NTT

Innovative R&D by NTT

Advances in Reducing Web Response Time

NTT Network Technology Laboratories

Noriaki Kamiyama

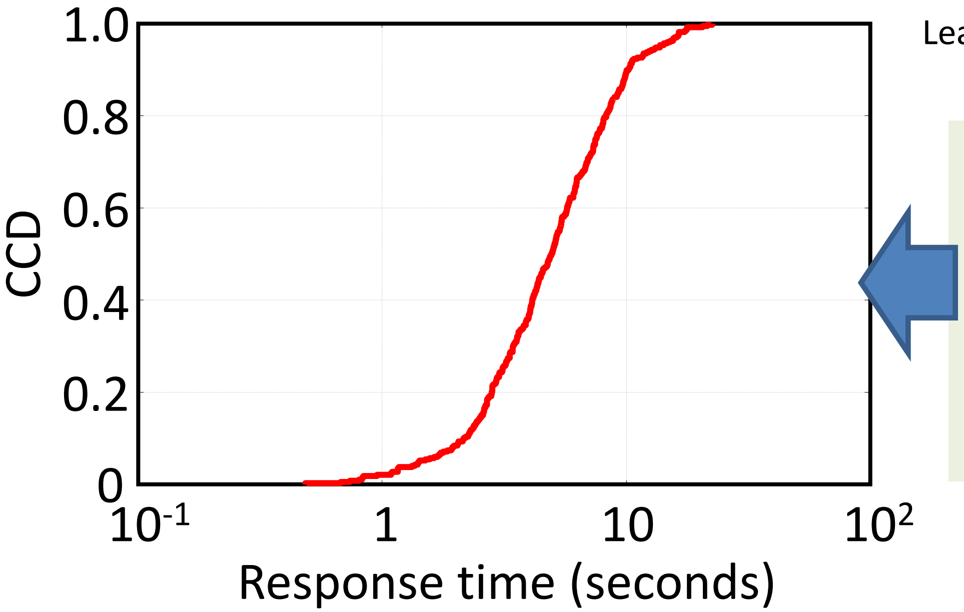
2016. 6. 14

Increase of Web Response Time

- Web response time*: longer than 5 seconds in 50% webpages, and longer than 10 seconds in 10% webpages
- Amazon increased revenue 1% for every 0.1 second reduction in web response time.**
- Need to reducing web response time

*Web response time: waiting time after clicking hyperlink until entire part of webpage is shown

**R. Kohavi and R. Longbotham, Online Experiments: Lessons Learned, IEEE Computer, Vol.40, No. 9, pp.103-105, Sep. 2007.



Complementary cumulative distribution (CCD) of web response time of most popular 1,000 websites when accessing from Tokyo, Japan, in June 2015

Tutorial Overview

- Purpose
  - Abstract basic technologies supporting web browsing services
  - Summarize possible factors degrading web response time
  - Analyze tendencies of web content deployment and effect of prioritizing in caching objects based on web categories
  - Overview various standard and advanced approaches reducing web response time

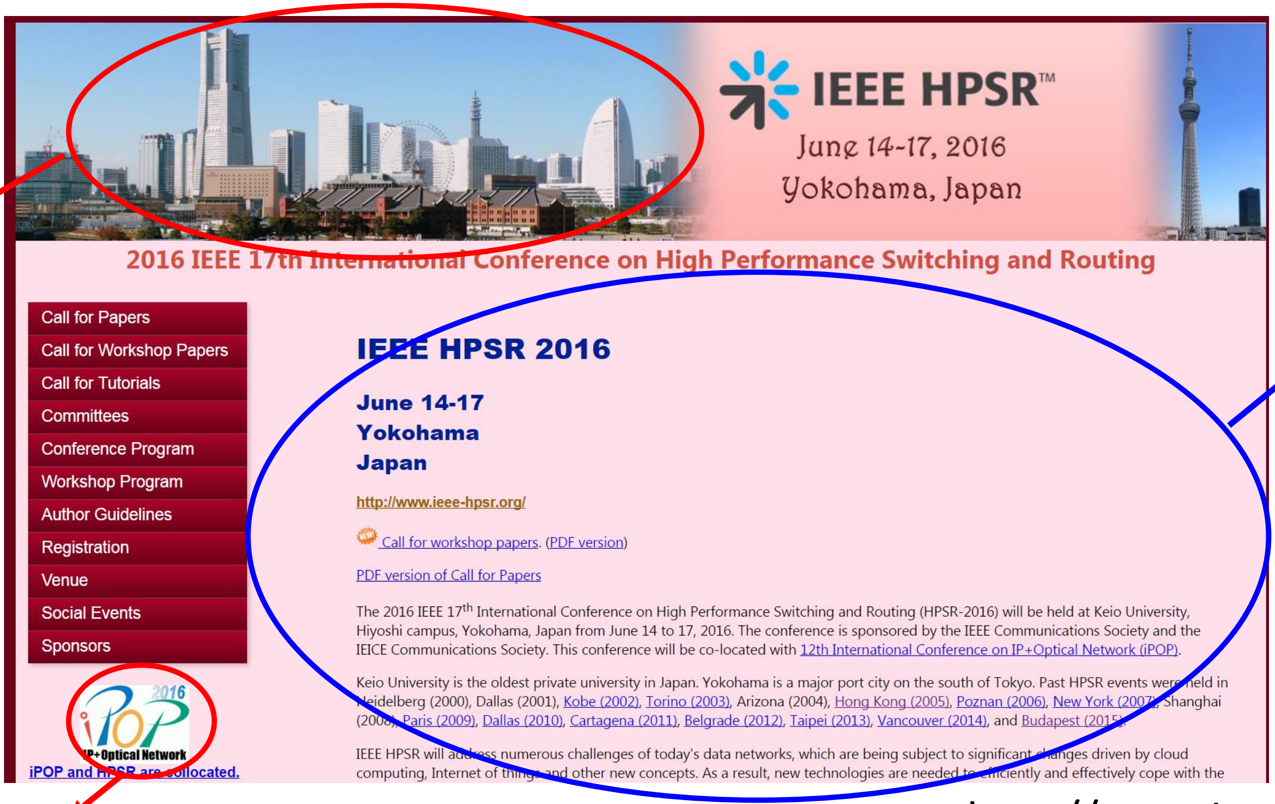
- Organization
  - Mechanism providing web browsing services
  - Possible factors degrading web response time
  - Geographical tendency of web content deployment
  - Approaches reducing web response time

- Mechanism providing web browsing services
  - Overview
  - HTTP
  - CDN
  - Objects

- Possible factors degrading web response time
- Geographical tendency of web content deployment
- Approaches reducing web response time

Multiple Objects Constructing Webpage

- One webpage consists of one main body file (HTML: HyperText Markup Language) and multiple data object files



2016 IEEE 17th International Conference on High Performance Switching and Routing

iPOP and HPSR are collocated.

IEEE HPSR 2016

June 14-17
Yokohama
Japan

http://www.ieee-hpsr.org/

Call for workshop papers. (PDF version)

PDF version of Call for Papers

The 2016 IEEE 17th International Conference on High Performance Switching and Routing (HPSR-2016) will be held at Keio University, Hiyoshi campus, Yokohama, Japan from June 14 to 17, 2016. The conference is sponsored by the IEEE Communications Society and the IEICE Communications Society. This conference will be co-located with 12th International Conference on IP+Optical Network (iPOP).

Keio University is the oldest private university in Japan. Yokohama is a major port city on the south of Tokyo. Past HPSR events were held in Heidelberg (2000), Dallas (2001), Kobe (2002), Torino (2003), Arizona (2004), Hong Kong (2005), Poznan (2006), New York (2007), Shanghai (2008), Paris (2009), Dallas (2010), Cartagena (2011), Belgrade (2012), Taipei (2013), Vancouver (2014), and Budapest (2015).

IEEE HPSR will address numerous challenges of today's data networks, which are being subject to significant changes driven by cloud computing, Internet of things and other new concepts. As a result, new technologies are needed to efficiently and effectively cope with the

Data object:
Photo of
Minatomirai

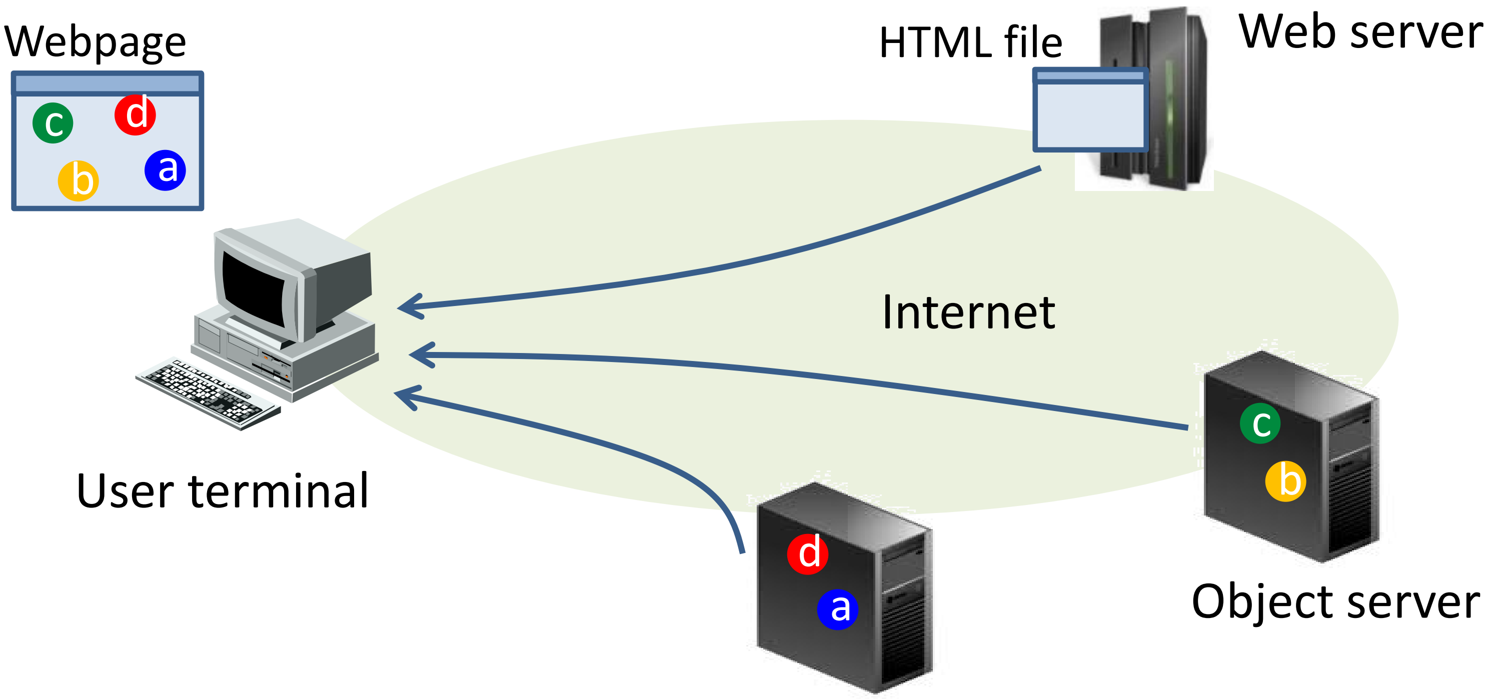
Main body

Data object:
logo of iPOP

http://www.ieee-hpsr.org/

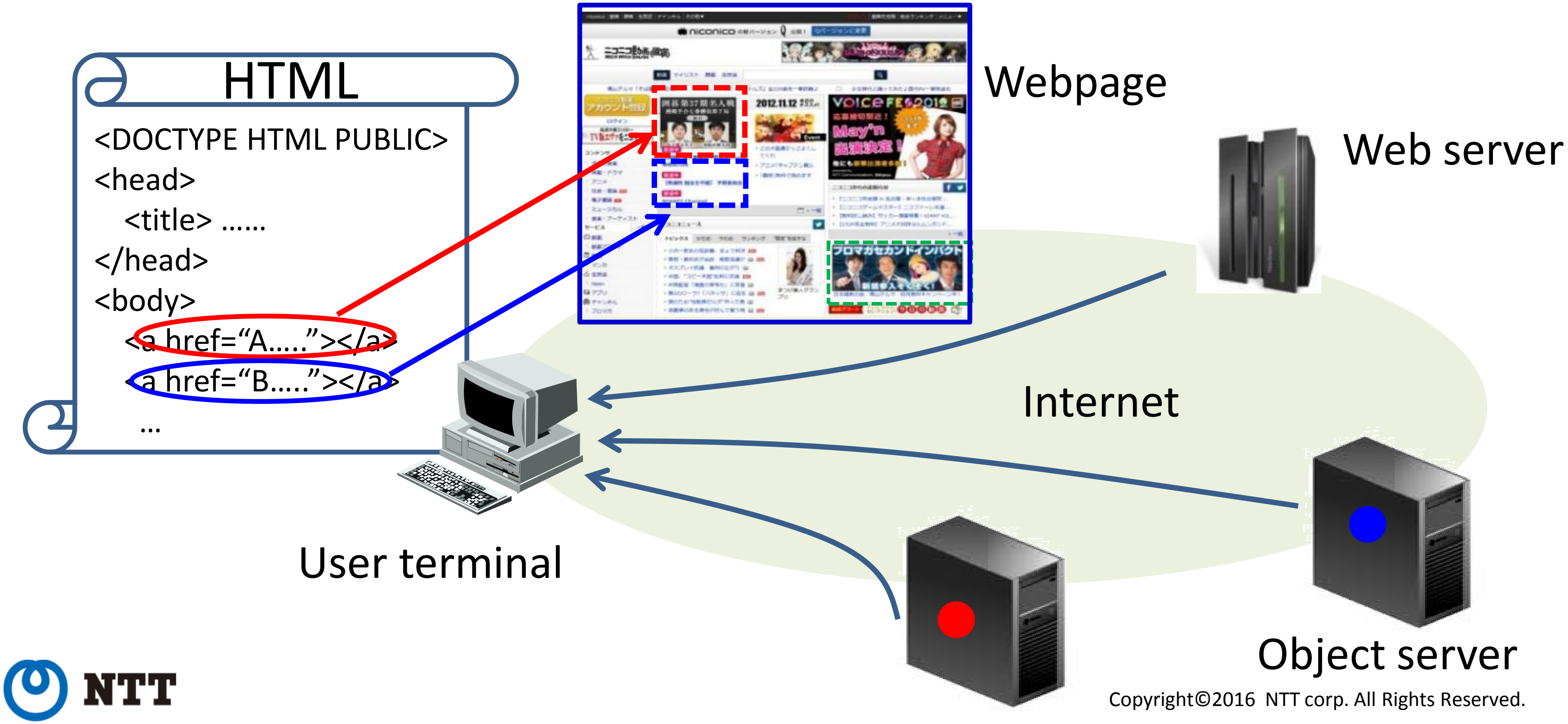
Components in Web Browsing Service

- Web server: providing HTML files and data objects of webpages
- Object server: providing data objects
- Web browser: obtaining data objects and displaying webpages at user terminal



Rough Overview of Procedure at Web browsers

- Process executed at Web browser
  - Getting HTML file from web server using HTTP (hypertext transfer protocol)
  - Pursing HTML file
  - Obtaining data objects embedded in HTML file from object servers using HTTP
  - Rendering webpage



```
<DOCTYPE HTML PUBLIC>
<head>
   <title> ……
</head>
<body>
   <a href="A….."></a>
   <a href="B….."></a>
   …
```

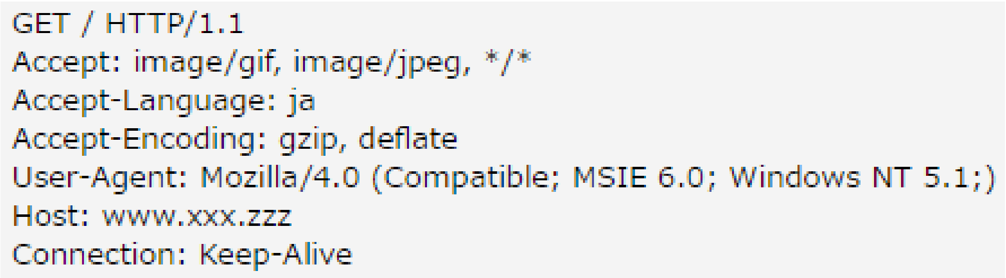
- Mechanism providing web browsing services
  - Overview
  - HTTP
  - CDN
  - Objects

- Possible factors degrading web response time
- Geographical tendency of web content deployment
- Approaches reducing web response time

Getting Objects on HTTP Sessions

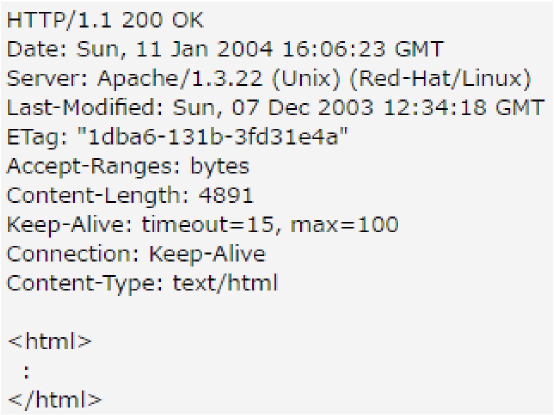
- Using HTTP on TCP to obtain HTML file and objects
  - Default port number: 80
  - User terminals request objects by sending “HTTP request” messages to object servers.
  - Object servers send objects to user terminals by “HTTP response” messages.

HTTP request



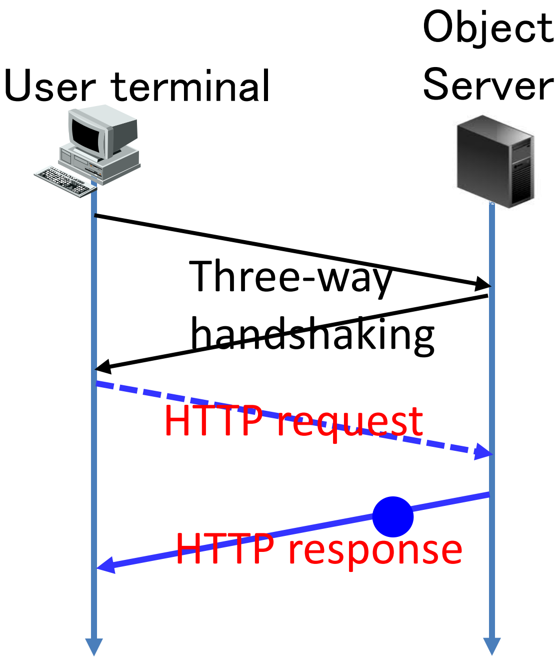
```
GET / HTTP/1.1
Accept: image/gif, image/jpeg, */*
Accept-Language: ja
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (Compatible; MSIE 6.0; Windows NT 5.1;)
Host: www.xxx.zzz
Connection: Keep-Alive
```

HTTP response



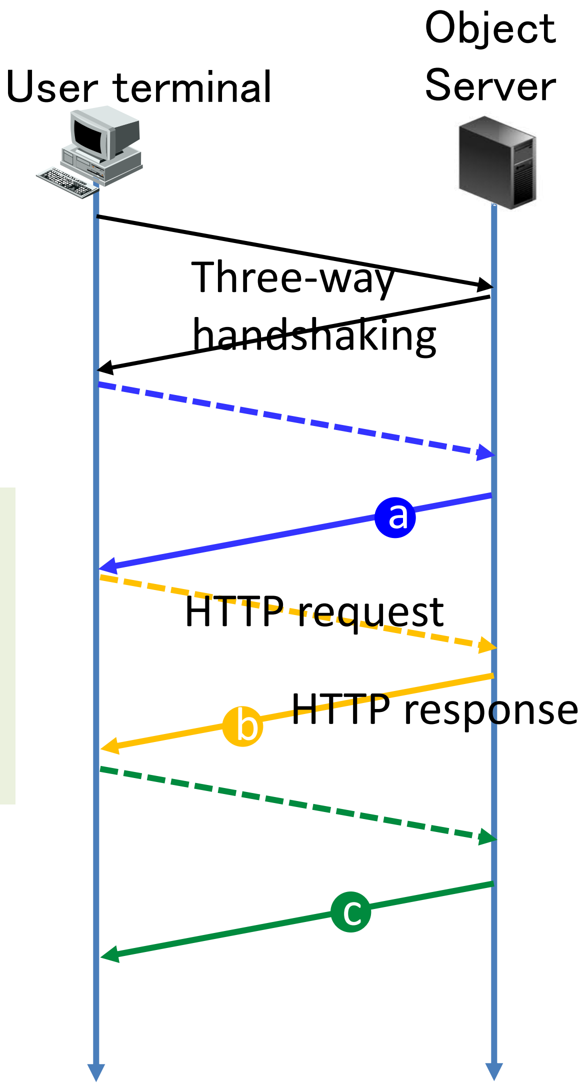
```
HTTP/1.1 200 OK
Date: Sun, 11 Jan 2004 16:06:23 GMT
Server: Apache/1.3.22 (Unix) (Red-Hat/Linux)
Last-Modified: Sun, 07 Dec 2003 12:34:18 GMT
ETag: "1dba6-131b-3fd31e4a"
Accept-Ranges: bytes
Content-Length: 4891
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html

<html>
  :
</html>
```



HTTP/1.1

- HTTP versions
  - HTTP/0.9: initial version
  - HTTP/1.0: supports HTTP cookie
  - HTTP/1.1: supports persistent connections and HTTP pipelining

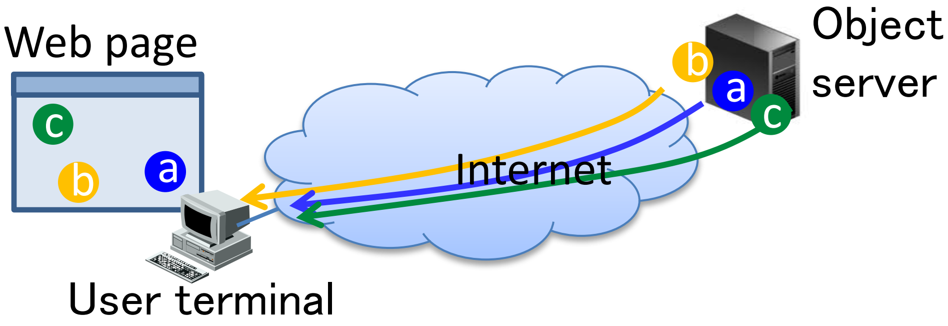


Persistent connections:

- Enabling user terminals to download multiple objects on a single TCP session
- No need to make three-way handshaking for obtaining second to last objects

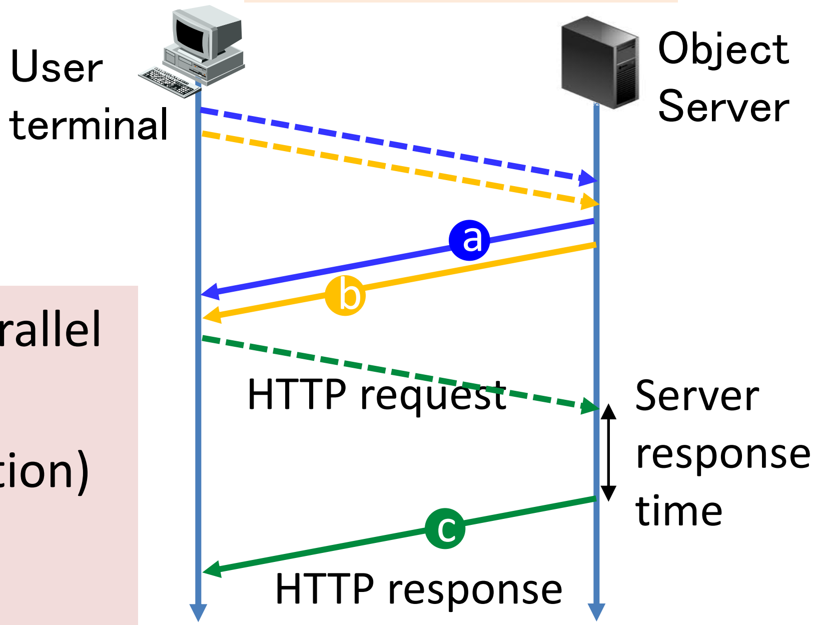
HTTP Pipelining

- Multiple objects are transmitted on a single TCP session between user terminal (UT) and object server. (Similar with HTTP persistent)
- UT can send multiple HTTP requests to same object server without waiting HTTP responses in parallel.
- UT cannot infer to which HTTP request each packet belongs, so server needs to send HTTP response in the order of received requests.



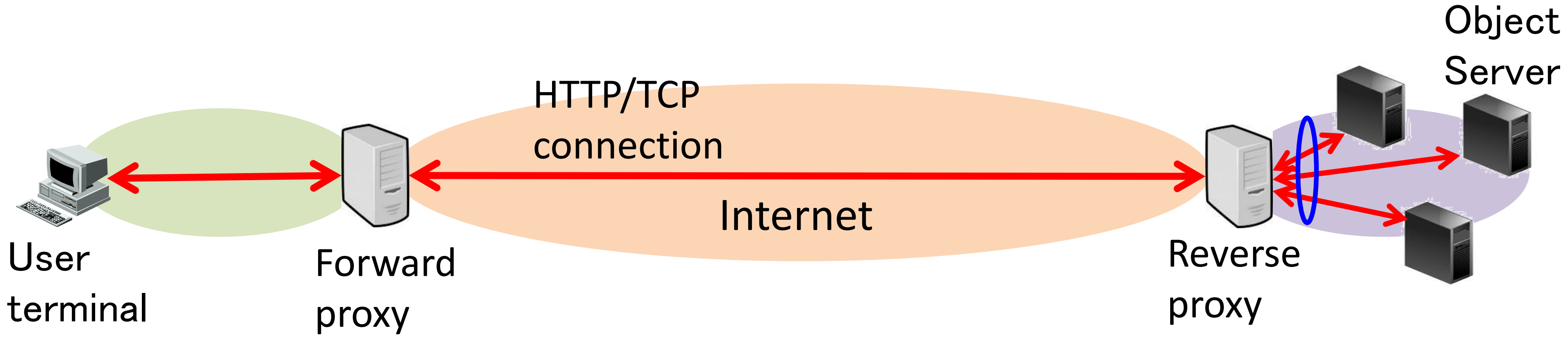
- To avoid overload at object servers, limit parallel transmission count ≤ P
  - P = 2 (suggested by HTTP/1.1 specification)
  - P = 4 (Safari 3, Opera 9)
  - P = 6 (Explore 8, Firefox 3)

Example when P = 2



Proxy Servers

- Two type of proxy servers:
  - Forward proxy:
    - Be placed in each access network and enterprise networks
    - Reducing response time by caching objects
  - Reverse proxy:
    - Be placed at gateway of server networks
    - Balancing load of web and object servers by connecting them in round robin
- Separate HTTP/TCP connections into two parts

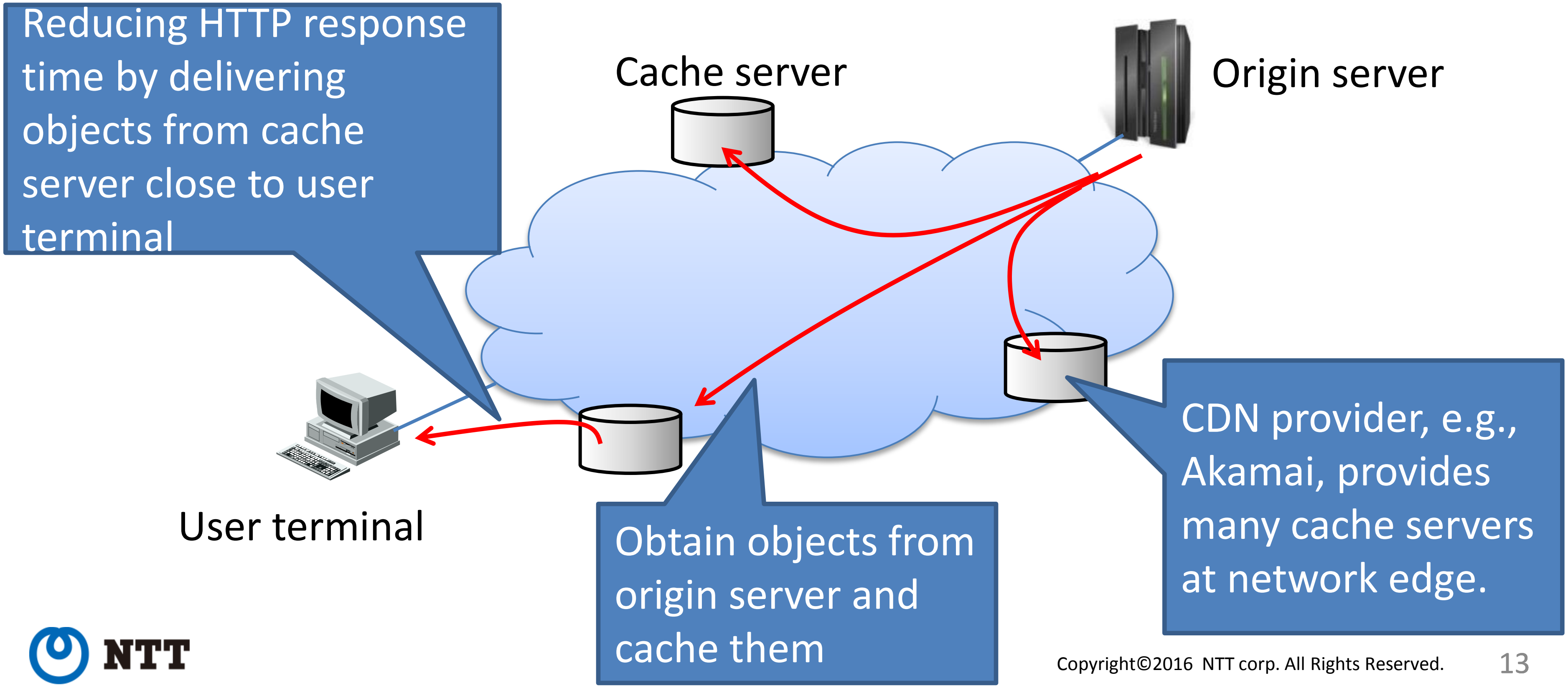


- Mechanism providing web browsing services
  - Overview
  - HTTP
  - CDN
  - Objects
- Possible factors degrading web response time
- Geographical tendency of web content deployment
- Approaches reducing web response time

CDN (content delivery network)

- CDN is a platform delivering web objects from cache servers located close to user terminals.
- 74% of 1,000 most popular websites use CDN*, and CDN is most common technique for reducing HTTP response time.

*J. Ott, et al., Content Delivery and the Natural Evolution of DNS, ACM IMC 2012



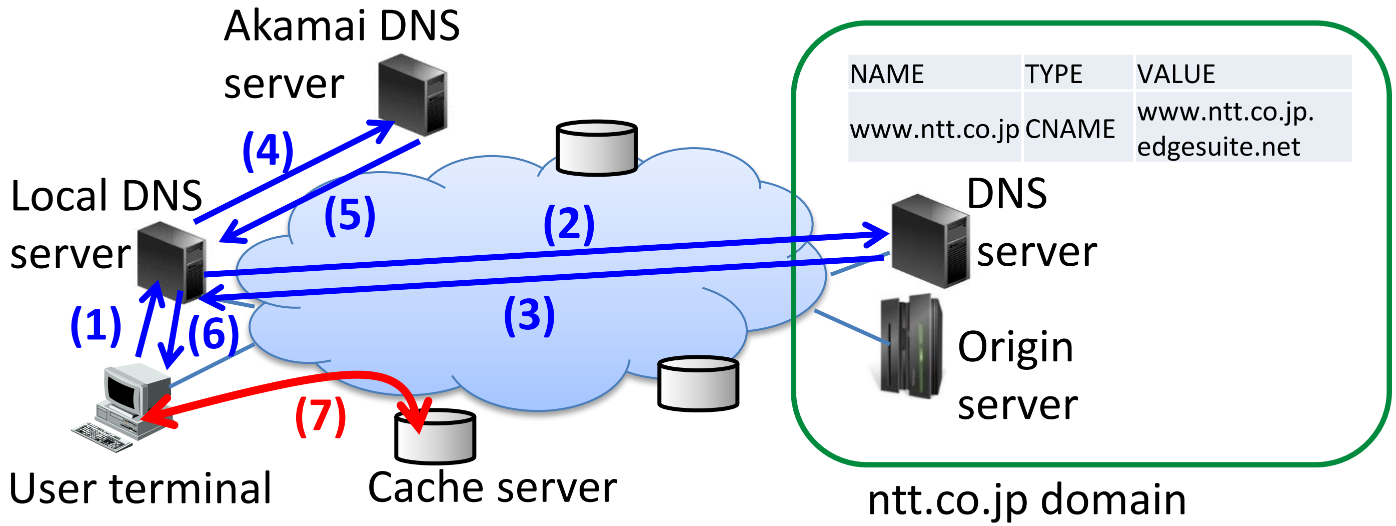
Reducing HTTP response time by delivering objects from cache server close to user terminal

Obtain objects from origin server and cache them

CDN provider, e.g., Akamai, provides many cache servers at network edge.

Process of Name Resolution of Akamai

- Cache server is selected through name resolution process by DNS.
- Selection policy of cache servers is concealed.



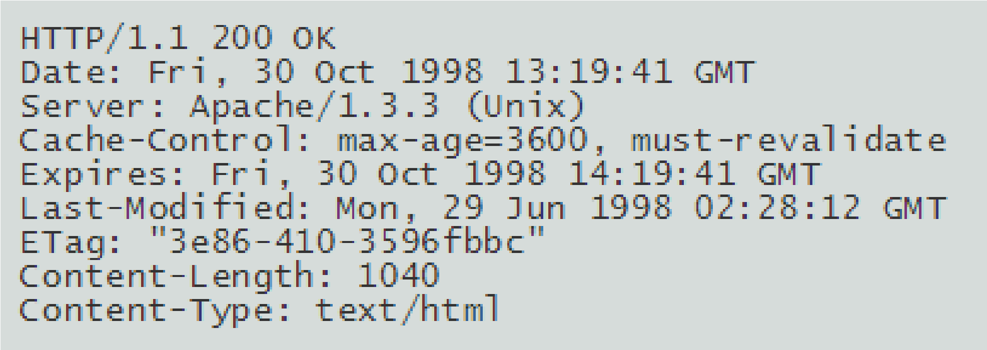
NAME	TYPE	VALUE
www.ntt.co.jp	CNAME	www.ntt.co.jp. edgesuite.net

1. UT requests name resolution (NR) of “www.ntt.co.jp” to local DNS (LDNS).
2. LDNS requests NR to DNS server of “ntt.co.jp” domain.
3. LDNS DNS server of “ntt.co.jp” returns its CNAME to LDNS.
4. LDNS requests NR of “www.ntt.co.jp.edgesuite.net” to Akamai DNS server.
5. Akamai DNS server returns IP address of selected cache server to LDNS.
6. LDNS returns IP address of selected cache server to UT.
7. UT downloads content from selected cache server.

Validation of Cached Objects

- Objects are cached with metadata including information for judging validation
  - Expires: explicitly indicating expiration date and time
  - Last-Modified: indicating last modified date and time
  - Etag (entity tag): code updated at modifying object
- Also applicable to proxy caches and browser caches

Comparing ones cached and ones at origin servers



```
HTTP/1.1 200 OK
Date: Fri, 30 Oct 1998 13:19:41 GMT
Server: Apache/1.3.3 (Unix)
Cache-Control: max-age=3600, must-revalidate
Expires: Fri, 30 Oct 1998 14:19:41 GMT
Last-Modified: Mon, 29 Jun 1998 02:28:12 GMT
ETag: "3e86-410-3596fbbc"
Content-Length: 1040
Content-Type: text/html
```

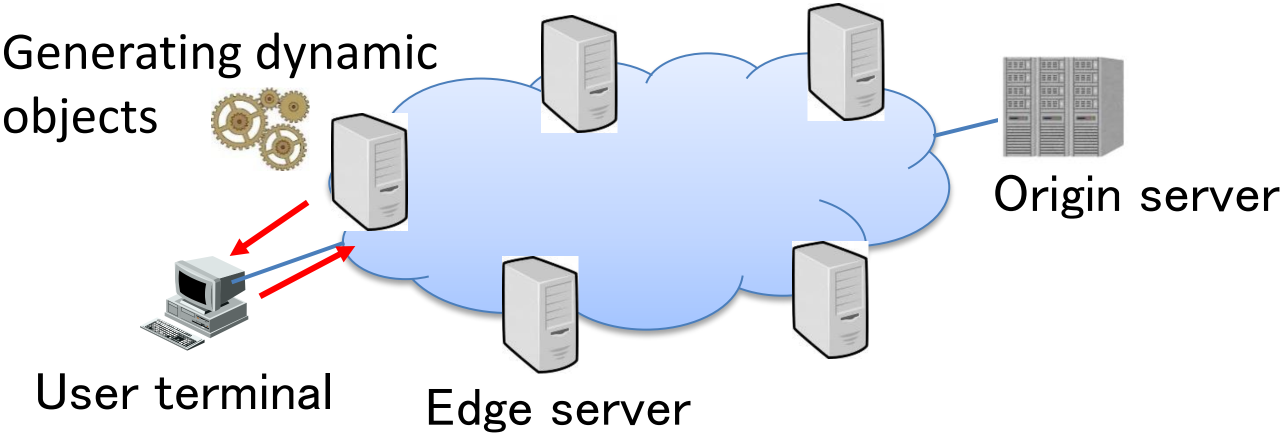
Edge Computing

- Many objects are dynamically generated in modern webpages.
- Edge computing* is effective to deliver dynamic objects efficiently.

*M. Rabinovich, et al., Computing on the Edge: A Platform for Replicating Internet Applications," WCW 2003.
A. Davis, et al., EdgeComputing: Extending Enterprise Applications to the Edge of the Internet, WWW 2004.

Edge servers located at edge nodes

- ■ Caches application codes for generating dynamic objects
- ■ Dynamically generates objects for user requests



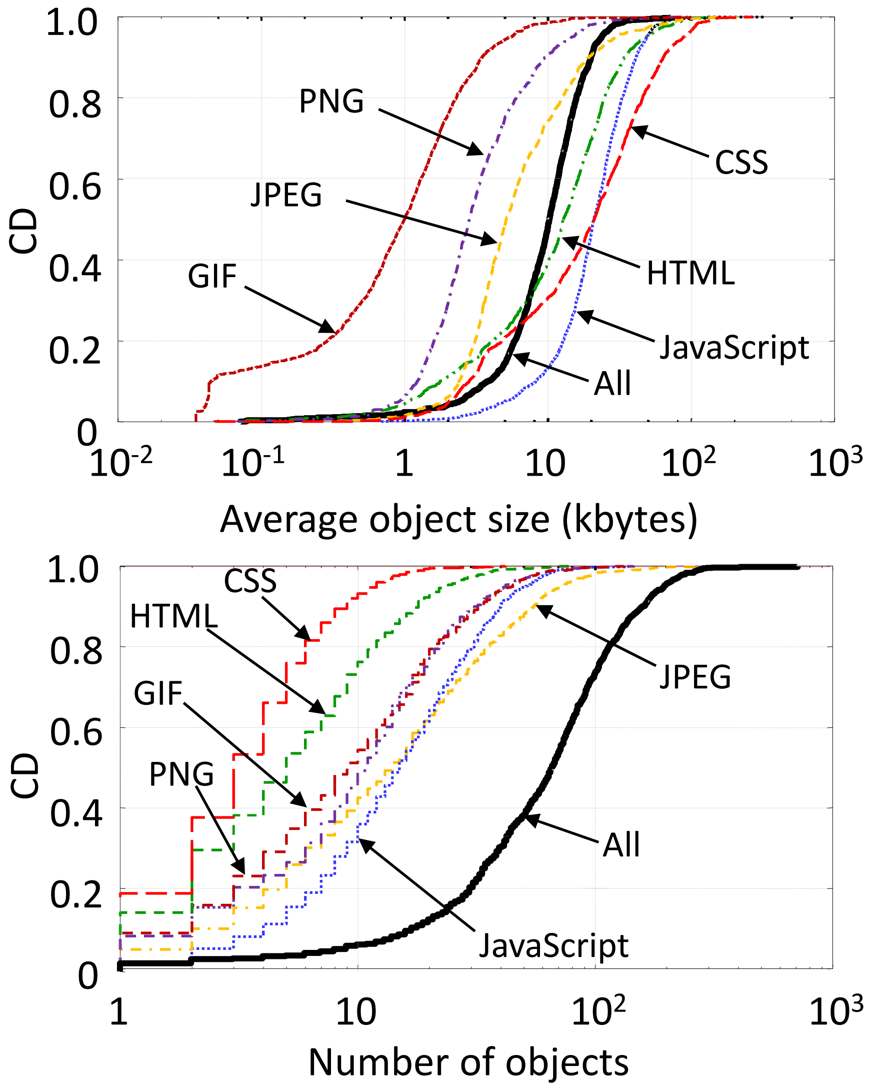
- Mechanism providing web browsing services
  - Overview
  - HTTP
  - CDN
  - Objects

- Possible factors degrading web response time
- Geographical tendency of web content deployment
- Approaches reducing web response time

Type of Objects

Data type	extension	MIME type
Plain text	.txt	text/plain
HTML	.html	text/html
XML	.xml	text/xml
JavaScript	.js	text/javascript
CSS	.css	text/css
GIF image	.gif	image/gif
JPEG image	.jpg	image/jpeg
PNG image	.png	image/png

- CSS (cascading style sheets): language indicating style of webpages
- JavaScript: programs generating dynamic objects at browsers
- Image objects tend to be smaller than text objects.

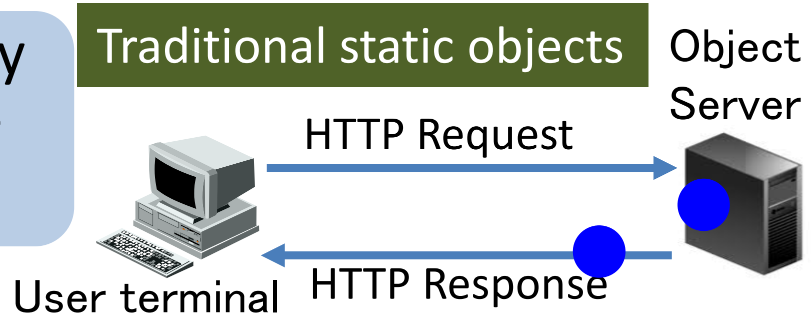


Browsing about 1,200 most popular webpages from Tokyo, Japan in Jan. 2013

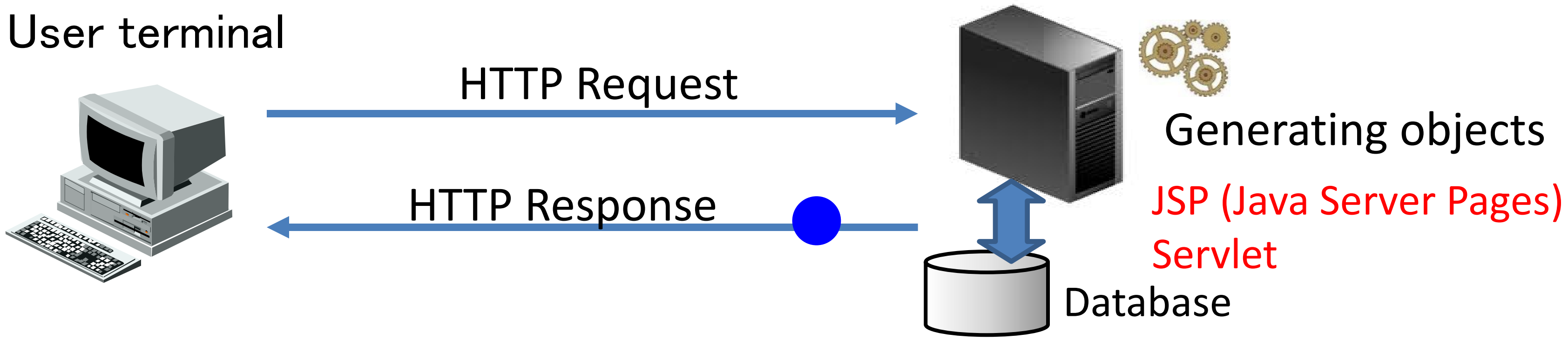
Dynamic Objects

Many modern webpages consist of many dynamic objects generated at servers or browsers at time of page browsing

Traditional static objects

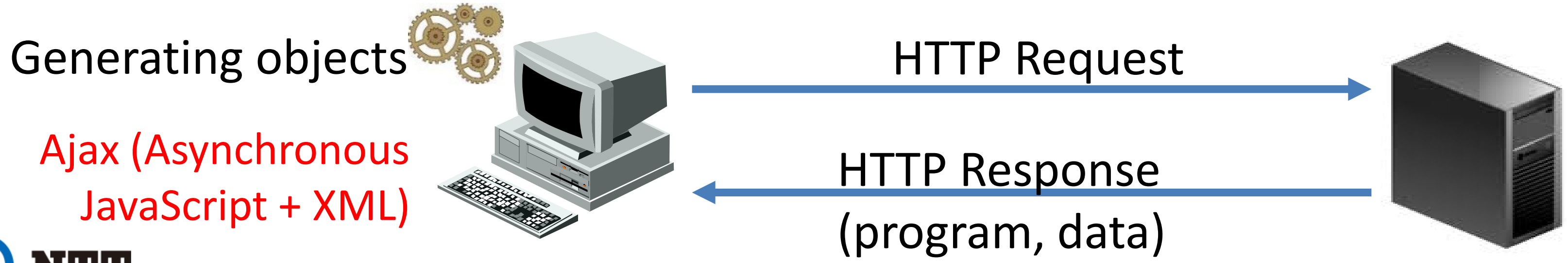


Dynamic objects generated at servers



JSP (Java Server Pages)
Servlet

Dynamic objects generated at browsers



Ajax (Asynchronous JavaScript + XML)

JSP (Java Server Pages)

- Directly embedding program called “scriplet” in HTML file
- Web servers executes scriplet, inserts obtained values into HTML file, and sends generated HTML file to user terminals as HTTP response.

HTML file

```
<html>
<head>
<title> Test </title>
</head>
<body>
<p>Price including 10% tax of 100 USD = <% out.println(100 * 1.1); %> USD </p>
</body>
</html>
```

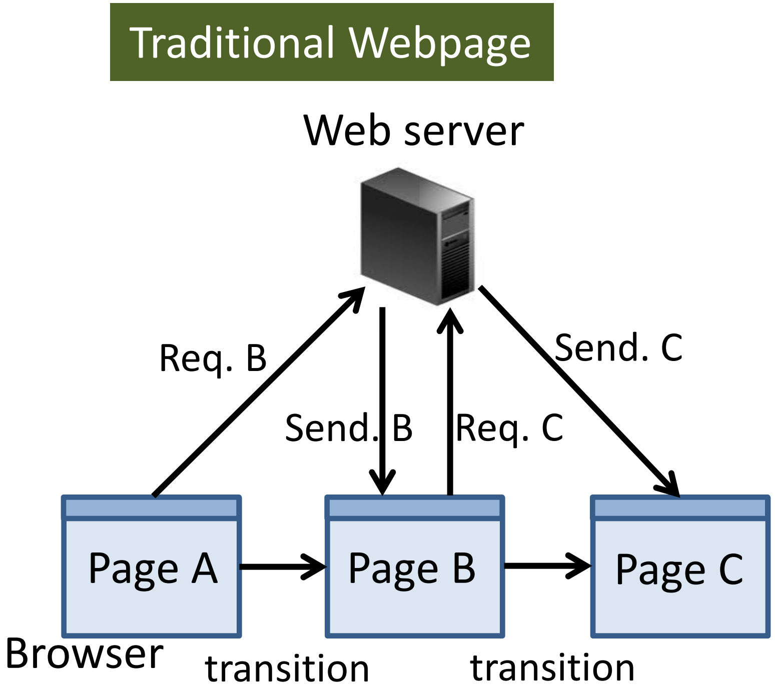
scriplet

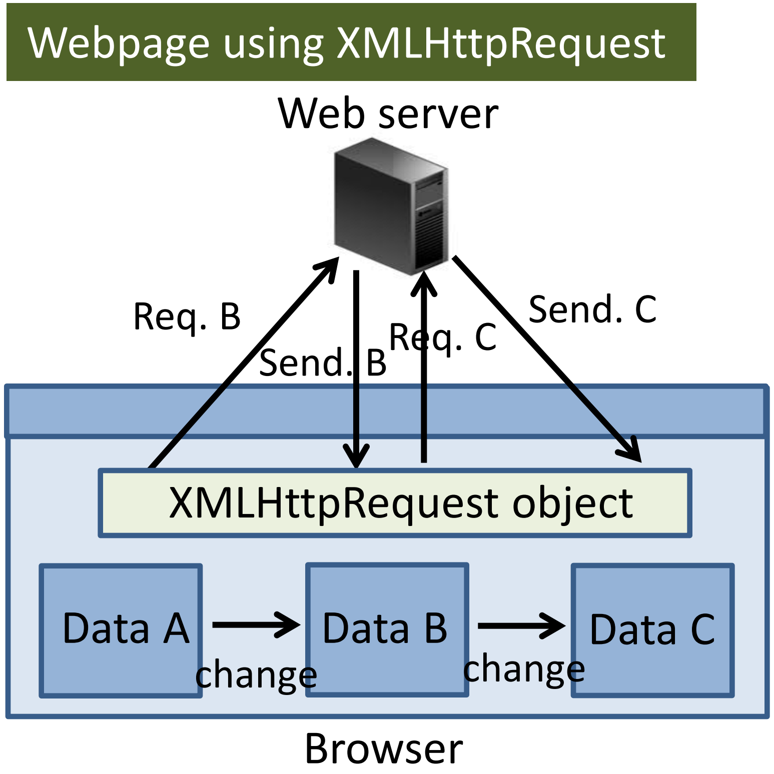
Output

```
Test
Price including 10% tax of 100 USD = 110 USD
```

Ajax (Asynchronous JavaScript + XML)

- Ajax enables web browsers to display various data without changing URLs. (Example: Google Map)
- Extracting various data included in XML data using XMLHttpRequest
- XMLHttpRequest is a kind of object of JavaScript.





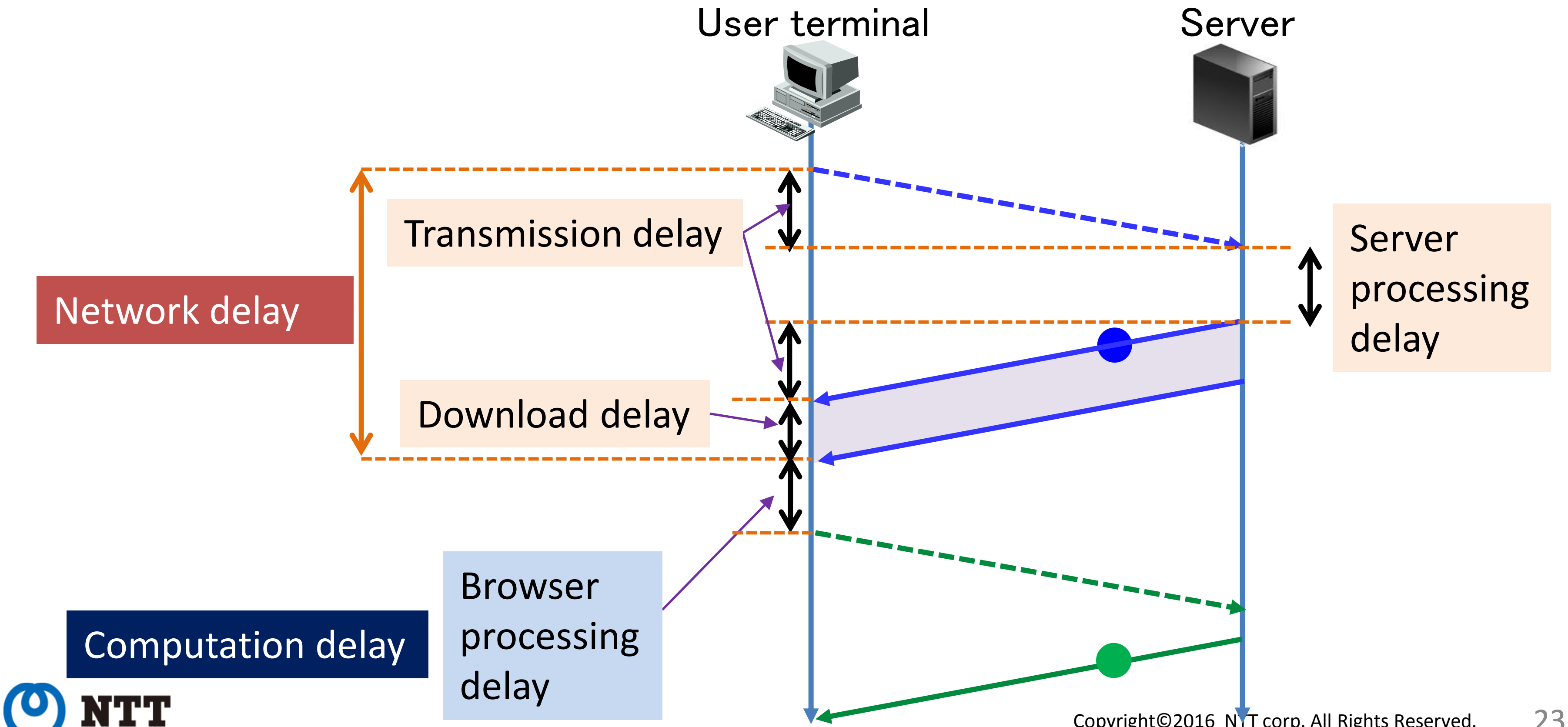
- Mechanism providing web browsing services

- Possible factors degrading web response time

- Geographical tendency of web content deployment
- Approaches reducing web response time

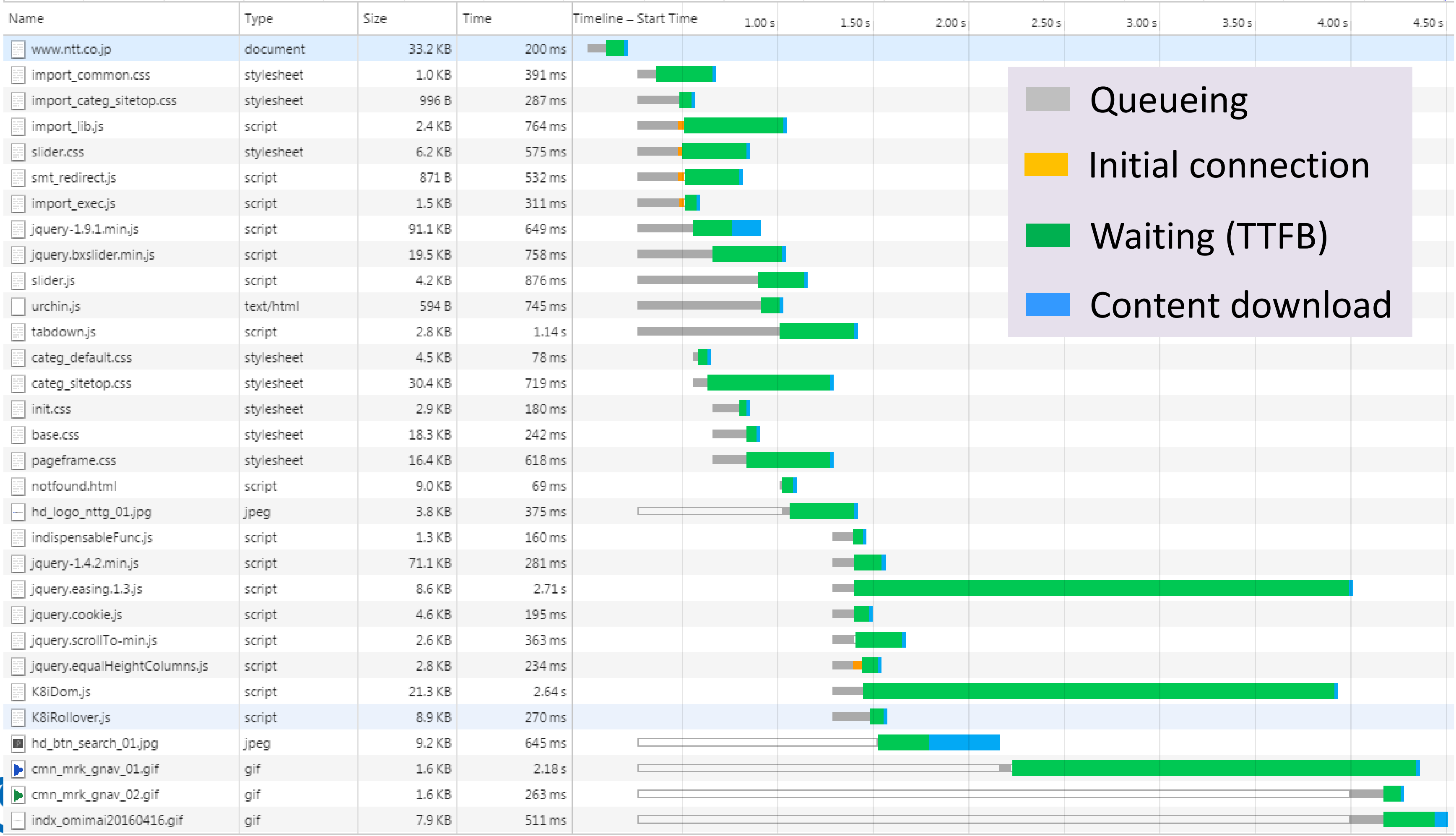
Components of Web Response Time

- Transmission delay: latency between user terminal and servers, i.e., RTT X 2
- Server processing delay: computation and waiting time at servers
- Download delay: delay required to download objects
- Browser processing delay: computation and waiting time at browser



Example of Networking Process

- Can check diagram of networking process by using developer tool of Google Chrome
- Example when accessing home page of NTT (www.ntt.co.jp)

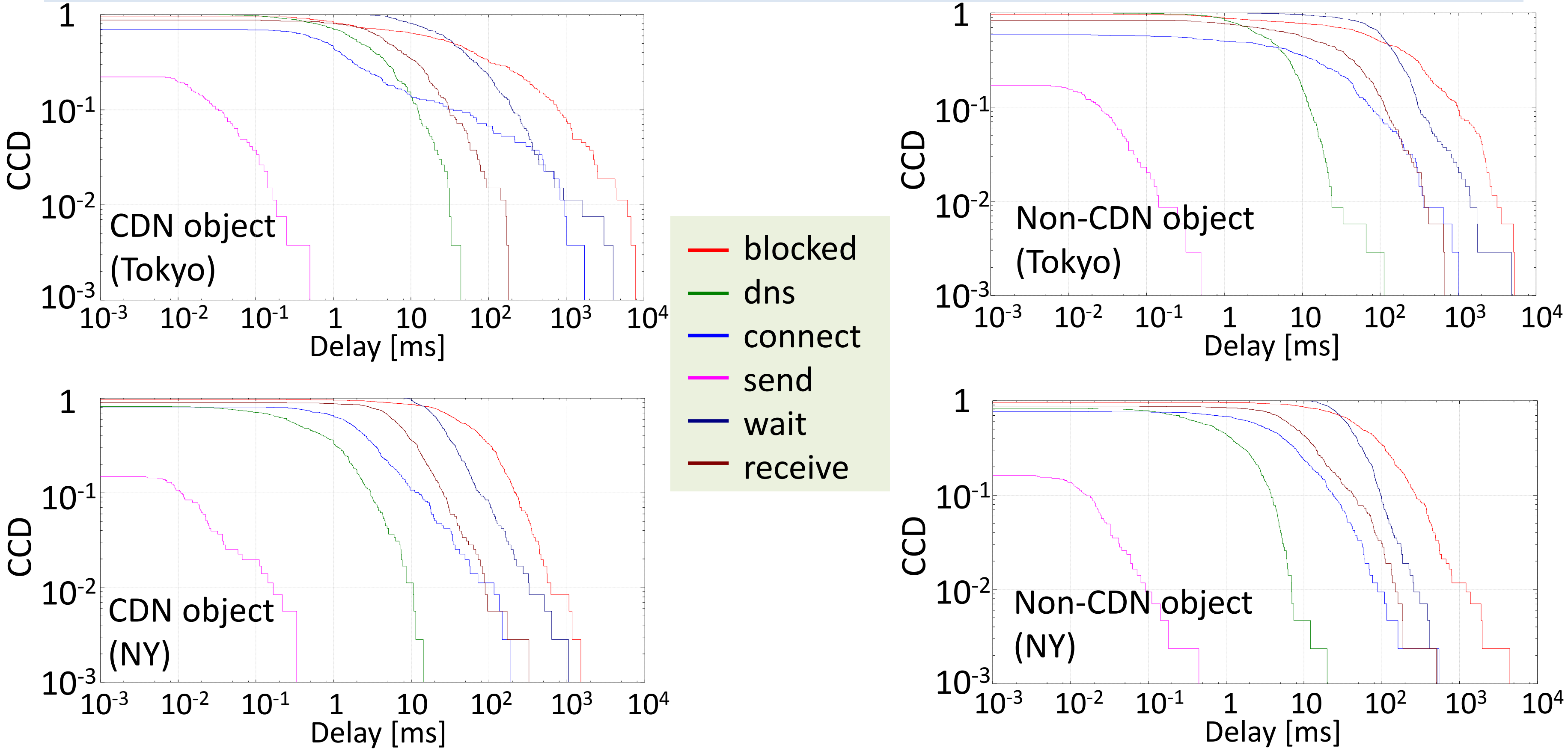


Name	Type	Size	Time
www.ntt.co.jp	document	33.2 KB	200 ms
import_common.css	stylesheet	1.0 KB	391 ms
import_categ_sitetop.css	stylesheet	996 B	287 ms
import_lib.js	script	2.4 KB	764 ms
slider.css	stylesheet	6.2 KB	575 ms
smt_redirect.js	script	871 B	532 ms
import_exec.js	script	1.5 KB	311 ms
jquery-1.9.1.min.js	script	91.1 KB	649 ms
jquery.bxslider.min.js	script	19.5 KB	758 ms
slider.js	script	4.2 KB	876 ms
urchin.js	text/html	594 B	745 ms
tabdown.js	script	2.8 KB	1.14 s
categ_default.css	stylesheet	4.5 KB	78 ms
categ_sitetop.css	stylesheet	30.4 KB	719 ms
init.css	stylesheet	2.9 KB	180 ms
base.css	stylesheet	18.3 KB	242 ms
pageframe.css	stylesheet	16.4 KB	618 ms
notfound.html	script	9.0 KB	69 ms
hd_logo_nttg_01.jpg	jpeg	3.8 KB	375 ms
indispensableFunc.js	script	1.3 KB	160 ms
jquery-1.4.2.min.js	script	71.1 KB	281 ms
jquery.easing.1.3.js	script	8.6 KB	2.71 s
jquery.cookie.js	script	4.6 KB	195 ms
jquery.scrollTo-min.js	script	2.6 KB	363 ms
jquery.equalHeightColumns.js	script	2.8 KB	234 ms
K8iDom.js	script	21.3 KB	2.64 s
K8iRollover.js	script	8.9 KB	270 ms
hd_btn_search_01.jpg	jpeg	9.2 KB	645 ms
cmn_mrk_gnav_01.gif	gif	1.6 KB	2.18 s
cmn_mrk_gnav_02.gif	gif	1.6 KB	263 ms
indx_omimai20160416.gif	gif	7.9 KB	511 ms

▨	Queueing
🟨	Initial connection
🟩	Waiting (TTFB)
🟦	Content download

CCD of Each Delay Component

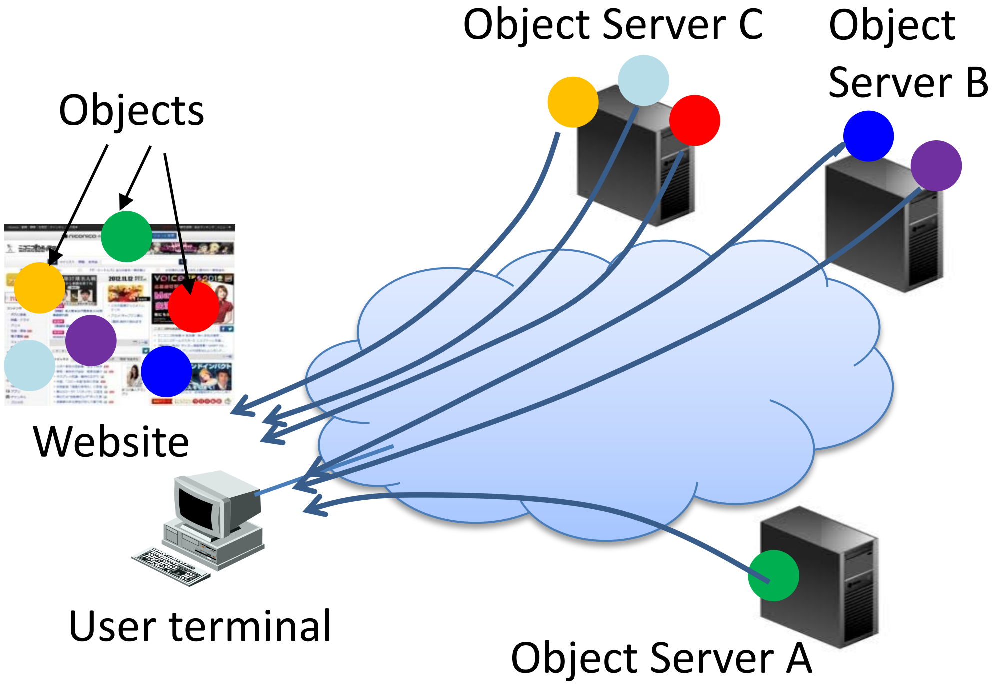
- Plots CCD (complementary cumulative distribution) of each delay component of HTTP response time
- Browsed about 1,000 most popular webpages at Tokyo and New York

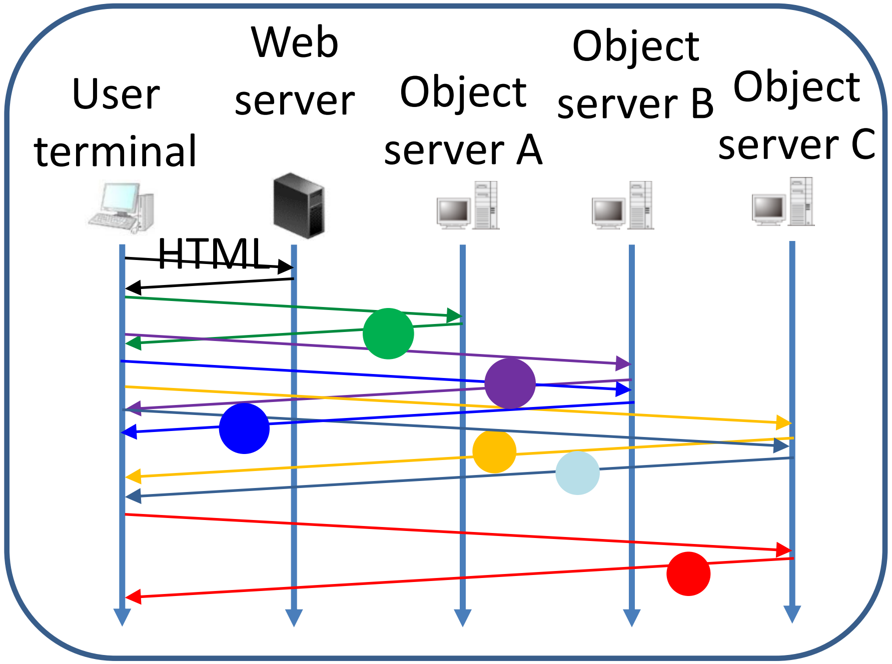


- (1)blocked, (2)wait, and (3)receive are three biggest factors in many webpages.
- In Japan, connect is the third factor in few webpages.

Complexity of Web Traffic Pattern

One webpage consists of multiple data objects which are delivered from various object servers using HTTP sessions.





- 50% webpages: obtain ≥ 70 objects from ≥ 15 servers at ≥ 7 locations
- 10% webpages: obtain ≥ 150 objects from ≥ 40 servers at ≥ 15 locations

Computations at Web Browsers

- HTML parsing:
  - Analyzing obtained HTML file and extracting embedded objects
- CSS evaluation:
  - Analyzing CSS file and setting structure and style of webpages, e.g., location, size, and color of each object
- JavaScript evaluation:
  - Executing JavaScript code and generating dynamic objects
- Rendering:
  - Drawing webpages from obtained and generated objects

Dependencies among Processes at Browsers

- Due to various dependencies among processes, cannot start all processes simultaneously

- **Flow dependency**:
  - Loading object ⇒ Parsing tag referencing object
  - Evaluating object ⇒ Loading object

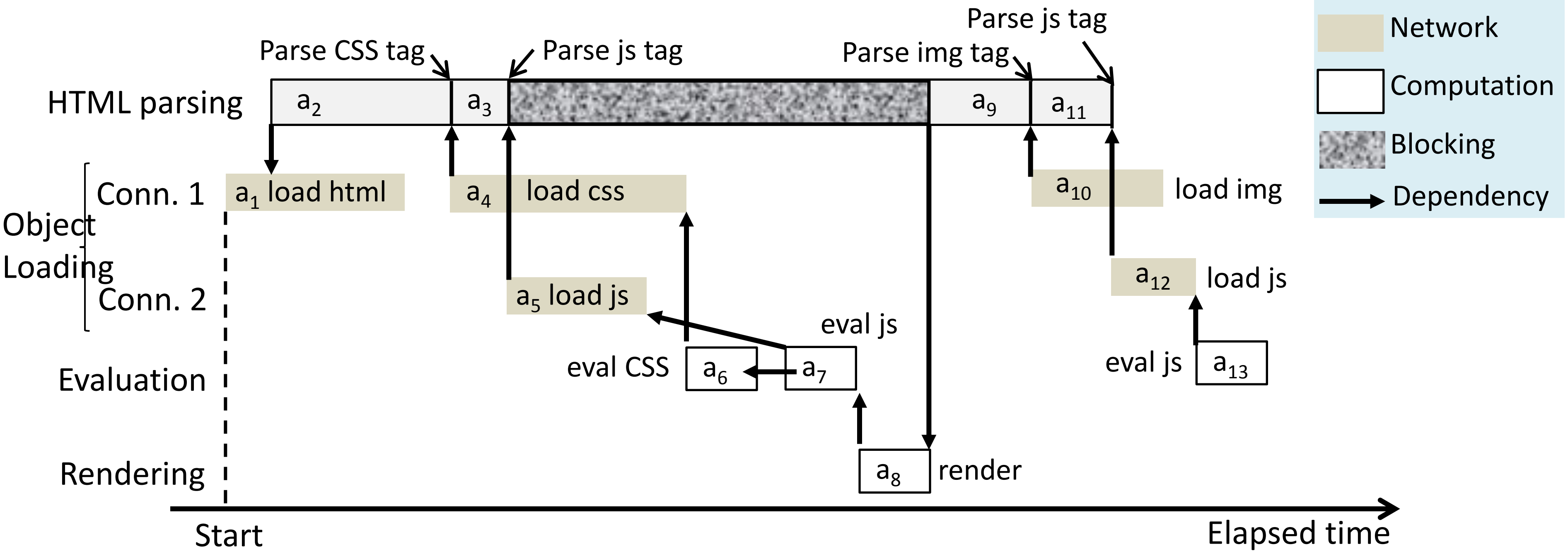
- **Output dependency**:
  - Need to sustain consistency of DOM (document object model) tree* which can be accessed from HTML parsing and JavaScript evaluation
  - Parsing next tag in HTML ⇒ Completing previous JavaScript download and evaluation

- **Resource constraints**:
  - Cannot exceed maximum number of HTTP sessions to same domain

*DOM tree: intermediate representation of webpage

Critical Path on Process at Browsers

Critical path exists because of dependencies between processes



- Starts HTML parsing (a_2)
- Continues HTML parsing while downloading CSS object (a_4)
- Suspends HTML parsing while downloading JS object (a_5)
- Downloads CSS and JS objects in parallel

- Starts eval of CSS obj after obtain (a_6)
- Starts eval of JS after eval of CSS (a_7)
- Starts render after obtaining CSS and JS objects (a_8)
- Resumes HTML parsing when completing rendering (a_9)

Ratio of Network and Computation Delay on Critical Path

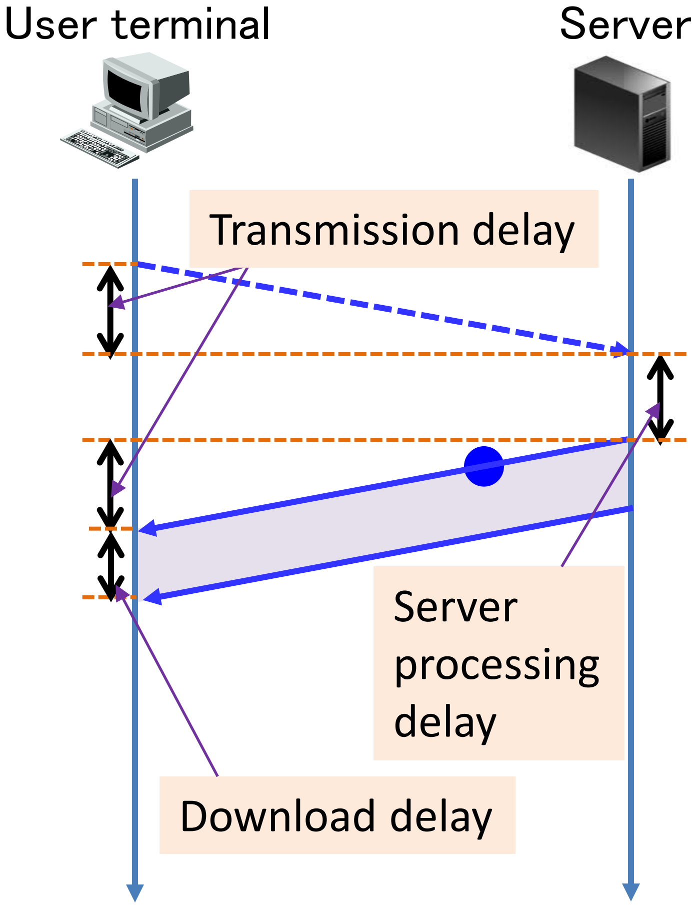
- Wang et al. analyzed CDF (cumulative distribution function) of ratio of network delay and computation delay on critical path when browsing 200 most popular webpages*

*X. Wang, et al. (Univ. Washington), Demystifying Page Load Performance with WProf, NSDI 2013

Network delay occupied 65% on critical path in median

Factors Increasing Network delay

- **Increase of HTTP connection count**
  - Increase of object count
- **Limitation in parallel download of objects**
  - Diversity of object servers accessed
  - Upper limit to avoid server congestion
- **Increase of name resolution time**
- **Increase of RTT**
  - Network congestion
  - Inadequate use of CDN
    - Selecting improper cache servers
    - Limitation of cacheable object count
- **Increase of server response time**
  - Increase of objects generated at servers
  - Congestion of web and object servers
- **Growth of object size**



Factors Increasing Computation Delay

- Increase of number of JavaScript blocking HTML parsing
- Improper writing of HTML file
  - Write JavaScript at top of HTML file
  - Duplication of same JavaScript objects
- Old version browser
- Low spec user terminal

- Mechanism providing web browsing services
- Possible factors degrading web response time

- Geographical tendency of web content deployment
  - Procedure of active measurement of web traffic
  - Comparison of traffic properties among web categories
  - Effect of cache control based on web category

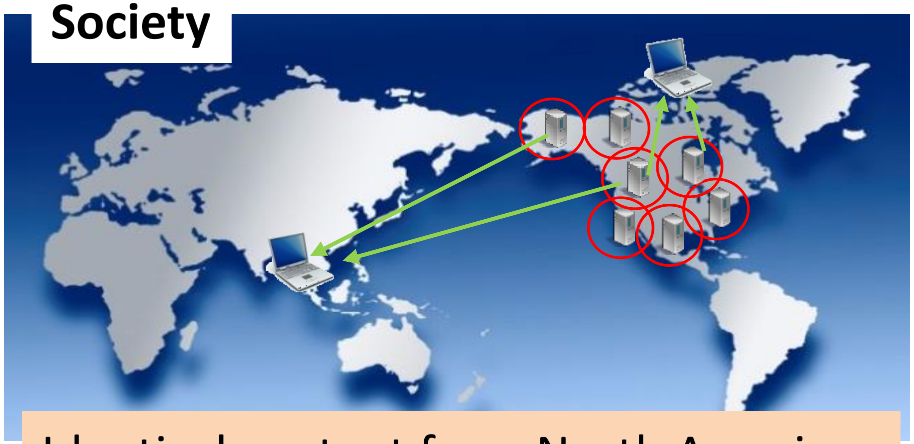
- Approaches reducing web response time

Effect of CDN and Edge Computing

Geographical deployment pattern may differ among website categories, e.g., Sports and News, and effect of CDN and edge computing will depend on website categories.

Yahoo Answers, McAfee SiteAdvisor, …

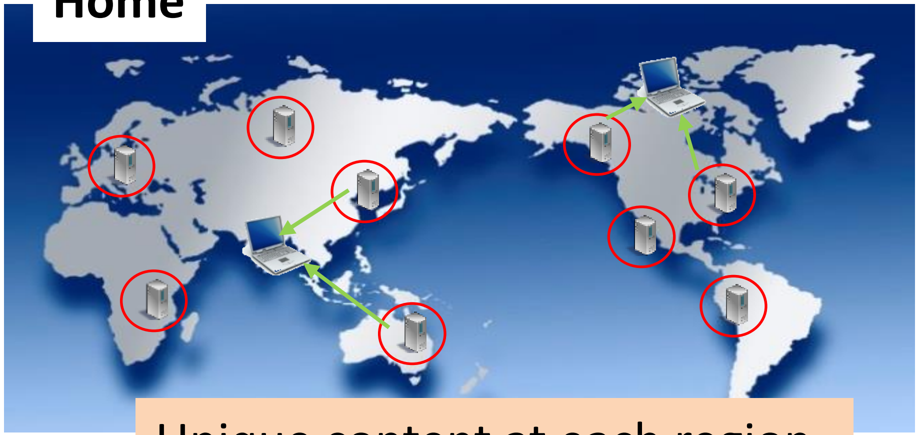
Society



Identical content from North America

Yelp, Groupon, …

Home



Unique content at each region

Our approach*

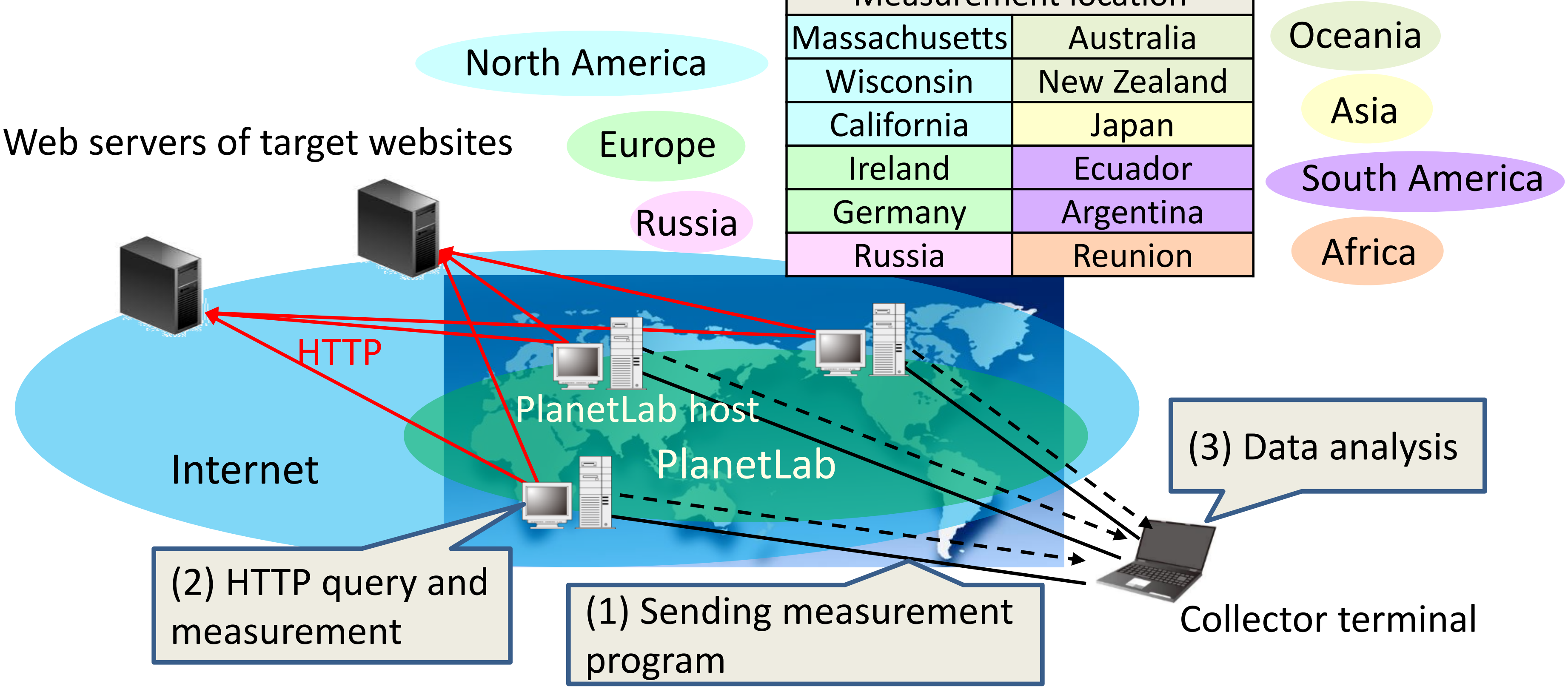
*N. Kamiyama, Y. Nakano, K. Shiomoto, G. Hasegawa, M. Murata, and H. Miyahara, “Priority Control Based on Website Categories in Edge Computing,” IEEE GIS 2016

1. Propose to differentiate caching priority among website categories
2. Roughly analyze effect of category-based priority control in CDN (edge computing) using active measurement data from 12 locations in world

Measurement Procedure

1. Selected 12 PlanetLab hosts as measurement terminals accessing various websites
2. Measured various properties, e.g., object count obtained and RTT, by executing program at each PlanetLab host to access various websites sequentially
3. Collected measurement results at collector terminal

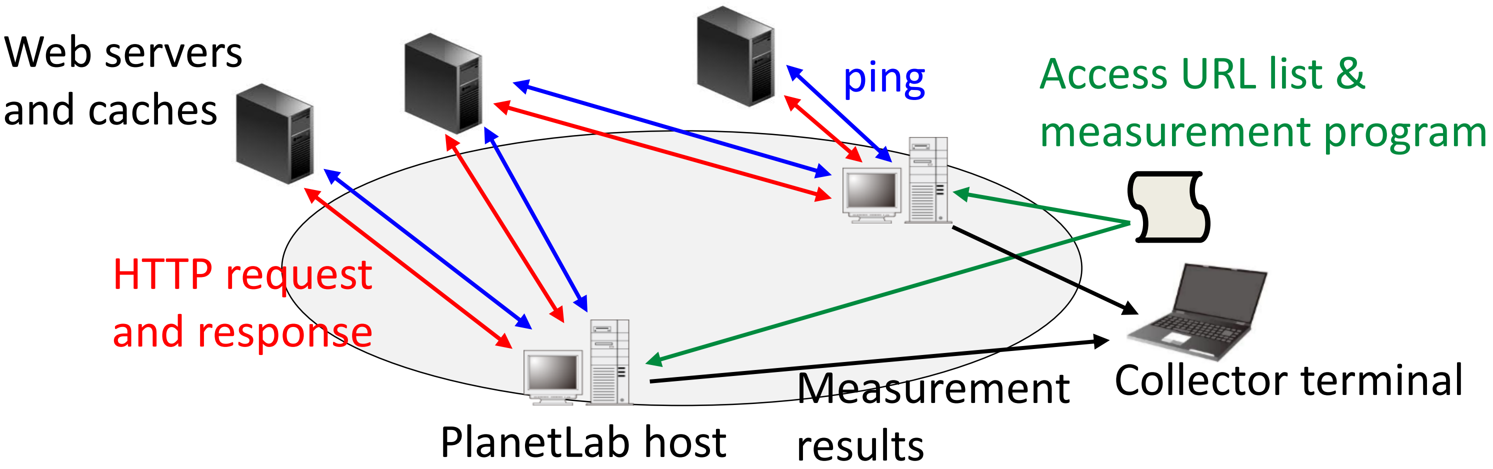
Measurement location	
Massachusetts	Australia
Wisconsin	New Zealand
California	Japan
Ireland	Ecuador
Germany	Argentina
Russia	Reunion



PlanetLab: overlay network consisting of over 500 hosts worldwide

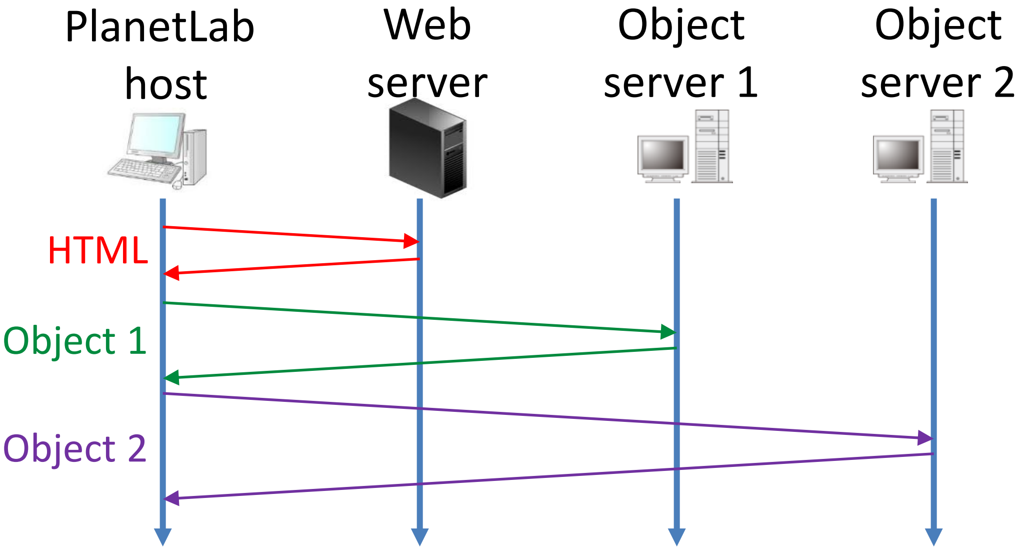
Measurement Program

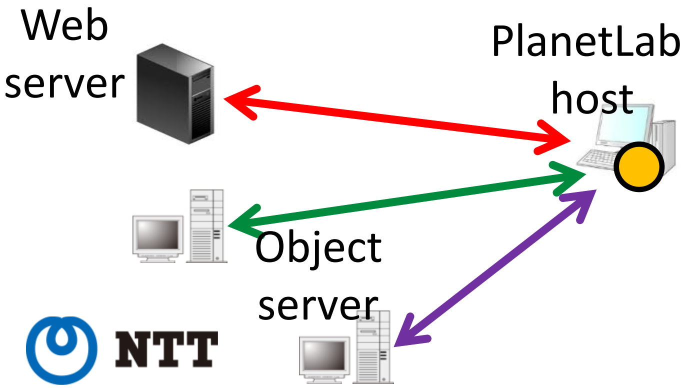
- Generated URL list and sent it to each PlanetLab host
- Starting from 0:00 (midnight) or 12:00 (noon), each PlanetLab host executed following procedures:
  1. Accessed websites according to URL list and obtained HAR (HTTP Archive) files
  2. Extracted information of HTTP response time from obtained HAR files
  3. Measured RTT to each object server by sending *ping*
  4. Obtained domain name of each object server using *dig* command
  5. Sent measurement results to collector terminal



Obtaining HAR Files

- Obtained HTML file initially, and obtained each object embedded in HTML file
- HAR (HTTP Archive) file: outputs various properties of each object in JSON (JavaScript Object Notation) format





HTLM:
Object 1
Object 2

HTML pursing

HAR file

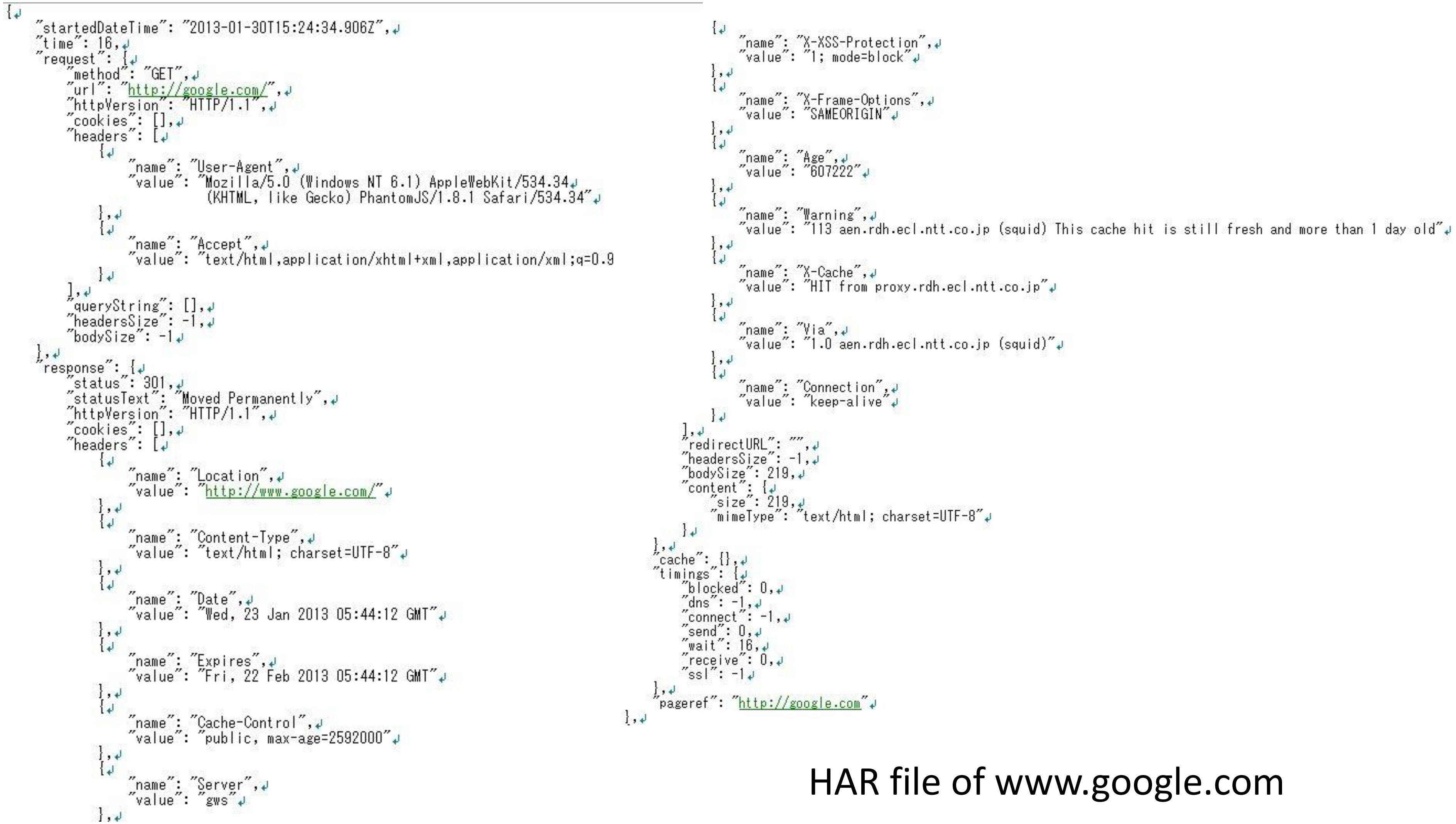
Object 1:
Size: 100
Delay: 50
MIME type: jpeg
Location: Osaka

Object 2:
Size: 500
Delay: 200
MIME type: javascript
Location: NY

…

Using phantomJS, providing browser function, and netsniff, extracting HAR files, obtained HAR files of many websites sequentially in batch process

Example of HAR File



```
{
  "startedDateTime": "2013-01-30T15:24:34.906Z",
  "time": 16,
  "request": {
    "method": "GET",
    "url": "http://google.com/",
    "httpVersion": "HTTP/1.1",
    "cookies": [],
    "headers": [
      {
        "name": "User-Agent",
        "value": "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/534.34
                  (KHTML, like Gecko) PhantomJS/1.8.1 Safari/534.34"
      },
      {
        "name": "Accept",
        "value": "text/html,application/xhtml+xml,application/xml;q=0.9
      }
    ],
    "queryString": [],
    "headersSize": -1,
    "bodySize": -1
  },
  "response": {
    "status": 301,
    "statusText": "Moved Permanently",
    "httpVersion": "HTTP/1.1",
    "cookies": [],
    "headers": [
      {
        "name": "Location",
        "value": "http://www.google.com/"
      },
      {
        "name": "Content-Type",
        "value": "text/html; charset=UTF-8"
      },
      {
        "name": "Date",
        "value": "Wed, 23 Jan 2013 05:44:12 GMT"
      },
      {
        "name": "Expires",
        "value": "Fri, 22 Feb 2013 05:44:12 GMT"
      },
      {
        "name": "Cache-Control",
        "value": "public, max-age=2592000"
      },
      {
        "name": "Server",
        "value": "gws"
      },
```

```
      {
        "name": "X-XSS-Protection",
        "value": "1; mode=block"
      },
      {
        "name": "X-Frame-Options",
        "value": "SAMEORIGIN"
      },
      {
        "name": "Age",
        "value": "607222"
      },
      {
        "name": "Warning",
        "value": "113 aen.rdh.ecl.ntt.co.jp (squid) This cache hit is still fresh and more than 1 day old"
      },
      {
        "name": "X-Cache",
        "value": "HIT from proxy.rdh.ecl.ntt.co.jp"
      },
      {
        "name": "Via",
        "value": "1.0 aen.rdh.ecl.ntt.co.jp (squid)"
      },
      {
        "name": "Connection",
        "value": "keep-alive"
      }
    ],
    "redirectURL": "",
    "headersSize": -1,
    "bodySize": 219,
    "content": {
      "size": 219,
      "mimeType": "text/html; charset=UTF-8"
    }
  },
  "cache": {},
  "timings": {
    "blocked": 0,
    "dns": -1,
    "connect": -1,
    "send": 0,
    "wait": 16,
    "receive": 0,
    "ssl": -1
  },
  "pageref": "http://google.com"
},
```

HAR file of www.google.com

URL List of Measurement Target

- Selected 300 most popular webpages in each of 16 categories based on public information of Alexa*
- Totally Selected 927 webpages from which measurement data were successfully obtained at all 12 measurement locations

*http://www.alexa.com/topsites

Category	#sites	Category	#sites
Business	40	Home	47
Computer	91	Shopping	68
News	27	Adult	102
Reference	109	Arts	60
Regional	73	Games	58
Science	86	Kids & teens	64
Society	83	Recreation	52
Health	52	Sports	53

Classifying Objects Based on CDN Use

- Classified objects into CDN objects delivered using CDN or non-CDN objects delivered without using CDN
  - Listed 44 second-level domains of various CDN providers by manually checking websites of various CDN providers
  - Obtained domain names of hosts actually delivering objects, e.g., www.akamai.com/qqq/rrr, by using dig command from URL names, e.g., www.google.com/xxx/yyy.jpg, of objects extracted from HAR files
  - Identified CDN objects by comparing second-level domain obtained by dig command with entries of generated list

List of second-level domains of CDN objects

profile.ak.fbcdn.net
static.ak.fbcdn.net
r.ddmcdn.com
s.cdn-care.com
cmscdn.staticcache.org
g-ecx.images-amazon.com
max.blurtitcdn.com
a.espncdn.com
ecx.images-amazon.com
edgekey.net
edgesuite.net

cloudfront.net
vo.msecnd.net
edgecastcdn.net
cdngc.net
bootstrapcdn.com
example.com
akadns.net
akam.net
akamaiedge.net
akamai.net
akamaiedge.net

akamaihd.net
edgesuite.net
edgekey.net
srip.net
akamaitechnologies.com
akamaitechnologies.fr
akamaitech.net
akadns.net
akam.net
akamaistream.net
edgekey.net

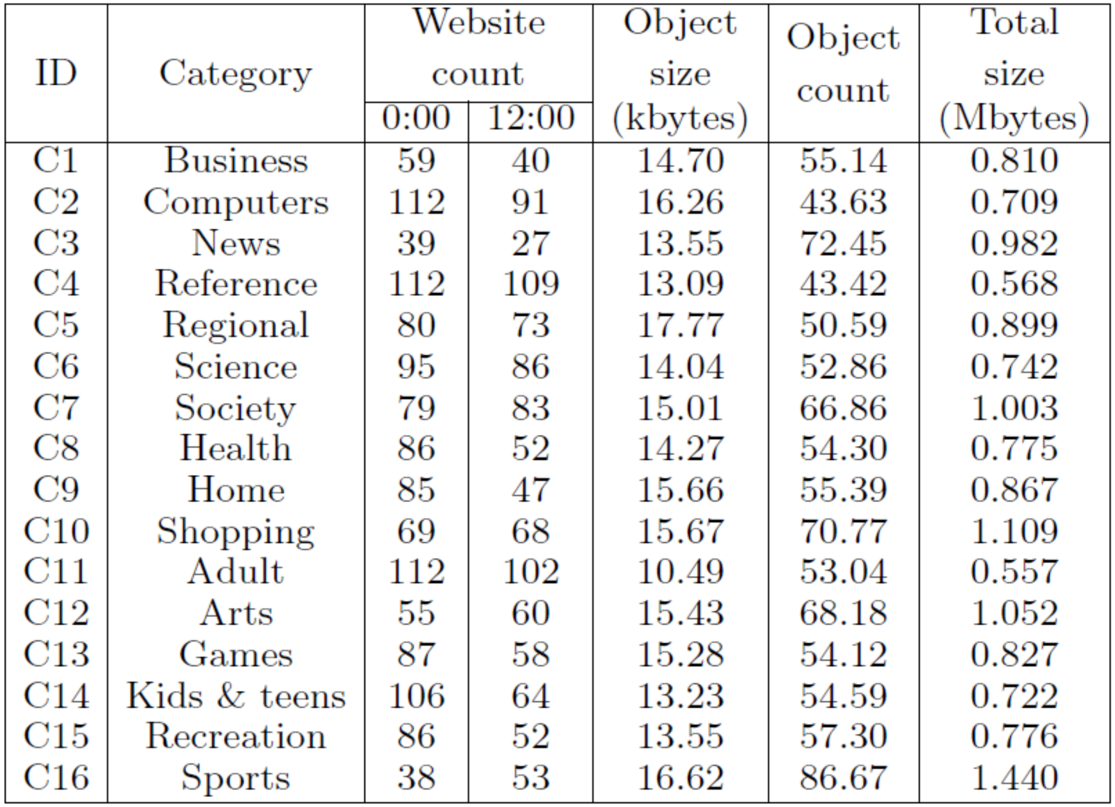
edgesuite.net
cloudfront.net
vo.msecnd.net
edgecastcdn.net
cdngc.net
push-11.cdnsun.com
ve14.fr3.atl1.llnw.net
hs-9.cdn77.com
nyud.net
CloudFlare
Incapsula

- Mechanism providing web browsing services
- Possible factors degrading web response time

- Geographical tendency of web content deployment
  - Procedure of active measurement of web traffic
  - Comparison of traffic properties among web categories
  - Effect of cache control based on web category

- Approaches reducing web response time

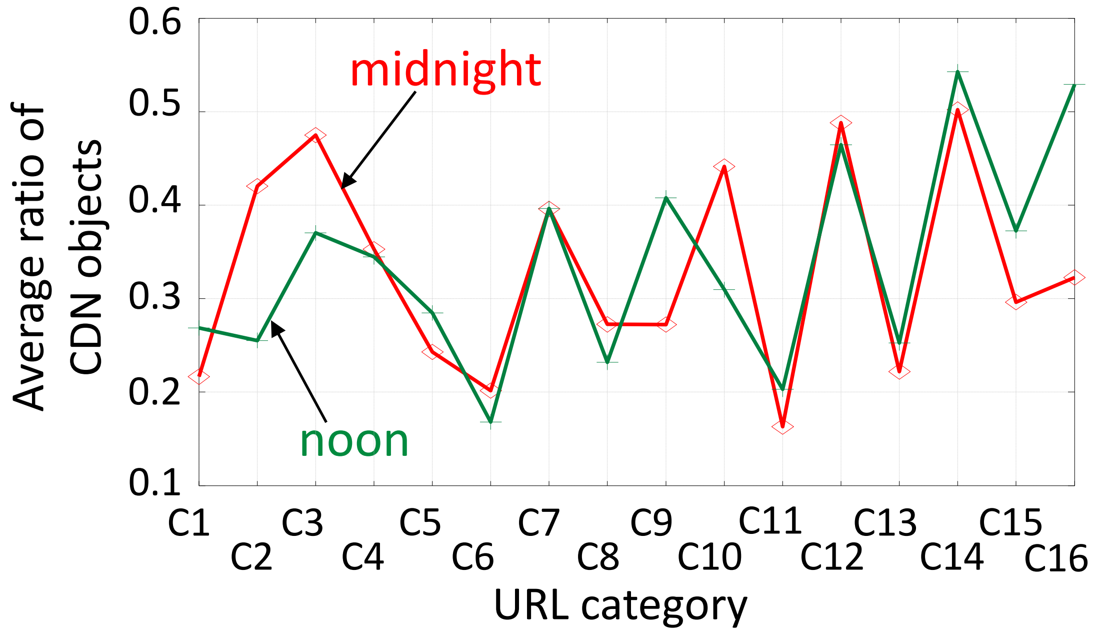
Basic Properties



ID	Category	Website count		Object size (kbytes)	Object count	Total size (Mbytes)
		0:00	12:00			
C1	Business	59	40	14.70	55.14	0.810
C2	Computers	112	91	16.26	43.63	0.709
C3	News	39	27	13.55	72.45	0.982
C4	Reference	112	109	13.09	43.42	0.568
C5	Regional	80	73	17.77	50.59	0.899
C6	Science	95	86	14.04	52.86	0.742
C7	Society	79	83	15.01	66.86	1.003
C8	Health	86	52	14.27	54.30	0.775
C9	Home	85	47	15.66	55.39	0.867
C10	Shopping	69	68	15.67	70.77	1.109
C11	Adult	112	102	10.49	53.04	0.557
C12	Arts	55	60	15.43	68.18	1.052
C13	Games	87	58	15.28	54.12	0.827
C14	Kids & teens	106	64	13.23	54.59	0.722
C15	Recreation	86	52	13.55	57.30	0.776
C16	Sports	38	53	16.62	86.67	1.440

- Entertainment webpages, e.g., Arts, Shopping, and Sport, tend to have more objects and larger total data size.
- Information webpages, e.g., Business, Computers, Health, and Reference, tend to have fewer objects and smaller total data size.

Ratio of CDN Objects



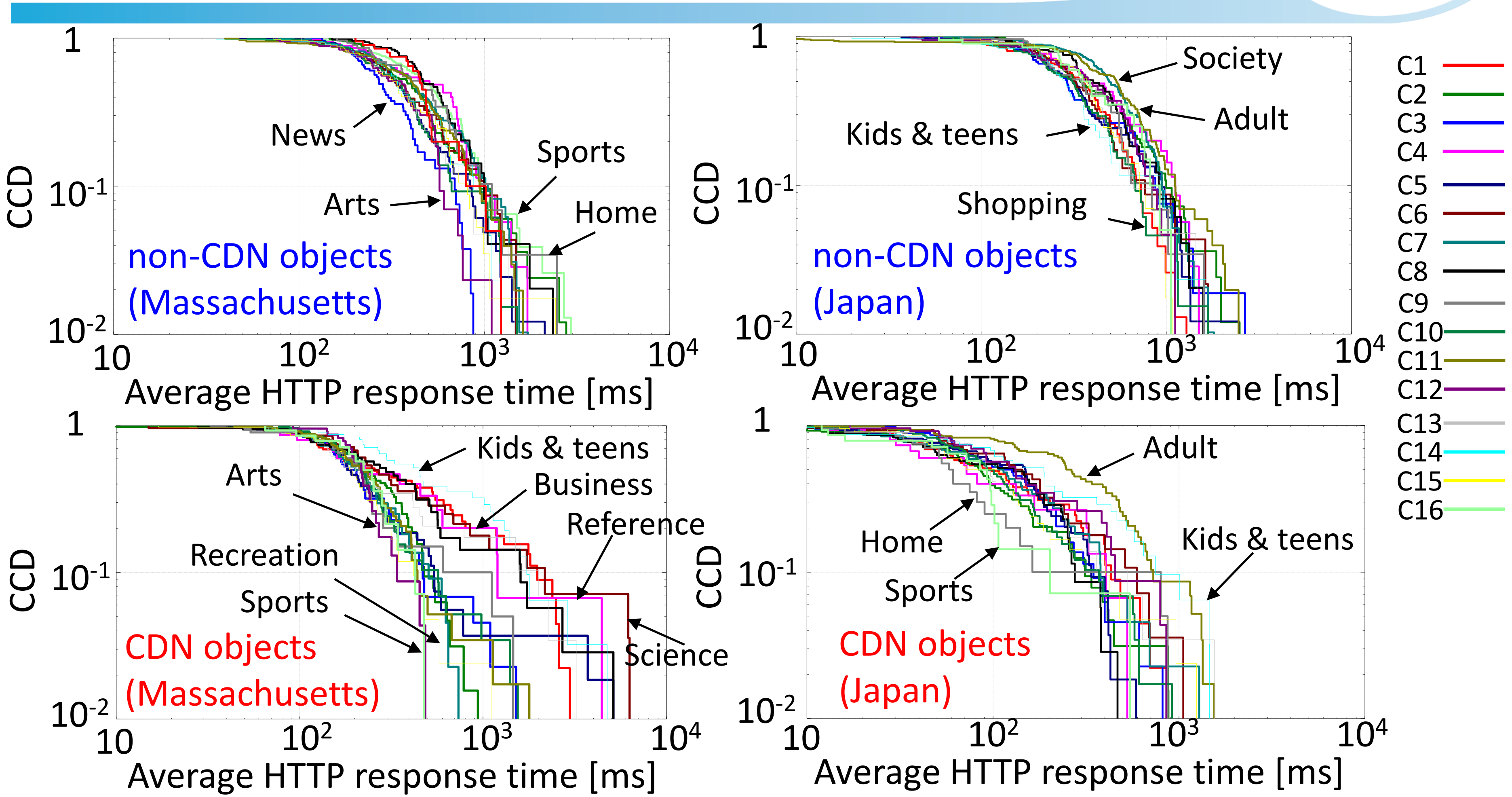
ID	Category	ID	Category
C1	Business	C5	Regional
C2	Computers	C6	Science
C3	News	C7	Society
C4	Reference	C8	Health

ID	Category	ID	Category
C9	Home	C13	Games
C10	Shopping	C14	Kids & teens
C11	Adult	C15	Recreation
C12	Arts	C16	Sports

- Having more CDN objects in websites of Computers, News, Society, Shopping, Arts, and Kids & teens
- Having fewer CDN objects in websites of Business, Regional, Science, Adult, and Games

Ratio of CDN objects differed among categories, between 0.2 and 0.5

Average HTTP Response Time at Noon



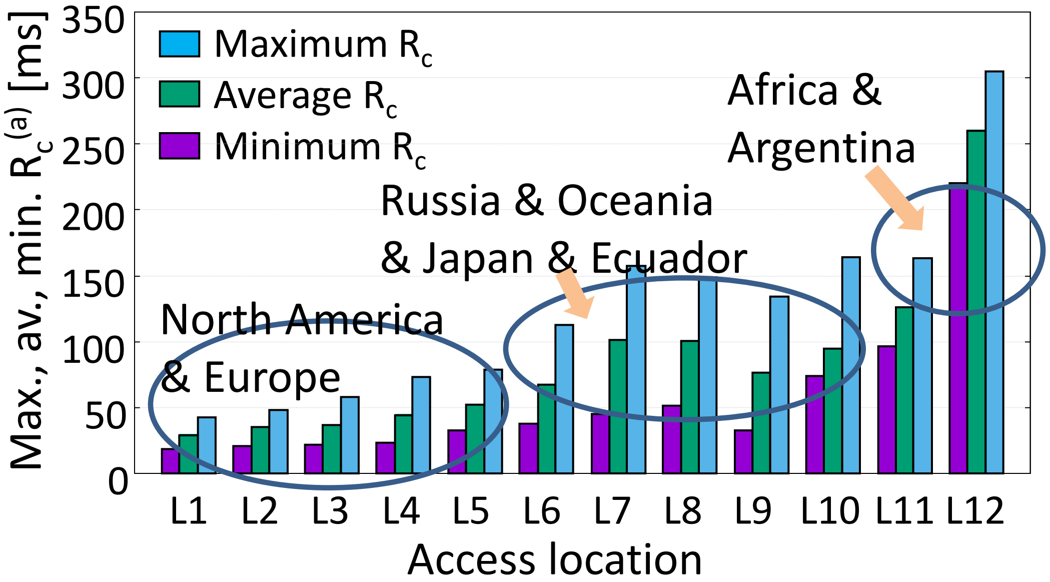
- Non-CDN objects: exceeded 500 ms in 20 - 80% websites
- CDN objects: exceeded 500 ms in just 2 to 40% websites

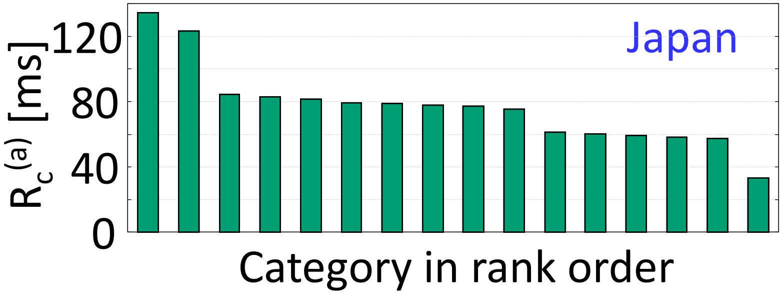
Confirmed effect of CDN in reducing HTTP response time

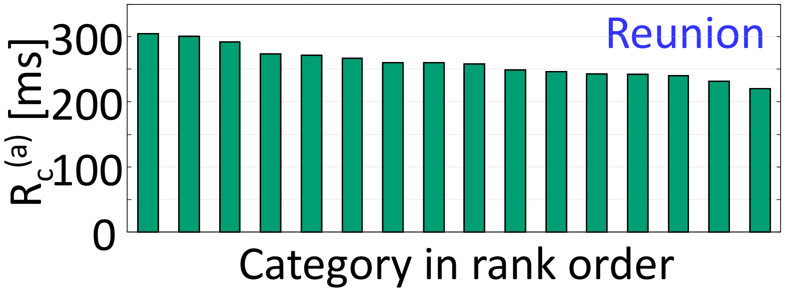
Average RTT to Servers in Each Web Category

$R_c^{(a)}$: average RTT in webpages of category c from location a

Max., average, min. of $\mathbf{R_c^{(a)}}$ at each location







L1	Massachusetts	L4	Ireland	L7	Australia	L10	Ecuador
L2	Wisconsin	L5	Germany	L8	New Zealand	L11	Argentina
L3	California	L6	Russia	L9	Japan	L12	Reunion

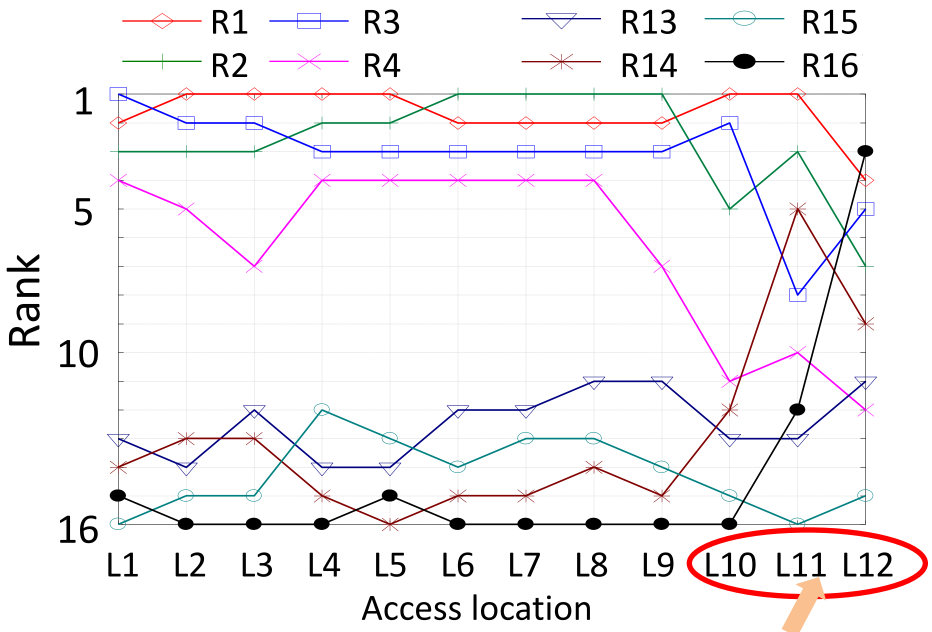
- Africa & Argentina: large RTT in all categories
- Other areas: large difference in RTT among categories

Rank of Average RTT among Web Categories

Category Rank in all areas (all) and 3 areas

Rank	All	California	Japan	Reunion
R1	Reference	Reference	Adult	Regional
R2	Adult	News	Reference	Business
R3	News	Adult	News	Shopping
R4	Games	Society	Science	Reference
R5	Computers	Business	Computers	News
R6	Science	Science	Society	Arts
R7	Society	Games	Games	Adult
R8	Regional	Kids&teens	Kids&teens	Computers
R9	Arts	Computers	Arts	Recreation
R10	Business	Regional	Health	Society
R11	Kids&teens	Health	Sports	Sports
R12	Health	Sports	Business	Games
R13	Sports	Recreation	Regional	Kids&teens
R14	Recreation	Arts	Home	Science
R15	Home	Home	Recreation	Home
R16	Shopping	Shopping	Shopping	Health

Rank of top and bottom 4 categories at at each location



South America & Africa

- Category rank of average RTT is common in all areas except South America and Africa
  - Objects of universal webpages, e.g., Reference (Stack overflow, Yahoo Answers, etc) and News (CNN, Yahoo News, etc) concentrate in North America. ⇒ Large RTT
  - Objects of webpages with high locality, e.g., Shopping (Amazon, Ebay, etc) and Home (Yelp, Groupon, etc) are unique in each area. ⇒ Small RTT
- In South America and Africa, objects of all categories exist remote location, and category rank of average rank is unique.

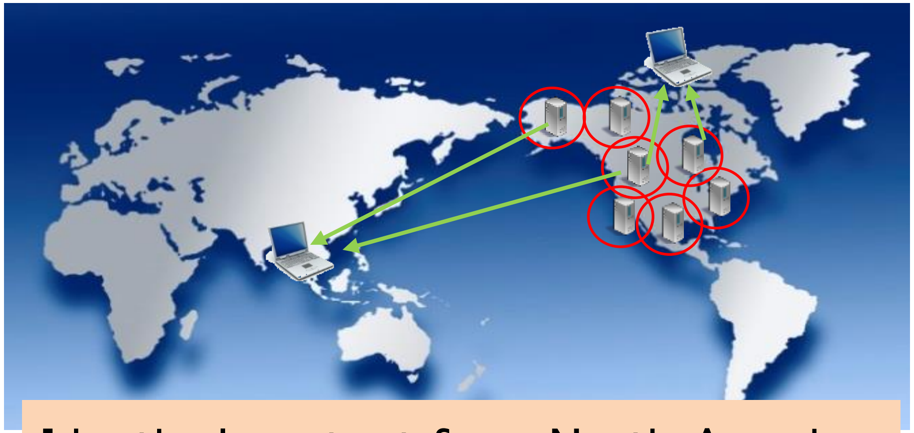
- Mechanism providing web browsing services
- Possible factors degrading web response time

- Geographical tendency of web content deployment
  - Procedure of active measurement of web traffic
  - Comparison of traffic properties among web categories
  - Effect of cache control based on web category

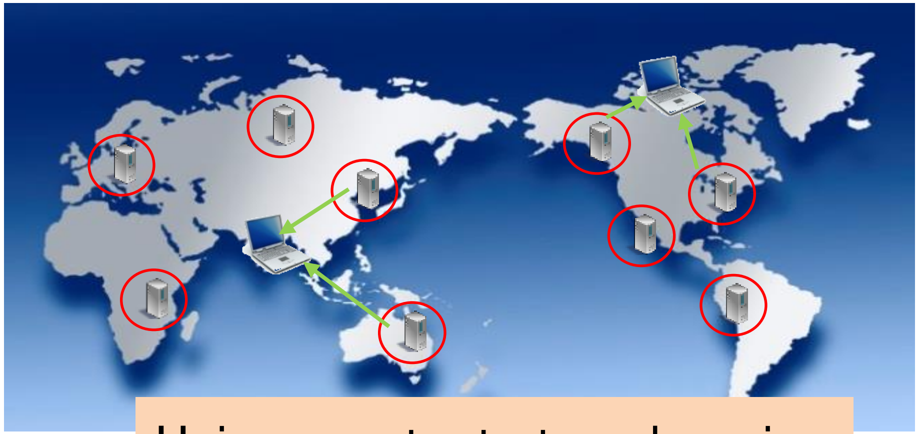
- Approaches reducing web response time

Identifying Tendencies of Object Deployment

- Geographical pattern of original objects, i.e., non-CDN objects, will differ among access locations even when accessing same website.
- Want to identify tendencies of geographical deployment of objects in each web category
- Try to classify webpages based on pattern of distance or RTT to objects servers from 12 access locations



Identical content from North America



Unique content at each region

Machine Learning

- Supervised learning
  - Naive bayes classifier
  - Decision tree learning
  - Neural network
  - Support vector machines
  - …

- Unsupervised learning
  - Hierarchical clustering
  - k-means clustering
  - DBSCAN
  - …

K-means Clustering

- Non-hierarchical clustering method classifying n members into k clusters $\mathbf{S}_1$, $\mathbf{S}_2$, …, $\mathbf{S}_k$ based on d-dimensional observation

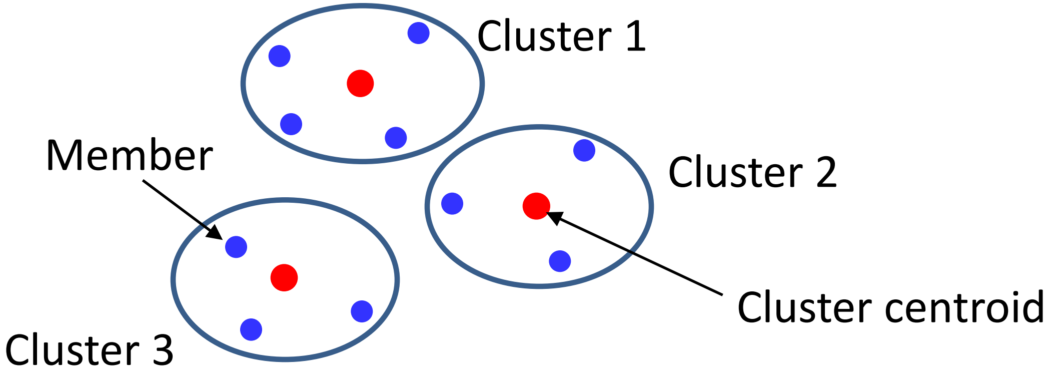
  $\mathbf{x}_j = (x_{j1}, x_{j2}, \ldots, x_{jd},)$ of each member j

- Classify n members into k clusters so as to minimize WCSS (within-cluster sum of squares)

$$\underset{\mathbf{S}}{\arg\min} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \left\| \mathbf{x} - \boldsymbol{\mu}_i \right\|^2$$

μ_i: centroid of cluster i

- Repeat two steps:
  - Assigns each member to cluster of closest centroid, minimizing WCSS
  - Updates centroid of each cluster



K-means++ Method

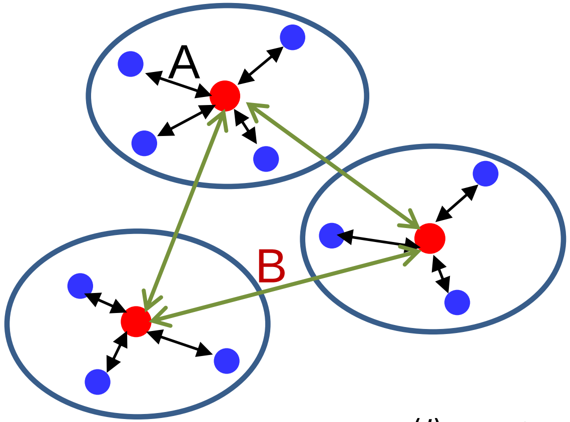
- Clustering results of k-means strongly depends on initial cluster assignment.

- k-means++ method*: improves clustering accuracy by selecting k members with distance vectors as initial centroid of clusters
  - Randomly selects one member as centroid of one cluster
  - Calculates distance D(x) of each member x to closest centroid already selected
  - Randomly selects another member proportionally with $D(x)^2$ as another centroid
  - Repeats above procedure until k centroids are selected.

*D. Arthur and S. Vassilvitskii, k-means++: the advantages of careful seeding, ACM SODA 2007

Optimally Setting Cluster Count K

- Jain-Dubes method*
  - Applies k-means for each k in range of $2 \leq k \leq 1+\log_2 n$
  - Selects k minimizing cost p(k)



$$p(k) = \frac{1}{k}\sum_{i=1}^{k} \max_{1 \leqq j \leqq k} \left\{ \frac{\eta_i + \eta_j}{\xi_{ij}} \right\}$$

$$\eta_j = \frac{1}{n_j}\sum_{i=1}^{n_j} D\left(\boldsymbol{x}_i^{(j)}, \boldsymbol{m}_j\right) \qquad \xi_{ij} = D(\boldsymbol{m}_i, \boldsymbol{m}_j)$$

$x_i^{(j)}$: i-th member of cluster j, n_j: number of members of cluster j

m_j: centroid of cluster j, D(**a**,**b**): distance b/w vectors **a** and **b**

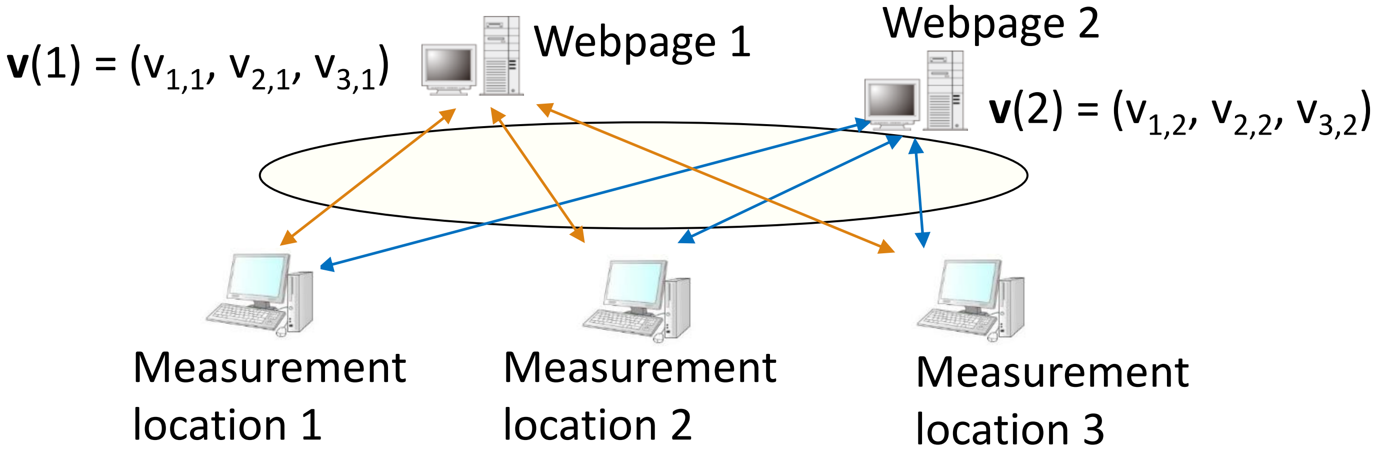
- Corresponds to minimize ratio of (A) average distance of members to centroid within each cluster against (B) average distance between centroids of any pair of clusters

*A. K. Jain and R. C. Dubes, Algorithms for clustering data, Prentice-Hall, 1988

Clustering Analysis of Webpages based on RTT

- Analyzed geographical tendencies by clustering webpages based on average RTT at 12 access locations

- Applied k-means method based on vectors **v**(y) with elements v_{xy}, average RTT b/w access location x and objects of webpage y.
- Optimally set cluster count k using JD method
- Set initial cluster using k-means++ method



$\mathbf{v}(1) = (v_{1,1}, v_{2,1}, v_{3,1})$

Webpage 1

Webpage 2

$\mathbf{v}(2) = (v_{1,2}, v_{2,2}, v_{3,2})$

Measurement location 1

Measurement location 2

Measurement location 3

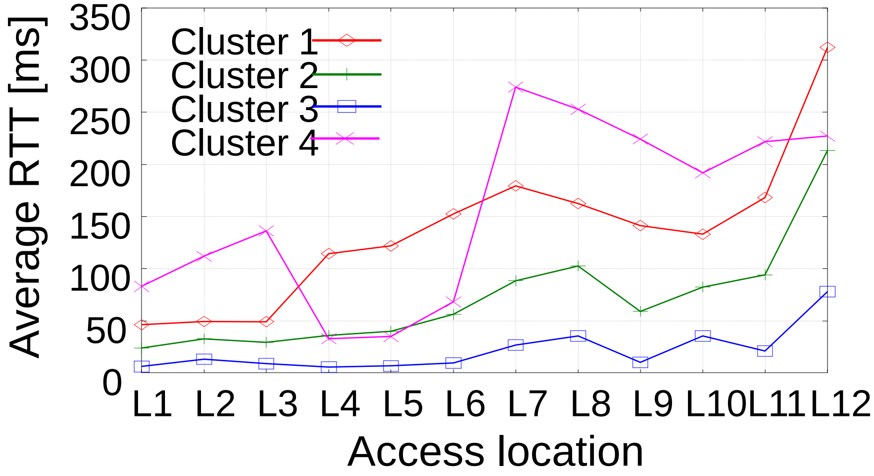
Geographical Distribution of Original Objects

L1	Massachusetts	L7	Australia
L2	Wisconsin	L8	New Zealand
L3	California	L9	Japan
L4	Ireland	L10	Ecuador
L5	Germany	L11	Argentina
L6	Russia	L12	Reunion

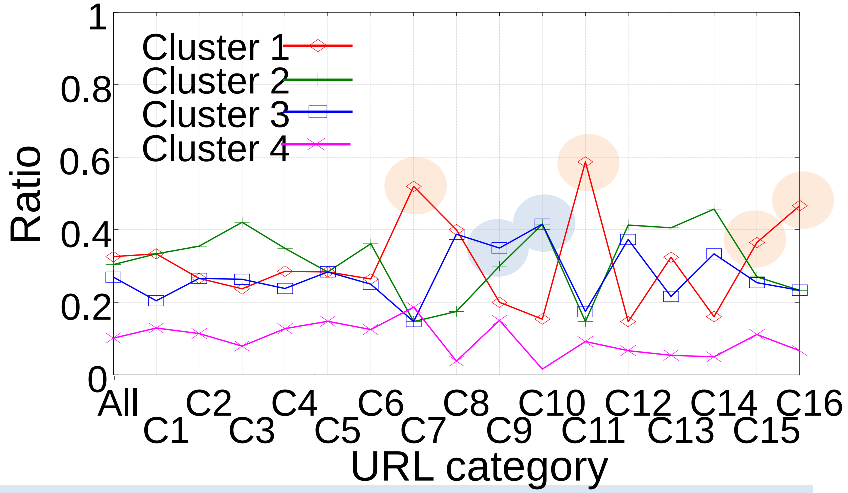
Clustering webpages based on RTT of non-CDN objects at midnight

C1	Business	C5	Regional	C9	Home	C13	Games
C2	Computers	C6	Science	C10	Shopping	C14	Kids & teens
C3	News	C7	Society	C11	Adult	C15	Recreation
C4	Reference	C8	Health	C12	Arts	C16	Sports

Av. RTT of non-CDN objects of each cluster from each access location



Ratio of websites of each category classified in each cluster



- Cluster 1: RTT was small only in North America. ⇒ Geographical locality is weak, and identical content are viewed from various regions.
- Cluster 3: RTT was small in all areas except Africa. ⇒ Geographical locality is strong, and unique content are viewed in each region.

Confirmed different tendencies of object deployment among web categories

Roughly Estimating Web Response Time

Assumption

- Starts obtaining objects on all TCP co. with all servers
- Fairly obtains objects over all TCP co. with each server
- Continuously receives objects on each TCP connection
- Obtains each object from edge servers with probability H with zero RTT

H: cache hit ratio

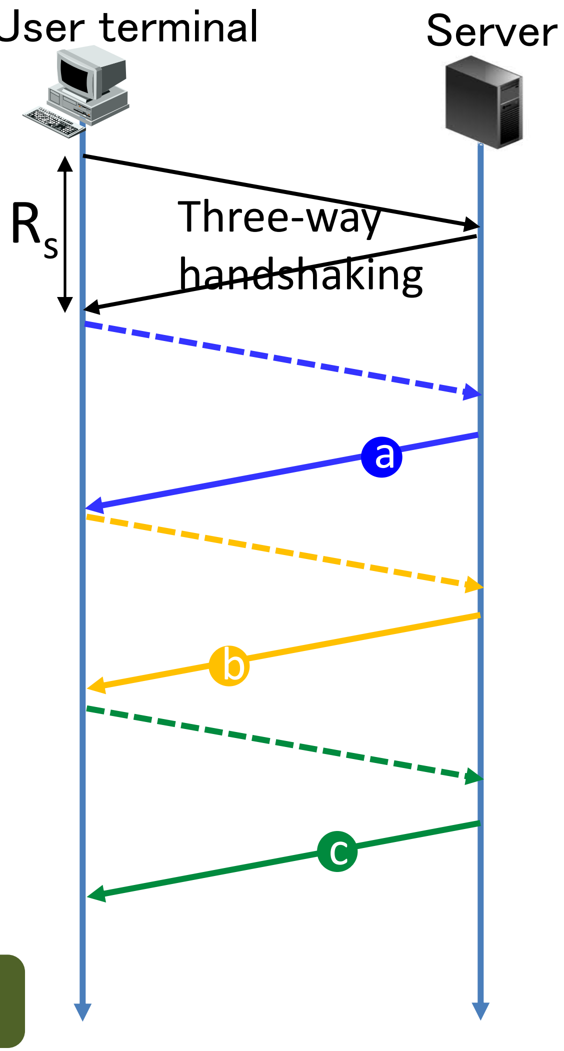
D_x: estimated time reduced by delivering objects of webpage x from edge servers

$$D_x = \max_{s \in \boldsymbol{S}_x} \left\{ \left\lceil \frac{M_s H}{P} \right\rceil + \lfloor H \rfloor \right\} R_s$$

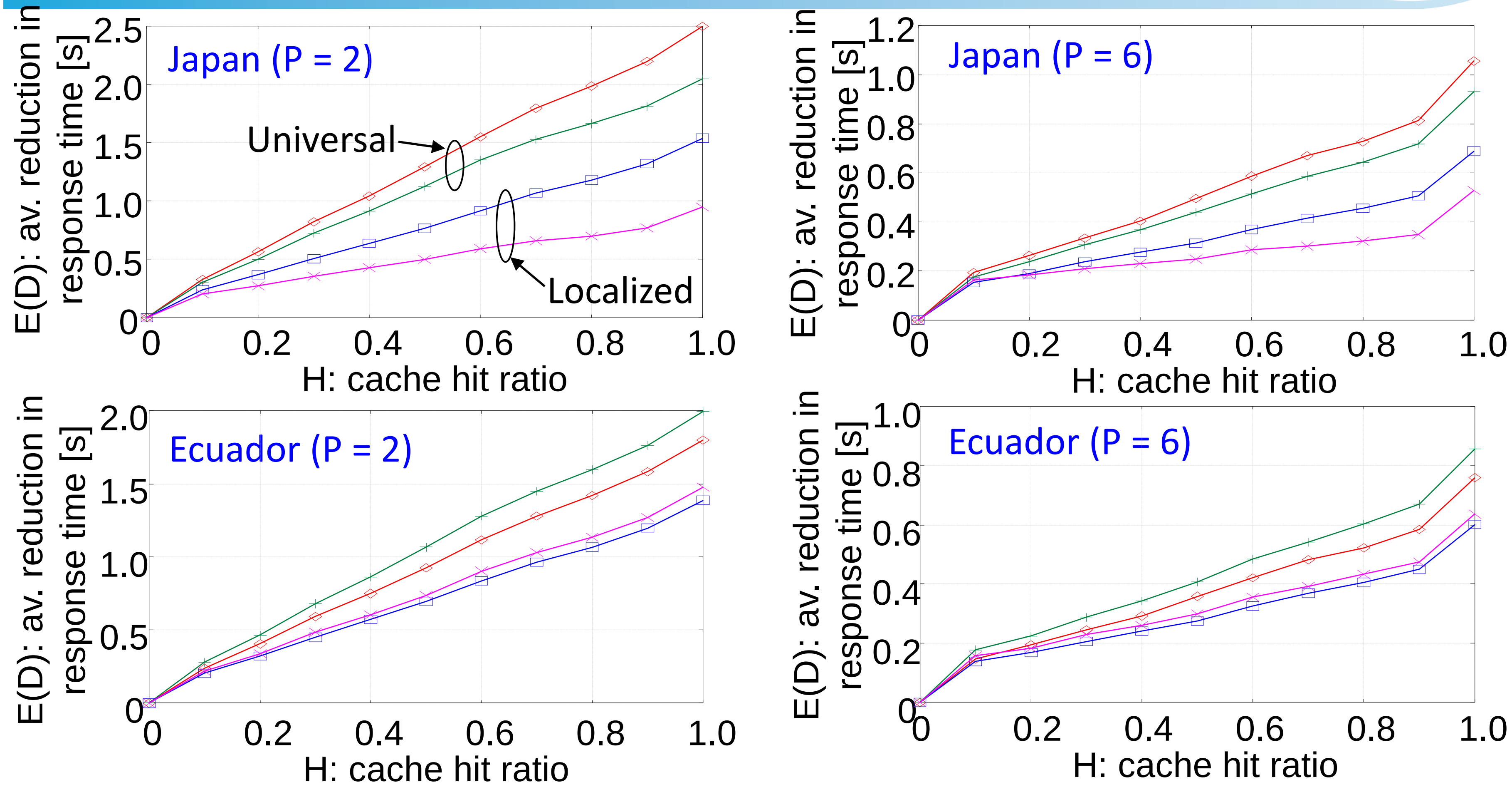
- $\mathbf{S}_x$: set of servers sending objects of page x
- M_s: number of objects obtained from server s
- R_s: average RTT b/w user terminal and sever s

Apply measured value

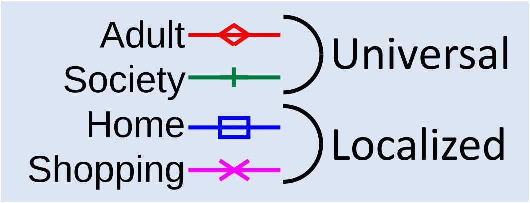
Flow sequence on TCP connection



Average Reduction in Response Time of Four Categories

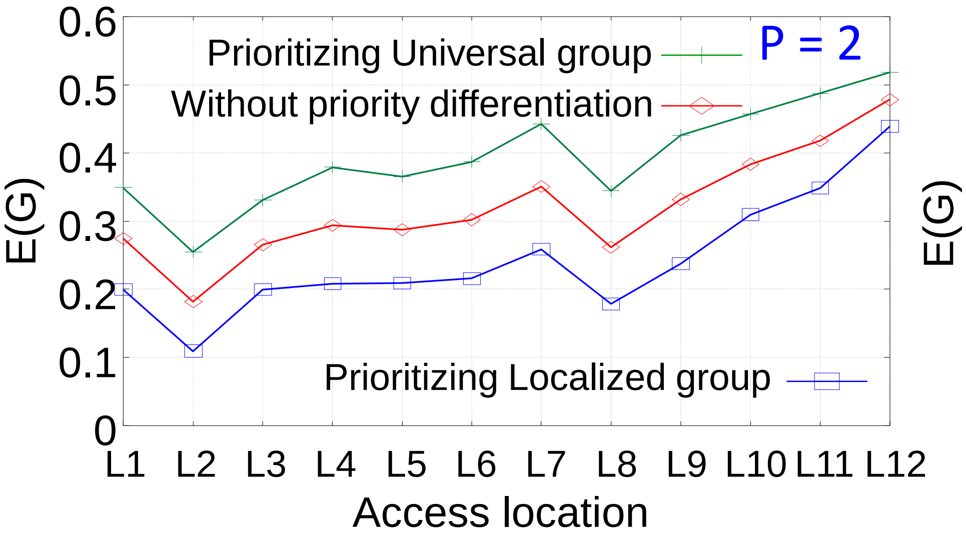


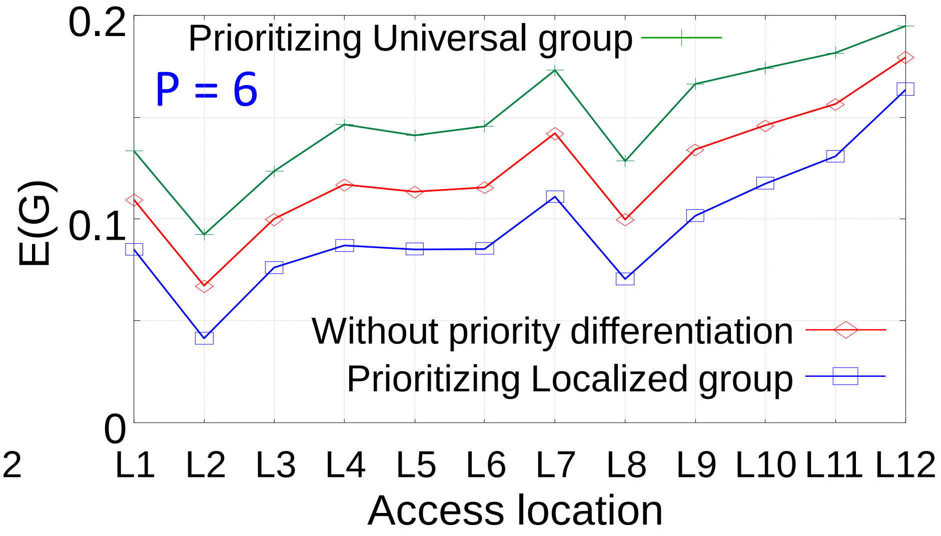
Confirm difference of E(D) between Universal websites (adult, society) and Localized websites (home, shopping)



Effect of Web Category Differentiation in Edge Computing

E(G), average reduction ratio of web response time:
E(G) = E(D)/(average response time without edge computing)





L1	Massachusetts	L4	Ireland	L7	Australia	L10	Ecuador
L2	Wisconsin	L5	Germany	L8	New Zealand	L11	Argentina
L3	California	L6	Russia	L9	Japan	L12	Reunion

- Compare E(G) among three caching policies:
  - Without priority differentiation: delivering 50% of objects of each category
  - Prioritizing universal group: delivering all objects of Adult and Society webpages
  - Prioritizing localized group: delivering all objects of Home and Shopping webpages

Can improve effect of edge computing by prioritizing universal webpages

- Mechanism providing web browsing services
- Possible factors degrading web response time
- Geographical tendency of web content deployment
- Approaches reducing web response time
  - Overview
  - Inlining, Prefetch, and SPDY and HTTP/2
  - CDN enhancement

Major Techniques Reducing Web Response Time

- Inlining
- Prefetch
- SPDY and HTTP/2
- CDN enhancement
  - For content providers:
    - Increases objects delivered using CDNs
    - Switches CDN provider
  - For CDN providers:
    - Increases investment for cache servers
    - Uses better selection policy of cache servers
    - Uses better cache replacement policy
    - Carefully distributes data objects over cache servers

14 Rules Improving Frontend Web Performance by Souders

S. Souders, High-Performance Web Sites, Communication of the ACM, 2008

- 1. Make fewer HTTP requests
- 2. Use a content delivery network
- 3. Add an Expires header
- 4. Gzip components
- 5. Put stylesheets at the top
- 6. Put scripts at the bottom
- 7. Avoid CSS expressions ⇒ Setting CSS dynamically is time consuming.
- 8. Make JavaScript and CSS external ⇒ Do not use Inlining.
- 9. Reduce DNS lookups ⇒ Use fewer hostnames
- 10. Minify JavaScript
- 11. Avoid redirects
- 12. Remove duplicate scripts
- 13. Configure ETags
- 14. Make Ajax cacheable

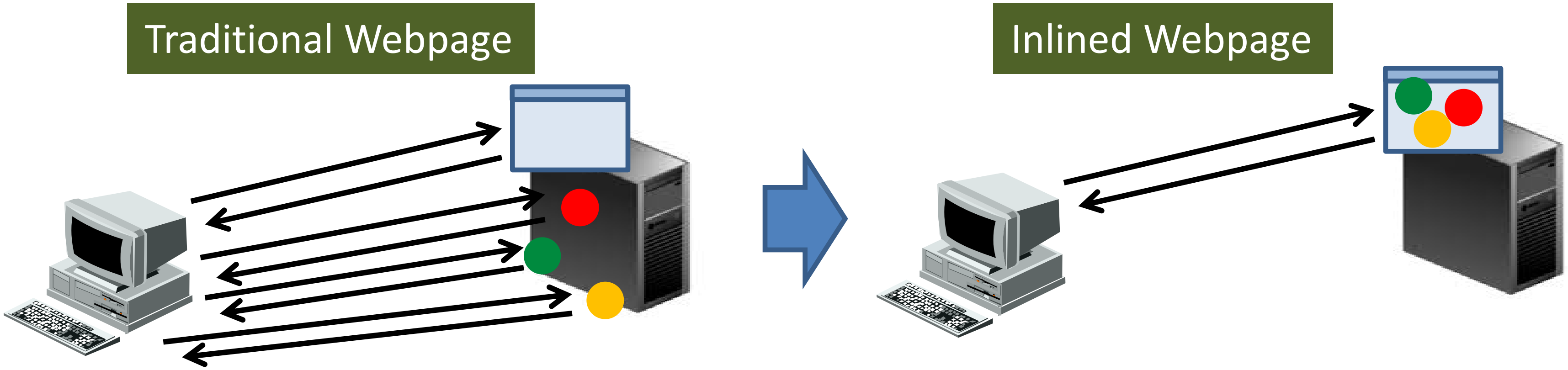
- Mechanism providing web browsing services
- Possible factors degrading web response time

- Geographical tendency of web content deployment

- Approaches reducing web response time
  - Overview
  - Inlining, Prefetch, and SPDY and HTTP/2
  - CDN enhancement

Inlining

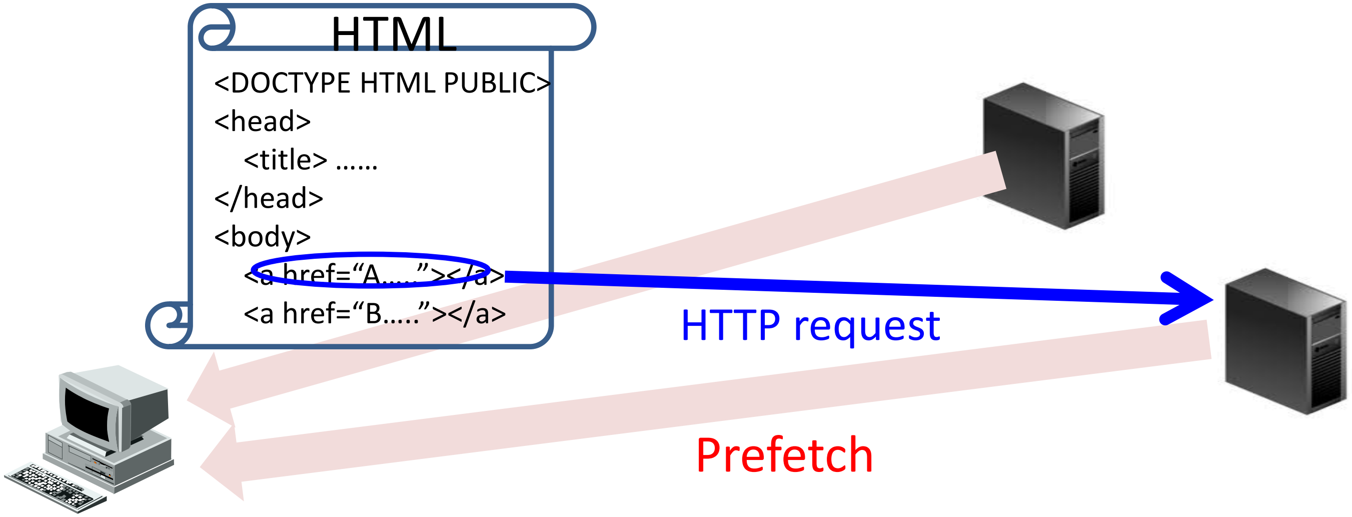
- Decreases number of HTTP connections by
  - Concatenating multiple JavaScript files into fewer JavaScript files or combining multiple style sheets into a single CSS file
  - Directly inserting them into HTML file



- Only applicable to objects provided by same server of HTML file
- Need to refetch entire part of inlined file even when just a single object is updated.
- Requires support of web servers

Prefetch

- Link prefetch: sending HTTP request to URL which is embedded in HTML file, i.e., hyperlink, and receive HTML file as well as data objects before user actually clicks hyperlinks
- DNS prefetch: resolving URL which is embedded in HTML file in advance



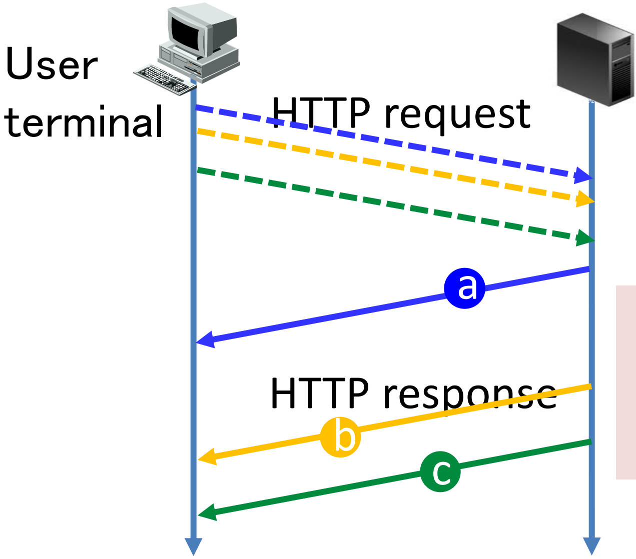
Pros: prefetched webpage can be displayed with short response time.

Cons: network and computation resources are wasted if prefetched wabpages are not actually browsed.

SPDY and HTTP/2

- SPDY was developed by Google to solve limitation of HTTP pipelining.
- SPDY is going to move to HTTP/2.

HTTP pipelining



User terminal

Object Server

HTTP request

a

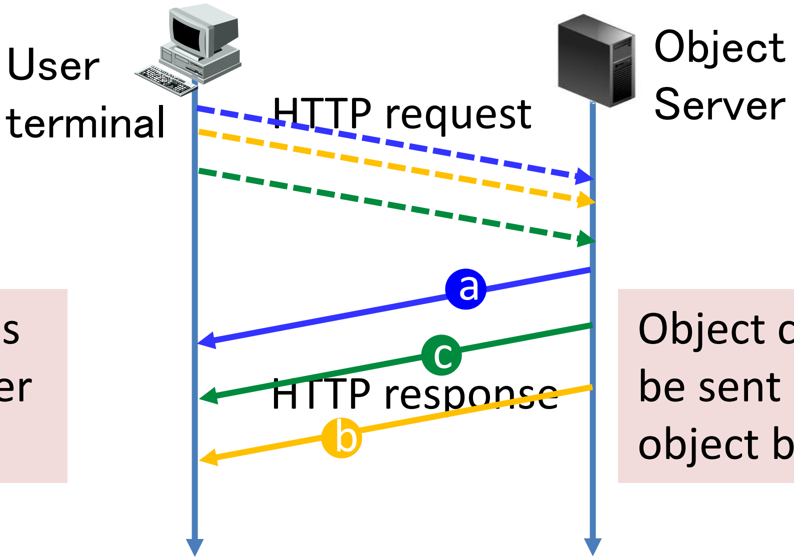
HTTP response

b

c

Object c needs to be sent after object b.

SPDY and HTTP/2



User terminal

Object Server

HTTP request

a

c

HTTP response

b

Object c can be sent before object b.

- Object servers need to send HTTP responses in the order of received HTTP requests.
  ⇒ HOL problem

- Keeps status of each HTTP session as "SPDY stream" and enables UT to distinguish each HTTP session
- Solves HOL problem of HTTP pipelining

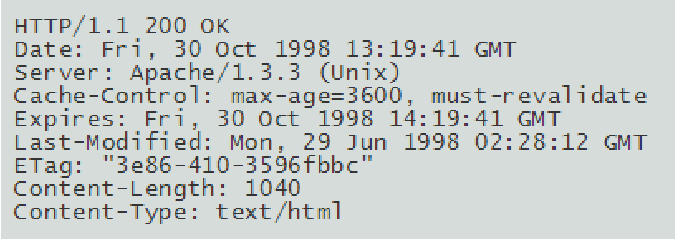
- Mechanism providing web browsing services
- Possible factors degrading web response time
- Geographical tendency of web content deployment
- Approaches reducing web response time
  - Overview
  - Inlining, Prefetch, and SPDY and HTTP/2
  - CDN enhancement

Increase Objects Delivered Using CDNs

- Solution for content providers to increase objects delivered using CDNs
  - Subscribes CDN providers (for static objects)
  - Increases number of cacheable objects (for static objects)
  - (Possibly) Uses edge computing (for dynamic objects)

- **Tips for increasing cacheable object count***

  (also applicable to proxy and browser caches)
  - Carefully sets HTTP header, i.e, expires, last-modified, and Etag, related to validation
  - Uses identical URL for identical content among various webpages and users
  - Not changes data files of objects unnecessarily
  - Minimizes use of cookies



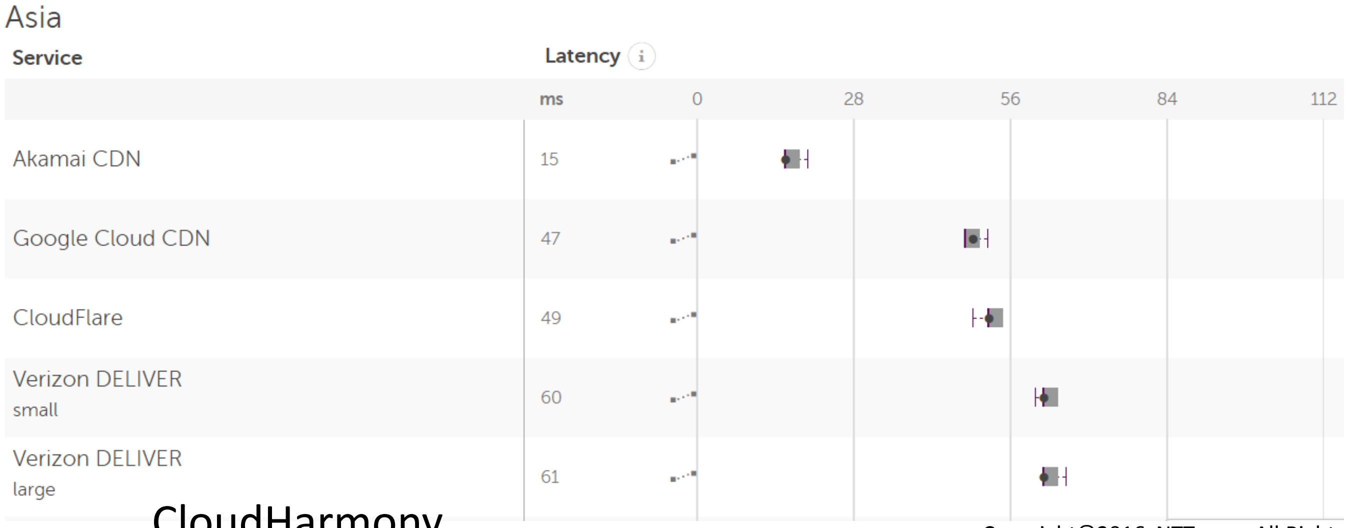
```
HTTP/1.1 200 OK
Date: Fri, 30 Oct 1998 13:19:41 GMT
Server: Apache/1.3.3 (Unix)
Cache-Control: max-age=3600, must-revalidate
Expires: Fri, 30 Oct 1998 14:19:41 GMT
Last-Modified: Mon, 29 Jun 1998 02:28:12 GMT
ETag: "3e86-410-3596fbbc"
Content-Length: 1040
Content-Type: text/html
```

*Caching Tutorial, https://www.mnot.net/cache_docs/

Switches CDN Providers

- Content providers can send their content using CDNs by subscribing CDN providers.
- Desirable to subscribe CDN providers with reasonable performance in areas of potential target users
- Can select CDN providers by using report service of third party

- **CloudHarmony**: Reports performance of various CDNs in various areas
- **Cedexis**: Can measure HTTP response time by inputting access area and CDN
- **Probe API**: Provides API responding measurement results of HTTP response time using probes over world



Asia

Service	Latency ms
Akamai CDN	15
Google Cloud CDN	47
CloudFlare	49
Verizon DELIVER small	60
Verizon DELIVER large	61

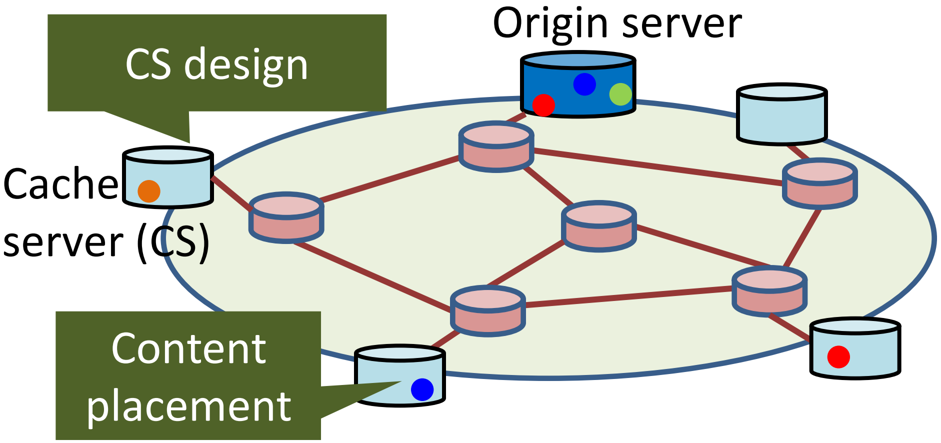
CloudHarmony

Design and Control Issues in CDNs

- CDN providers face various design and control issues to operate CDNs.

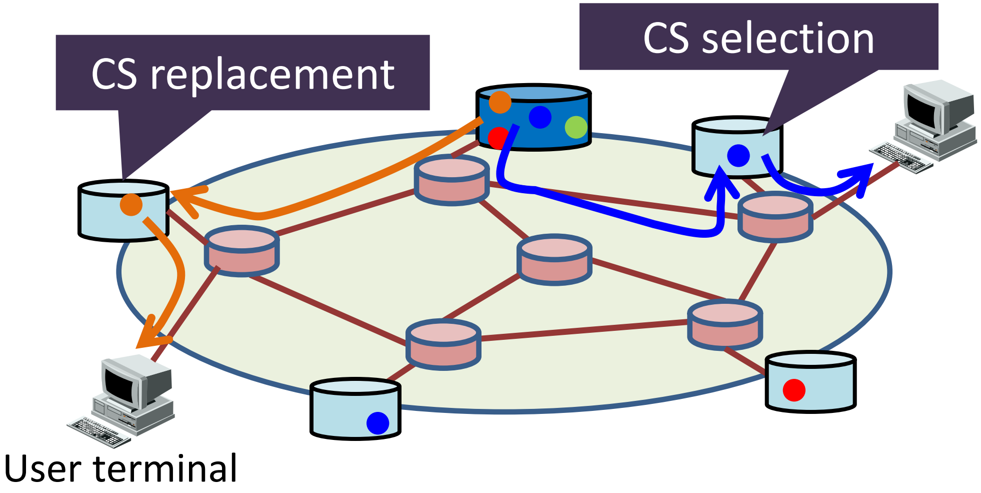
Related to content deployment

- **Cache server design**: determining location and capacity of cache servers
- **Content placement**: determining location of placing each content



Related to content delivery

- **Cache server selection**: selecting cache server for each request
- **Cache replacement**: selecting content to be removed at capacity shortage of cache servers



Optimum Design of Cache Server Locations

- Location of cache servers should be determined considering various factors including total cost and CDN performance, e.g., average latency to users.
- Min K-Center: graph-theoretic algorithm finding a set of center nodes minimizing maximum distance between a node and its closest center

- *l*-Greedy: approximate min K-Center*
  - Places cache servers on network iteratively in greedy fashion
  - Exhaustively checks each node to determine node that best satisfies optimization condition, i.e., RTT
  - Allows for *l* steps backtracking, i.e., checking all possible combinations of removing *l* of already placed cache servers and replacing them with *l*+1 new cache servers

*P. Krishnan, D. Raz, and Y. Shavitt, "The cache location problem," ACM/IEEE Trans. Networking, vol. 8, pp. 568–582, Oct. 2000.

Cache Replacement

- When cache miss occurs, cache server obtains content from origin server, caches obtained content, and sends it to requesting user.
- When available storage capacity is insufficient, some content items are removed from cache server using cache replacement algorithm.

Existing cache replacement algorithms

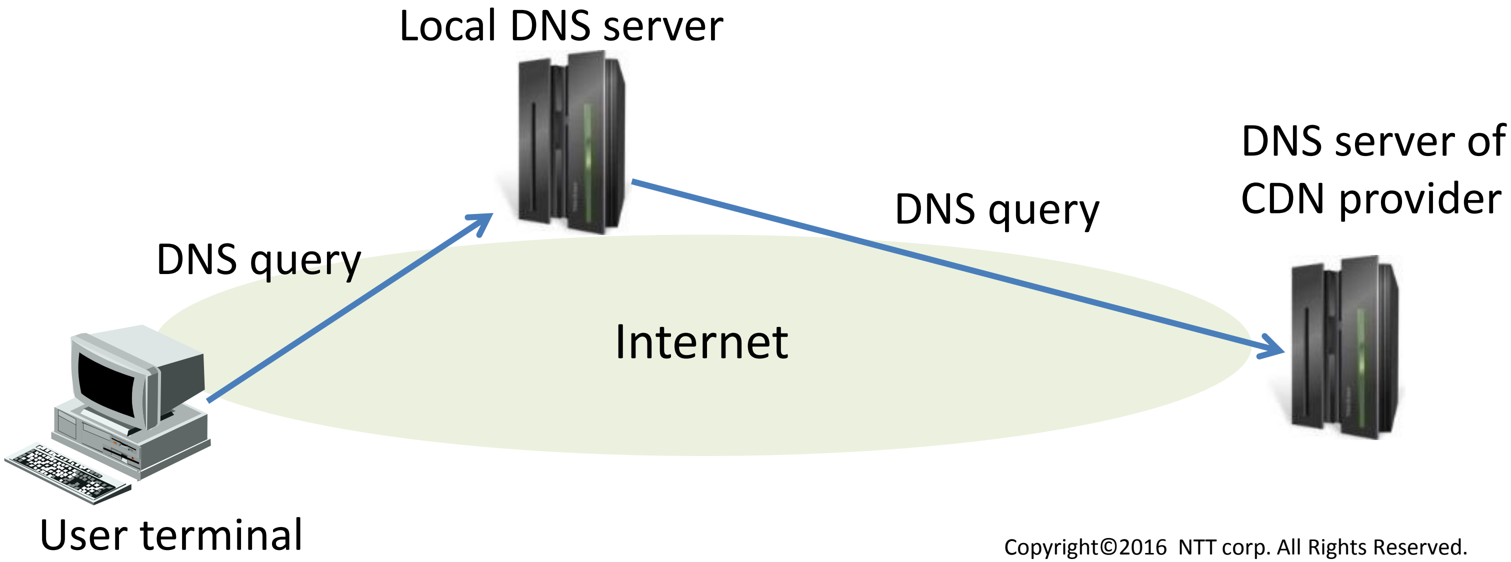
- LRU (least recently used):
  - Removes content item which is least recently accessed
- LFU (least frequently used):
  - Removes content item with smallest access frequency

Existing cache replacement algorithms only consider popularity of content.

HTTP response time will be possibly minimized by considering other factors, e.g., server load and server distance.

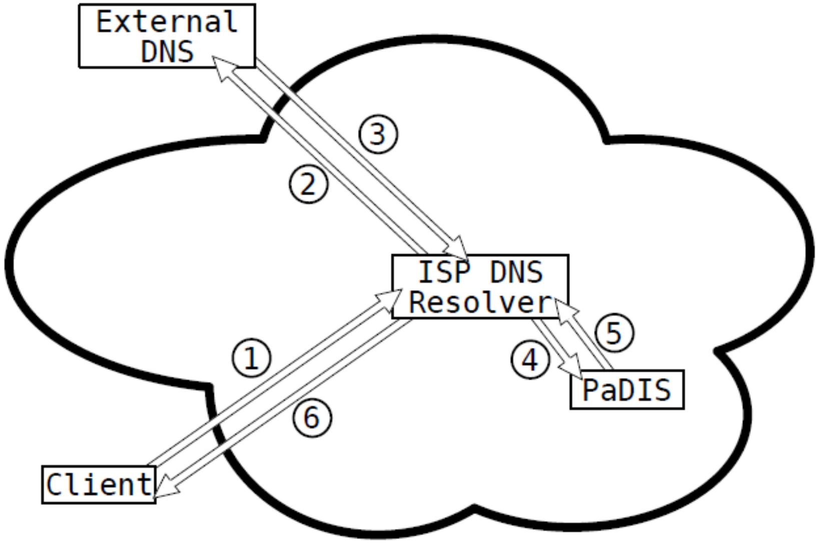
Difficulties in Selecting Optimum Cache Servers

- CDN providers cannot use network information, e.g., topology, link load, and network congestion, because they are confidential information of network providers.
- Without using network information, many CDN providers simply select cache servers based on location of user terminals estimated by source IP addresses.
- However, source IP addresses are for local DNS (LDNS) servers querying name resolution, not for user terminals.



Provider-aided Distance Information System (PaDIS)

- ISP manages PaDIS server storing rank of path candidates for each domain name based on network information.
- After receiving name resolution results from External DNS server operated by CDN providers, LDNS queries to PaDIS server.
- PaDIS server returns best path based on rank information for domain name and updates PaDIS server using retuned addresses from External DNS.



I. Poese, B. Frank, B. Ager, G. Smaragdakis, and A. Feldmann, Improving Content Delivery Using Provider-aided Distance Information, ACM IMC 2010

Conclusion

- Abstracted basic technologies supporting web browsing services, i.e., HTTP, CDN, and objects

- Summarized possible factors degrading web response time from two aspects: network delay and computation delay

- Analyzed tendencies of web content deployment and effect of prioritizing in caching objects based on web categories through active measurement of most popular 1,000 webpages from 12 locations

- Overviewed various standard and advanced approaches reducing web response time, i.e., Inlining, Prefetch, SPDY (HTTP/2), and CDN enhancement