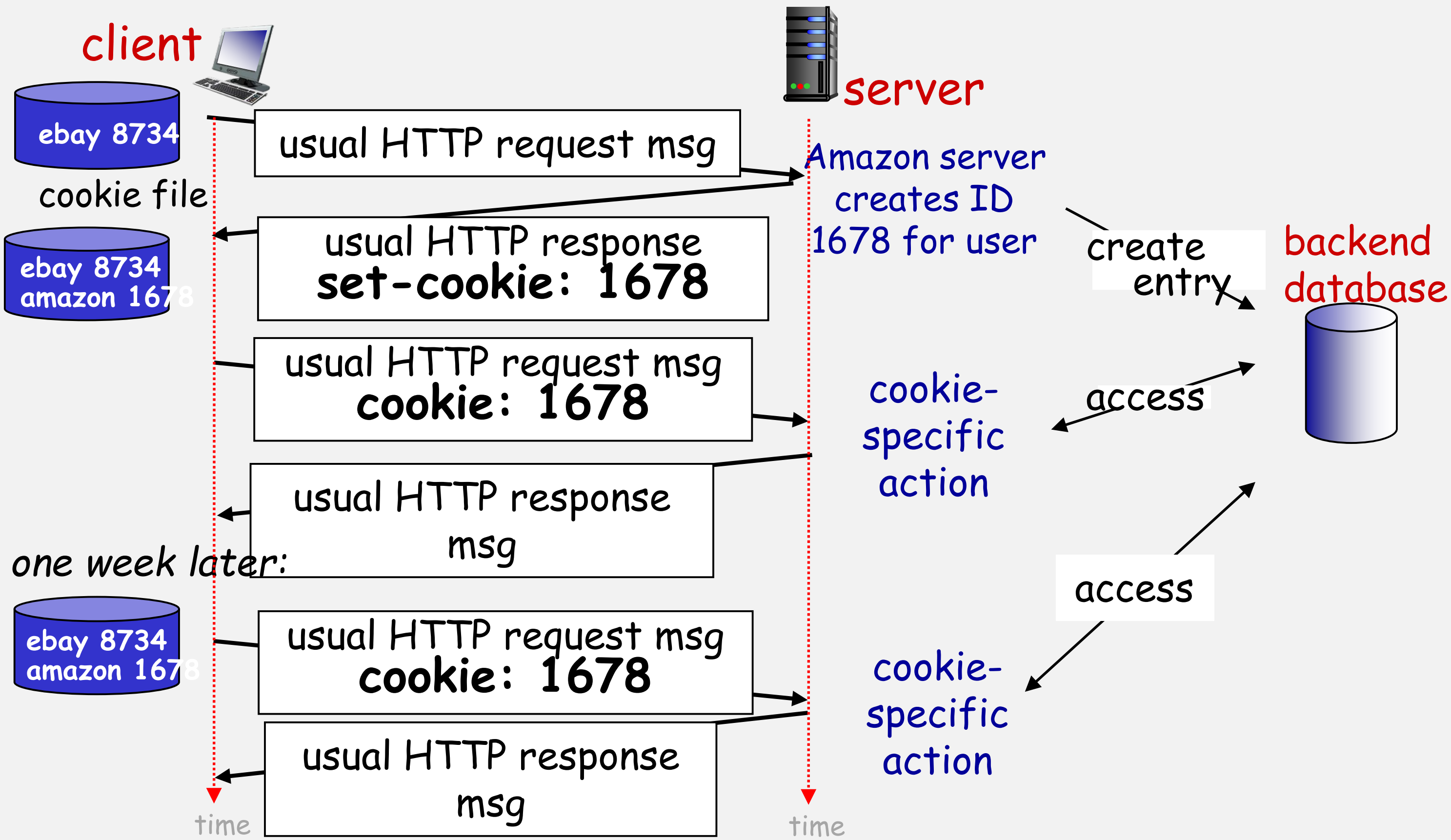
CS5222 Computer Networks and Internets Tutorial (Week 3)

Prof Weifa Liang

Weifa.liang@cityu.edu.hk

Slides based on book *Computer Networking: A Top-Down Approach.*

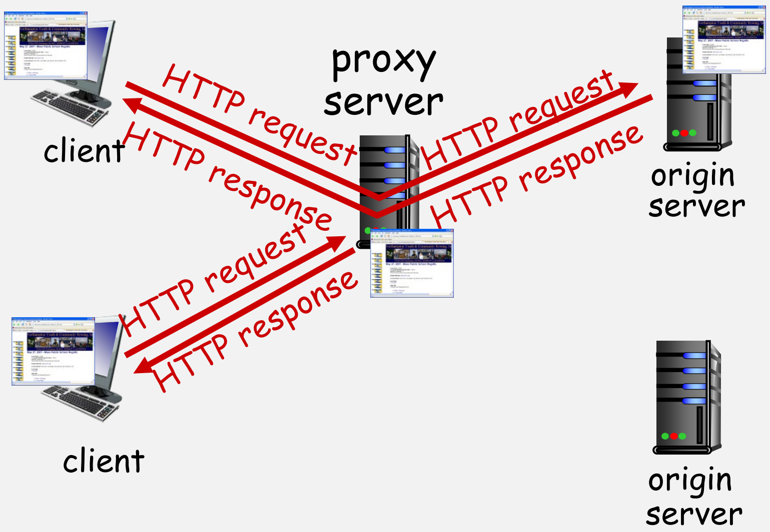
Maintaining user/server state: cookies



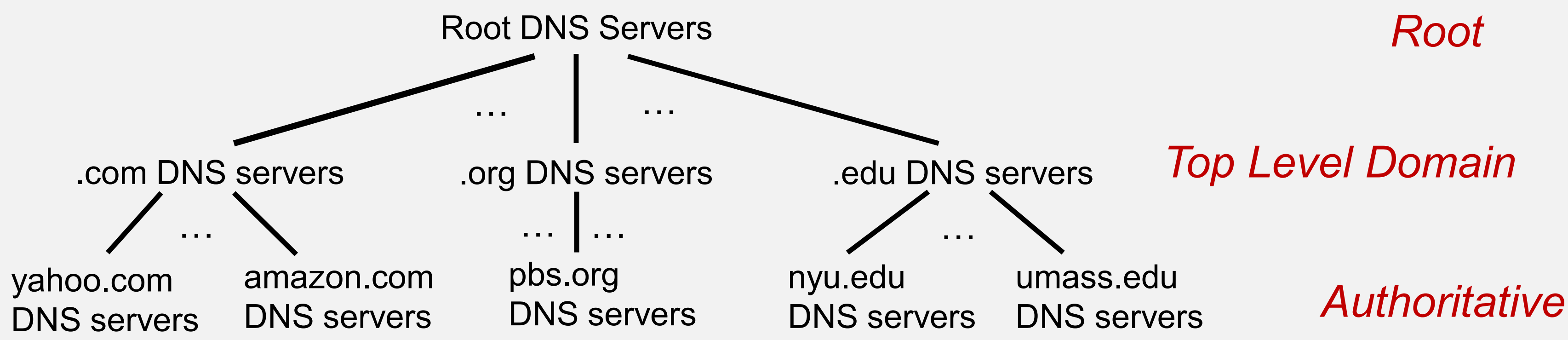
Web caches (proxy servers)

Goal: satisfy client request without involving origin server

- user configures browser to point to a *Web cache*
- browser sends all HTTP requests to cache
  - *if* an object is in cache: cache returns the object to the client
  - *else* cache requests the object from origin server, caches the received object, then returns the object to the client



DNS: a distributed, hierarchical database



Client wants IP address for www.amazon.com; (1st approximation):

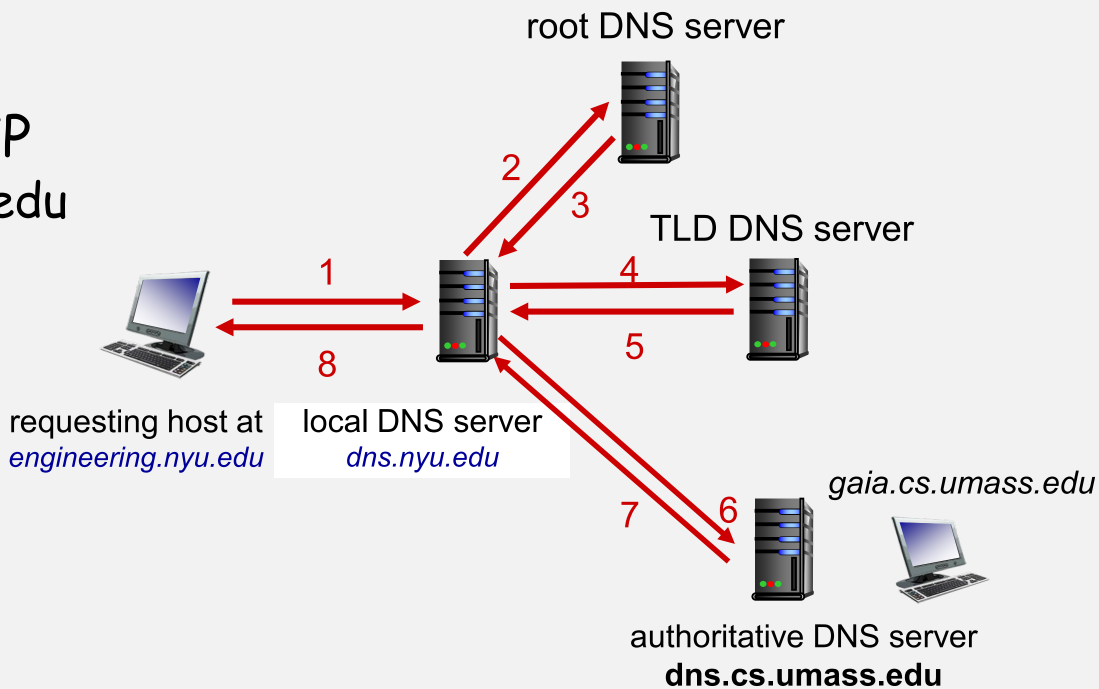
- client queries root server to find .com DNS server
- client queries .com DNS server to get amazon.com DNS server
- client queries amazon.com DNS server to get IP address for www.amazon.com

DNS name resolution: iterated query

Example: host at engineering.nyu.edu wants IP address for gaia.cs.umass.edu

Iterated query:

- contacted server replies with name of server to contact
- "I don't know this name, but ask this server"

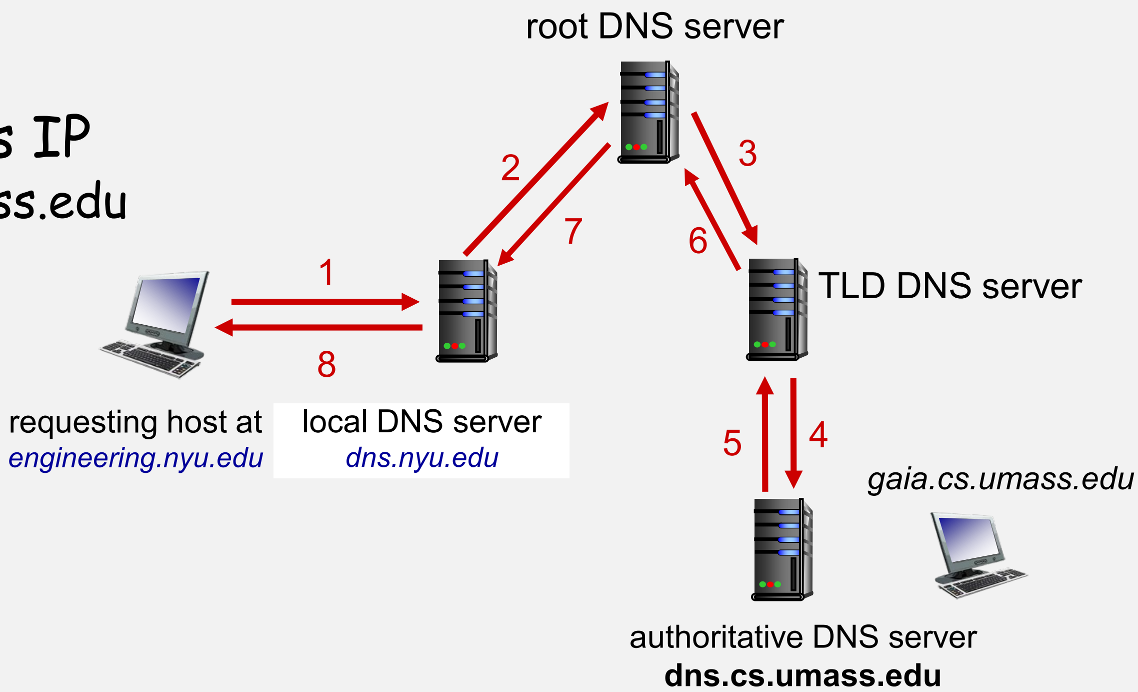


DNS name resolution: recursive query

Example: host at engineering.nyu.edu wants IP address for gaia.cs.umass.edu

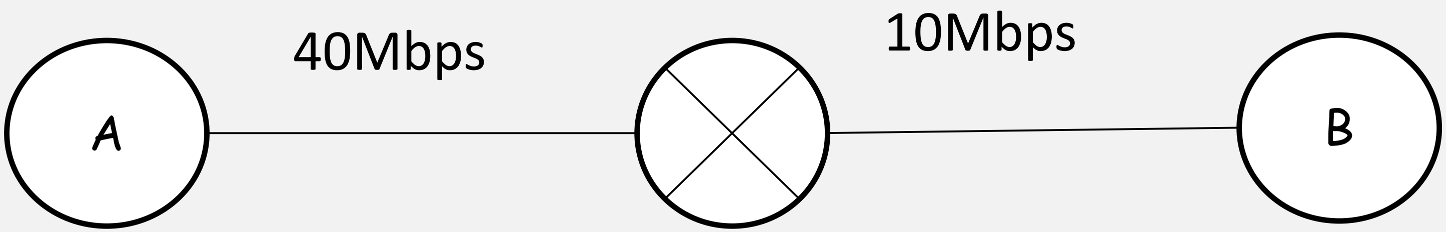
Recursive query:

- puts burden of name resolution on contacted name server
- heavy load at upper levels of hierarchy?



Work on questions

1. Suppose that A has a file with size of 1 Gbits to send to B through the following path. **How much time (in sec) will pass from the time when B receives the first bit of the file until B has received the whole file?**



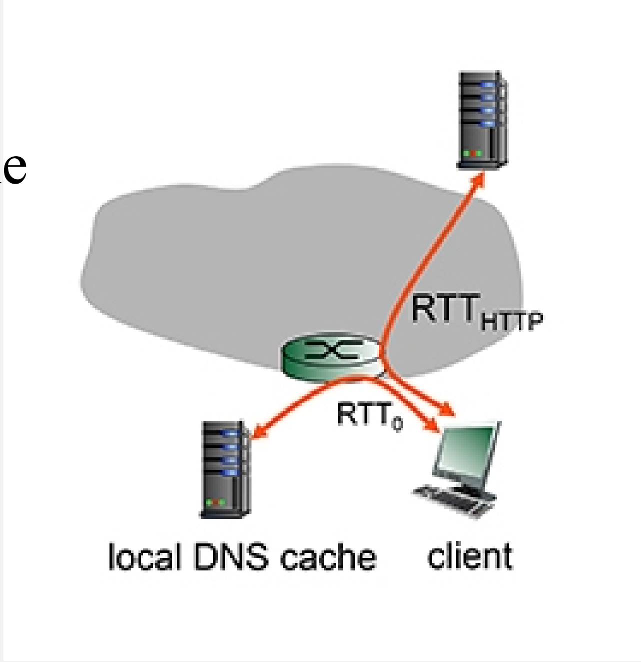
Answer:

1Gbits=1000M bits

- The throughput of the routing path is 10Mbps, i.e., B can receive 10M bits per sec from A.
- It thus takes 1Gbits/10Mbps=100 sec for B to receive the file.

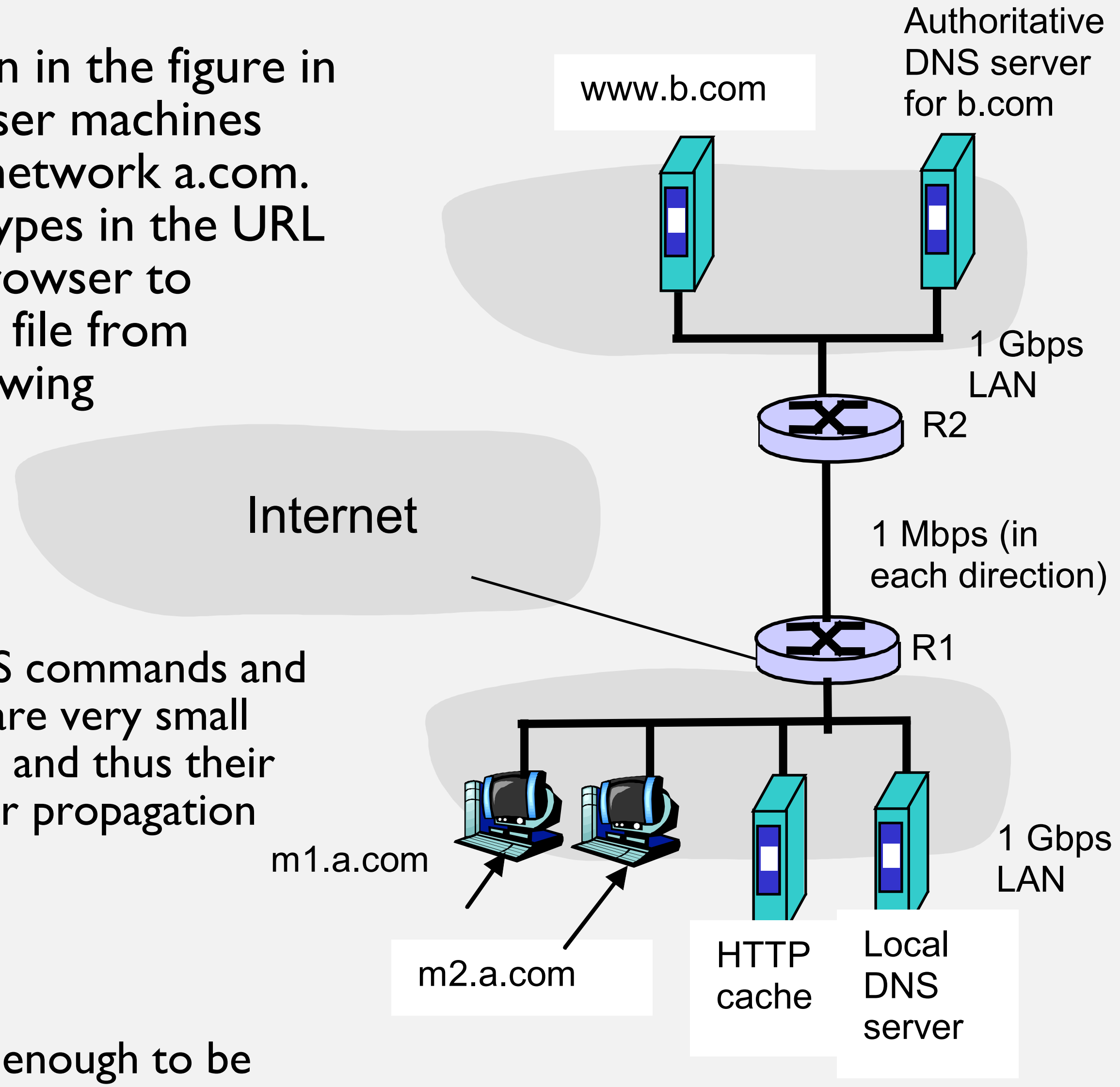
2. Suppose within your Web browser you click on a link to obtain a Web page. The IP address for the associated URL is not cached in your local host, so a DNS lookup is necessary to obtain the IP address. It takes **RTT_0 = 5 msecs** for the host to send a DNS lookup and get the IP address. The Web page associated with the link does not reference any other objects. The RTT between the local host and the Web server containing the object is **RTT_{HTTP} = 15** msecs.

Assuming it takes zero transmission time for the HTML object, how much time elapses from when the client clicks on the link until the client receives the object?



- Answer:
- The time from when the Web request is made in the browser until the page is displayed in the browser is
- $RTT_0 + 2*RTT_{HTTP} = 5 + 2*15 = 35$ msecs.
- Note that 2 RTT_{HTTP}s are needed to fetch the HTML object - one RTT_{HTTP} to establish the TCP connection, and the other RTT_{HTTP} to perform the HTTP GET/response over that TCP connection.

3. Consider the networks shown in the figure in the right side. There are two user machines m1.a.com and m2.a.com in the network a.com. Suppose the user at m1.a.com types in the URL www.b.com/bigfile.htm into a browser to retrieve a 1Gbit (or 1,000 Mbit) file from www.b.com. We have the following **assumptions:**



- The packets containing any DNS commands and HTTP commands such as GET are very small compared to the size of the file, and thus their transmission times (but not their propagation times) can be neglected.
- Propagation delays within the LAN are small enough to be ignored. The propagation from router R1 to router R2 is small enough to be ignored.
- The propagation delay from anywhere in a.com to any other site in the Internet (except b.com) is 500 ms (=0.5 seconds).

List the sequence of DNS and HTTP messages sent/received from/by m1.a.com as well as any other messages that leave/enter the a.com network that are not directly sent/received by m1.a.com from the point that the URL is entered into the browser until the file is completely received. Indicate the source and destination of each message.

You can assume that every HTTP request by m1.a.com is first directed to the HTTP cache in a.com. Assume that all caches (HTTP cache and local DNS server) are initially empty. Moreover, all DNS requests are iterated queries. Calculate the time for m1.a.com to receive the file.

Answer:

Name resolution messages and delay:

- *M1.a.com needs to resolve the name www.b.com to an IP address so it sends a DNS REQUEST message to its local DNS resolver*
  - *(this takes no time given the assumptions)*
- *Local DNS server does not have any information so it contacts a root DNS server with a REQUEST message*
  - *(this take 500 ms given the assumptions)*
- *Root DNS server returns name of DNS Top Level Domain server for .com*
  - *(this takes 500 ms given the assumptions)*
- *Local DNS server contacts .com TLD*
  - *(this takes 500 ms given the assumptions)*
- *TLD .com server returns the authoritative name server for b.com*

  (this takes 500 ms given the assumptions)
- *Local DNS server contacts the authoritative name server for b.com*
  - *(this takes no time given the assumptions)*
- *Authoritative name server for b.com returns IP address of www.b.com.*
  - *(this takes no time given the assumptions)*

Answer (continued):

HTTP messages and delay

- *Since we ignore the propagation delay and transmission delay for short messages, RTT for TCP connection is ignored.*
- *HTTP client sends HTTP GET message to www.b.com, which it sends to the HTTP cache in the a.com network (this takes no time given the assumptions).*
- *The HTTP cache does not find the requested document in its cache, so it sends the GET request to www.b.com. (this takes no time given the assumptions)*
- *www.b.com receives the GET request. It takes 1000 seconds to send a file of size of 1Gbps from www.b.com to the cache, and then 1 second to send it m1.a.com.*
- *The total delay is thus: .5 + .5 + .5 +.5 + 1,000 +1= 1,003 sec.*

Now assume that machine m2.a.com makes a request to exactly the same URL that m1.a.com made.

List the sequence of DNS and HTTP messages sent/received from/by m2.a.com as well as any other messages that leave/enter the a.com network that are not directly sent/received by m2.a.com from the point that the URL is entered into the browser until the file is completely received. Indicate the source and destination of each message.

Answer:

- *m2.a.com needs to resolve the name www.b.com to an IP address, so it sends a DNS REQUEST message to its local DNS resolver*
  - *this takes no time given the assumptions*
- *The local DNS server looks in its cache and finds the IP address for www.b.com, since m1.a.com had just requested that name which has been resolved already, and returns the IP address to m2.a.com.*
  - *this takes no time given the assumptions*
- *HTTP client at m2.a.com sends a HTTP GET message to www.b.com, which it sends to the HTTP cache in the a.com network (this takes no time given the assumptions).*
- *The HTTP cache finds the requested document in its cache, so it sends a GET request with an* ***If-Modified-Since*** *to www.b.com. (this takes no time given the assumptions)*
- *www.b.com receives the GET request. The document has not changed, so www.b.com sends a short HTTP RESPONSE message to the HTTP cache in a.com indicating that the cached copy is valid. (this takes no time given the assumptions)*
- *There is a 1 sec delay to send the 1Gbps file from the HTTP cache to m2.a.com.*
- *The total delay is thus: 1 sec*

4. You accessed Amazon.com before. When you access Amazon.com again, the website lists the items that you browsed before and provide some recommendations to you. Explain how this happens.

Answer:

- Amazon’s server has a backend database.
- When a client accesses Amazon’s web server for the first time, a cookie (for the user) will be generated and an entry will be added to its backend database.
- This cookie can be stored at the client’s computer permanently.
- When a client browses an item, the cookie and the item’s id are saved in the backend database of Amazon’s web server.
- When the client visits Amazon’s web server again, the cookie is included in the HTTP request sent to Amazon’s web server.
- By using the cookie id, all items that the client browsed previously can be retrieved from the backend database of the Amazon’s web server.

5. Suppose that a web browser wants to display a web page that contains references to 10 objects. Assume that the web page and its referenced objects are very small, hence their transmission times can be ignored.

For each one of the scenarios (a-d) stated below, answer the following two questions:

- How many HTTP request messages does the web browser need to send (in total) to retrieve all objects?
- How many RTTs does it take until the client has received all objects?

- **Scenarios:**

  a) The web browser can open up to 5 **parallel TCP connections** to the server over which it can send/receive HTTP messages. Assume that **non-persistent HTTP** is used.

  - **Answer**: For all scenarios, the number of requests is the same.
  - Non-persistent HTTP means we need to create a new TCP connection for each object. Therefore,
  - 2 RTTs for the base HTML file (one for TCP and one for HTML).
  - 2 RTTs for the first 5 objects since we can perform these in parallel (through the 5 parallel TCP connections).
  - Similarly, 2 RTTs for the remaining 5 objects.
  - In total: 6 RTTs.

  b) The web browser can open up to 5 **parallel TCP connections** to the server over which it can send/receive HTTP messages. Assume that **persistent HTTP** is used.

  - **Answer:** Since we are using the persistent HTTP, all requests/responses can be sent back-to-back, so we can just use a single TCP connection (each parallel connection transmits 2 objects). In total 3 RTTs (2 RTTs for the base HTML file + 1 RTT for 10 objects)

- **Scenarios:**

  c) The web browser can create a **single TCP connection** to the server over which it can send/receive HTTP messages. Assume that **non-persistent HTTP** is used.

  Answer:

  - Each HTTP request/response requires a separate TCP connection and these are established sequentially.
  - In total, we need 2*11 = 22 RTTs, where 2RTTs for a TCP connection and the base HTML file, and
  - 2 RTTS for each object downloading with one RTT for the TCP connection and another for request/response messages between the client and the server.

  d) The web browser can create a **single TCP connection** to the server over which it can send/receive HTTP messages. Assume that **persistent HTTP** is used.

  - **Answer:** Same as for scenario b). 3RTTs