CONTENT DELIVERY NETWORKS

CLOUD COMPUTING

Dr. Atm Shafiul Alam

a.alam@qmul.ac.uk

Queen Mary University of London

School of Electronic Engineering and Computer Science

Thanks to Dr. Gareth Tyson and Prof. Steve Uhlig who created the original version of these slides

QUICK RECAP…

In last lecture, we...

- Learnt about in-memory processing
- Understood how Spark works
- Learnt about stream processing

Map/Reduce is not a good fit for every case

- Rigid structure: Map, Shuffle Sort, Reduce
- No support for iterations
- Only one synchronization barrier
- See graph processing as an example…

Resillient Distributed Datasets (RDDs)

- Immutable collections **distributed** across the cluster
- Created generally by reading data from HDFS
- Can be **transformed** into other RDDs for processing
  - E.g. filter
- Can have **actions** performed on them
  - E.g. count
- Can be saved back to HDFS/ other programs with actions

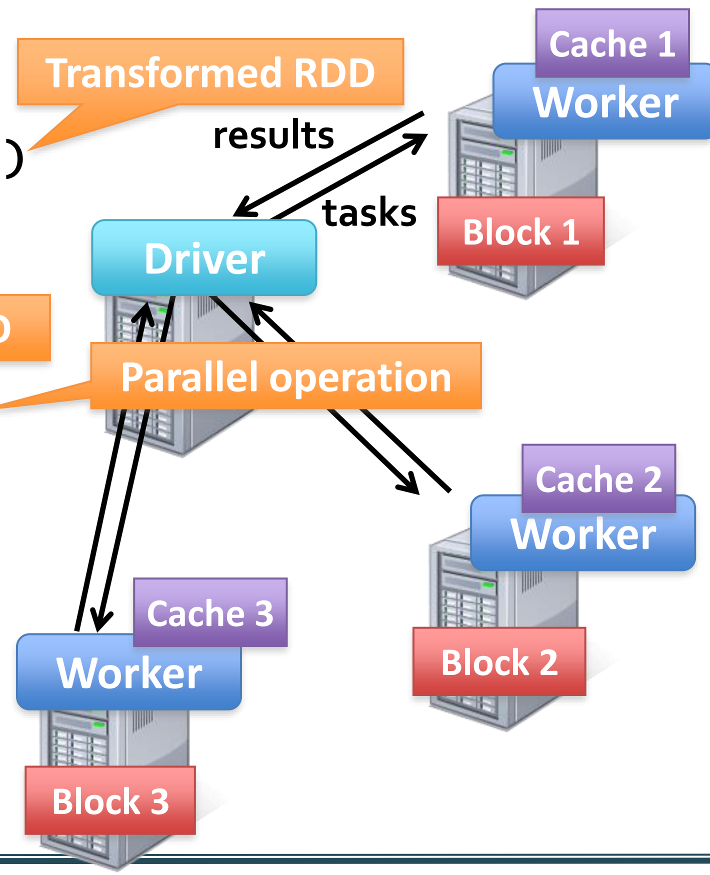
Example: Log Mining

- Load error messages from a log into memory, then interactively search for various patterns

Base RDD

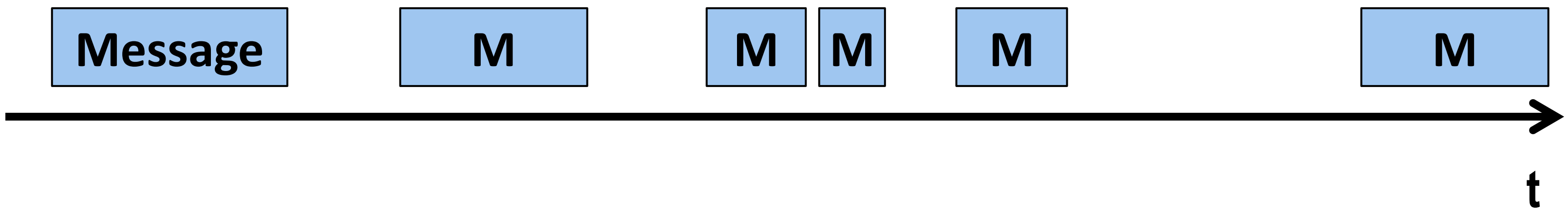
```
lines = spark.textFile("hdfs://...")
errors = lines.filter(_.startsWith("ERROR"))
messages = errors.map(_.split('\t')(2))
cachedMsgs = messages.cache()

cachedMsgs.filter(_.contains("foo")).count
cachedMsgs.filter(_.contains("bar")).count
. . .
```

Cached RDD



Result: full-text search of Wikipedia in <1 sec (vs 20 sec for on-disk data)

Information Streams



Unbounded sequence of messages
Arrival time is not fixed

Spark Streaming: Discretized Streams

- Unlike **pure** stream processing, we process the incoming messages on **micro batches**

Chop up the live stream into batches of X seconds

Spark treats each batch of data as RDDs and processes them using RDD operations

Finally, the processed results of the RDD operations are returned in batches

Content Delivery Networks (CDNs)

- Let's set up a website..
  - Write some HTML
  - Buy a domain (e.g. `www.gareth.com`)
  - Buy a web server

- What happens if it becomes popular in China?
- What happens if it becomes popular across the world?
- How can we serve millions of requests?

Agenda

- DNS
- P2P
- Content delivery
- CDNs
- Deployment
- Future

DNS: Domain name system

- Used to translate domain names to IP addresses
  - E.g. www.qmul.ac.uk -> 161.23.10.1
- The DNS is a distributed database
- Consists of many interconnected servers
- Many applications rely on DNS
  - Including **WWW**, email, etc.

DNS: services, structure

- *DNS services*
  - Hostname to IP address translation
  - Often used to manage load distribution
    - Replicated Web servers: many IP addresses correspond to one domain

Why not centralize DNS?

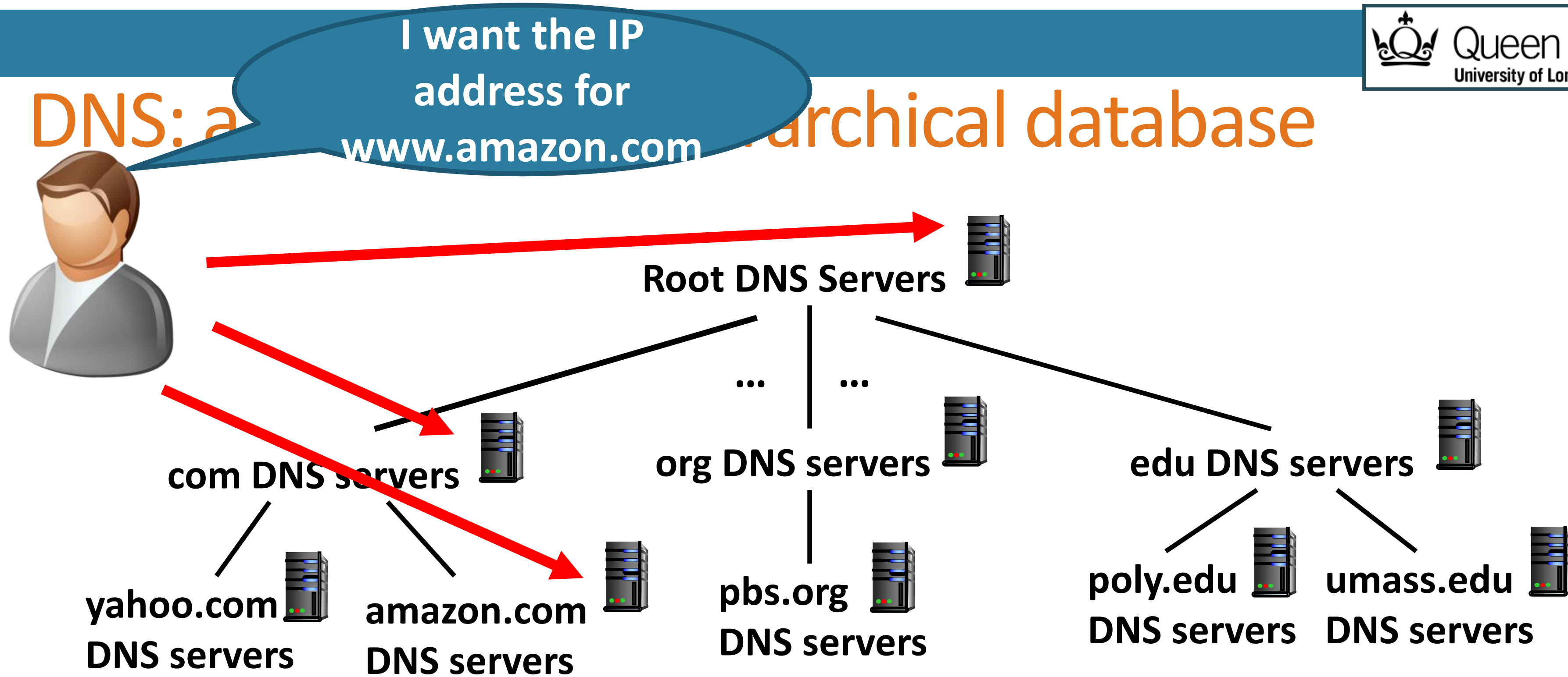
- Single point of failure
- Traffic volume
- Distant centralized database
- Maintenance

A*: *doesn't scale!

Content Delivery Network use DNS to "redirect" clients to the nearest web server


I want www.gareth.com
www.gareth.com

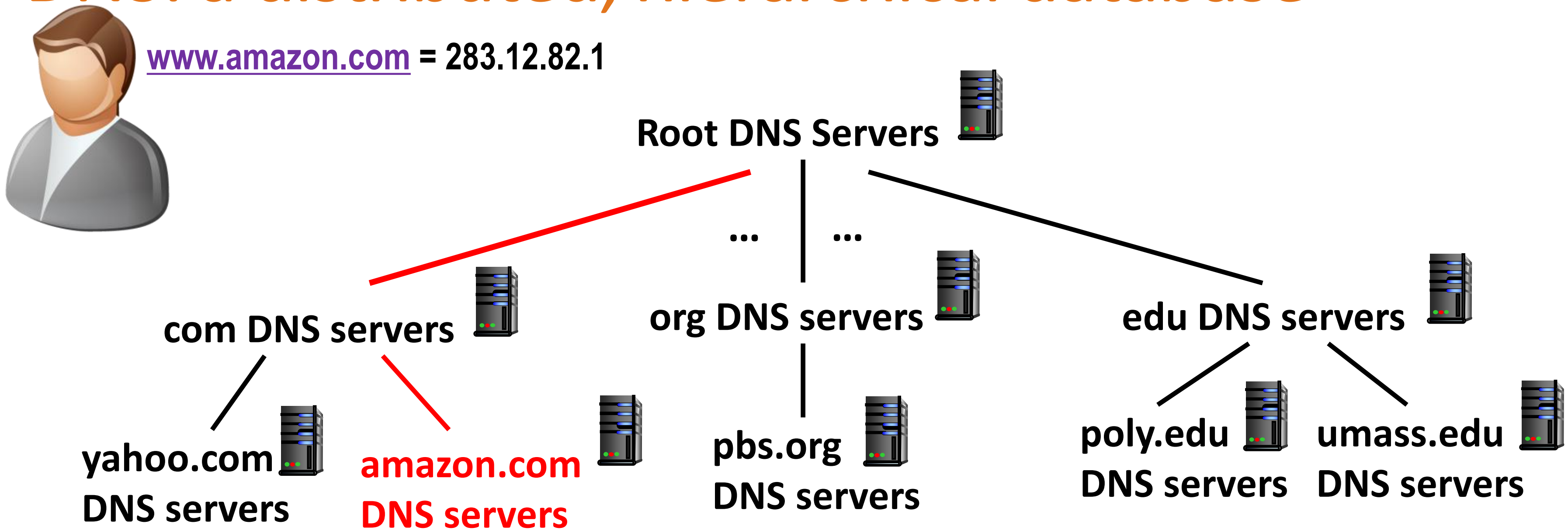
DNS: a … archical database



Client wants IP for www.amazon.com:

- client queries root server to find com DNS server
- client queries .com DNS server to get amazon.com DNS server
- client queries amazon.com DNS server to get IP address for www.amazon.com

DNS: a distributed, hierarchical database

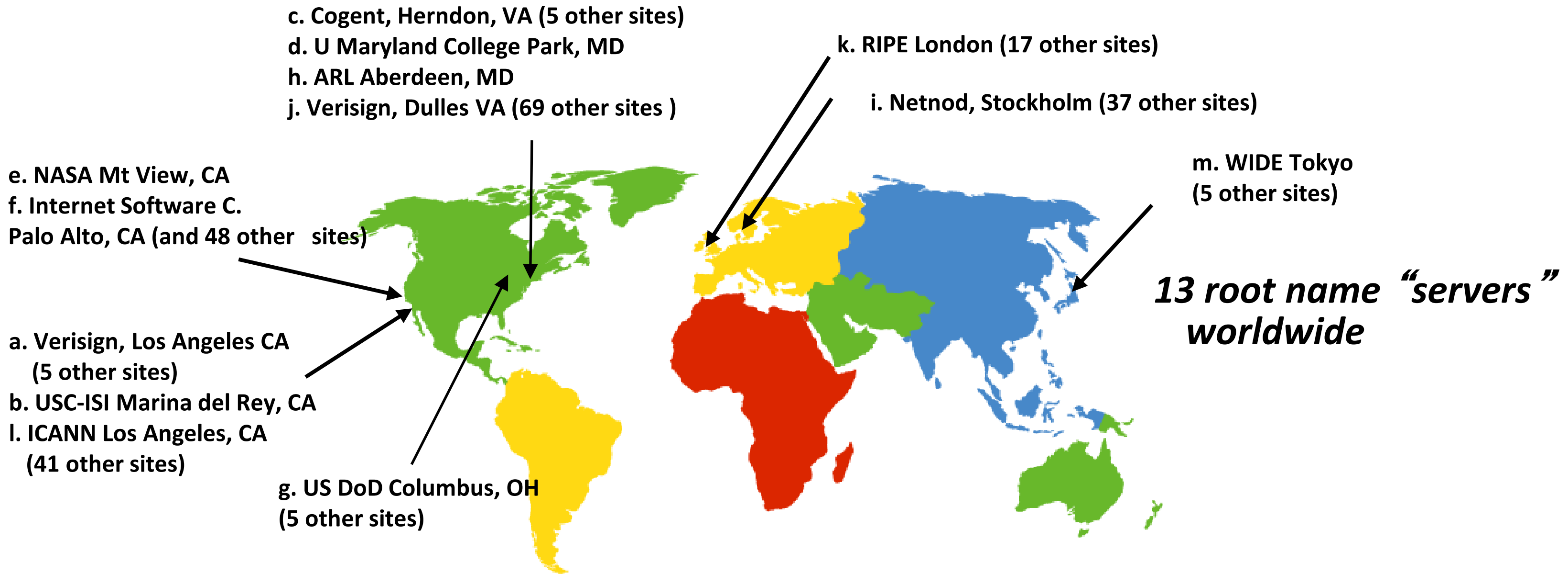


Client wants IP for www.amazon.com:

- client queries root server to find com DNS server
- client queries .com DNS server to get amazon.com DNS server
- client queries amazon.com DNS server to get IP address for www.amazon.com

DNS: root name servers

- First point of contact when performing DNS resolution
- Root name server:
  - Contacts authoritative name server if name mapping not known
  - Gets mapping
  - Returns mapping to local name server



TLD & authoritative servers

- *Top-level domain (TLD) servers:*
  - Responsible for .com, .org, .net, .edu, etc.
  - And top-level country domains, e.g. .uk, .fr, .cn
  - Network Solutions maintains servers for .com TLD
  - Educause for .edu TLD
- *Authoritative DNS servers:*
  - Organization's own DNS server(s), providing authoritative hostname to IP mappings for organization's named hosts

Local DNS resolver

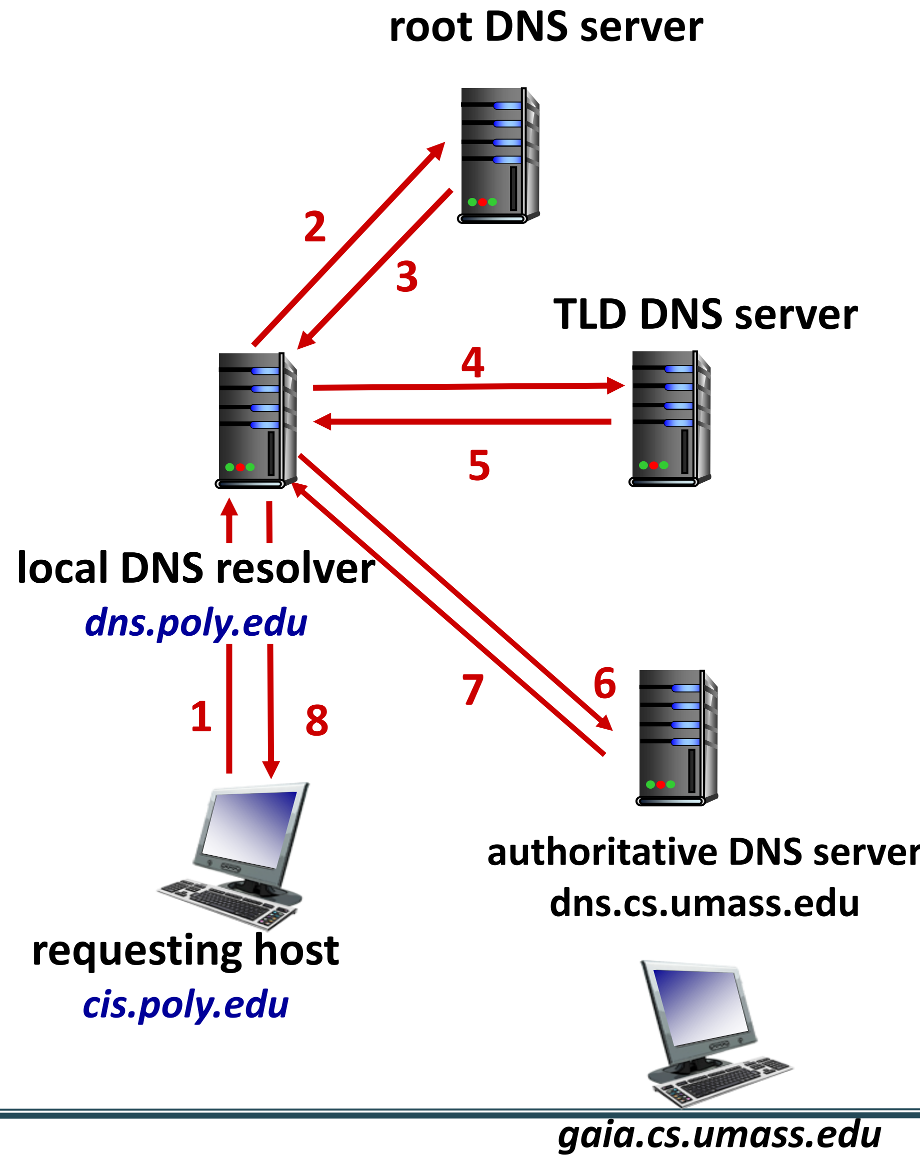
- Client, not part of DNS hierarchy
- Each ISP (residential ISP, company, university) has one
- When host makes DNS query, query is sent to its local DNS resolver
  - Has local cache of recent name-to-address translation pairs (but may be out of date!)
  - Acts as a proxy, forwards query into DNS hierarchy

DNS name resolution example

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

Iterative query:

- Contacted server replies with name of server to contact (referral) = "I don't know this name, but ask this server"
- Resolver iterates until it finds the answer



DNS caching

- Once (any) name server/resolver learns mapping, it *caches* mapping
  - Cache entries timeout (disappear) after some time (TTL)
  - TLD servers typically cached in local name servers
    - Root name servers not often visited
- Cached entries may be *out-of-date* (best effort name-to-address translation!)
  - If name host changes IP address, may not be known Internet-wide until all TTLs expire

DNS Caching

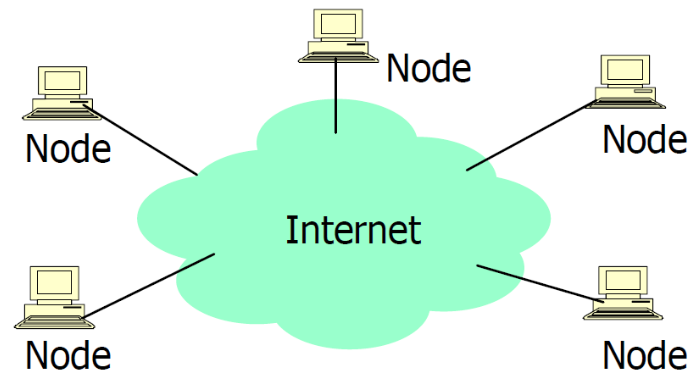
- Caching is used in DNS name servers
- A name server can store DNS query results for a period of time determined in the TTL field of the resource record
- Benefits of caching
  - Reduces the burden on the root servers
  - Reduces DNS traffic across the Internet
  - Increases performance in Internet applications

Agenda

- DNS
- **P2P**
- Content delivery
- CDNs
- Deployment
- Future

Peer-to-Peer (P2P) Systems

- A distributed system architecture
- In a P2P network, every node is both a client and a server
  - Provide and consume data
  - Any node can initiate a connection
- No centralized data source
  - *The ultimate form of democracy on the Internet*
- As no. of clients increases, no. of servers also increases
  - Perfectly scalable
  - Distributed costs
  - Increased privacy



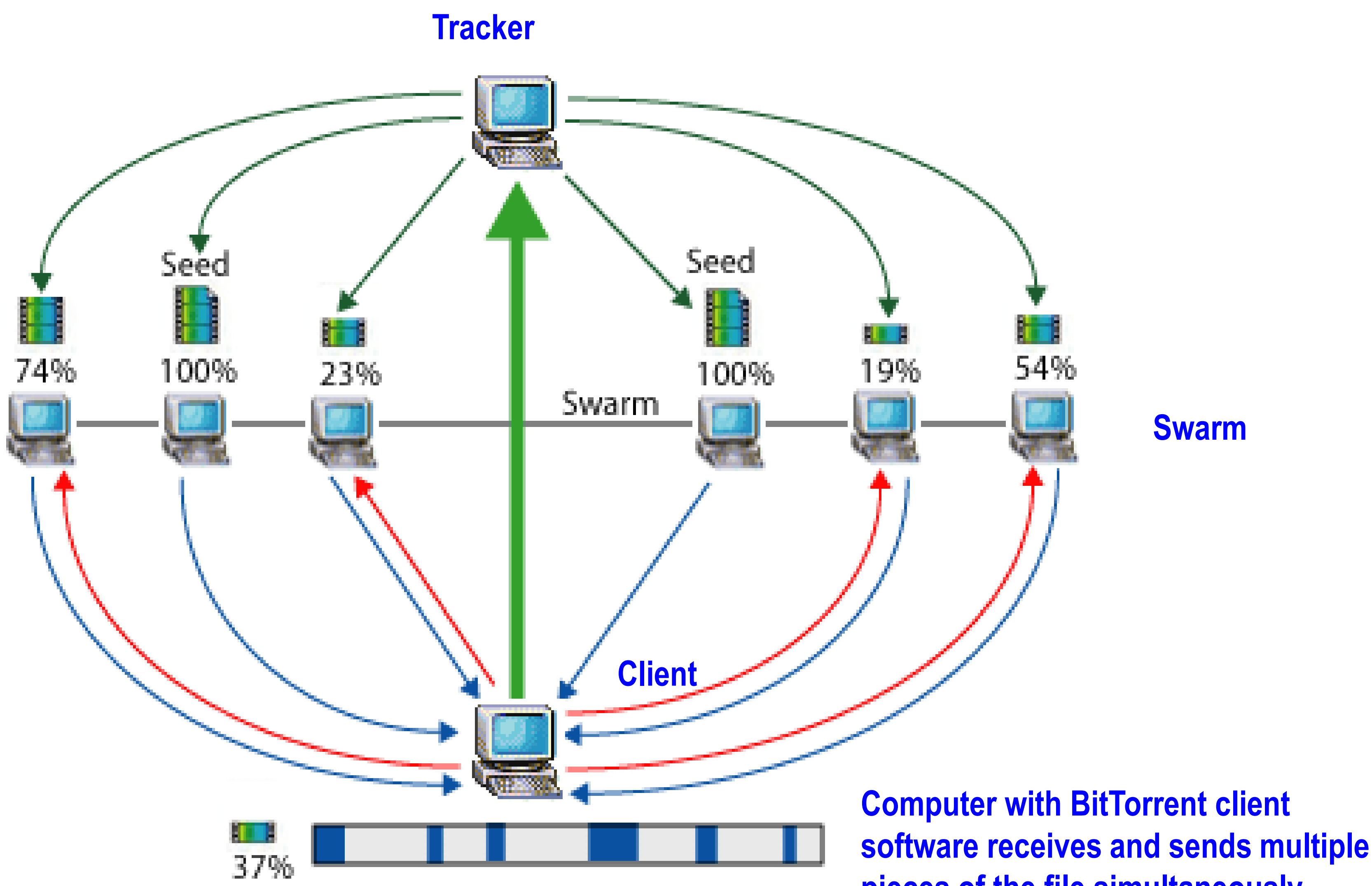
Categories of P2P systems

- Unstructured
  - No restriction on overlay structures and data placement
  - Representative systems
    - Napster, Gnutella, Kazza, Freenet, Bittorrent
- Structured
  - Distributed hash tables (DHTs)
    - put(k,v) and v=get(k) interface
  - Place restrictions on overlay structures and data placement
  - Representative systems
    - Chord, Pastry, Tapestry, and CAN

P2P Network Benefits

- Efficient use of resources
  - B/W, storage, and processing power at the edge of the network
- Scalability
  - Consumers of resources also donate resources
  - Aggregate resources grow naturally, as more peers join
- Reliability
  - Replicas
  - Geographic distribution
  - No single point of failure
- Ease of administration
  - Self-organization
  - No need for server deployment and provisioning
  - Built-in fault tolerance, replication, and load balancing

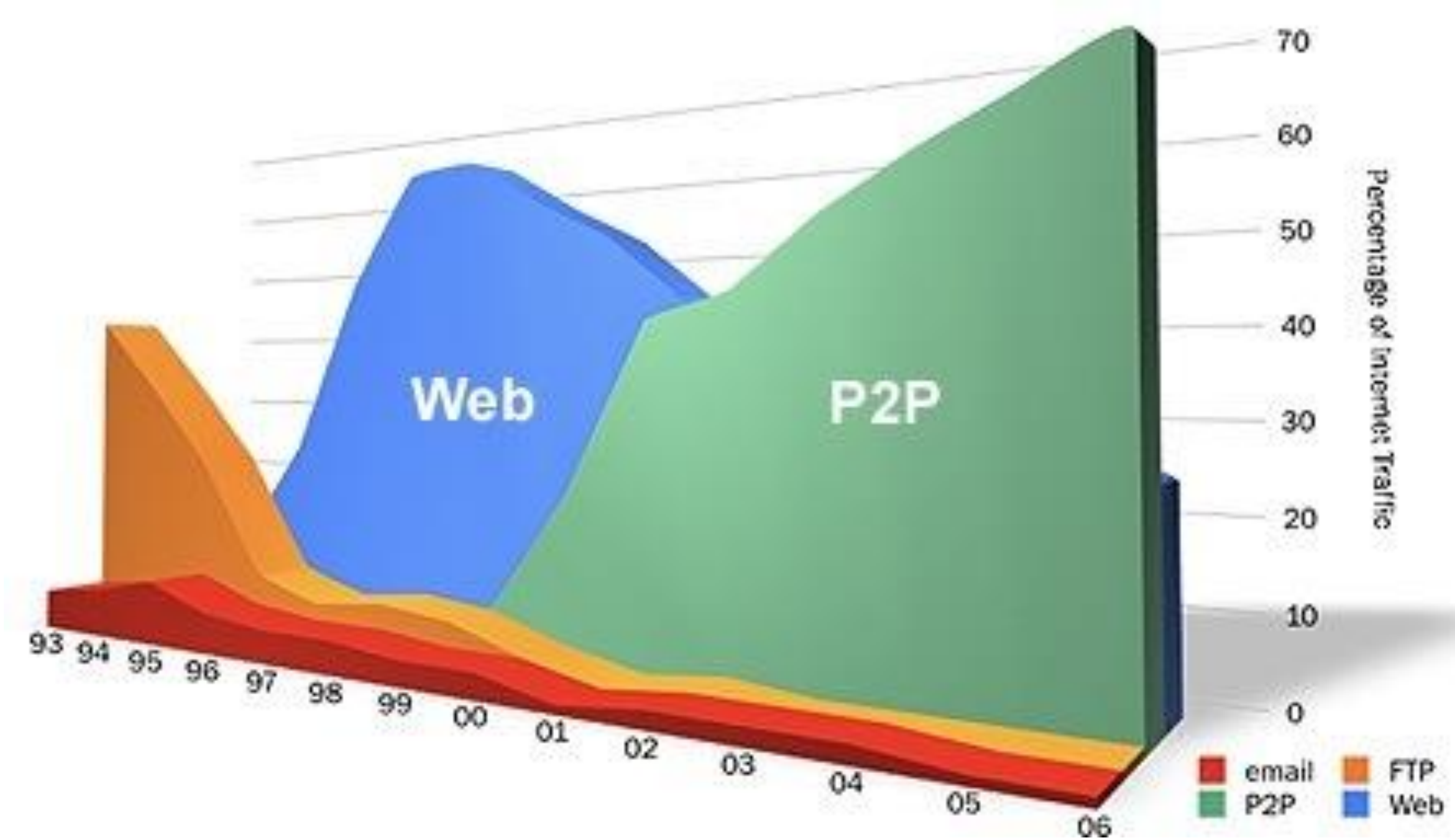
P2P: Server selection



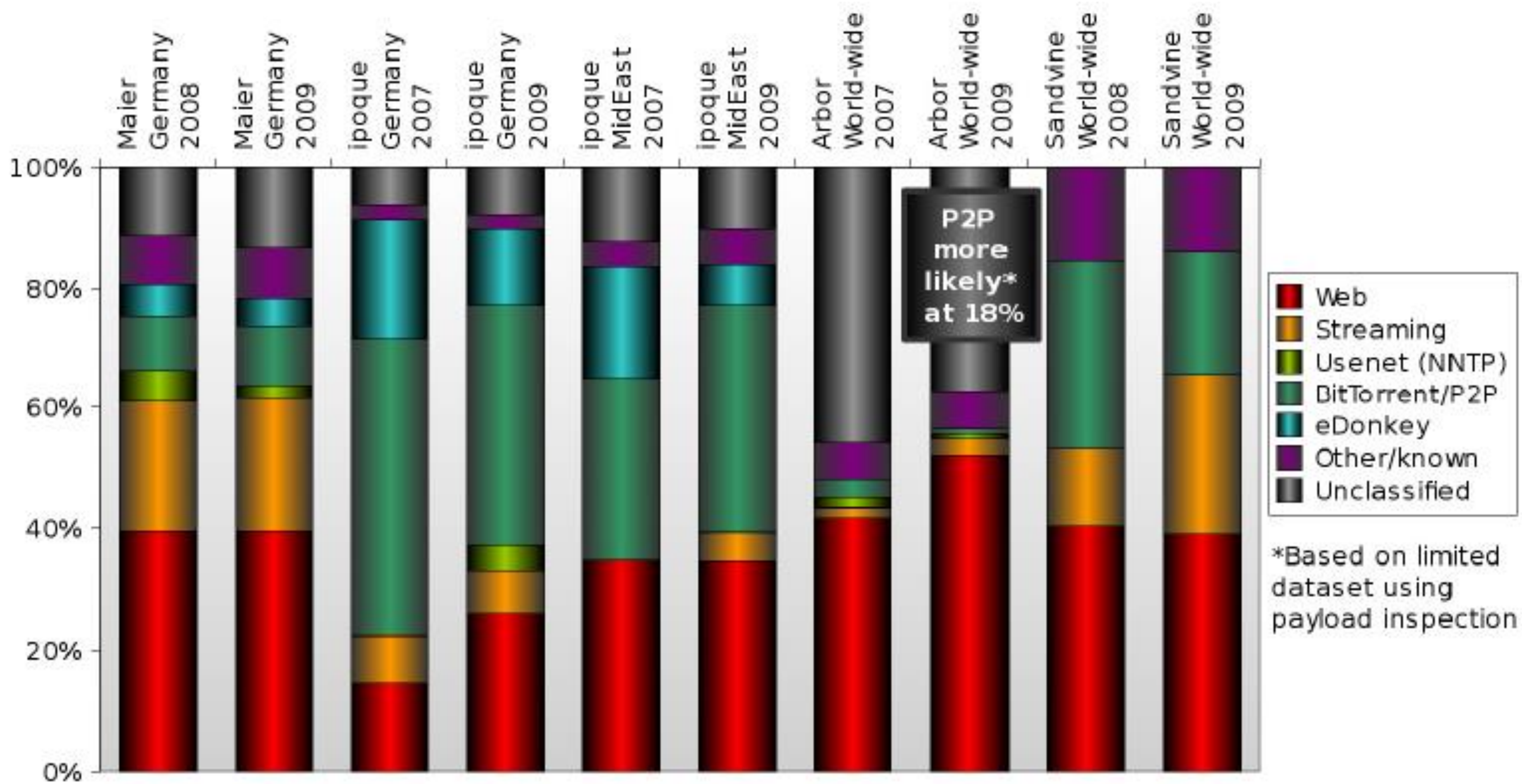
Why is P2P popular?

- Easy to share content
- Storage scales cheaply
- Issues:
  - Reliability
  - Performance
  - Control (lots of copyright content)

P2P vs. Web: who dominates?



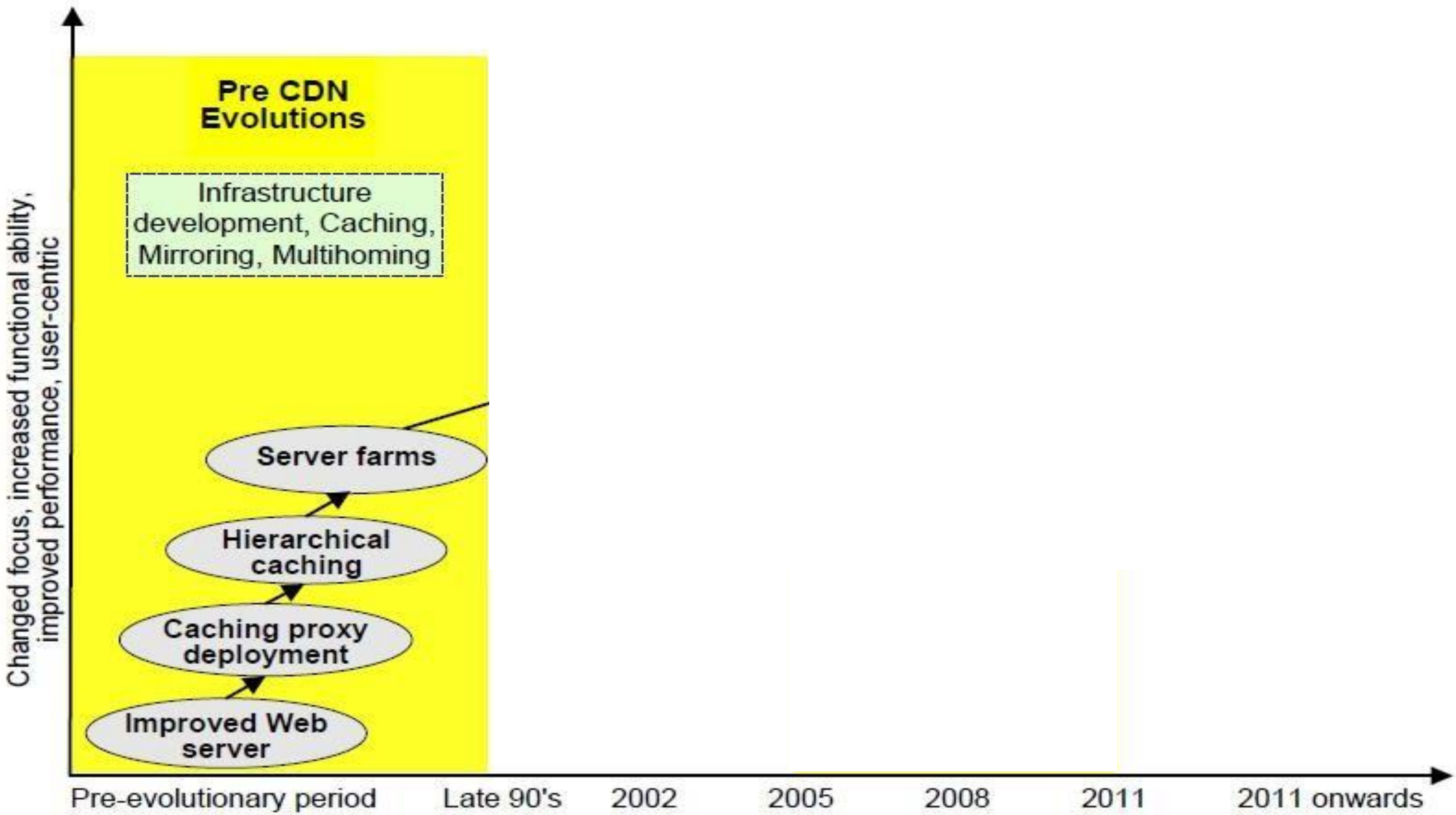
P2P vs. Web: reality



Agenda

- DNS
- P2P
- **Content delivery**
- CDNs
- Deployment
- Future

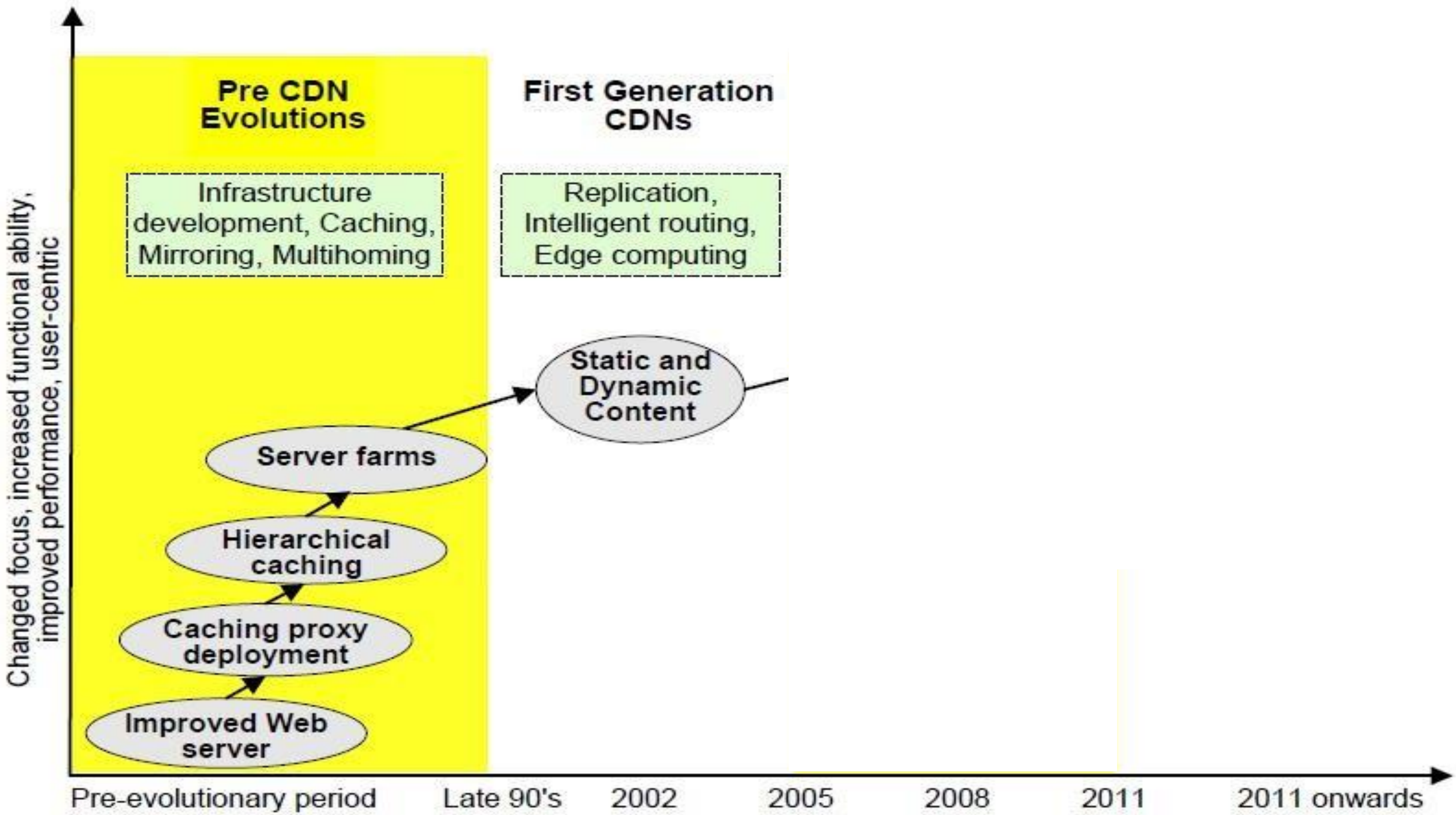
Web 1.0



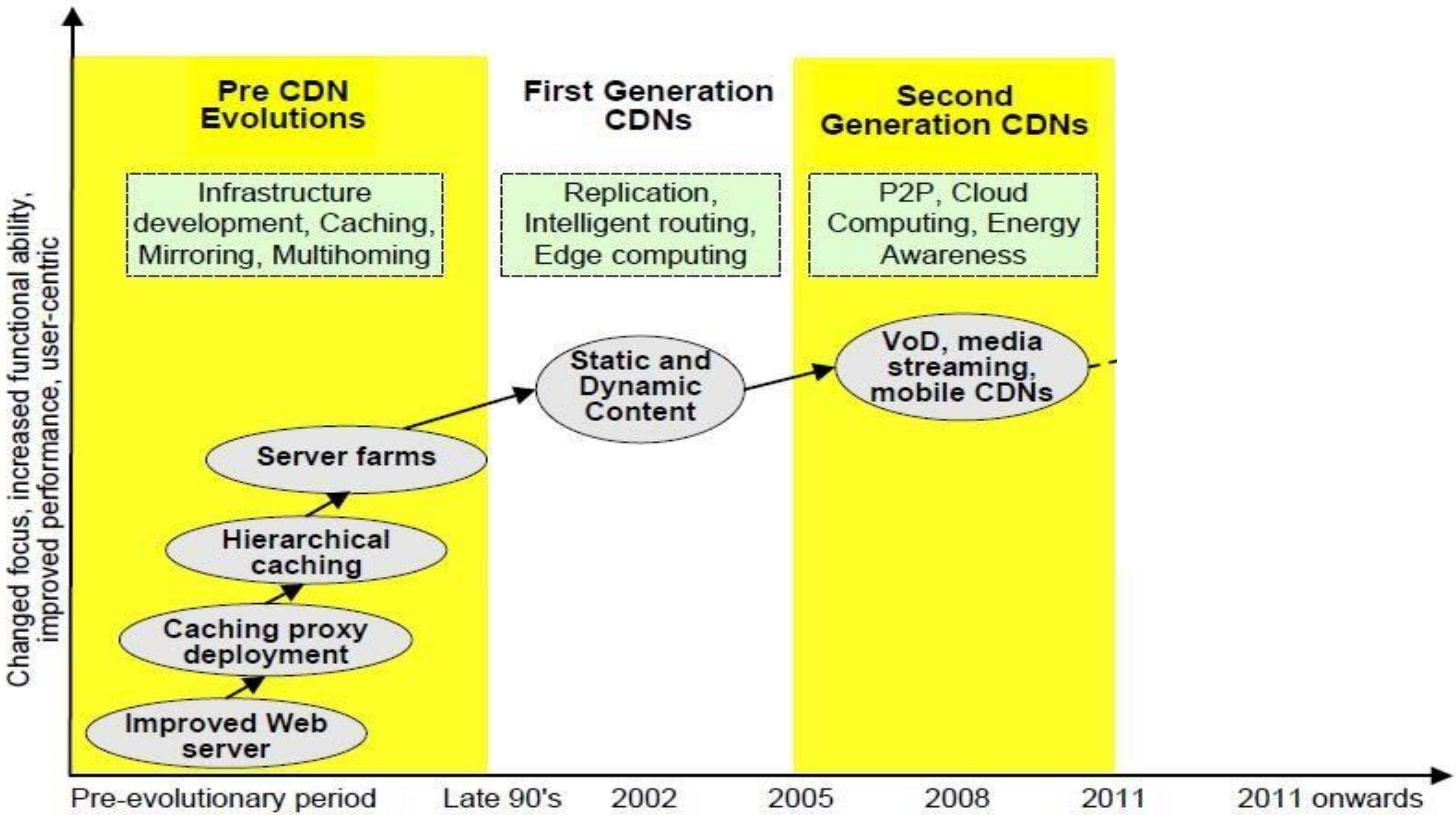
Pathan Mukaddim. Ongoing Trends and Future Directions in Content Delivery Networks (CDNs)

http://amkpathan.wordpress.com/article/ongoing-trends-and-future-directions-in-3uxfz2buz8z1w-2/

CDN 1.0

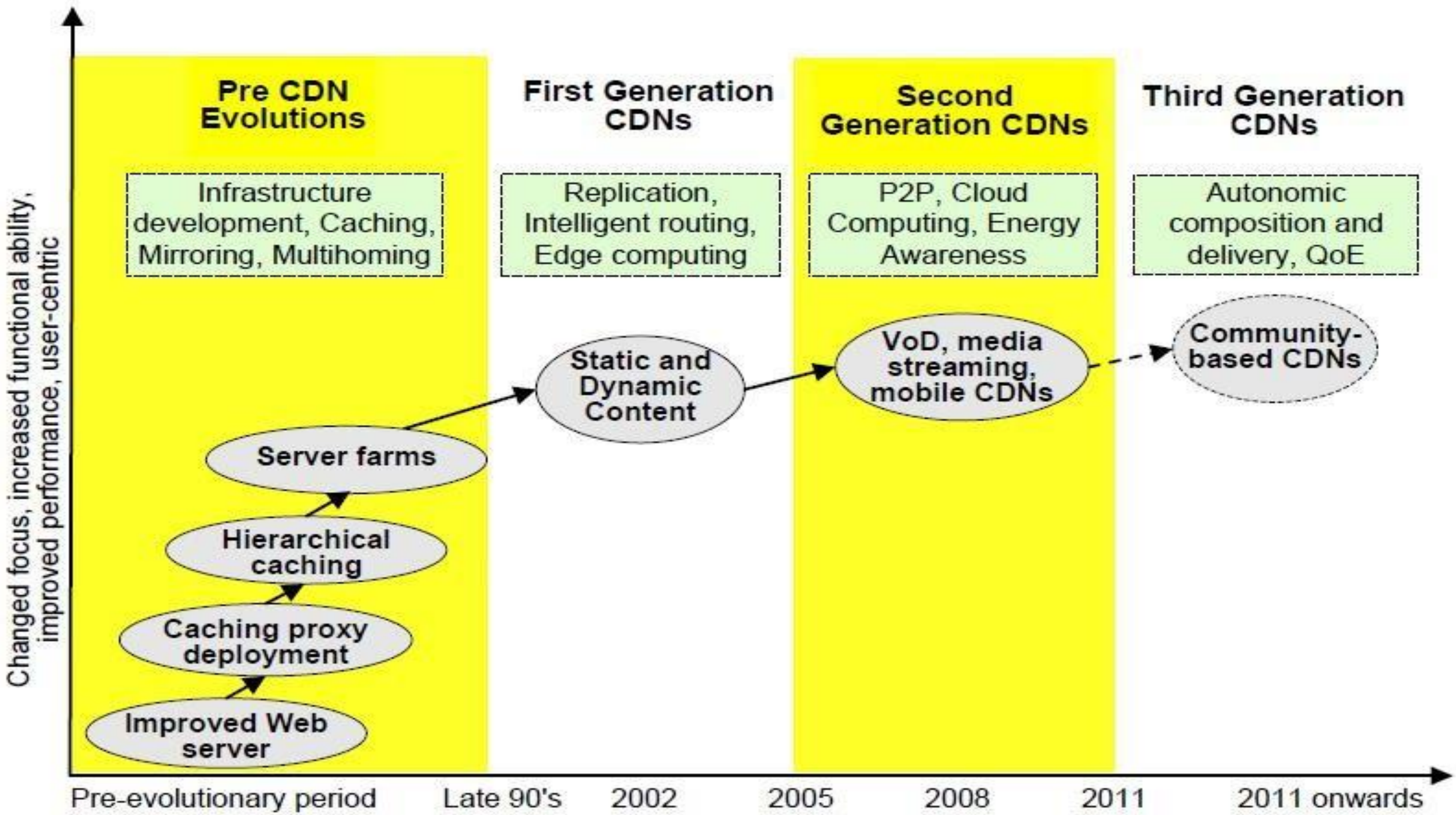

Changed focus, increased functional ability, improved performance, user-centric
Pre CDN Evolutions
First Generation CDNs
Infrastructure development, Caching, Mirroring, Multihoming
Replication, Intelligent routing, Edge computing
Static and Dynamic Content
Server farms
Hierarchical caching
Caching proxy deployment
Improved Web server
Pre-evolutionary period
Late 90's
2002
2005
2008
2011
2011 onwards

CDN 2.0



Changed focus, increased functional ability, improved performance, user-centric
Pre CDN Evolutions
First Generation CDNs
Second Generation CDNs
Infrastructure development, Caching, Mirroring, Multihoming
Replication, Intelligent routing, Edge computing
P2P, Cloud Computing, Energy Awareness
VoD, media streaming, mobile CDNs
Static and Dynamic Content
Server farms
Hierarchical caching
Caching proxy deployment
Improved Web server
Pre-evolutionary period
Late 90's
2002
2005
2008
2011
2011 onwards

CDN 3.0


Changed focus, increased functional ability, improved performance, user-centric
Pre CDN Evolutions
First Generation CDNs
Second Generation CDNs
Third Generation CDNs
Infrastructure development, Caching, Mirroring, Multihoming
Replication, Intelligent routing, Edge computing
P2P, Cloud Computing, Energy Awareness
Autonomic composition and delivery, QoE
Improved Web server
Caching proxy deployment
Hierarchical caching
Server farms
Static and Dynamic Content
VoD, media streaming, mobile CDNs
Community-based CDNs
Pre-evolutionary period
Late 90's
2002
2005
2008
2011
2011 onwards

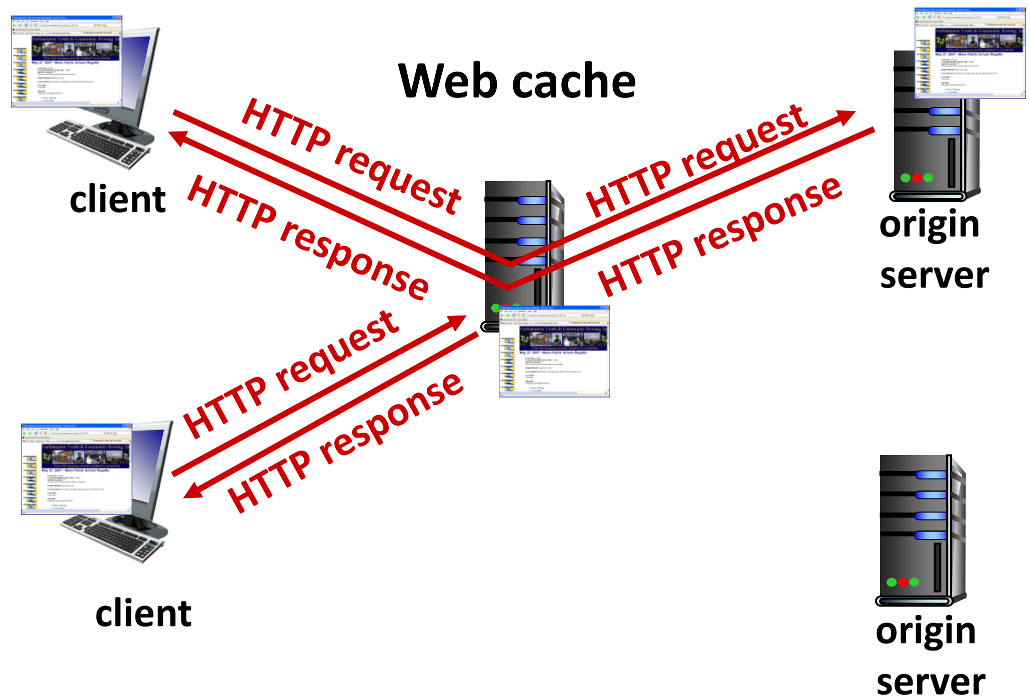
Agenda

- DNS
- P2P
- Content delivery
- **CDNs**
- Deployment
- Future

Web caches

- **Cache** servers to improve efficiency – the precursor to CDNs

- Browser sends all HTTP requests to cache
  - **If** object in cache: cache returns object
  - **else** cache requests object from origin server



Web caches

- Cache acts as both client and server
  - server for original requesting client
  - client to origin server
- Typically cache is installed by ISP (university, company, residential ISP)

Why Web caching?

- Reduce response time for client request
- Reduce traffic in the access network

Problems with web caching

- Caching proxies do not serve all Internet users
  - Size of all Web content is too large
  - Over 50% of all HTTP objects are uncacheable
- Content providers (say, Web servers) cannot rely on existence and correct implementation of caching proxies.
- Web content is **dynamic** and **customized**
  - E.g., can't cache banking content
- Accounting issues with caching proxies:
  - Example: www.cnn.com needs to know the number of hits to the advertisements displayed on the web page.

Content Delivery Networks (CDNs)

- Also sometimes called Content Distribution Network
- **Infrastructure**: large distributed system of servers deployed in multiple data centers across the Internet
- **Goal**: distribute content to end-users on a large-scale with high availability and high performance
- Simply put…

  A CDN provides a mechanism for

  - Replicating content on multiple servers in the Internet
  - Providing clients with a means to determine the servers that can deliver the content fastest.

Content Delivery Networks (CDNs)

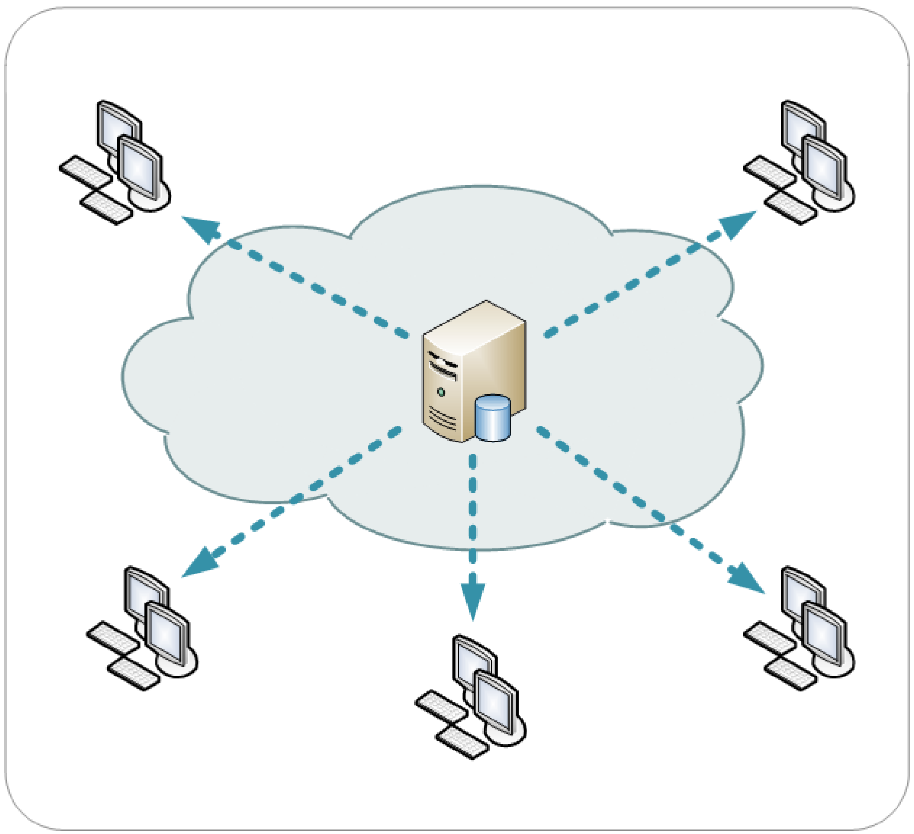
- The content providers are the CDN customers.
  - Media companies and e-commerce vendors pay CDN operators to deliver their content
  - CDN pays ISPs, carriers, and network operators for hosting its servers
- Usually associated with large web platforms
  - Netflix, Facebook, Google etc.

Contents served by CDNs

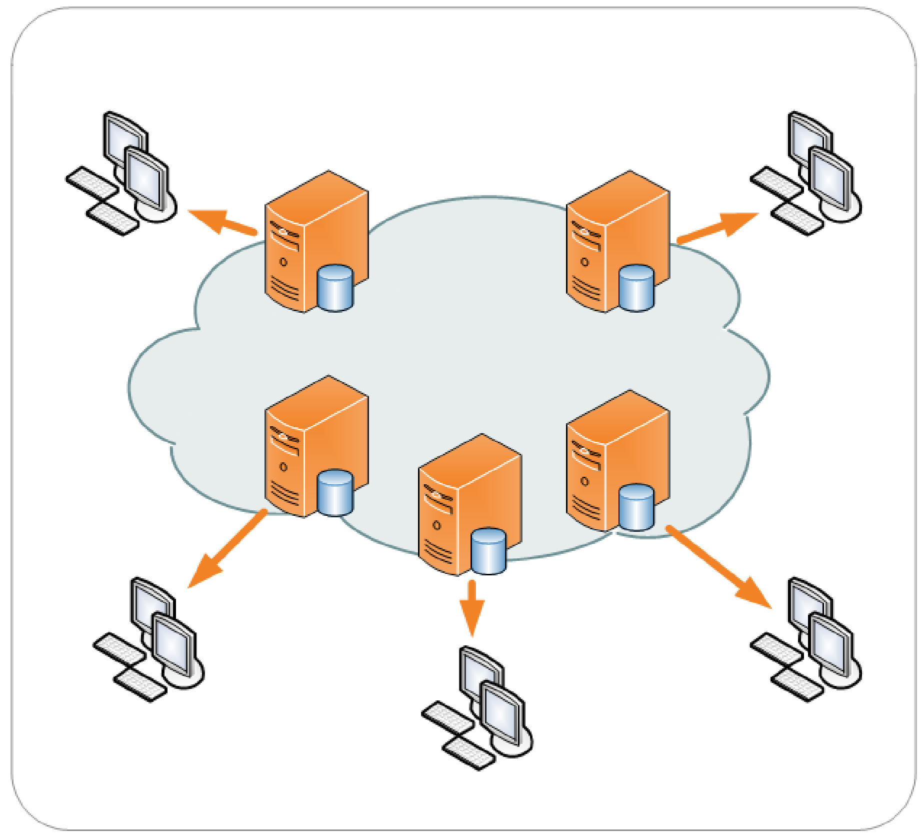
- Serve a large fraction of the Internet content:
  - Web objects (text, graphics and scripts)
  - Downloadable objects (media files, software, documents)
  - Applications (e-commerce, portals)
  - Live/on-demand streaming media (movies)
  - Social networks (images)
- At least half of the world's bits are delivered by CDNs
  - Probably closer to 80/90%

CDN vs Non-CDN model

Non-CDN model

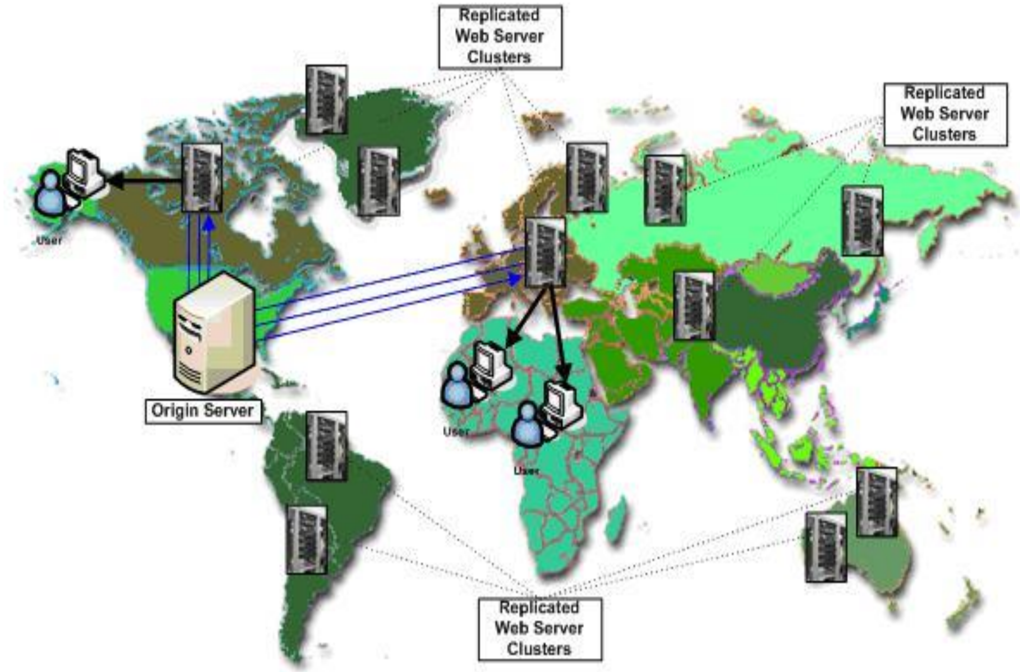


CDN model



CDN deployments

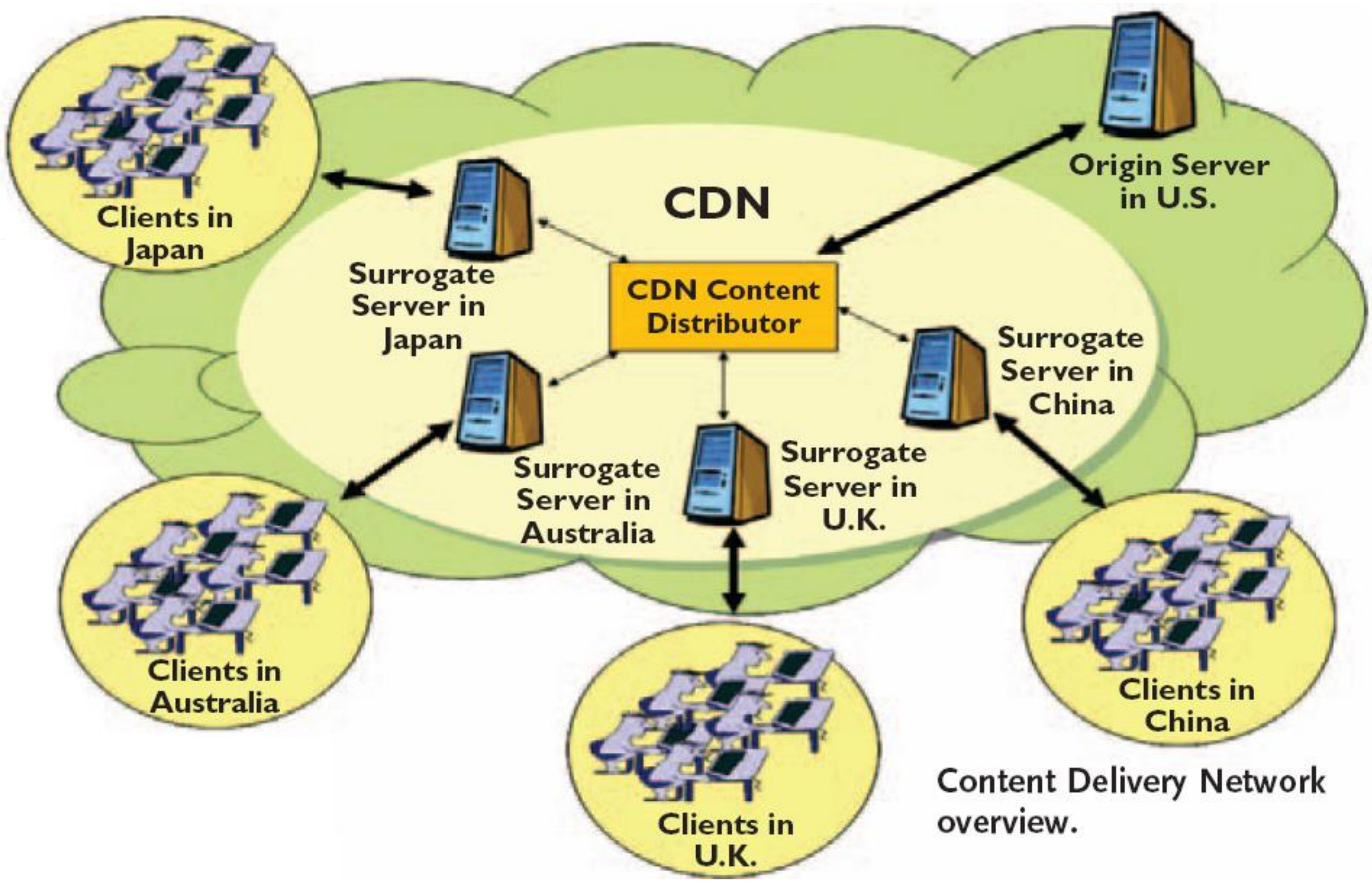
- CDN company deploys hundreds of CDN servers around the world, often deep inside ISP networks
  - Close to users
- CDN replicates its customers' content in CDN servers.
- When provider updates content, CDN updates servers



Examples of CDNs

- Akamai
  - 147K+ servers, 1200+ networks, 650+ cities, 92 countries
- Limelight
  - Well provisioned delivery centers, interconnected via a private fiber-optic connected to 700+ access networks
- ChinaCache
  - 30K+ servers, 150+ major cities
- Edgecast
  - 30+ PoPs, 5 continents, 2000+ direct connections
- Others
  - Google, Facebook, AWS, AT&T, Level3, Brokers

A Big Picture



Content Delivery Network overview.

Why use a CDN?

- Reduces latency to users
- Increase security against Denial of Service attacks
- Increases scalability
- Can be cheaper and easier to manage
- Bypass traffic jams on the web
  - Requested data is close to the clients
  - Avoid traversing bottleneck links
- Reduce load on the original server - easy load balancing

How does CDN work?

- Users send requests to origin server
- Requests somehow intercepted by redirection service
- Redirection service forwards user’s request to the “best” CDN content server
- Content served from the CDN content server

Server Selection

- Which server?
  - Lowest load: to balance load on servers
  - Best performance: to improve client performance
    - Based on Geography? RTT? Throughput? Load?
  - Any alive node: to provide fault tolerance
- How to direct clients to a particular server?
  - As part of routing: anycast, cluster load balancing
  - As part of application: HTTP redirect
  - As part of naming: DNS

Trade-offs between approaches

- Routing based (IP anycast)
  - **Pros**: Transparent to clients, works when browsers cache failed addresses, circumvents many routing issues
  - **Cons**: Little control, complex, scalability, TCP can't recover,
- Application based (HTTP redirects)
  - **Pros**: Application-level, fine-grained control
  - **Cons**: Additional load and RTTs, hard to cache
- Naming based (DNS selection)
  - **Pros**: Well-suitable for caching, reduce RTTs
  - **Cons**: Request by resolver not client, request for domain not URL, hidden load factor of resolver's population
  - Much of this data can be estimated "over time"

CNAME redirection via DNS

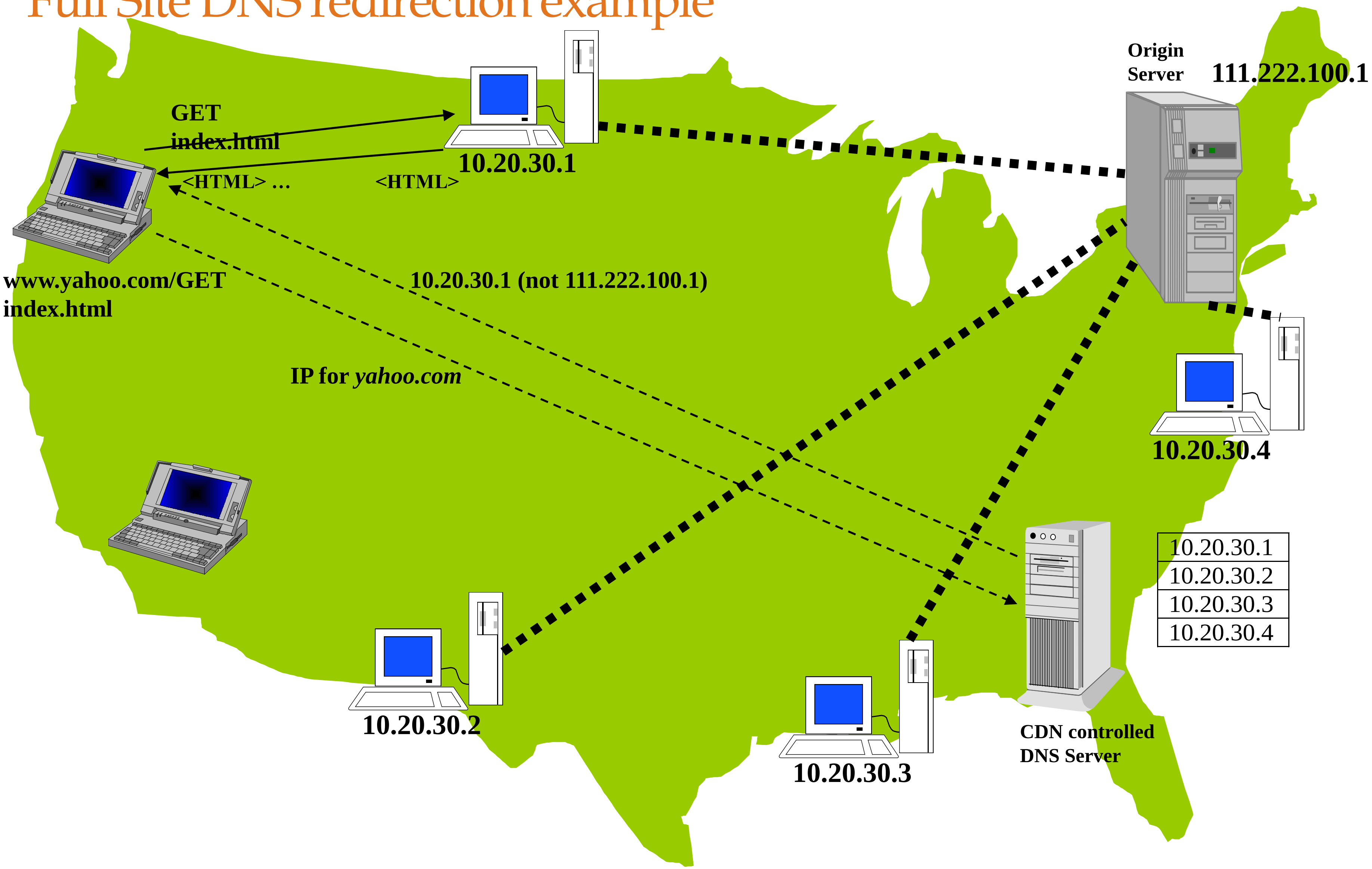
- Used to **redirect** clients to “nearby” server
- Based on many criteria
  - Latency to client
  - Load balancing
  - Available servers
- Load balancing tries to balance clients across many servers to avoid hotspot

CNAME redirection via DNS

- CNAME = canonical name
- Popular for large CDNs, e.g., Akamai

```
; <<>> DiG 9.8.3-P1 <<>> www.whitehouse.gov
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 11921
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;www.whitehouse.gov.                IN      A
;; ANSWER SECTION:
www.whitehouse.gov.     3596        IN      CNAME   www.whitehouse.gov.edgesuite.net.
www.whitehouse.gov.edgesuite.net. 896 IN    CNAME   www.eop-edge-lb.akadns.net.
www.eop-edge-lb.akadns.net. 300     IN      CNAME   a1128.dsch.akamai.net.
a1128.dsch.akamai.net.  20          IN      A       195.59.54.235
a1128.dsch.akamai.net.  20          IN      A       195.59.126.161
```

Full Site DNS redirection example



Vendors: Adero(Full), Akami and Digital Island (Partial)

Two DNS Redirection Types

- Full redirection
  - Any request for origin server is redirected to CDN
  - Basically, CDN takes control of content provider's DNS zone
  - **Benefit**: All requests are automatically redirected
  - **Disadvantage**: May send lots of traffic to CDN, hence expensive for the content provider, $ per byte
- Partial redirection
  - Content provider marks which objects are to be served from CDN
  - Typically, larger objects like images are selected
  - Refer to images as: <img src=http://cdn.com/foo/bar/img.gif>
  - When client wants to retrieve image, DNS request for cdn.com gets resolved by CDN and image is fetched from the selected content server
  - **Pro**: Fine-grained control over what gets delivered
  - **Con**: Have to (manually) mark content for CDN

Agenda

- DNS
- P2P
- Content delivery
- CDNs
- **Deployment**
- Future

Where is popular content hosted?

- Rely on active measurements:
- Sample popular hostnames from Alexa.com
- Ask DNS from multiple vantage points (150+ ISPs around the world)
- Sample different types of “objects”
  - Popular hostnames
  - Embedded files
  - Unpopular

Where is popular content?

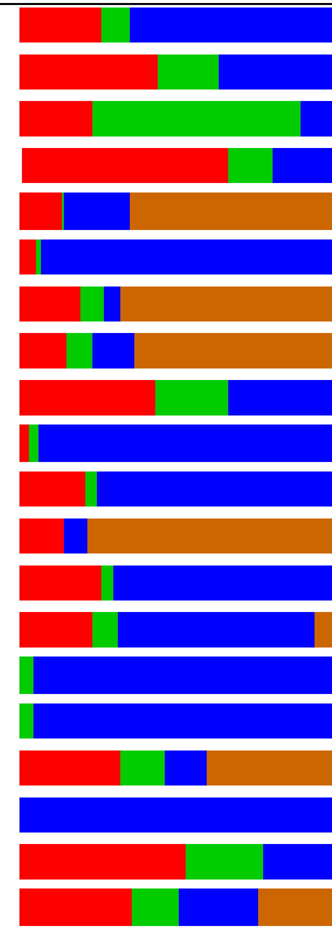
- 1st: California
- 2nd: China
- USA: 9 among top 20
- Other Westerns countries: limited own content

Rank	Country	Potential	Normalized potential
1	USA (CA)	0.254	0.108
2	China	0.128	0.107
3	USA (TX)	0.190	0.061
4	Germany	0.183	0.058
5	Japan	0.163	0.051
6	France	0.146	0.034
7	Great Britain	0.157	0.030
8	Netherlands	0.144	0.029
9	USA (WA)	0.135	0.027
10	USA (unknown)	0.164	0.027
11	Russia	0.038	0.027
12	USA (NY)	0.130	0.026
13	Italy	0.122	0.018
14	USA (NJ)	0.125	0.016
15	Canada	0.028	0.015
16	USA (IL)	0.116	0.014
17	Australia	0.118	0.013
18	Spain	0.116	0.013
19	USA (UT)	0.111	0.012
20	USA (CO)	0.113	0.012

Top western CDNs "clusters"

CDN "cluster" ~ set of servers that serve the same hostnames

Rank	# hostnames	# ASes	# prefixes	owner	content mix
1	476	79	294	Akamai	
2	161	70	216	Akamai	
3	108	1	45	Google	
4	70	35	137	Akamai	
5	70	1	45	Google	
6	57	6	15	Limelight	
7	57	1	1	ThePlanet	
8	53	1	1	ThePlanet	
9	49	34	123	Akamai	
10	34	1	2	Skyrock	
11	29	6	17	Cotendo	
12	28	4	5	Wordpress	
13	27	6	21	Footprint	
14	26	1	1	Ravand	
15	23	1	1	Xanga	
16	22	1	4	Edgecast	
17	22	1	1	ThePlanet	
18	21	1	1	ivwbox.de	
19	21	1	5	AOL	
20	20	1	1	Leaseweb	



only on TOP, both on TOP and EMBEDDED, only on EMBEDDED, TAIL.

Who has power?

Who serves content?

Rank	AS name	CMI
1	NTT America	0.070
2	Tinet	0.029
3	Global Crossing	0.034
4	KDDI	0.025
5	Akamai Europe	0.019
6	TeliaNet Global	0.027
7	Deutsche Telekom	0.033
8	Korea Telecom	0.030
9	Qwest	0.036
10	Bandcon	0.045
11	Cable and Wireless	0.021
12	SingTel Optus	0.019
13	Akamai	0.018
14	France Telecom - Orange	0.017
15	Internode	0.017
16	Comcast	0.017
17	StarHub	0.018
18	nLayer	0.020
19	Beyond The Network	0.018
20	TATA	0.023

Who serves content that no one else has?

Rank	AS name	CMI
1	Chinanet	0.699
2	Google	0.996
3	ThePlanet.com	0.985
4	SoftLayer	0.967
5	China169 Backbone	0.576
6	Level 3	0.109
7	China Telecom	0.470
8	Rackspace	0.954
9	1&1 Internet	0.969
10	OVH	0.969
11	NTT America	0.070
12	EdgeCast	0.688
13	GoDaddy.com	0.969
14	Savvis	0.785
15	China169 Beijing	0.706
16	Amazon.com	0.895
17	LEASEWEB	0.942
18	Cogent	0.687
19	Hetzner Online	0.962
20	AOL	0.932

The "P2P CDN"

#	Country	U. Peers
1	US	1379462
2	ES	887480
3	GB	800308
4	CA	551820
5	IN	514246
6	AU	322009
7	BR	318294
8	IT	295339
9	PL	288780
10	PT	220124

TOP 10 ISPS (BT VIDEO USER)

	AS#	Peers	AS Name-Internet Service Provider
1	3352	165469	TELEFONICA-DATA-ESPANA(TDE)
2	3662	129047	DNEO-OSP7-COMCAST CABLE
3	6461	127297	MFNX MFN-METROMEIDA FIBER
4	2119	113597	TELENOR-NEXTEL T.NET
5	19262	101390	VZGNI-TRANSIT-Verizon ISP
6	3301	97658	TELIANET-SWEDEN TELIANET
7	3462	96564	HINET-DATA CBG
8	4134	87392	CHINANET-BACKBONE
9	6327	86964	SHAW-SHAW COMMUNICATION
10	174	74453	COGENT COGENT/PSI

LET'S LOOK AT SOME MAPS…

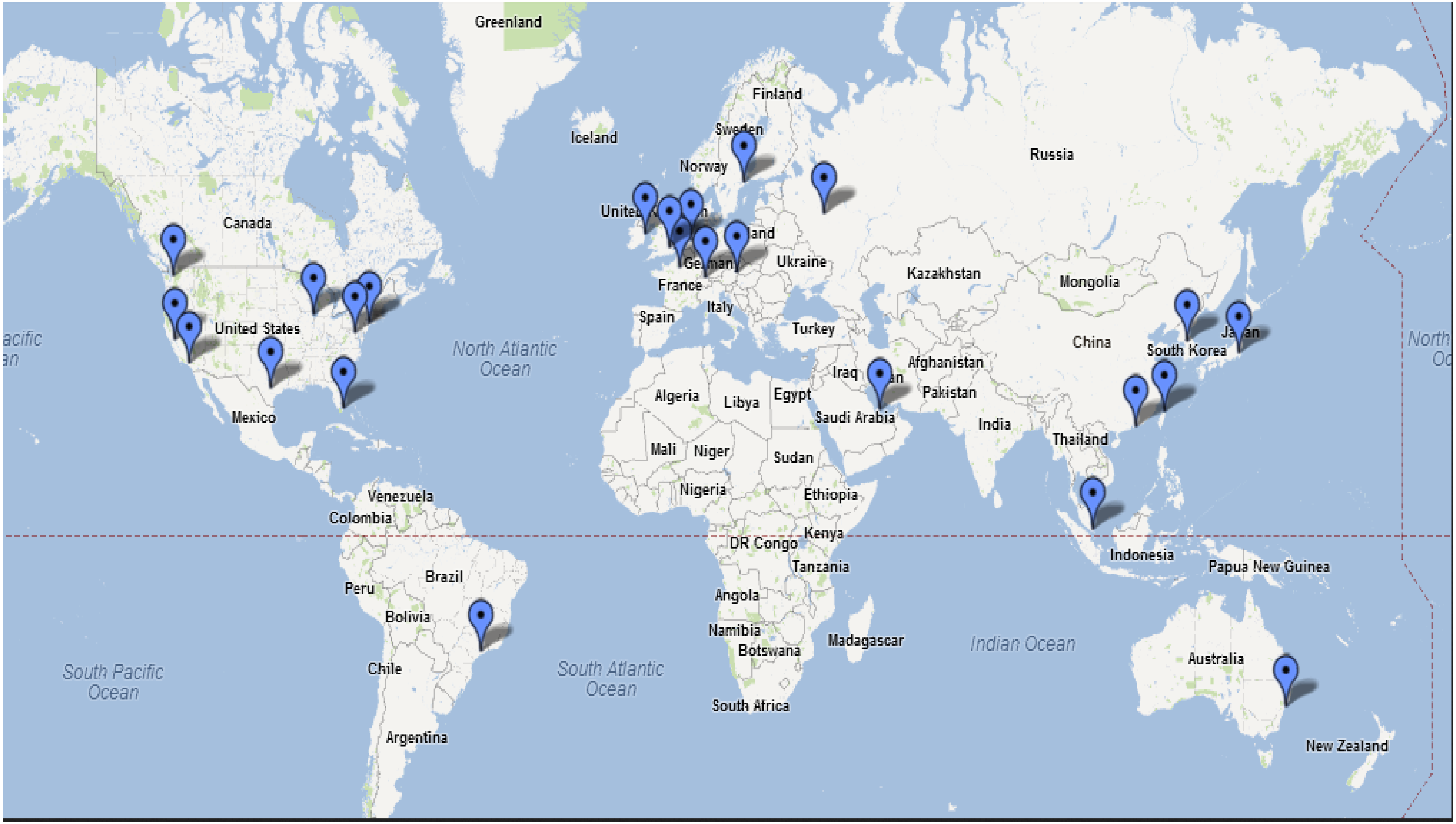
World data centers

http://www.datacentermap.com/

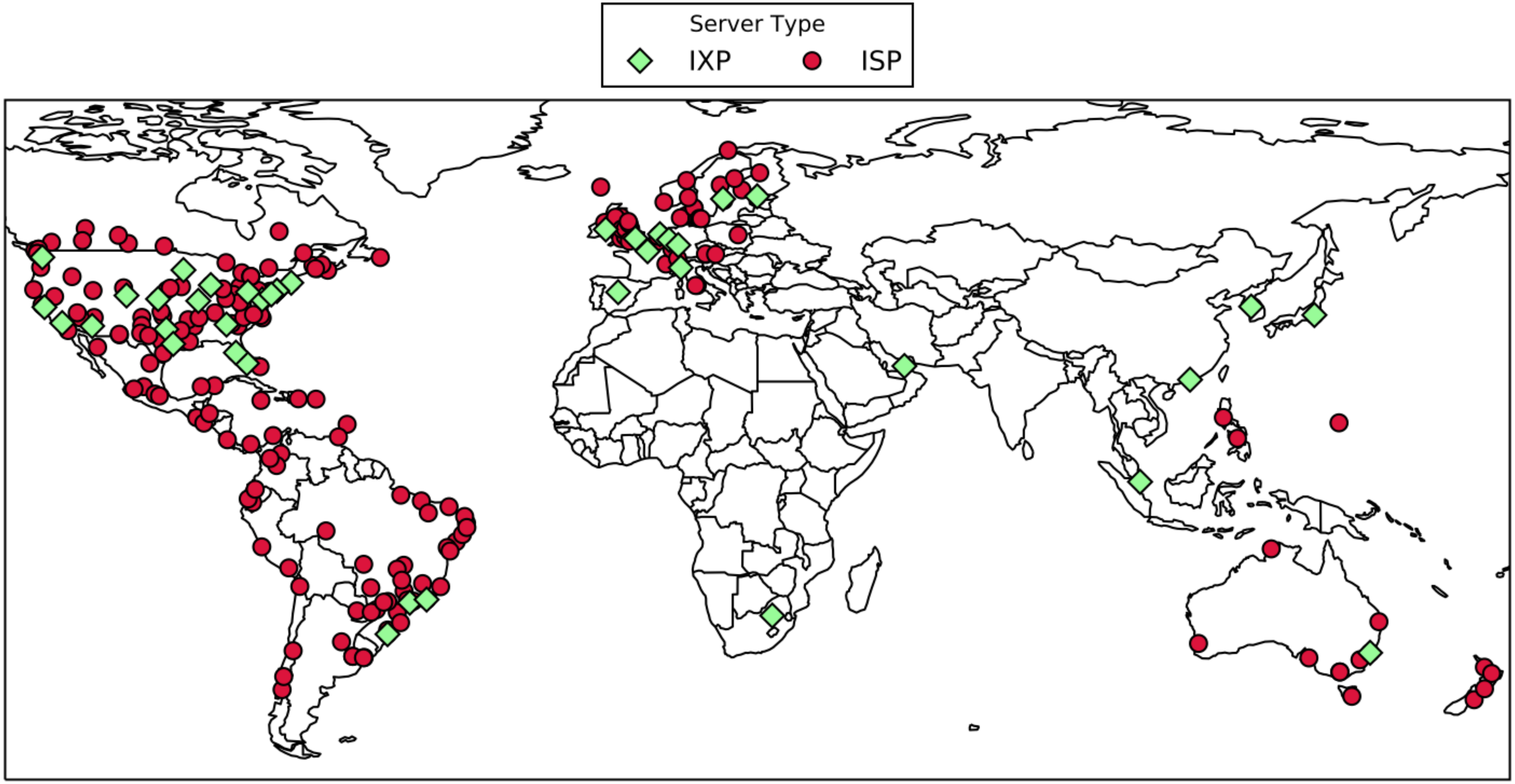
Google

http://royal.pingdom.com/2008/04/11/map-of-all-google-data-center-locations/

Windows Azure

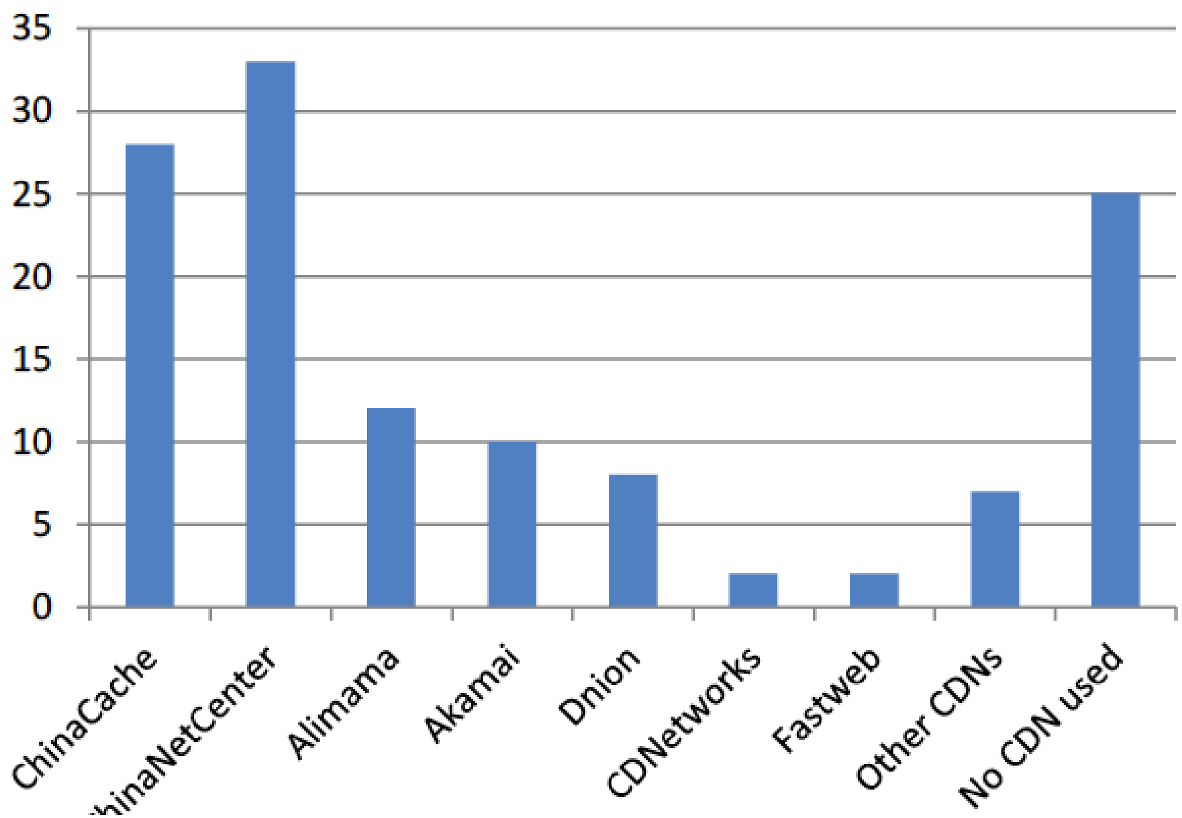


Netflix



CDNs in China

- **Selected top 100 Chinese websites (Alexa.com)**
- **Which CDNs serve content from these websites?**



ChinaCache

Largest China Network by Coverage and Capacity

ChinaCache's content delivery network covers over 120 cities throughout China with over 350 nodes and 12,000 servers.

We're 15x larger in both capacity and reach compared to all other global content delivery network providers in China combined.

ChinaNetCenter

Agenda

- DNS
- P2P
- Content delivery
- CDNs
- Deployment
- **Future**

Key future challenges

- Mobile networks
  - Latency in cell networks is higher
  - Internal network structure is more opaque
- Video
  - 4k/8k UHD = 16-30K Kbps compressed
  - 25K Tbps projected
  - Big data center networks not enough (5 Tbps each)
  - Multicast (from end systems) potential solution

CDN 2.0

- Hybrid CDNs: Akamai, PPTV
- Cloud-based VoD: NetFlix
- Meta-CDNs, e.g., Conviva
- Virtual CDNs through ISP micro-datacenters