CSC358 Tutorial 3

Julian Sequeira and KyoKeun Park

January 31, 2020

University of Toronto Mississauga

Question 1: Concept Review

(a) What happens to the user agents and the SMTP server when Alice sends an email to Bob?

(b) What's the difference between "*`RCPT TO:`*" and "*`TO:`*" in SMTP?

(c) What is at each level of the hierarchical structure of DNS? Why is it designed as a hierarchical structure?

(d) Why do we say "P2P is more scalable than client-server?"

(e) What's the point of "rarest-first" and "tit-for-tat"?

(f) Describe, as detailed as possible, everything that's happening behind the scene when you watch a movie on a video-streaming service, such as Netflix.

For the answers, review the lectures, books, go to office hours, and use the discussion board!

Question 2: All the RTT

Suppose you just unboxed a brand new laptop and are using a vanilla web browser (no browser cache, no parallel requests) to visit a new start-up's web page that has not been visited by anyone in the world yet. The web page consists of one HTML file containing references to one CSS file, one Javascript file, and 10 image files, all hosted on the same web server as the HTML file. Assume that the end-to-end delay for transmitting any application-layer request or response is a constant T. What is the total delay until you see the complete web page in your browser? Try to take **everything** happening behind the scene (as far as we have learned) into consideration. Discuss with the TA and others in the tutorial.

For the first object requested:

- 2T for round trip from client to local DNS server
- 2T for round trip from local DNS server to root DNS server
- 2T for round trip from local DNS to TLD DNS server (assuming iterative query, what about recursive queries?)
- 2T for round trip from local DNS server to authoritative name server of the start-up
- 2T for establishing HTTP connection
- 2T for getting the requested object

So 12T for the first object in total!

All the RTT

And then for all the other objects:

- 2T for round trip from client to local DNS server
- 2T for establishing HTTP connection
- 2T for getting the requested object
- 12 * 6 = 72T total for all 12 objects

So it takes 12 + 72 = 84T to get all objects.

Note- this is assuming non-persistent HTTP connections. How does a persistent connection change the calculation? What if there was a web cache involved?

File Distribution in P2P

Consider distributing a file of F bits to N peers using a P2P architecture. Assume the “fluid model”, i.e., the server (where the file is initially) can simultaneously transmit to multiple peers, transmitting to each peer at different rates, as long as the combined rate does not exceed the server upload bandwidth, u_s. For simplicity, assume that the peer download bandwidth, d_{min}, is very large so that it is never a bottleneck. Let u_c be a peer upload bandwidth of each of the N peers.

(a) Suppose that $u_s \leq u_c \cdot \frac{N}{N-1}$. Specify a distribution scheme (e.g., dividing the file in a certain way and sending certain parts to certain peers, etc.) that has a file distribution time (the time until every peer has a complete copy of the file) of

$$t = \frac{F}{u_s}$$

(b) Suppose that $u_s \geq u_c \cdot \frac{N}{N-1}$. Specify a distribution scheme that has a file distribution time of

$$t = \frac{N \cdot F}{u_s + N \cdot u_c}$$

(c) **Challenge Question**: What if each peer has a different upload bandwidth? Let u_i be the upload bandwidth of peer i where $1 \leq i \leq N$. Rethink part (a) and (b). How would the file distribution scheme change?

File Distribution in P2P (a)

(a) Suppose that $u_S \leq u_C \cdot \frac{N}{N-1}$. Specify a distribution scheme (e.g., dividing the file in a certain way and sending certain parts to certain peers, etc.) that has a file distribution time (the time until every peer has a complete copy of the file) of $t = \frac{F}{u_S}$

Basic idea: We know that the server upload bandwidth is slow, so let the peers handle most of it themselves

File Distribution in P2P (a)

(a) Suppose that $u_s \leq u_c \cdot \frac{N}{N-1}$*. Specify a distribution scheme (e.g., dividing the file in a certain way and sending certain parts to certain peers, etc.) that has a file distribution time (the time until every peer has a complete copy of the file) of* $t = \frac{F}{u_s}$

Basic idea: We know that the server upload bandwidth is slow, so let the peers handle most of it themselves

Divide the file into N parts of size $\frac{F}{N}$. The server will then upload each part to a different peer, at a rate of $r = \frac{u_s}{N}$

Note: $N \cdot r = u_s$, so the total bandwidth of the server is not exceeded *(a) Suppose that $u_s \leq u_c \cdot \frac{N}{N-1}$. Specify a distribution scheme (e.g., dividing the file in a certain way and sending certain parts to certain peers, etc.) that has a file distribution time (the time until every peer has a complete copy of the file) of $t = \frac{F}{u_s}$*

Basic idea: We know that the server upload bandwidth is slow, so let the peers handle most of it themselves

Divide the file into N parts of size $\frac{F}{N}$. The server will then upload each part to a different peer, at a rate of $r = \frac{u_s}{N}$

Note: $N \cdot r = u_s$, so the total bandwidth of the server is not exceeded

Each peer forwards the bits it receives to its $N-1$ peers at the same rate, r. So the aggregate forwarding rate is:

$$(N-1) \cdot r = (N-1) \cdot \frac{u_s}{N}$$

File Distribution in P2P (a)

(a) Suppose that $u_s \leq u_c \cdot \frac{N}{N-1}$. Specify a distribution scheme (e.g., dividing the file in a certain way and sending certain parts to certain peers, etc.) that has a file distribution time (the time until every peer has a complete copy of the file) of $t = \frac{F}{u_s}$

We still need to make sure that the upload bandwidth of each peer isn't exceeded. From the question we had:

$$u_s \leq u_c \cdot \frac{N}{N-1}$$

File Distribution in P2P (a)

(a) Suppose that $u_S \leq u_C \cdot \frac{N}{N-1}$. Specify a distribution scheme (e.g., dividing the file in a certain way and sending certain parts to certain peers, etc.) that has a file distribution time (the time until every peer has a complete copy of the file) of $t = \frac{F}{u_S}$

We still need to make sure that the upload bandwidth of each peer isn't exceeded. From the question we had:

$$u_S \leq u_C \cdot \frac{N}{N-1}$$

$$u_S \cdot \frac{N-1}{N} \leq u_C \cdot \frac{N}{N-1} \cdot \frac{N-1}{N}$$

File Distribution in P2P (a)

(a) Suppose that $u_S \leq u_C \cdot \frac{N}{N-1}$. Specify a distribution scheme (e.g., dividing the file in a certain way and sending certain parts to certain peers, etc.) that has a file distribution time (the time until every peer has a complete copy of the file) of $t = \frac{F}{u_S}$

We still need to make sure that the upload bandwidth of each peer isn't exceeded. From the question we had:

$$u_S \leq u_C \cdot \frac{N}{N-1}$$
$$u_S \cdot \frac{N-1}{N} \leq u_C \cdot \frac{N}{N-1} \cdot \frac{N-1}{N}$$
$$u_S \cdot \frac{N-1}{N} \leq u_C$$

File Distribution in P2P (a)

(a) Suppose that $u_S \leq u_C \cdot \frac{N}{N-1}$. Specify a distribution scheme (e.g., dividing the file in a certain way and sending certain parts to certain peers, etc.) that has a file distribution time (the time until every peer has a complete copy of the file) of $t = \frac{F}{u_S}$

We still need to make sure that the upload bandwidth of each peer isn't exceeded. From the question we had:

$$u_S \leq u_C \cdot \frac{N}{N-1}$$
$$u_S \cdot \frac{N-1}{N} \leq u_C \cdot \frac{N}{N-1} \cdot \frac{N-1}{N}$$
$$u_S \cdot \frac{N-1}{N} \leq u_C$$
$$(N-1) \cdot r \leq u_C$$

File Distribution in P2P (a)

(a) Suppose that $u_S \leq u_C \cdot \frac{N}{N-1}$. Specify a distribution scheme (e.g., dividing the file in a certain way and sending certain parts to certain peers, etc.) that has a file distribution time (the time until every peer has a complete copy of the file) of $t = \frac{F}{u_S}$

We still need to make sure that the upload bandwidth of each peer isn't exceeded. From the question we had:

$$
\begin{aligned}
u_s &\leq u_c \cdot \frac{N}{N-1} \\
u_s \cdot \frac{N-1}{N} &\leq u_c \cdot \frac{N}{N-1} \cdot \frac{N-1}{N} \\
u_s \cdot \frac{N-1}{N} &\leq u_c \\
(N-1) \cdot r &\leq u_c
\end{aligned}
$$

Then, each peer receives bits at an aggregate rate of:

$$r + (N-1) \cdot r = N \cdot r = u_s$$

Each peer then gets the file in $\frac{F}{u_s}$ time.
Remember the formula for transmission delay!

File Distribution in P2P (b)

Suppose that $u_S \geq u_C \cdot \frac{N}{N-1}$*. Specify a distribution scheme that has a file distribution time of* $t = \frac{N \cdot F}{u_S + N \cdot u_C}$

In this case, the server is (relatively) more powerful than the peers. We should try to offload the work to the server as much as possible to utilize the bandwidth!

File Distribution in P2P (b)

Suppose that $u_S \geq u_C \cdot \frac{N}{N-1}$. Specify a distribution scheme that has a file distribution time of $t = \frac{N \cdot F}{u_S + N \cdot u_C}$

In this case, the server is (relatively) more powerful than the peers. We should try to offload the work to the server as much as possible to utilize the bandwidth!

The file gets broken into $N+1$ parts. The server uploads bits from the first N parts to the N peers, one part for each peer, at rate $r = \frac{u_C}{N-1}$.

File Distribution in P2P (b)

Suppose that $u_S \geq u_C \cdot \frac{N}{N-1}$. Specify a distribution scheme that has a file distribution time of $t = \frac{N \cdot F}{u_S + N \cdot u_C}$

In this case, the server is (relatively) more powerful than the peers. We should try to offload the work to the server as much as possible to utilize the bandwidth!

The file gets broken into $N+1$ parts. The server uploads bits from the first N parts to the N peers, one part for each peer, at rate $r = \frac{u_C}{N-1}$.

Then, each peers need to forward the bits at a rate of r to each of the other $N-1$ peers. Hence, the upload rate of each peer is $r \cdot (N-1) = u_c$ (max upload capacity for peers)

Now, we need to consider the $(N+1)$st packet which server is taking care of.

File Distribution in P2P (b)

Suppose that $u_S \geq u_C \cdot \frac{N}{N-1}$. Specify a distribution scheme that has a file distribution time of $t = \frac{N \cdot F}{u_S + N \cdot u_C}$

Recall that our server is simultaneously forwarding bits to each peers! After considering the bandwidth used for peers, we have leftover upload rate of:

$$s = u_s - N \cdot r = u_s - u_c \cdot \frac{N}{N-1}$$

Would we have enough upload bandwidth for the last packet?

File Distribution in P2P (b)

Suppose that $u_S \geq u_C \cdot \frac{N}{N-1}$. Specify a distribution scheme that has a file distribution time of $t = \frac{N \cdot F}{u_S + N \cdot u_C}$

Recall that our server is simultaneously forwarding bits to each peers! After considering the bandwidth used for peers, we have leftover upload rate of:

$$s = u_s - N \cdot r = u_s - u_c \cdot \frac{N}{N-1}$$

Would we have enough upload bandwidth for the last packet? Yes, because of the assumption we make in the question.

The server uploads to each of the N peers at rate $\frac{s}{N}$. Note that the peers cannot forward the bits from $(N+1)$st part, since they have already used all of their bandwidth.

File Distribution in P2P (b)

Suppose that $u_S \geq u_C \cdot \frac{N}{N-1}$. Specify a distribution scheme that has a file distribution time of $t = \frac{N \cdot F}{u_S + N \cdot u_C}$

Recall that our server is simultaneously forwarding bits to each peers! After considering the bandwidth used for peers, we have leftover upload rate of:

$$s = u_s - N \cdot r = u_s - u_c \cdot \frac{N}{N-1}$$

Would we have enough upload bandwidth for the last packet? Yes, because of the assumption we make in the question.

The server uploads to each of the N peers at rate $\frac{s}{N}$. Note that the peers cannot forward the bits from $(N+1)$st part, since they have already used all of their bandwidth.

Putting the upload rate of file chunk 1 to N, as well as upload rate for $(N-1)$st chunk together, we get:

$$r \cdot N + s = (u_c \cdot \frac{N}{N-1}) + (u_s - u_c \cdot \frac{N}{N-1}) = u_s$$

So, the server upload and peer uploads are all maxed.

File Distribution in P2P (b)

Suppose that $u_s \geq u_c \cdot \frac{N}{N-1}$. Specify a distribution scheme that has a file distribution time of $t = \frac{NF}{u_S + N \cdot u_C}$

In this distribution scheme, each peer receives bits at an aggregate rate of:

$$d = r + (N-1) \cdot r + \frac{s}{N} = N \cdot r + \frac{N \cdot u_c}{N-1} + \frac{u_s}{N} - \frac{u_c}{N-1} = \frac{u_s}{N} + u_c$$

File Distribution in P2P (b)

Suppose that $u_s \geq u_c \cdot \frac{N}{N-1}$. Specify a distribution scheme that has a file distribution time of $t = \frac{NF}{u_s + N \cdot u_c}$

In this distribution scheme, each peer receives bits at an aggregate rate of:

$$d = r + (N-1) \cdot r + \frac{s}{N} = N \cdot r + \frac{N \cdot u_c}{N-1} + \frac{u_s}{N} - \frac{u_c}{N-1} = \frac{u_s}{N} + u_c$$

Therefore, the time taken to obtain the file is:

$$\frac{F}{d} = \frac{F}{u_s/N + u_c} = \frac{N \cdot F}{u_s + N \cdot u_c}$$