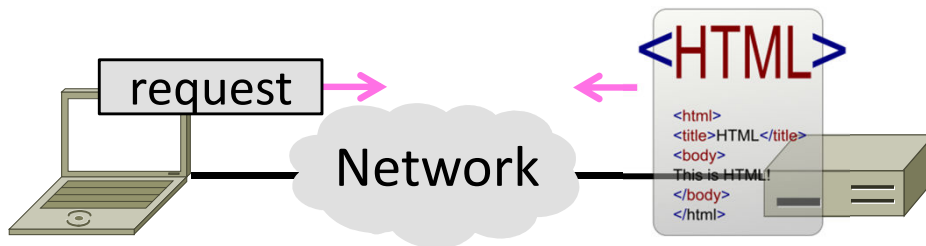# Computer Networks

HTTP Performance (§7.3.4)

David Wetherall (djw@uw.edu)
Professor of Computer Science & Engineering

UNIVERSITY *of* WASHINGTON

Topic

- Performance of HTTP
 - Parallel and persistent connections

PLT (Page Load Time)

- PLT is the key measure of web performance
  - From click until user sees page
  - Small increases in PLT decrease sales

- PLT depends on many factors
  - Structure of page/content
  - HTTP (and TCP!) protocol
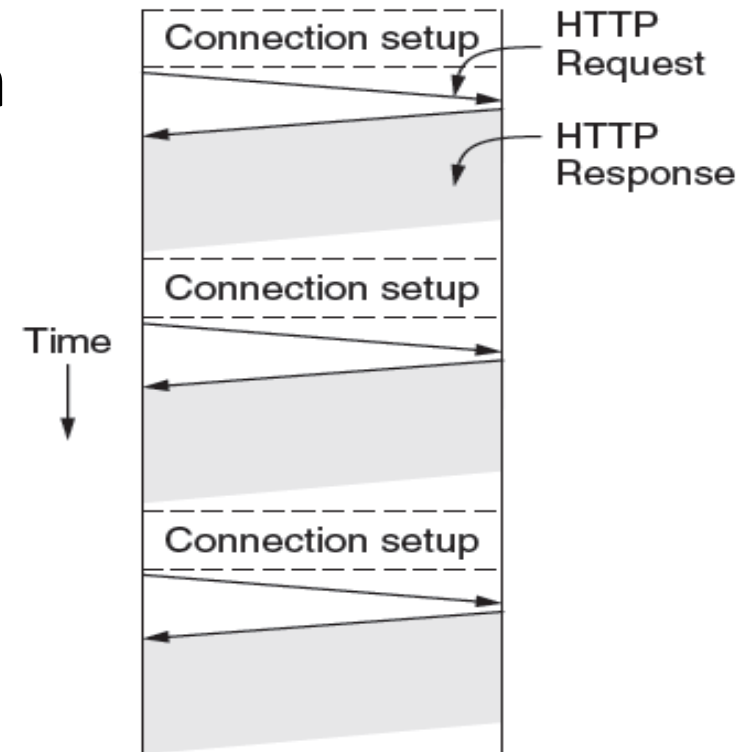 - Network RTT and bandwidth

Early Performance

- HTTP/1.0 uses one TCP connection to fetch one web resource
 - Made HTTP very easy to build
  - But gave fairly poor PLT …

Client · · · · Server

connect
req
resp

Time
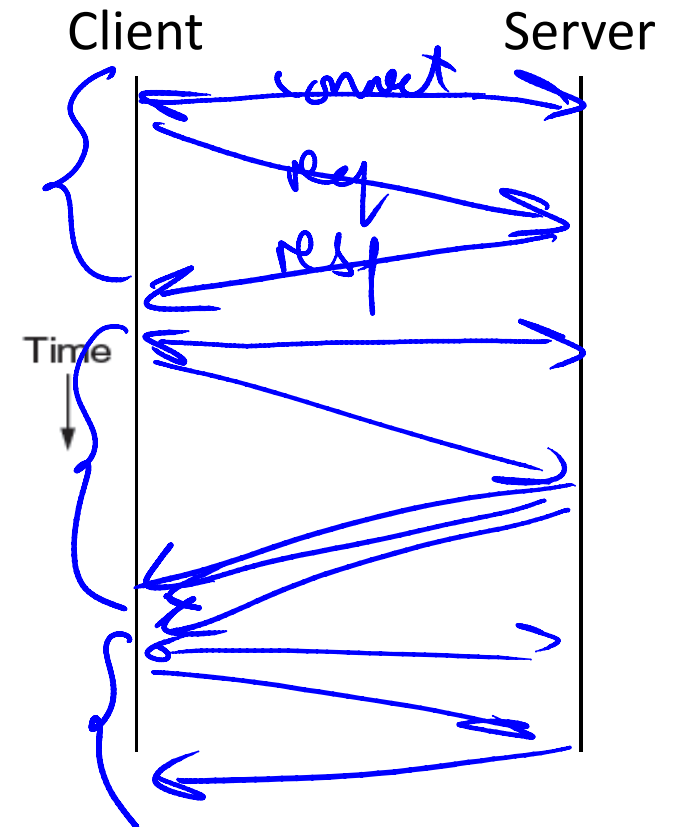
Early Performance (2)

- HTTP/1.0 used one TCP connection to fetch one web resource

  – Made HTTP very easy to build

  – But gave fairly poor PLT…

Early Performance (3)

- Many reasons why PLT is larger than necessary
  - Sequential request/responses, even when to different servers
  - Multiple TCP connection setups to the same server
  - Multiple TCP slow-start phases

- Network is not used effectively
  - Worse with many small resources / page
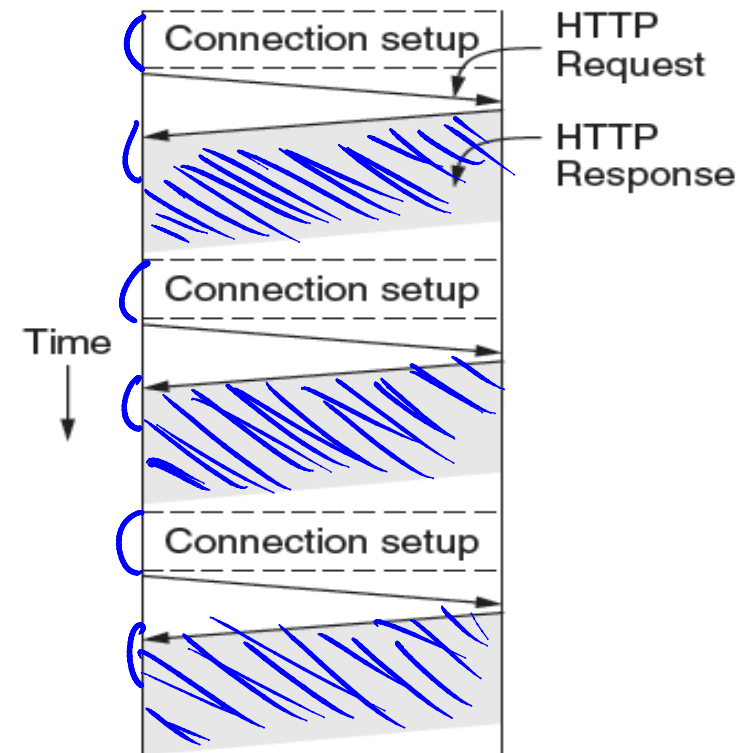
Ways to Decrease PLT

1. Reduce content size for transfer
   - Smaller images, gzip
2. Change HTTP to make better use of available bandwidth   } This time
3. Change HTTP to avoid repeated transfers of the same content   } Next time
   - Caching, and proxies
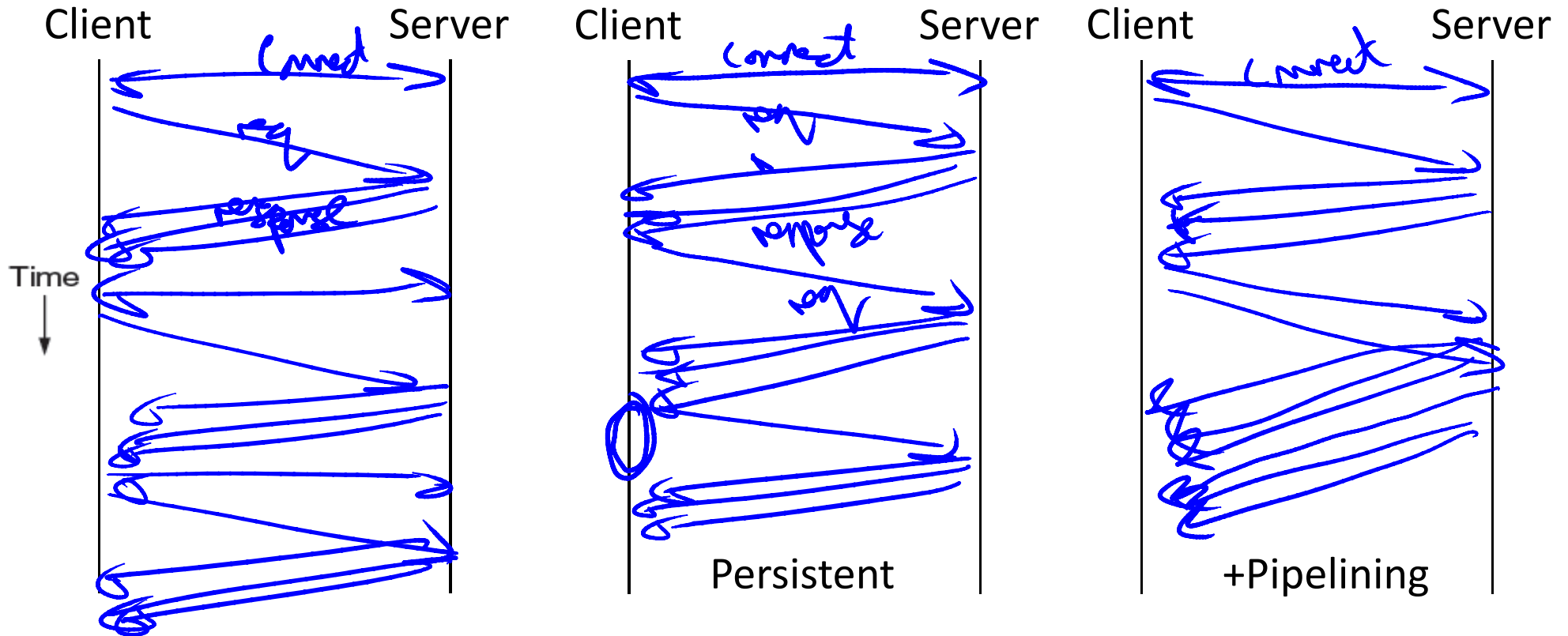4. Move content closer to client   } Later
   - CDNs [later]

Parallel Connections

- One simple way to reduce PLT

  – Browser runs multiple (8, say) HTTP instances in parallel

  Server is unchanged; already handled concurrent requests for many clients

- How does this help?

  Single HTTP wasn't using network much …

  So parallel connections aren't slowed much

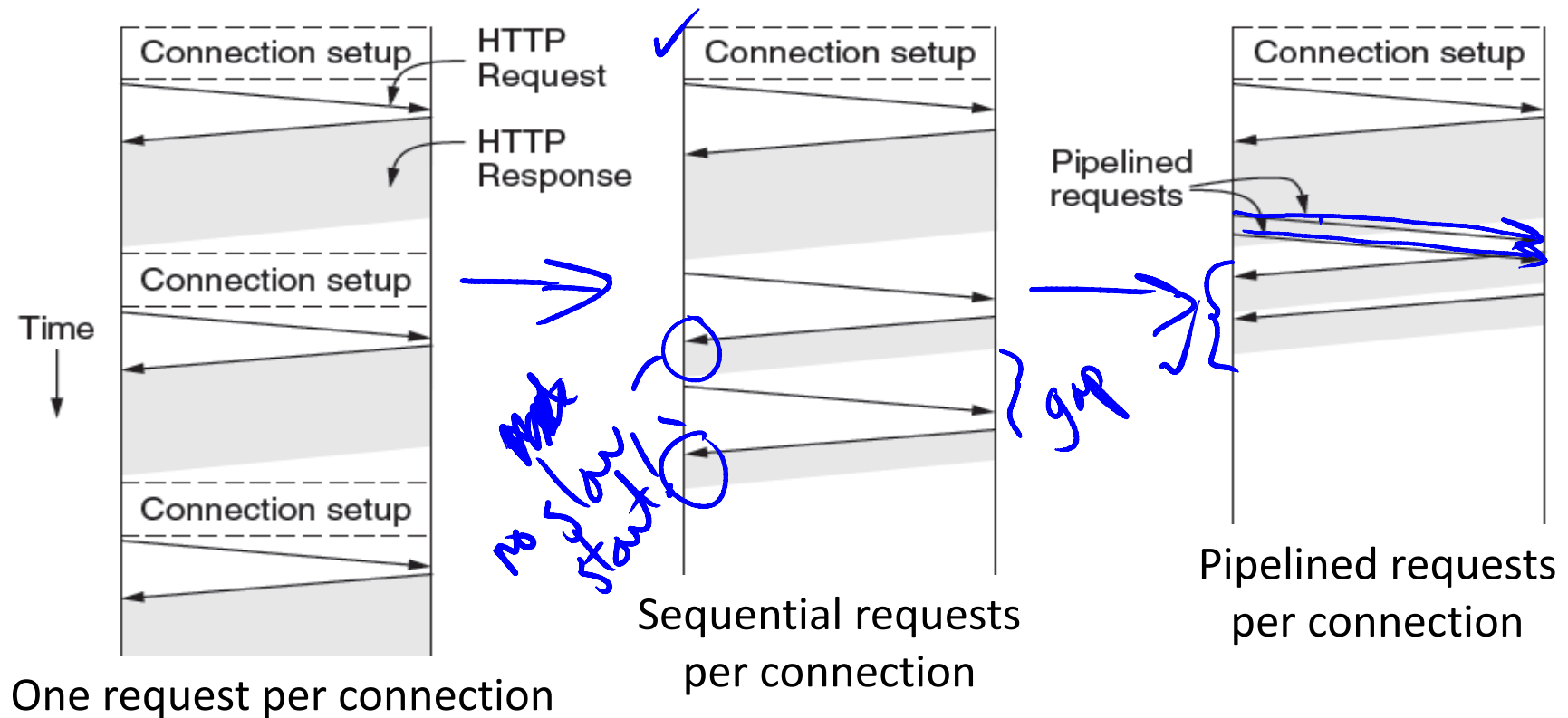 Pulls in completion time of last fetch

Persistent Connections

- Parallel connections compete with each other for network resources
  - 1 parallel client ≈ 8 sequential clients?
  - Exacerbates network bursts, and loss

- Persistent connection alternative
  - Make 1 TCP connection to 1 server
  - Use it for multiple HTTP requests

Persistent Connections (2)

Client   Server   Client   Server   Client   Server

Time

Persistent   +Pipelining

Persistent Connections (3)

One request per connection

Sequential requests per connection

Pipelined requests per connection

Persistent Connections (4)

- Widely used as part of HTTP/1.1

  - Supports optional pipelining
  - PLT benefits depending on page structure, but easy on network

- Issues with persistent connections

  – How long to keep TCP connection?

  – Can it be slower? (Yes. But why?)

END

© 2013 D. Wetherall