Computer Networking

Objective for this week:
- High-level understanding of TCP/IP Networking
- Appreciate why IP is so important

The notion of a Stack
- TCP/IP Protocol Stack

Application Layer Protocols
- SMTP, HTTP, IMAP, Telnet,FTP

Calling a local procedure

Remote Procedure Calls
- CGI
- SOAP
- REST

Buzzword Bingo

Latency
- Jitter

Bandwidth
- Downstream
- Upstream

Multiplexing

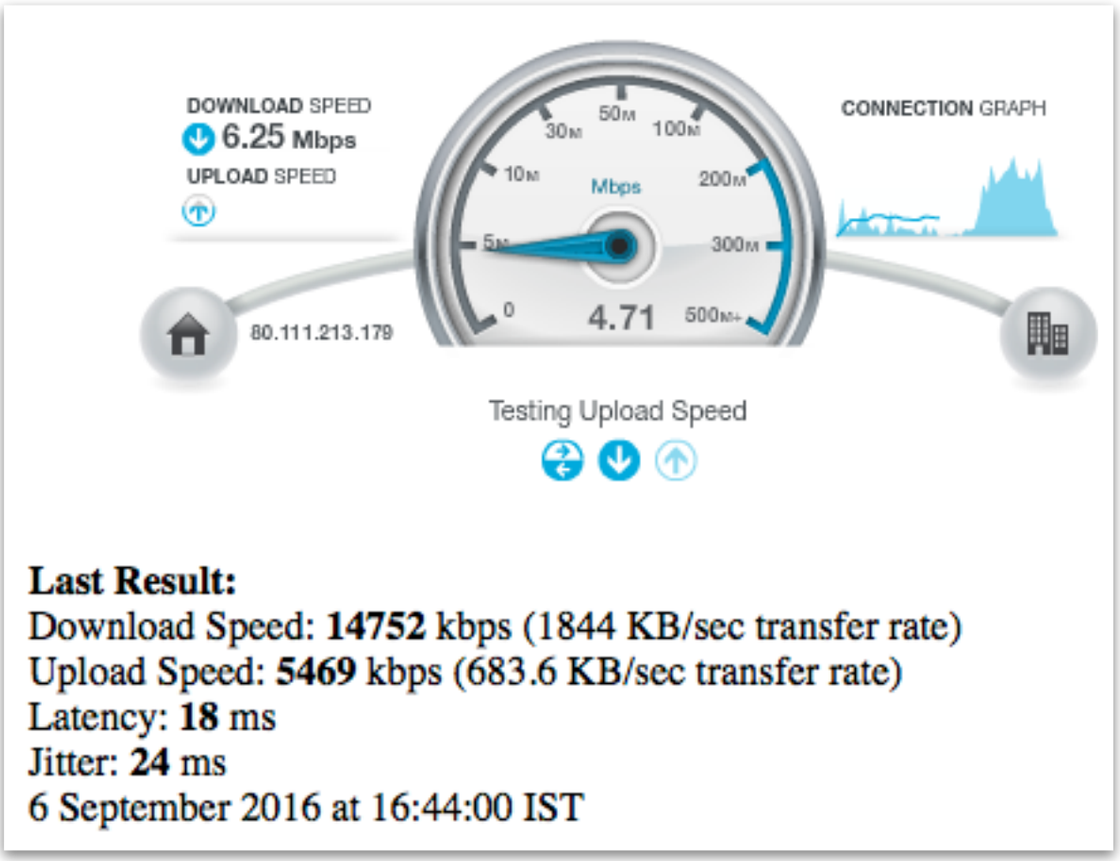
Contention

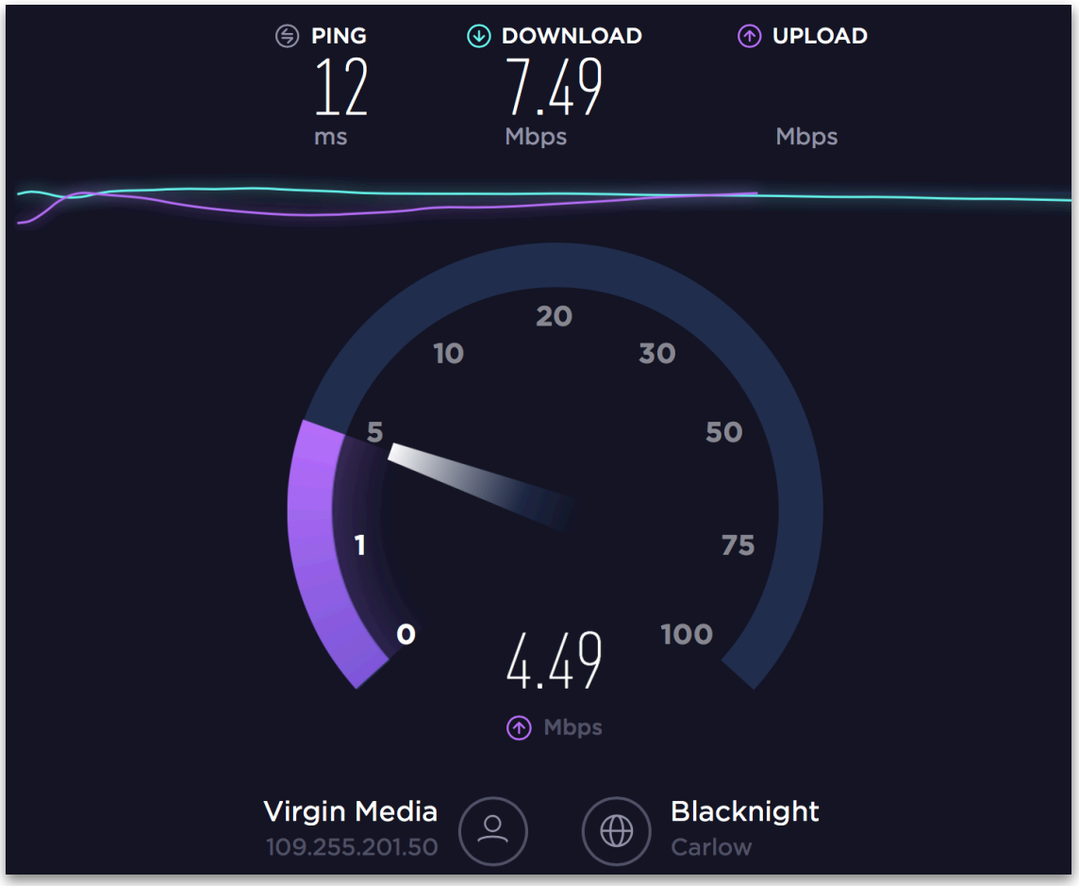
Ping

Traceroute

Performance metrics

Speedtest
http://www.speedtest.net



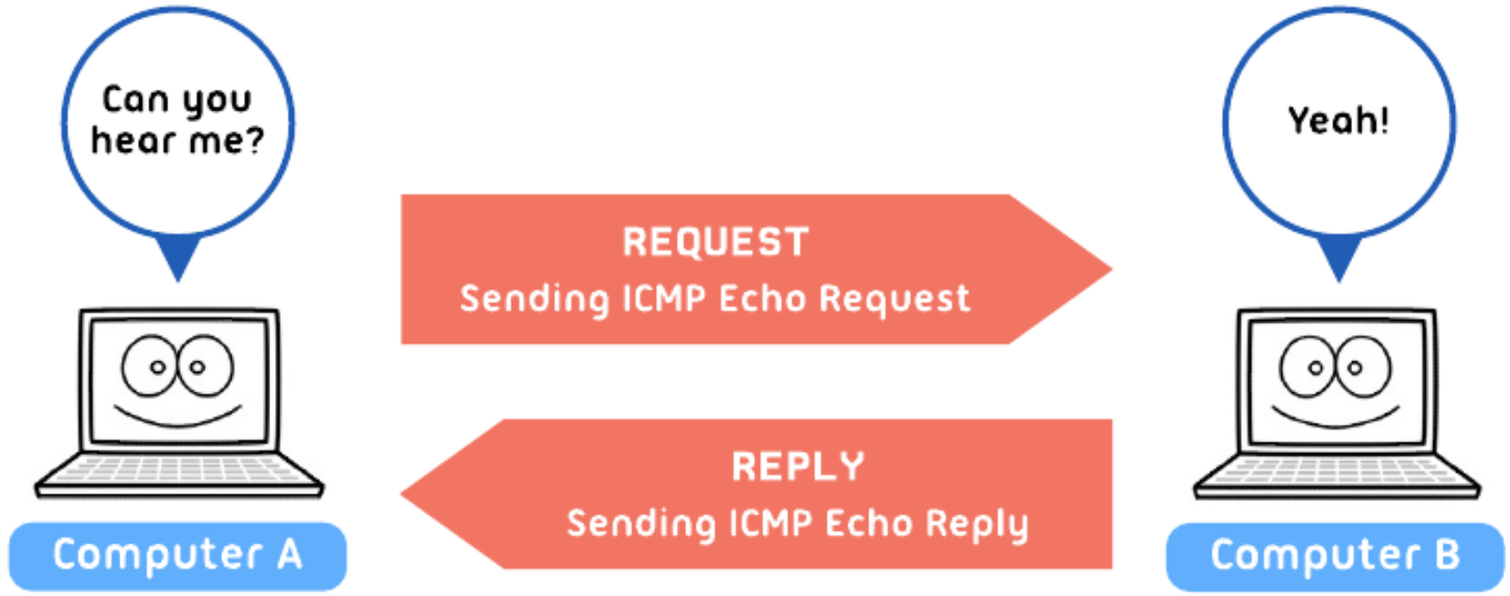


Ping

Test reachability of host

Measure round-trip time

ICMP
(Internet Control Message Protocol)
Echo request

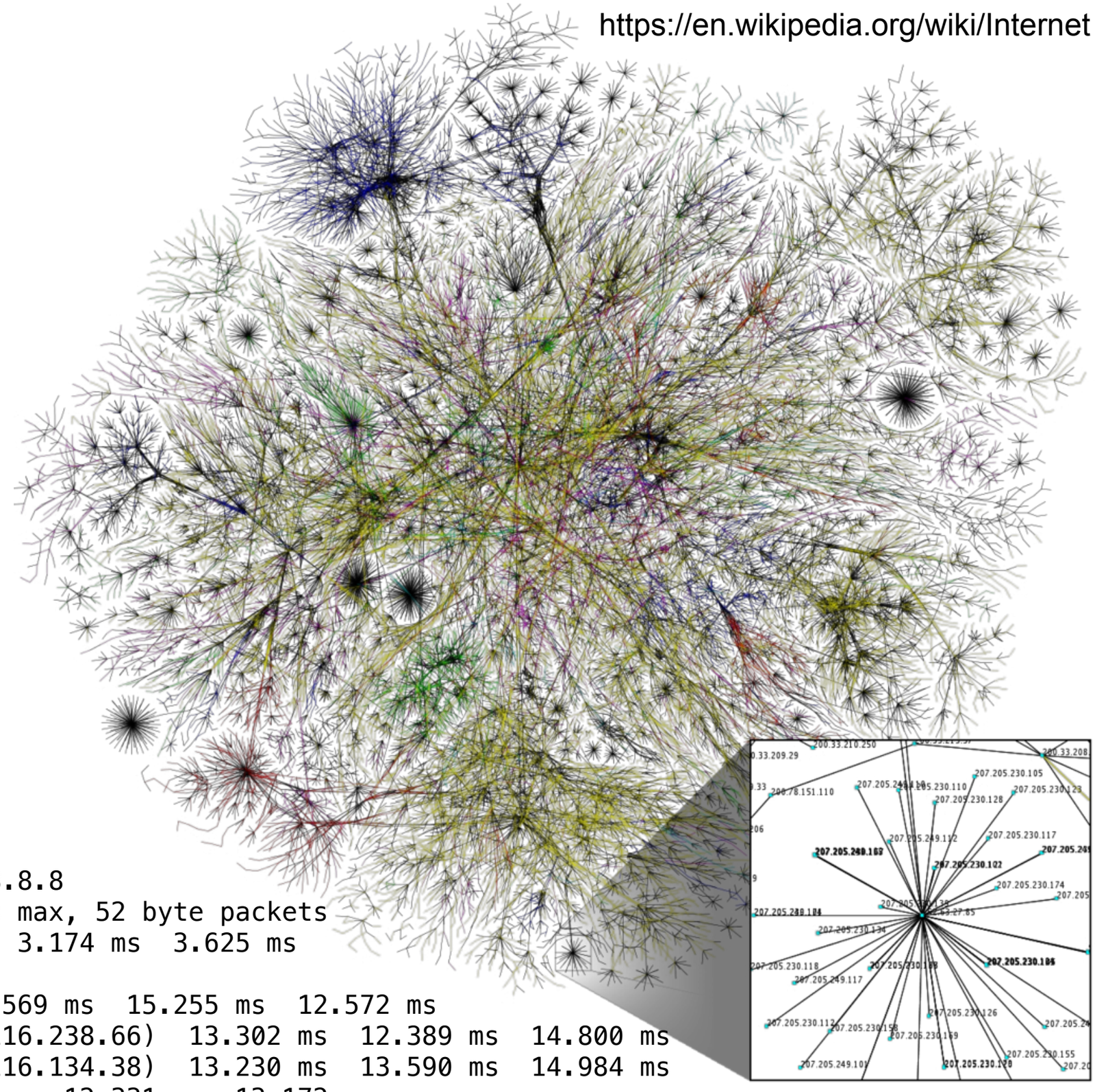



```
Last login: Wed Apr 24 13:48:36 on ttys000
MacBook-Air-2:~ ladymead$ ping -c 5 wikipedia.org
PING wikipedia.org (91.198.174.192): 56 data bytes
64 bytes from 91.198.174.192: icmp_seq=0 ttl=56 time=38.973 ms
64 bytes from 91.198.174.192: icmp_seq=1 ttl=56 time=33.249 ms
64 bytes from 91.198.174.192: icmp_seq=2 ttl=56 time=36.123 ms
64 bytes from 91.198.174.192: icmp_seq=3 ttl=56 time=34.377 ms
64 bytes from 91.198.174.192: icmp_seq=4 ttl=56 time=60.204 ms

--- wikipedia.org ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 33.249/40.585/60.204/9.998 ms
MacBook-Air-2:~ ladymead$
```

Traceroute

https://en.wikipedia.org/wiki/Internet



Diagnostic tool
- display route (path)
- timing delays at each hop

```
MacBook-Air-2:~ ladymead$ traceroute 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 64 hops max, 52 byte packets
 1  192.168.0.1 (192.168.0.1)  9.355 ms  3.174 ms  3.625 ms
 2  * * *
 3  109.255.251.29 (109.255.251.29)  29.569 ms  15.255 ms  12.572 ms
 4  ie-dub02a-rc1-ae35-0.aorta.net (84.116.238.66)  13.302 ms  12.389 ms  14.800 ms
 5  ie-dub02a-ri1-ae74-0.aorta.net (84.116.134.38)  13.230 ms  13.590 ms  14.984 ms
 6  74.125.118.8 (74.125.118.8)  12.953 ms  12.331 ms  13.172 ms
 7  * * *
 8  google-public-dns-a.google.com (8.8.8.8)  21.987 ms  12.506 ms  12.241 ms
```

Some Definitions

Bandwidth

The amount of data that can be transmitted over a given communications channel in a given period of time.

- Based on a more formal analysis where the ***width of a frequency band*** determines the ***speed*** of the connection.
- / www.speedtest.net

Latency

The delay in the transmission of a message from its source to a destination.

- Can be measured ***one-way*** or ***round-trip***.
- ping www.google.com, traceroute www.google.com

Contention

If users ***share*** a communications link performance will degrade with the number of users.

Exercise with ping and traceroute

Use **`ping`** to find the average and max round-trip time to wikipedia.com over 10 attempts

If you encounter firewall problems use data on your phone

`ping` reveals the IP address for wikipedia.com, use ipfind.co to find where it is located

Use **`traceroute`** and ipfind.co to identify the gateways between UCD and wikipedia.com

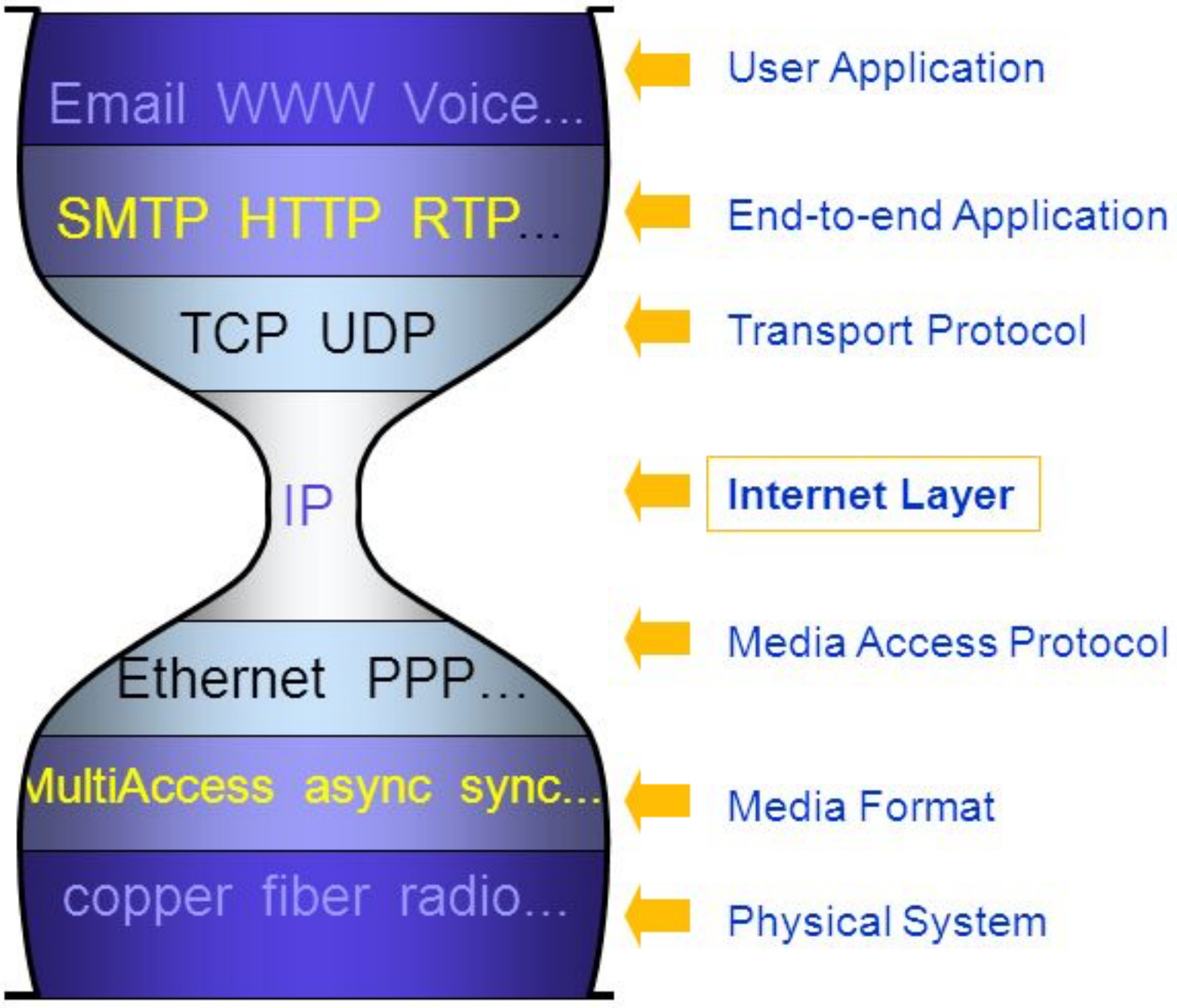
If `traceroute` returns `* * *` a gateway is not responding

Try `traceroute -I`

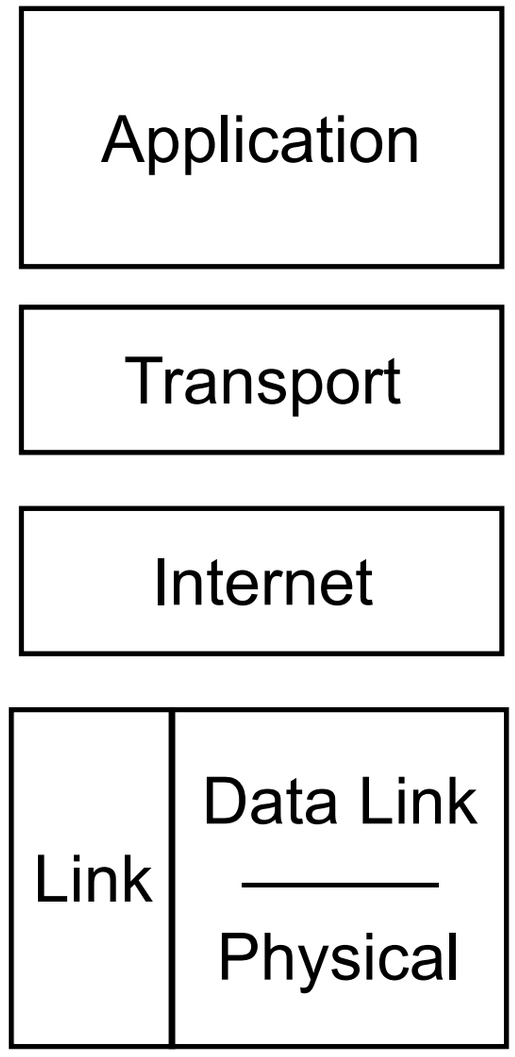
`If traceroute continues to return * * * try`

http://ping.eu/traceroute/

The IP Hourglass



TCP/IP Protocol Layers



TCP/IP Stack

Application Layer:

contains the logic needed to support the various user applications. Separate module are required for each application.

Transport Layer:

manage the connections across networks as information is passed from source to destination (e.g. ***TCP***).

Network/Internet Layer:

IP provides the routing functions across the multiple networks

Data Link layer:

concerned with access to and routing data across a network for two end systems attached to the same network (***MAC*** in Ethernet).

Physical Layer:

covers physical interface between PC or workstation and a transmission medium or network

Application Layer protocols

HTTP (HyperText Transfer Protocol)

This protocol, the core of the WWW, facilitates retrieval and transfer of hypertext (mixed media) documents.

Secure Shell (SSH)

A cryptographic network protocol for operating network services securely over an unsecured network.[1] The best known example application is for remote login to computer systems by users.

SNMP

Used to remotely manage network devices. Stands for the Simple Network Management Protocol.

DNS

Provides meaningful names like wikipedia.com for computers to replace numerical addresses like 123.45.67.89. Stands for the Domain Name System.

IMAP

Internet Message Access Protocol (IMAP) is an Internet standard protocol used by e-mail clients to retrieve e-mail messages from a mail server over a TCP/IP connection

Excellent Wikipedia Pages: https://en.wikipedia.org/wiki/Internet_protocol_suite

HTTP HyperText Transfer Protocol

Application Layer Protocol for the Web

Client Server Protocol

Stateless Protocol

Server maintains no information on past client interaction

- Maintaining state is hard to do properly
- Hard to ensure that client and server have same state after crash

Cookies can give some ***persistence*** to interactions

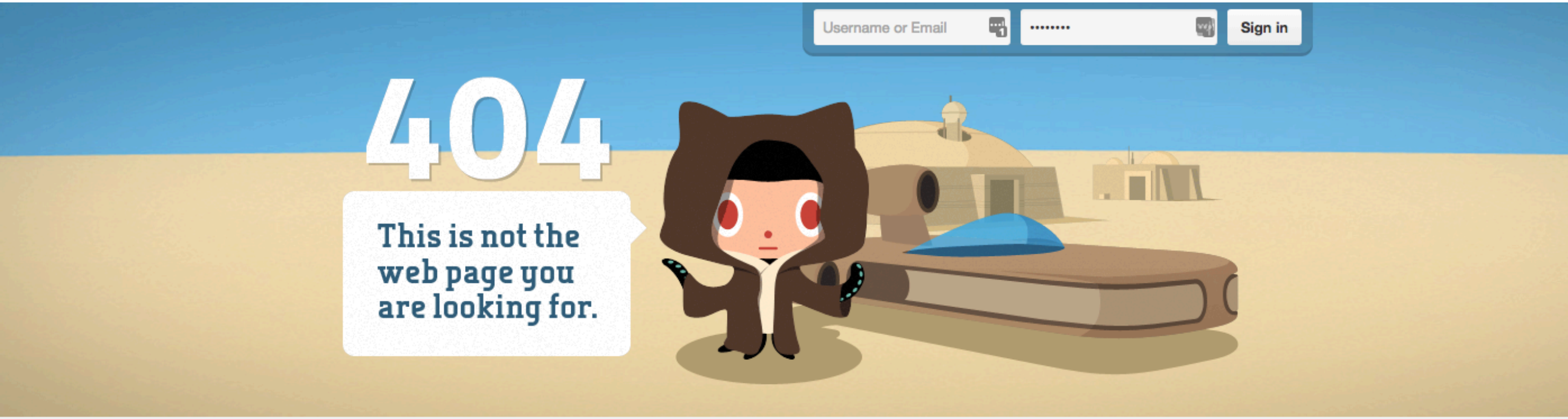
Establish the notion of a ***session***

HTTPS

HTTP + ***encryption*** and ***authentication*** between HTTP and TCP

- typically on port 443

https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol



Find code, projects, and people on GitHub:

Search

Contact Support — GitHub Status — @githubstatus

TCP Example

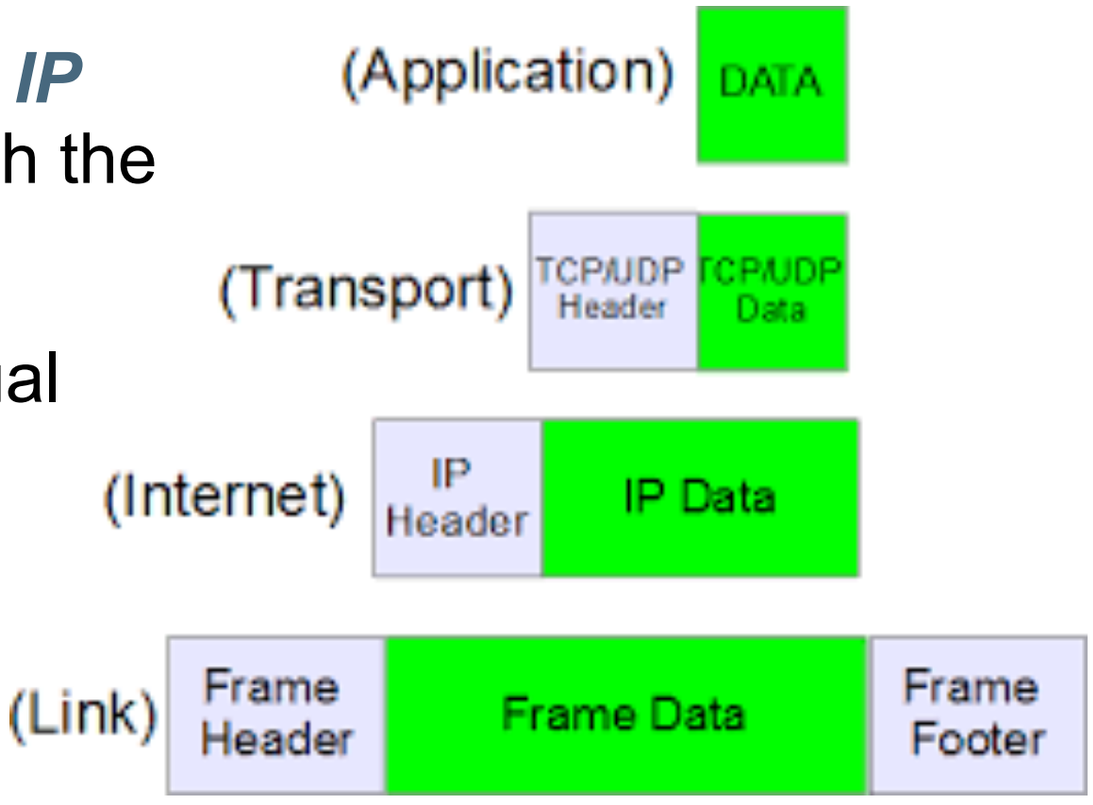
Web Server: serves HTML pages

TCP layer in the server divides the file into one or more packets, numbers the packet, then forward packets individually to ***IP***.

Note: each packet has the same destination ***IP address***, it may get routed differently through the network.

TCP (on the client) reassembles the individual packets and waits until they have arrived to forward them as a single file.

Connection-oriented protocol



By contrast IP is connectionless

HTTP Connection over TCP

Client initiates TCP connection
- socket to server
- typically port 80

Server accepts TCP connection
- HTTP messages exchanged
  - GET, HEAD
  - POST, PUT, DELETE

TCP connection closed

HTTP Status Codes

200 Ok

401 Unauthorized

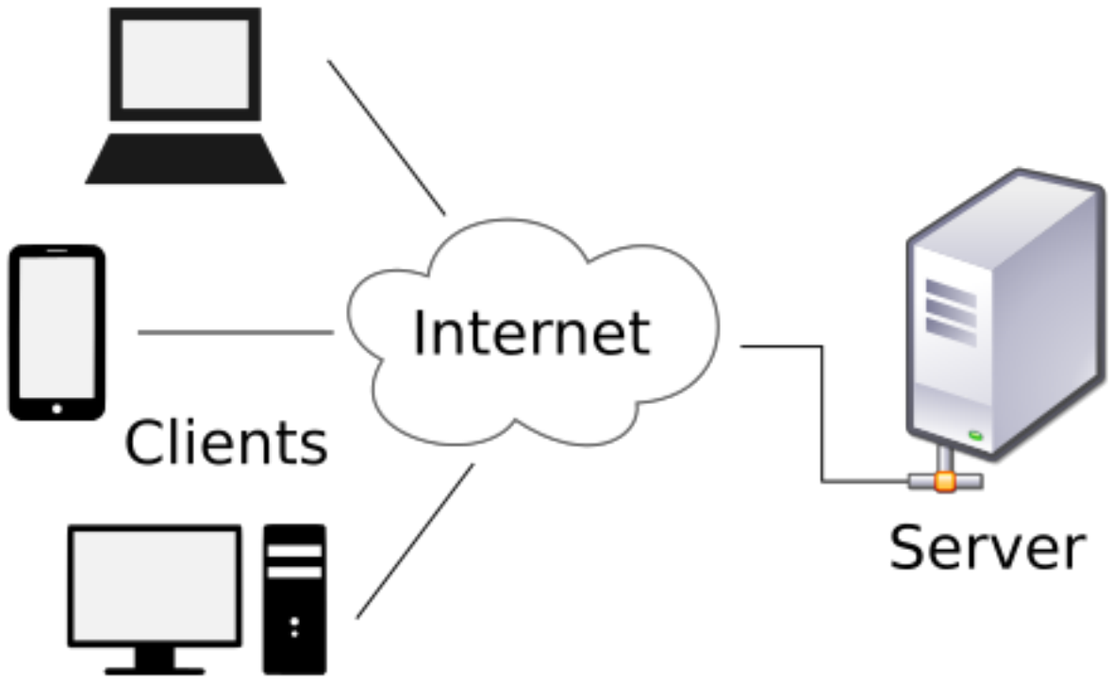
403 Forbidden

404 Not Found

504 Gateway Timeout

Client Server model

A distributed application structure that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients.

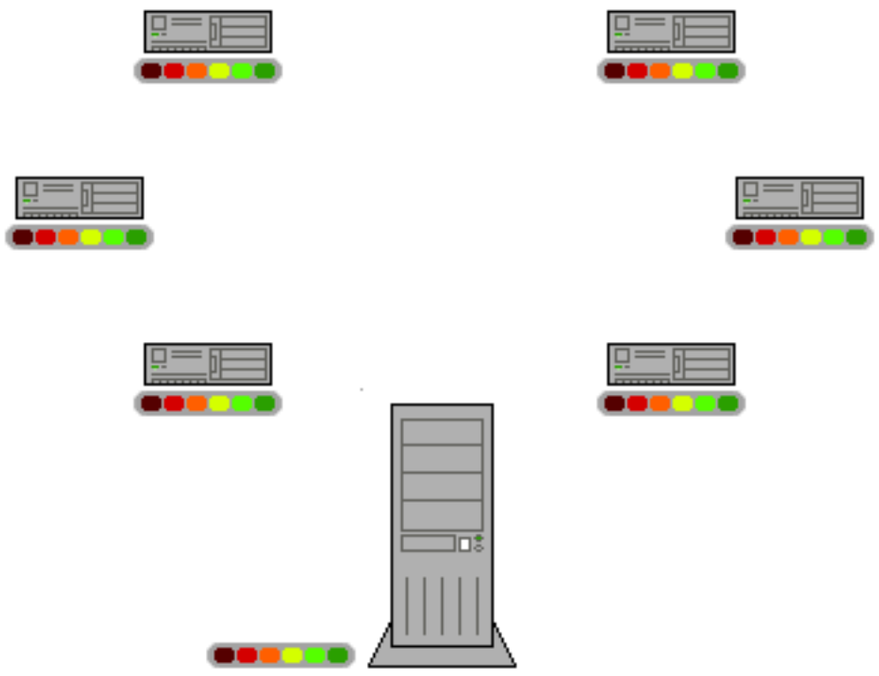


P2P Peer-to-peer

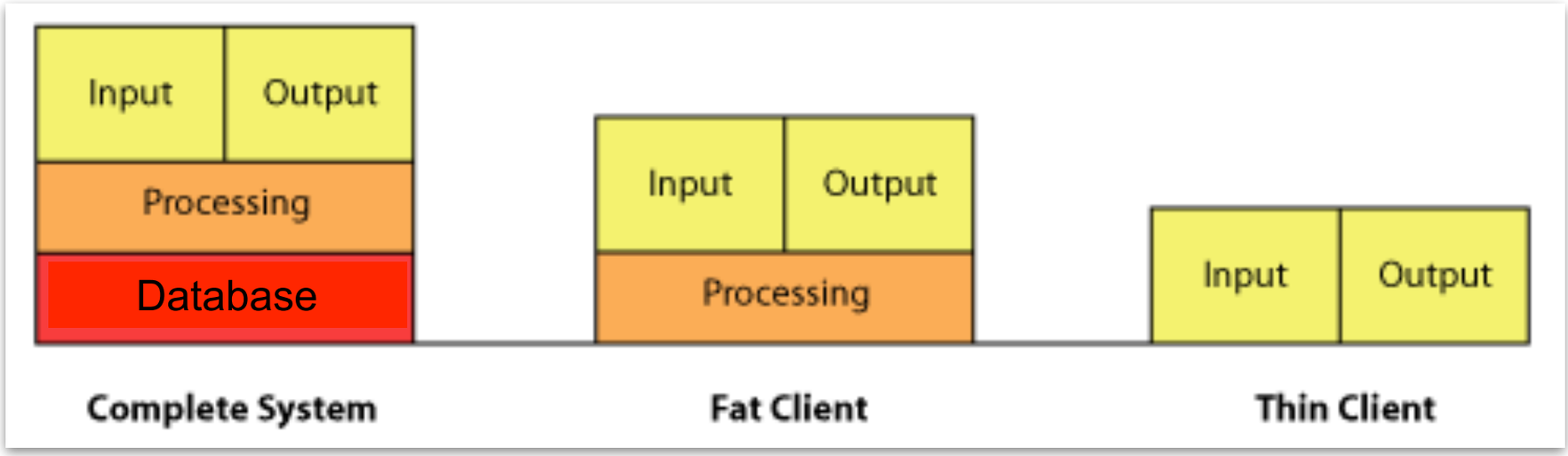
A distributed application architecture that partitions tasks or work loads between peers. Peers are equally privileged, equipotent participants in the application.

BitTorrent

P2P file sharing system



Client Server



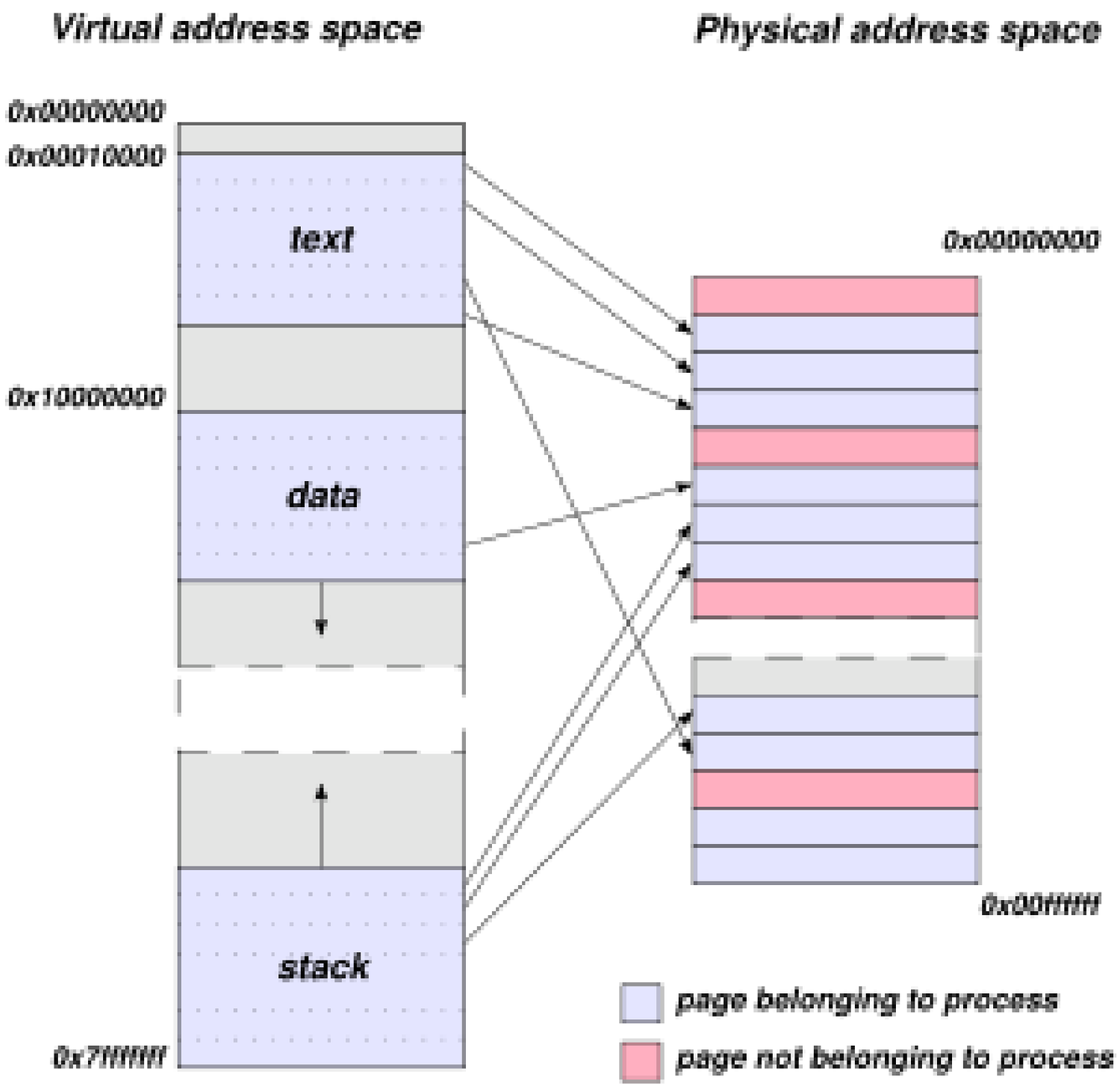
Where does the Processing go?

Come up with examples of:

- Complete System at client side
- Fat Client
- Thin Client

Calling a Local Procedure

Calling a function in its own address space.



Won't work calling a function on a server.

Send the call as a message.

Server Side Technologies

Server Side Technologies are those that run at the server side

The benefit of using Server Side technologies is the control over the execution environment

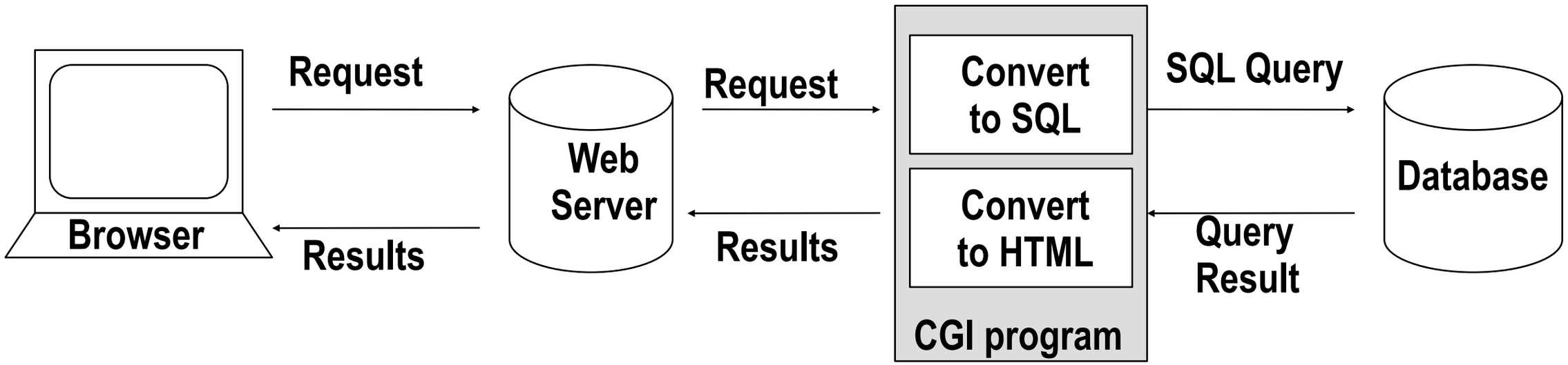
The drawback is too much responsibility on server

Often the server side technology acts as middleware to provide access to databases

CGI - Common Gateway Interface

One of the first practical techniques for creating dynamic content

Can be written in any? language



CGI - Common Gateway Interface

This is a ***protocol standard*** which specifies how information can be passed between a server and an application

CGI itself is not an application - simply a specification

CGI defines how data is passed
- received from browser in name-value pair environment variables
- written out for a browser using specific headers, etc.

SOAP (Simple Object Access Protocol)

Protocol specification for **exchanging structured information** in the implementation of web services in computer networks.

It uses **Extensible Markup Language** (XML) as its message format, and relies on application layer protocols, most often **Hypertext Transfer Protocol** (HTTP) or **Simple Mail Transfer Protocol** (SMTP), for message negotiation and transmission.

Allows processes running on **disparate operating systems** (such as Windows and Linux) to communicate.

Since Web protocols like HTTP are installed and running on all Operating systems, **SOAP allows clients to invoke web services and receive responses independent of language and platforms**.

RESTful API + JSON

example: JSON API
https://jsonapi.org/examples/
Representational state transfer (REST) or RESTful web services interoperability between computer systems on the Internet.

Term coined by Roy Fielding in his doctoral dissertation in 2000

Typically uses HTTP verbs: Get, Put, Post, Delete

6 principles

- Client-Server
- Stateless
- Cacheable
- Layered system
- Code on demand (optional)
- Uniform interface

Typically returns data in JSON format

https://www.servage.net/blog/2013/04/08/rest-principles-explained/

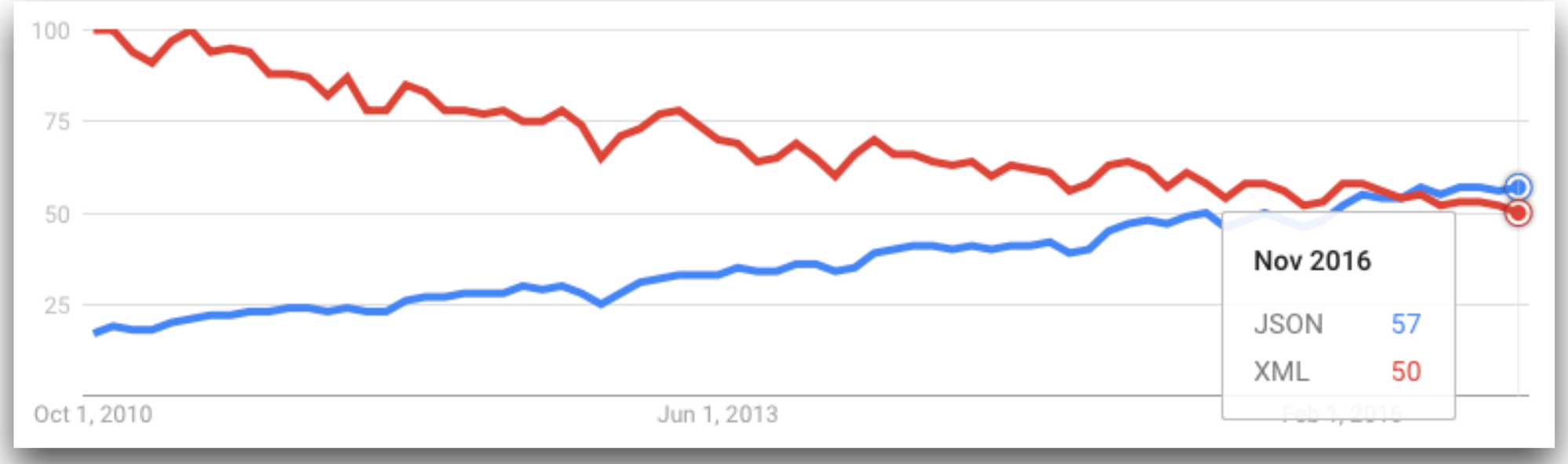
JSON JavaScript Object Notation

Standard for transmitting data objects

- Human readable
- Attribute pairs
- Syntax
  - {} delimits object
  - [] delimits array
  - , separates attributes
  - : separates key & value

Less verbose than XML

```
{
   "Image": {
       "Width":  800,
       "Height": 600,
       "Title":  "View from 15th Floor”
       "Thumbnail": {
           "Url":    "http://www.example.com/image/48",
           "Height": 125
           "Width":  100
       },
       "Animated" : false,
       "IDs": [116, 943, 234, 38793]
     }
}
```



If you pronounce it *jay-sawn* - people **will** judge you.

Computer Networking

Objective for this week:
- High-level understanding of TCP/IP Networking
- Appreciate why IP is so important

The notion of a Stack
- TCP/IP Protocol Stack

Application Layer Protocols
- SMTP, HTTP, IMAP, Telnet,FTP

TCP/IP layered architecture

Calling a local procedure

Remote Procedure Calls
- CGI
- SOAP
  - XML
- REST
  - JSON

Local Procedure Calls

RPC: Send a message