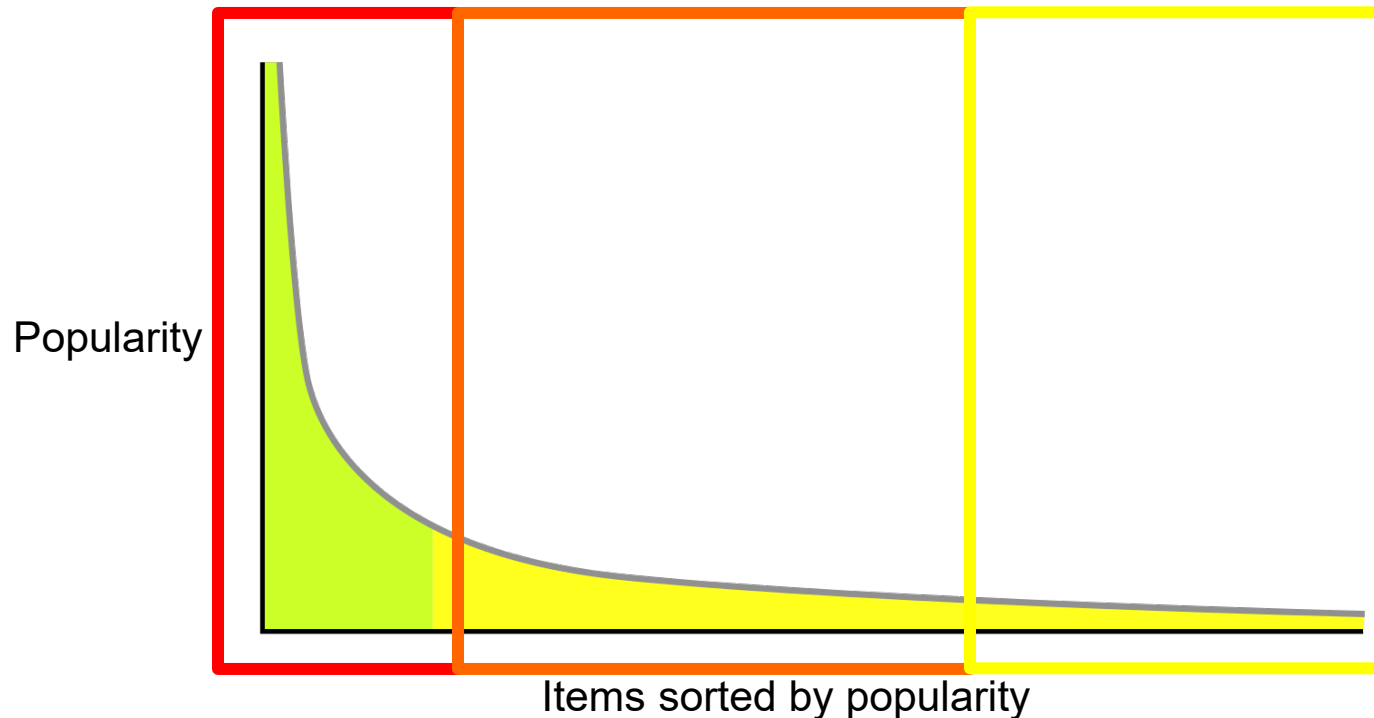


Popularity Comes with Age



Facebook Photo Distribution

- “Hot” vs. “warm” vs. “cold” photos
 - Hot: Popular, a lot of views (approx. 90% of views)
 - Warm: Somewhat popular, but still a lot of views in aggregate
 - Cold: Unpopular, occasional views



Handling Different Types of Photos

- Hot photos
 - Facebook uses a CDN (Content Distribution Network) for these.
 - Very good performance, but no reliability guarantee
 - CDN is a cache, not a permanent storage.
- Warm photos
 - Facebook has designed its own storage called Haystack.
 - Balances performance and reliability
- Cold photos
 - Facebook has designed an “archival” storage called f4.
 - Aims for storage efficiency when storing replicated photos (but not high performance)

CSE 486/586 Administtrivia

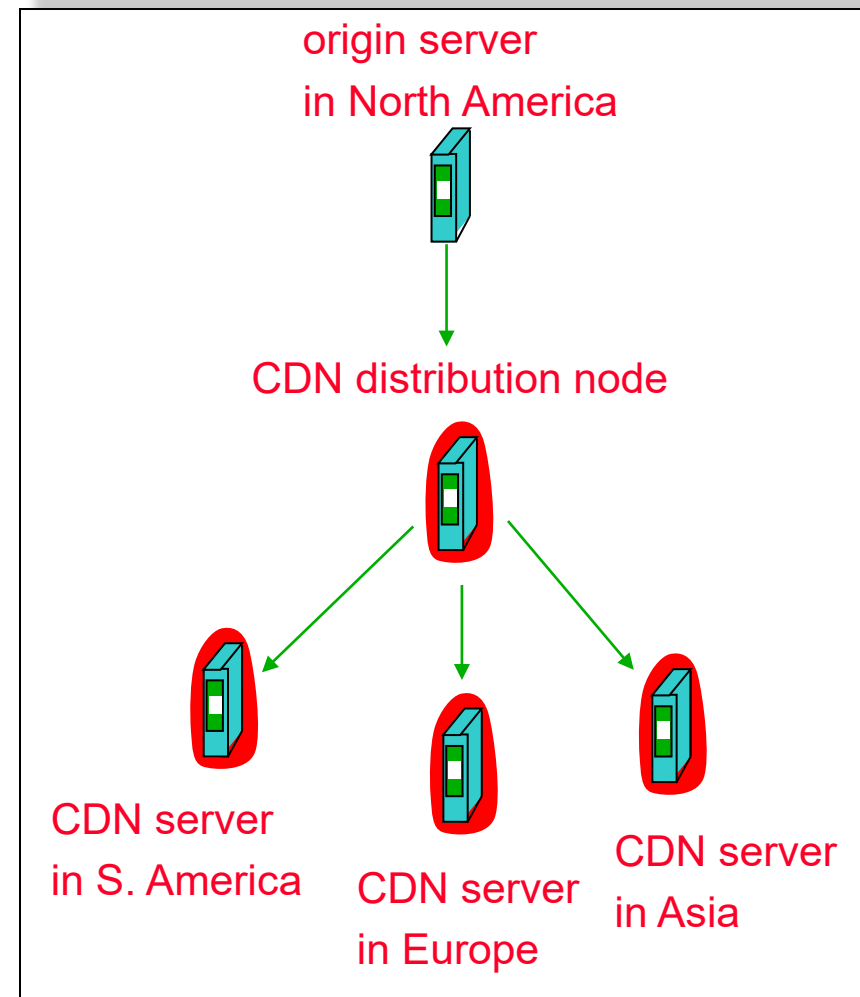
- PA4 deadline: 5/10
- Survey & course evaluation
 - Survey: <https://forms.gle/eg1wHN2G8S6GVz3e9>
 - Course evaluation:
<https://www.smartevals.com/login.aspx?s=buffalo>
- If **both** have 80% or more participation,
 - For each of you, I'll take the better one between the midterm and the final, and give the 30% weight for the better one and the 20% weight for the other one.
 - (Currently, it's 20% for the midterm and 30% for the final.)
- No recitation this week; replaced with office hours

CDN for Hot Photos

- Content providers are CDN customers

Content replication

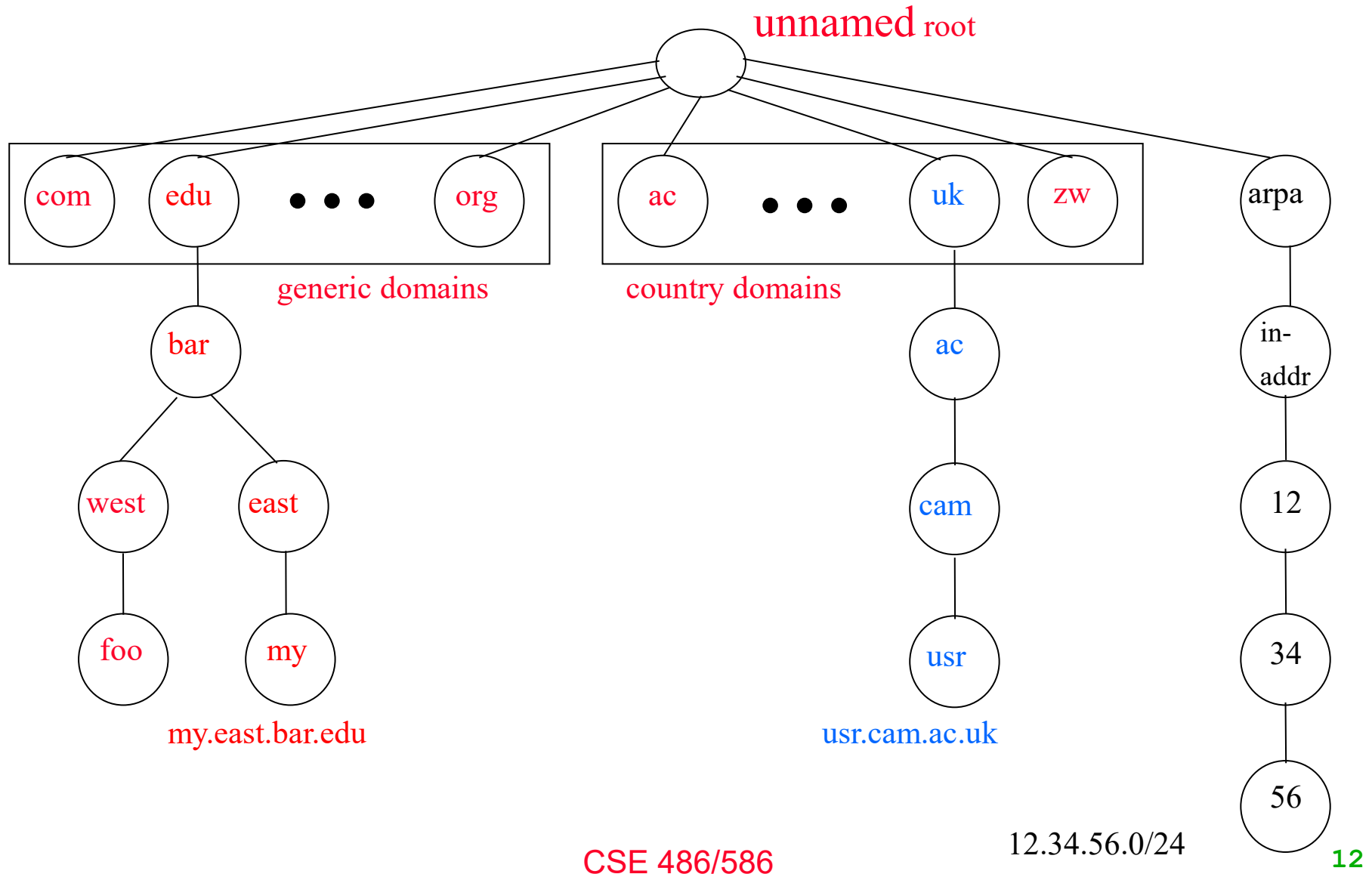
- CDN company (e.g., Akamai) installs thousands of servers throughout Internet
 - In large datacenters close to users
- CDN replicates customers' content
- When provider updates content, CDN updates servers



Domain Name System

- For a given user, how to locate a close server?
- Many CDNs rely on Domain Name System (DNS)
 - DNS maps a DNS name to an **IP address** or another DNS name (**alias**).
 - E.g., **www.cse.buffalo.edu**
 - » **Domain**: registrar for each top-level domain
 - » **Host name**: local administrator assigns to each host
- Properties of DNS
 - Hierarchical name space
 - Distributed over a collection of DNS servers
- Hierarchy of DNS servers
 - Root servers
 - Top-level domain (TLD) servers
 - Authoritative DNS servers

Distributed Hierarchical Database



DNS Root Servers

- 1088 instances operated by the 12 independent root server operators (see <http://www.root-servers.org/>)
- Labeled A through M

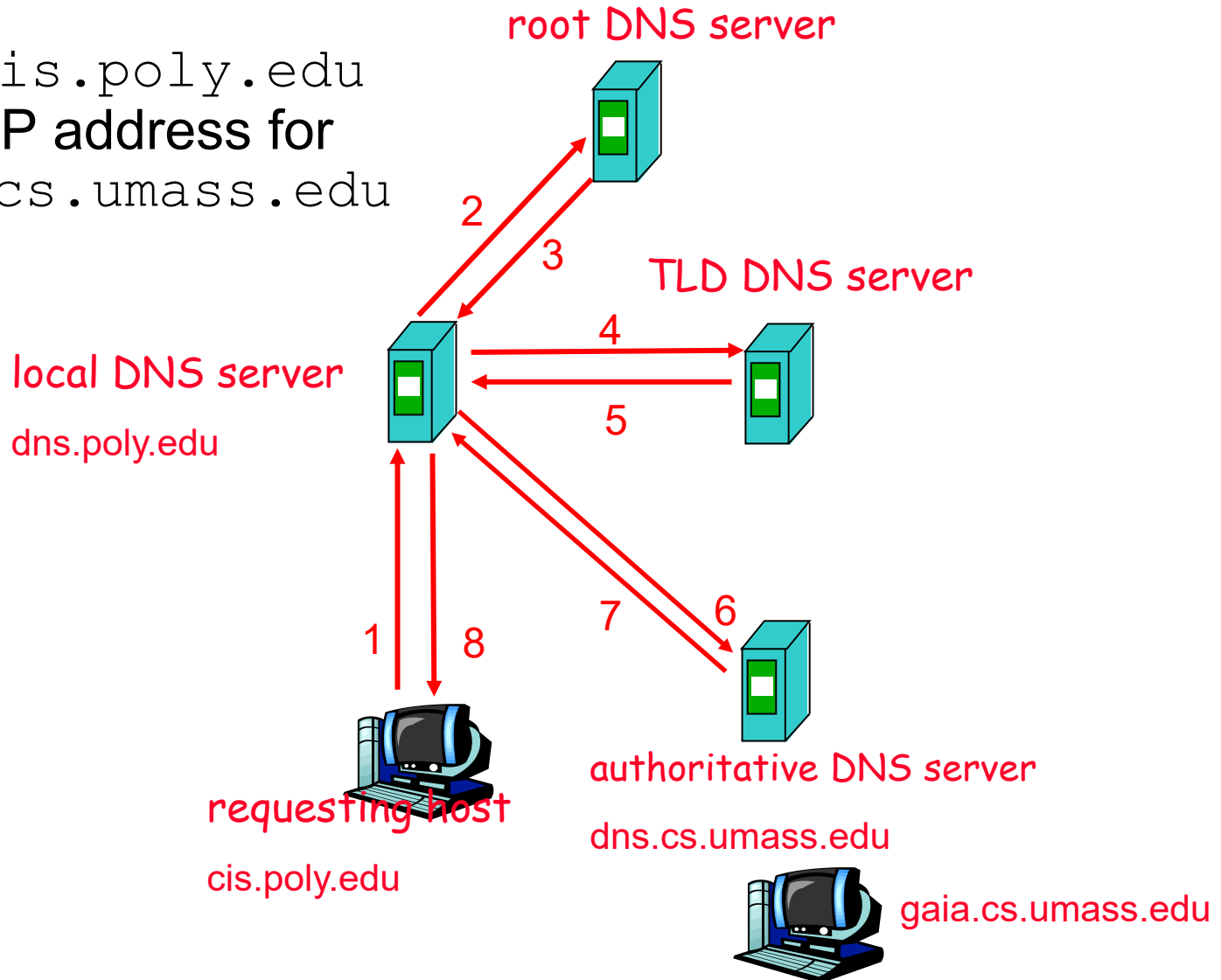


TLD and Authoritative DNS Servers

- Top-level domain (TLD) servers
 - Generic domains (e.g., com, org, edu)
 - Country domains (e.g., uk, fr, ca, jp)
 - Typically managed professionally
 - » Network Solutions maintains servers for “com”
 - » Educause maintains servers for “edu”
- Authoritative DNS servers
 - Provide public records for hosts at an organization
 - For the organization’s servers (e.g., Web and mail)
 - Can be maintained locally or by a service provider

Example

Host at `cis.poly.edu`
wants IP address for
`gaia.cs.umass.edu`



How a CDN Works

facebook.com (content provider)

DNS root server

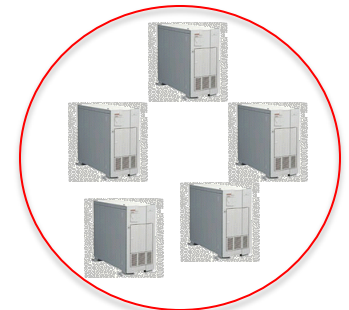


GET
index.
html

1 <http://cache.facebook.com/facebook.com/foo.jpg>
2



Akamai global
DNS server



Akamai
cluster



Akamai regional
DNS server



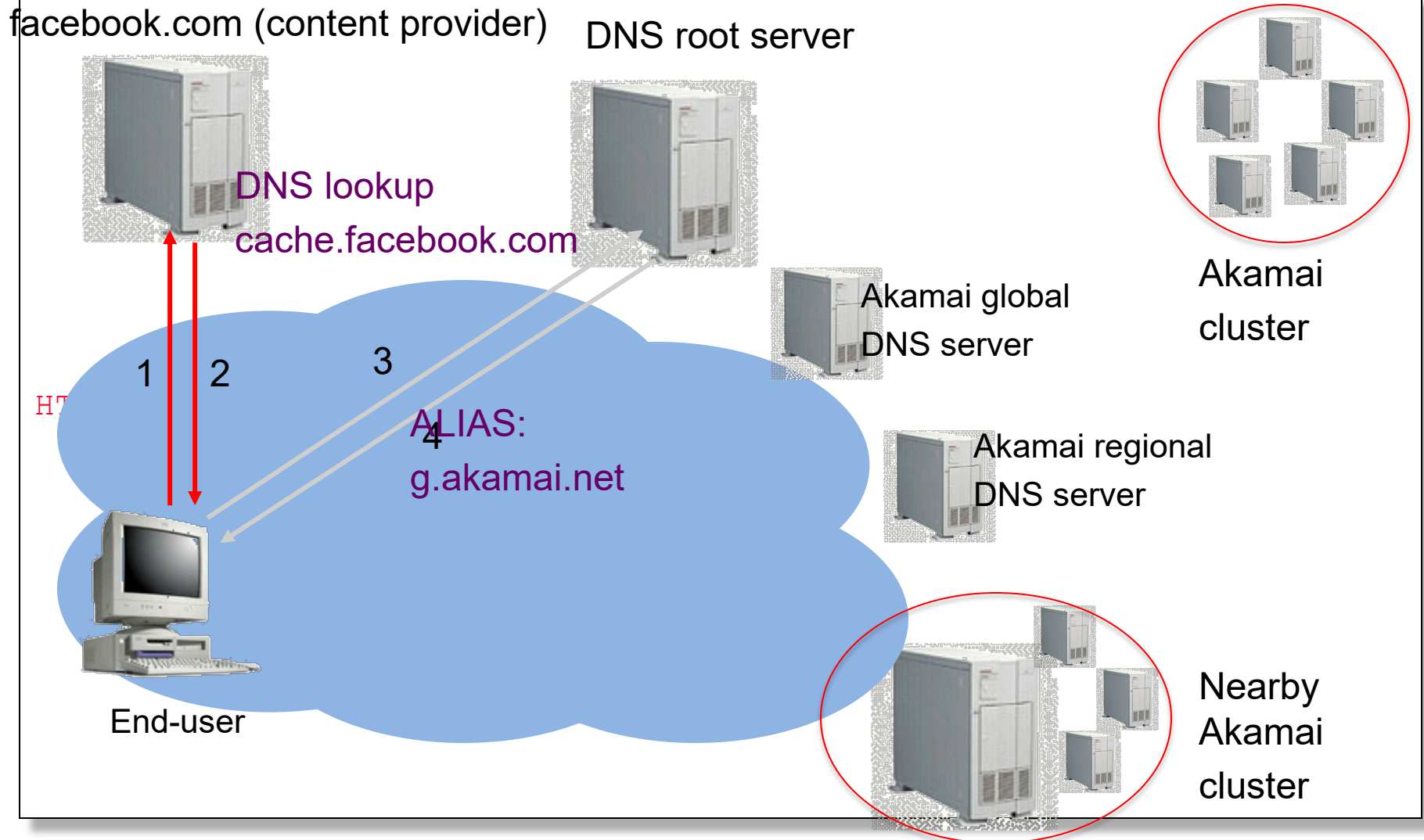
Nearby
Akamai
cluster

HTTP
HTTP

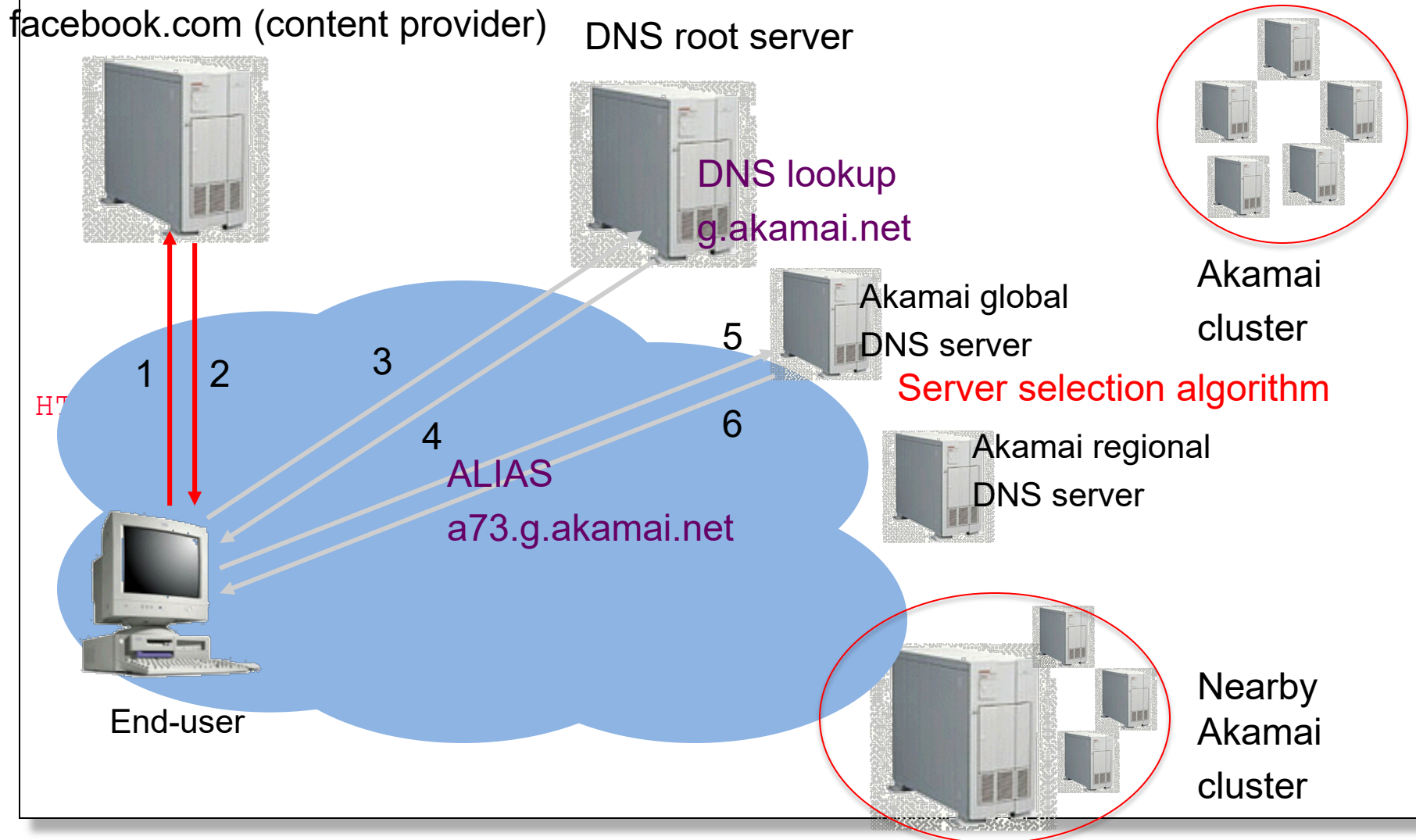


End-user

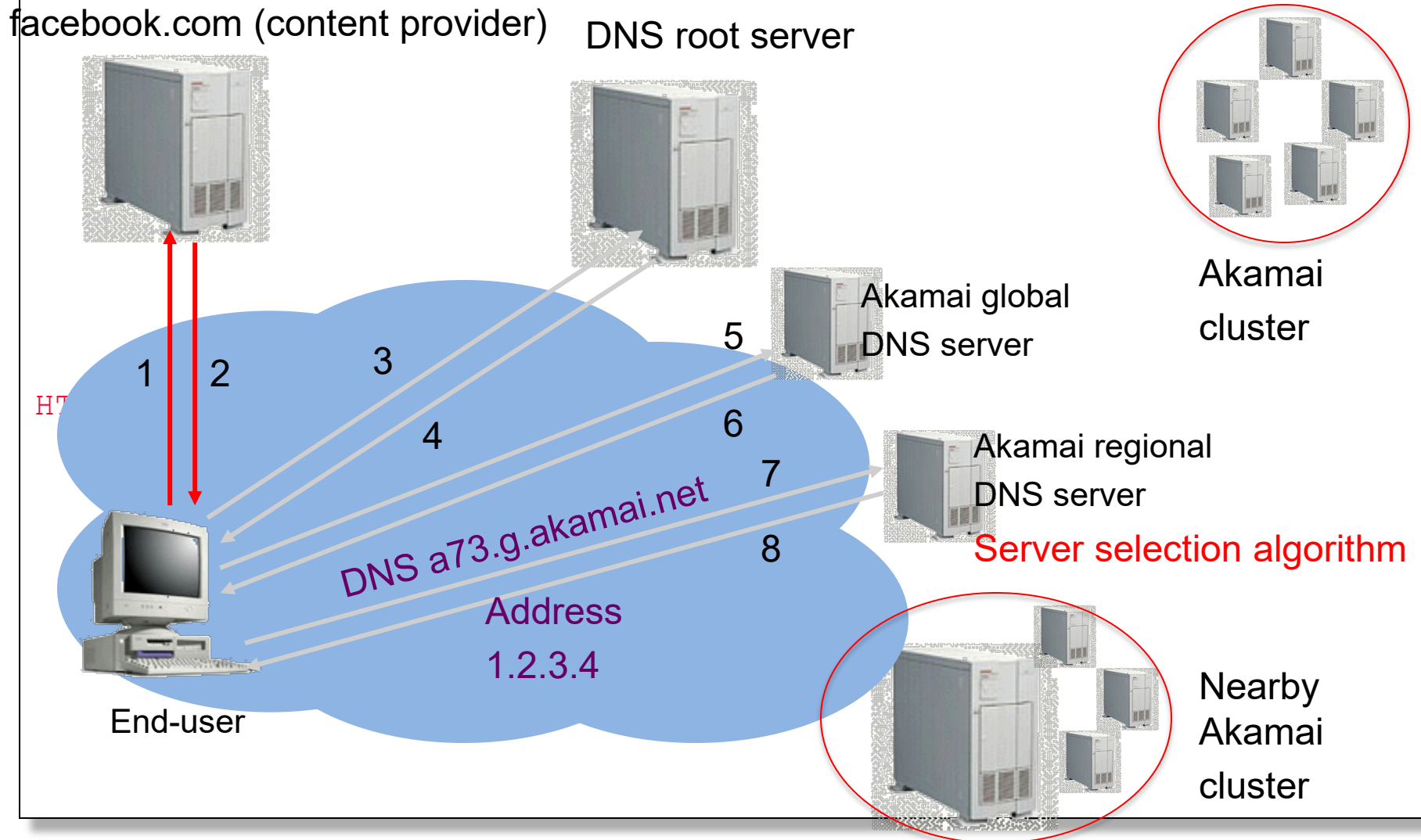
How a CDN Works



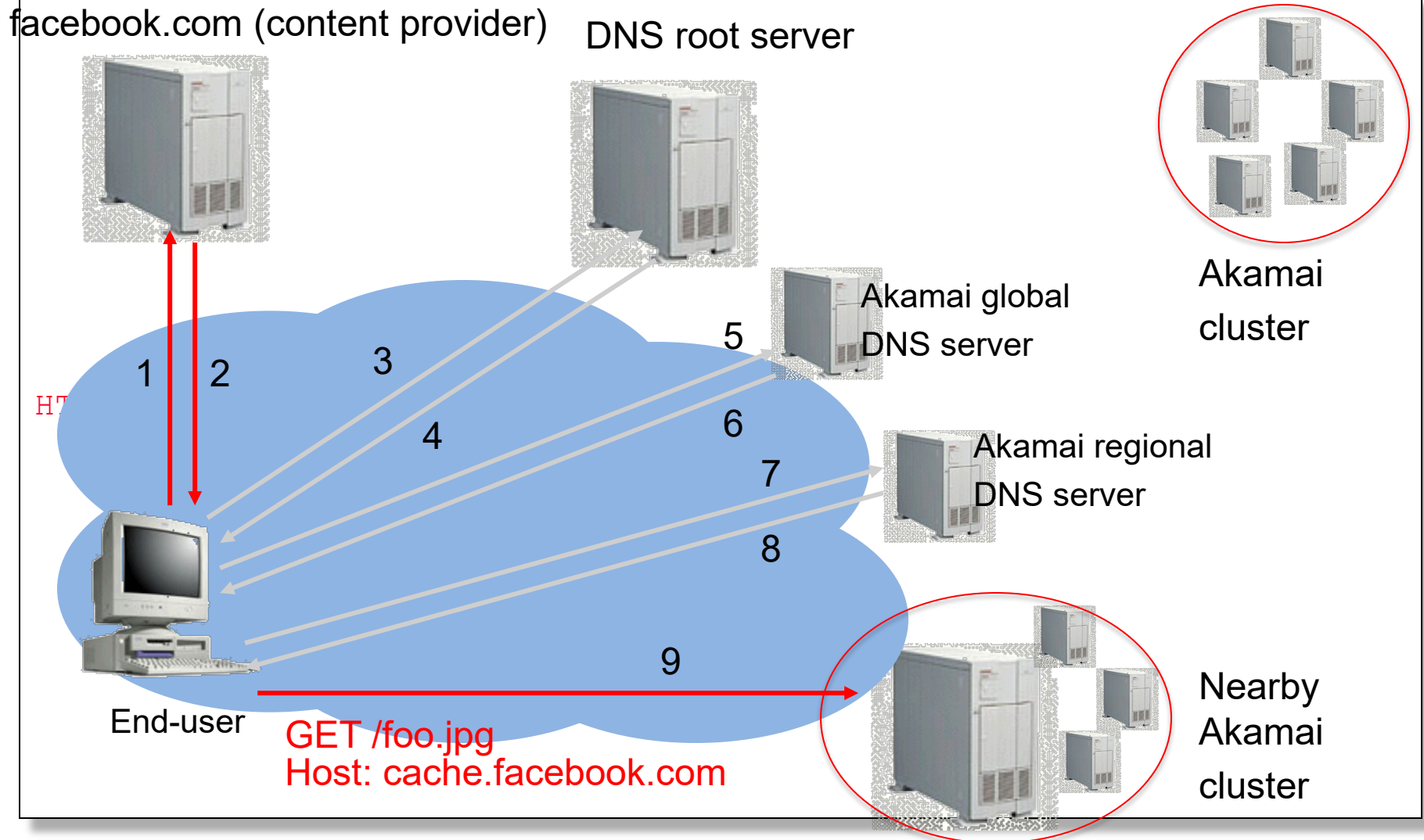
How a CDN Works



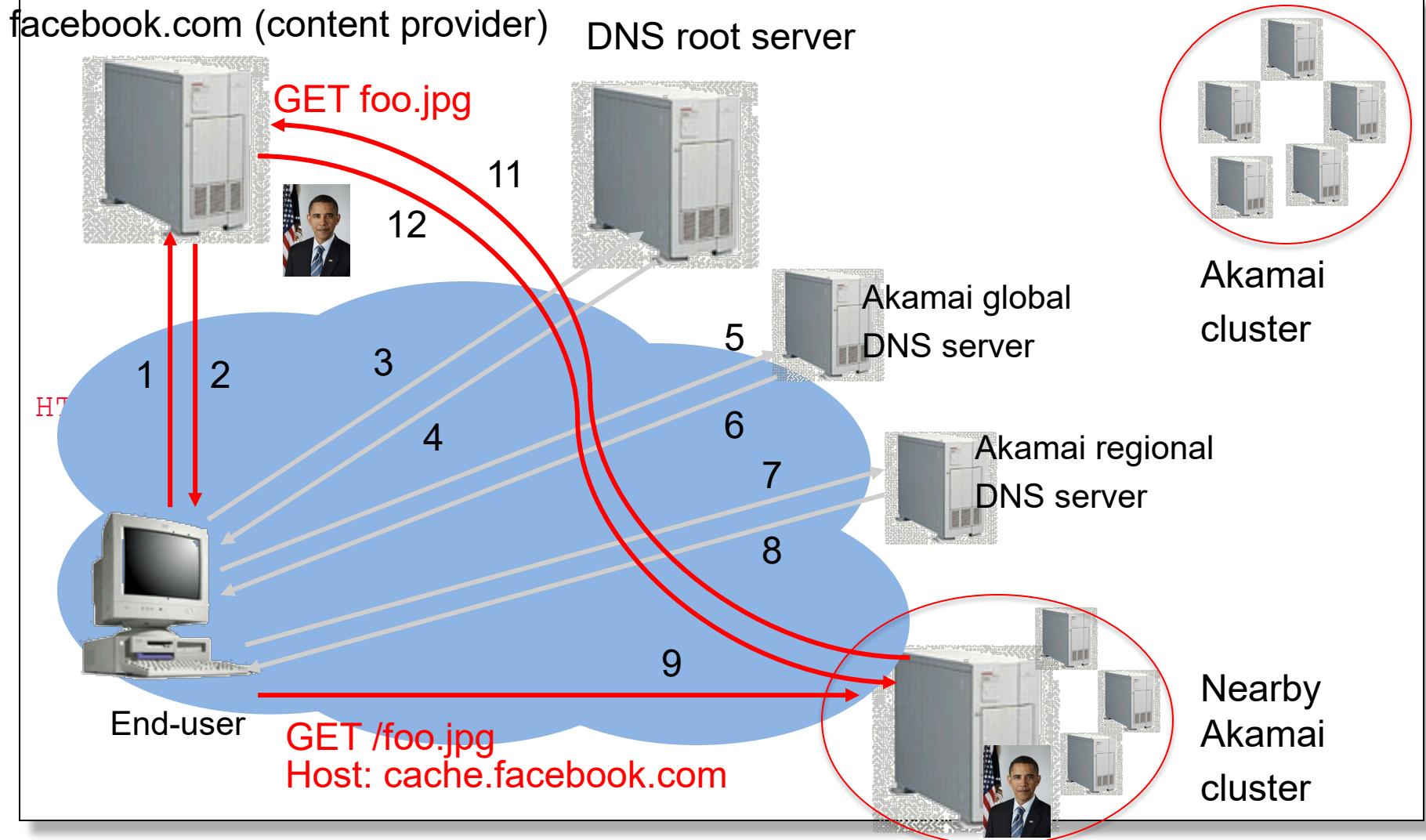
How a CDN Works



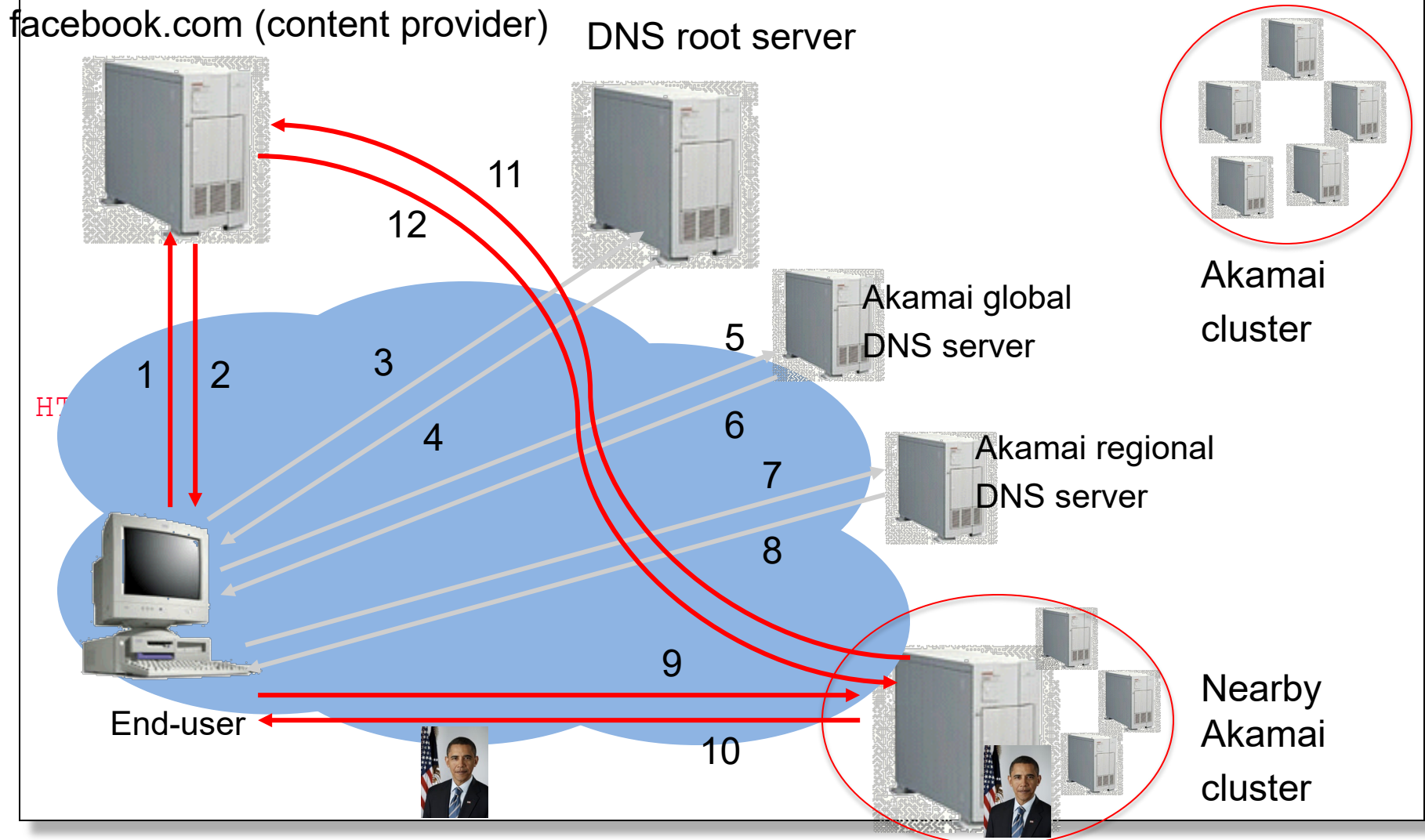
How a CDN Works



How a CDN Works



How a CDN Works



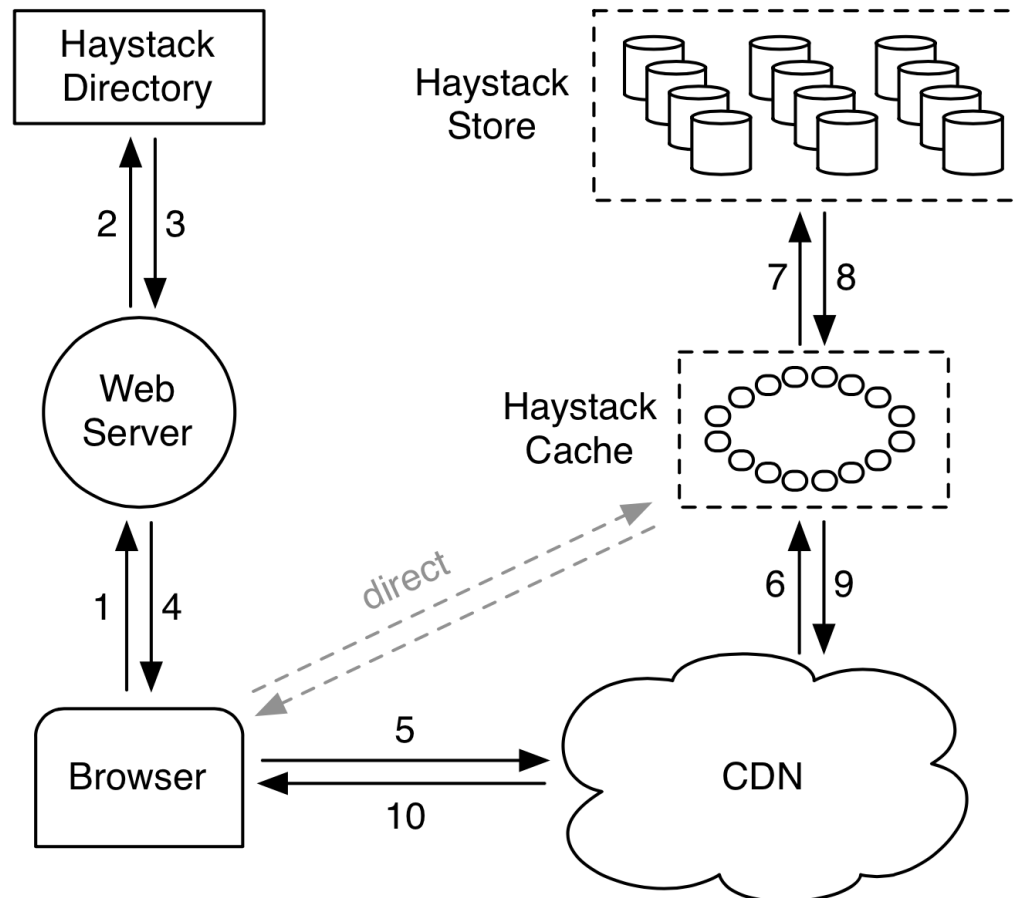
Facebook Photo Distribution

- “Hot” vs. “warm” vs. “cold” photos
 - Hot: Popular, a lot of views (approx. 90% of views)
 - Warm: Somewhat popular, but still a lot of views in aggregate
 - Cold: Unpopular, occasional views



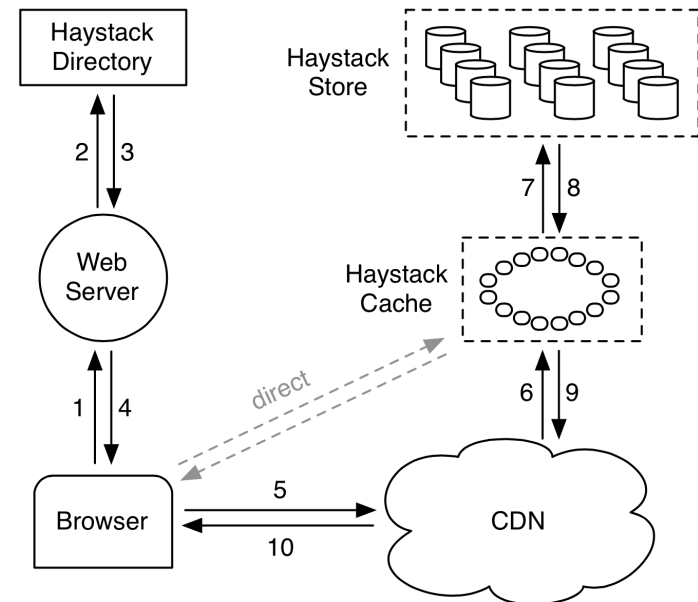
Handling Warm Photos: Haystack

- Designed for performance and reliability
- “Default” photo storage



Haystack Directory

- Helps the URL construction for an image
 - `http://<CDN>/<Cache>/<Machine id>/<Logical volume, Photo>`
 - Staged lookup
 - CDN strips out its portion.
 - Cache strips out its portion.
 - Machine strips out its portion



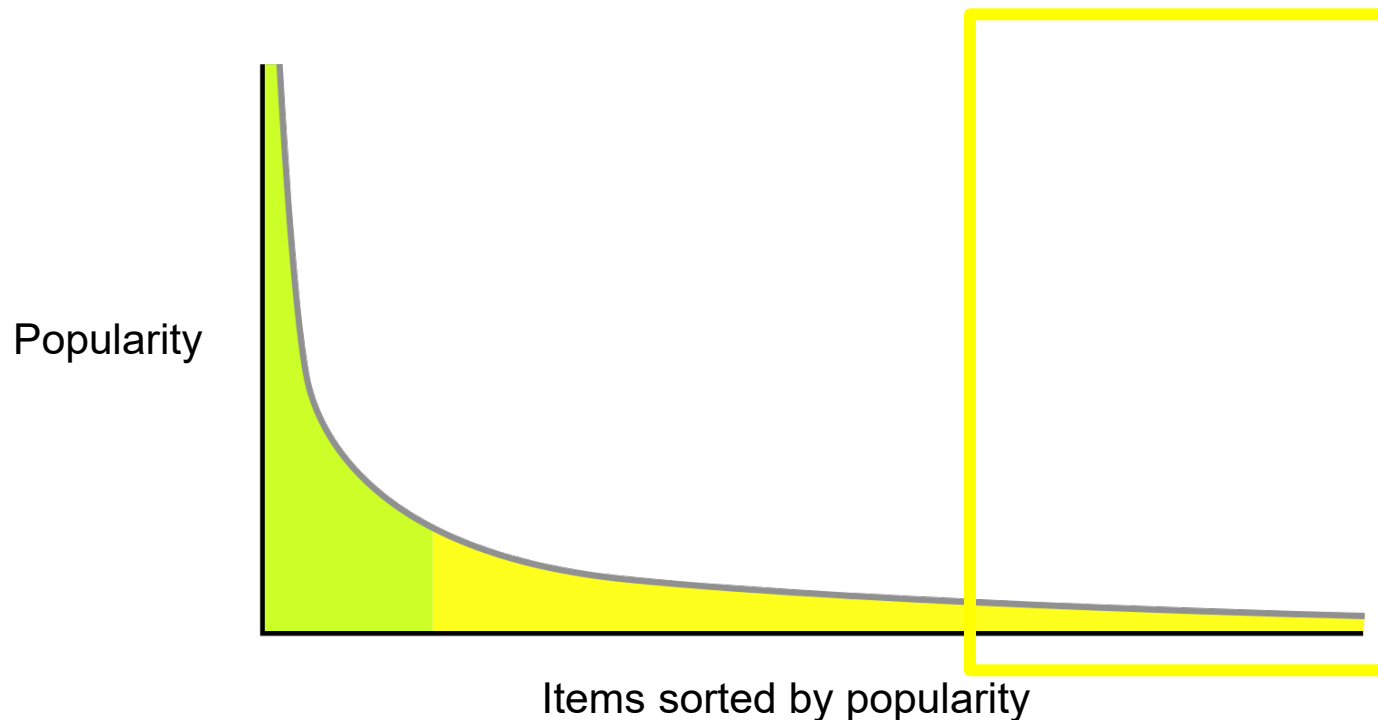
- Logical & physical volumes
 - A logical volume is replicated as multiple physical volumes
 - Physical volumes are stored.
 - Each volume contains multiple photos.

Haystack Cache & Store

- Haystack cache
 - Facebook-operated second-level cache using DHT
 - Photo IDs as keys
 - Further removes traffic to Store
- Haystack store
 - Maintains physical volumes
 - One volume is a single large file (100GB) with many photos (needles)
 - Performance-optimized: requires a single disk read for image retrieval

Facebook Photo Distribution

- “Hot” vs. “warm” vs. “cold” photos
 - Hot: Popular, a lot of views (approx. 90% of views)
 - Warm: Somewhat popular, but still a lot of views in aggregate
 - Cold: Unpopular, occasional views



CDN / Haystack / f4

- CDN absorbs much traffic for hot photos.
- Haystack's tradeoff: good **throughput and reliability**, but somewhat **inefficient storage space usage** (mainly due to replication).
- f4's tradeoff: **less throughput**, but **more storage efficient**.
 - ~ 1 month after upload, photos/videos are moved to f4.
 - f4 uses *an error-correcting coding scheme* to efficiently replicate data.

f4's Replication

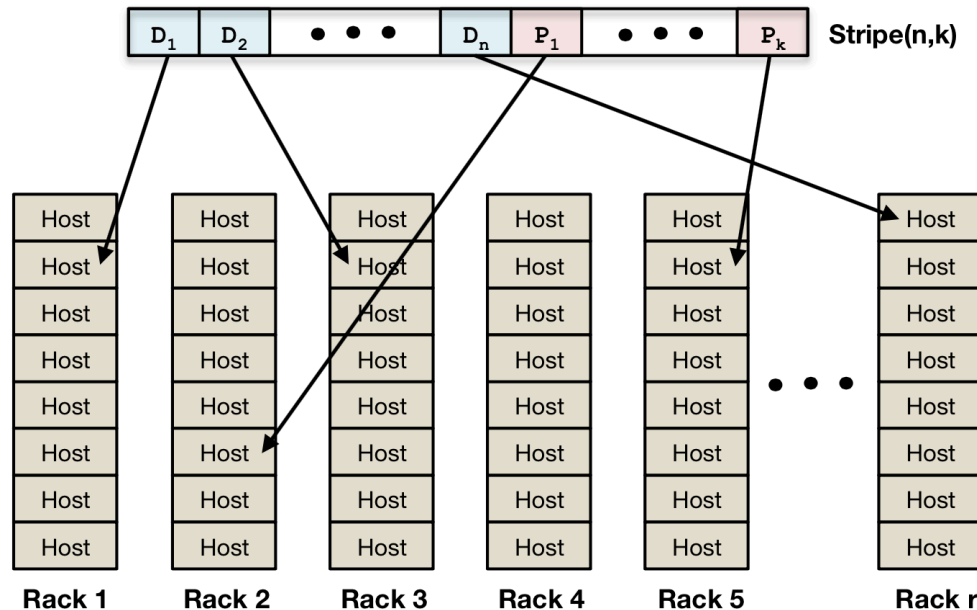
- (n, k) Reed-Solomon code
 - k data blocks, $f = (n - k)$ parity blocks, n total blocks
 - Upon a failure, any k blocks can reconstruct the lost block.
 - Can tolerate up to f block failures
 - Need to go through coder/decoder for read/write, which affects the throughput



- Parity example: XOR
 - (Reed-Solomon uses something more complicated than this.)
 - XOR bits, e.g., $(0, 1, 1, 0) \rightarrow P: 0$
 - Reconstruction after failures: $(0, 1, \text{X}, 0) \rightarrow P: 0$

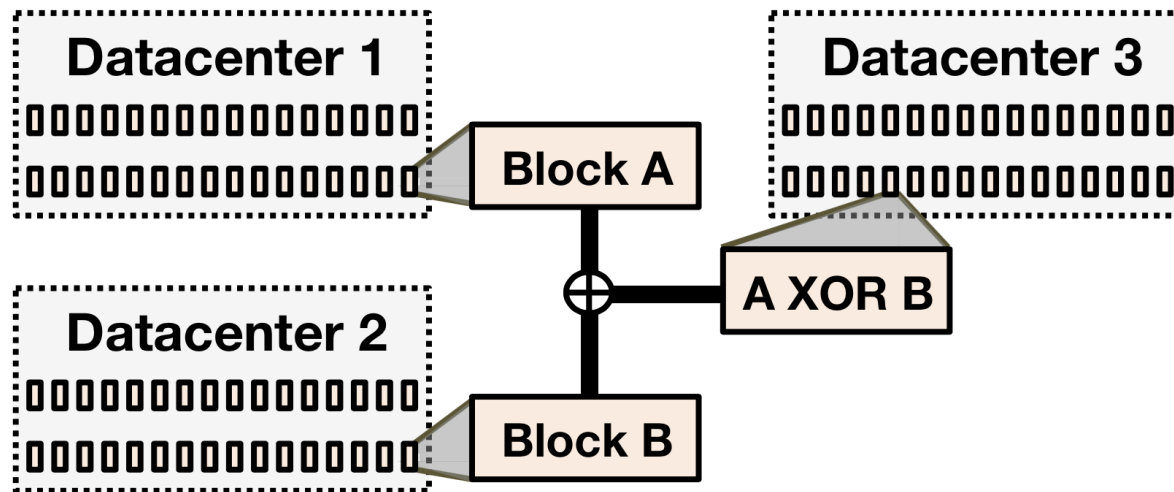
f4: Single Datacenter

- Within a single data center, (14, 10) Reed-Solomon code
 - This tolerates up to 4 block failures
 - 1.4X storage usage per block
- Distribute blocks across different racks
 - This tolerates four host/rack failures



f4: Cross-Datcenter

- Additional parity block
 - Can tolerate a single datacenter failure



- Overall average space usage per block: 2.1X
 - E.g., average for block A & B: $(1.4 \times 2 + 1.4) / 2 = 2.1$
- With 2.1X space usage,
 - 4 host/rack failures tolerated
 - 1 datacenter failure tolerated

Summary

- Engineering a system needs workload understanding.
- Facebook photo workload
 - Hot, warm, and cold.
- CDN for hot photos
 - Performance
- Haystack for warm photos
 - Performance & reliability
- f4 for cold photos
 - Reliability and storage efficiency

Acknowledgements

- These slides contain material developed and copyrighted by Indranil Gupta (UIUC), Michael Freedman (Princeton), and Jennifer Rexford (Princeton).