



TECHNION

Azrieli Continuing Education and
External Studies Division

Module 5.7.3

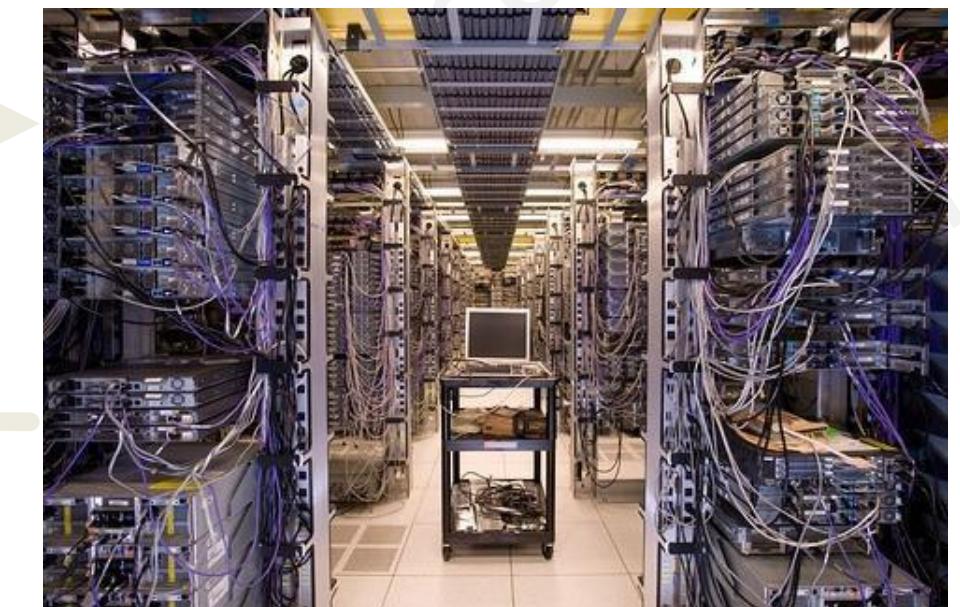
HTTP Connections

ALL RIGHTS RESERVED © COPYRIGHT 2022
DO NOT DISTRIBUTE WITHOUT WRITTEN PERMISSION

Client



Server



Internet

ALL RIGHTS RESERVED © COPYRIGHT 2022 | DO NOT DISTRIBUTE WITHOUT
WRITTEN PERMISSION



HTML

JavaScript

AJAX

CSS



HTTP

Response

socket

Request

GET

POST



Python

Templates



Data Store

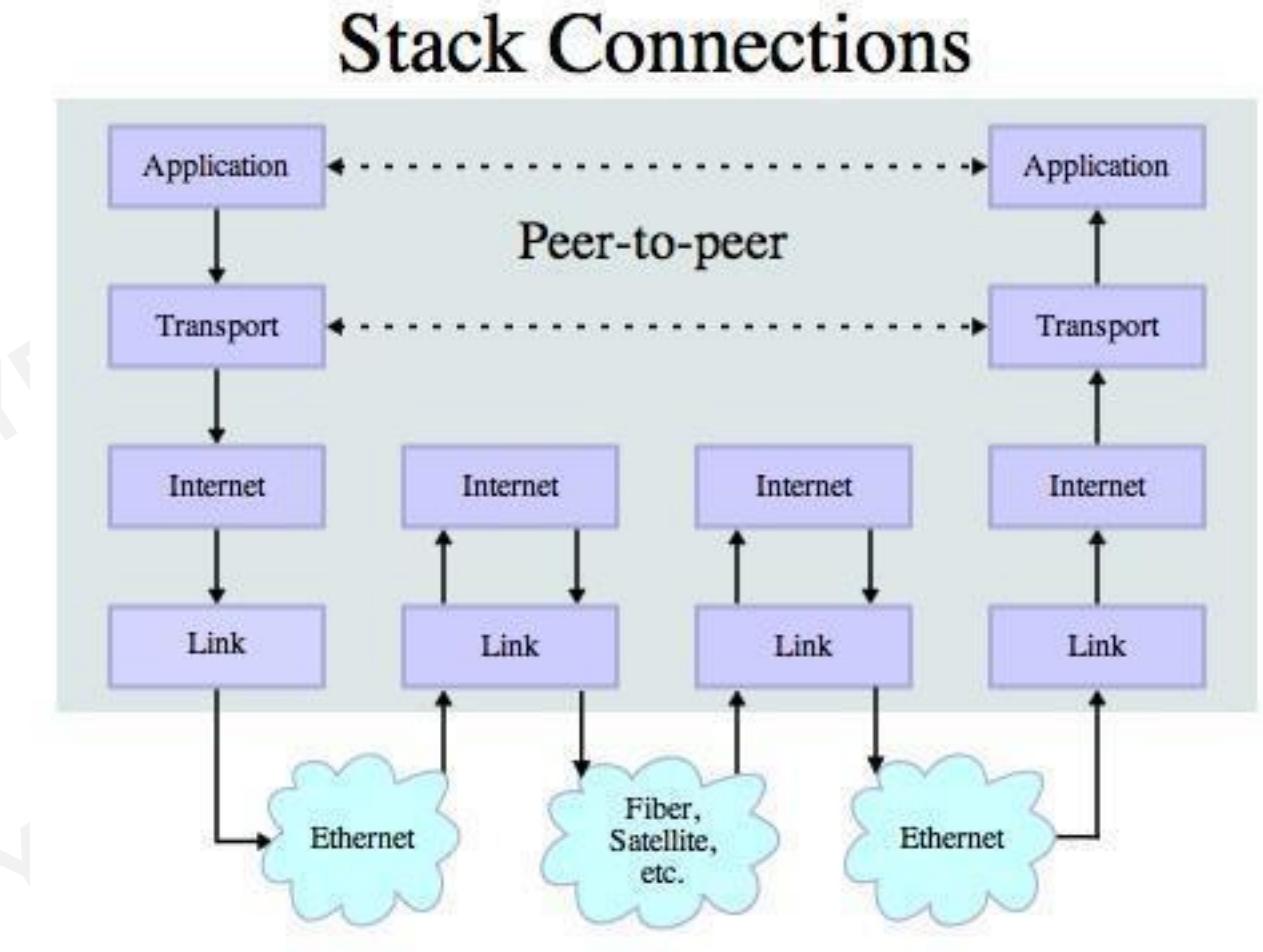
memcache

Network Architecture....

ALL RIGHTS RESERVED © COPYRIGHT 2022 | DO NOT DISTRIBUTE WITHOUT
WRITTEN PERMISSION

Transport Control Protocol (TCP)

- Built on top of IP (Internet Protocol).
- Assumes IP might lose some data - stores and retransmits data if it seems to be lost.
- Handles “flow control” using a transmit window.
- Provides a nice reliable pipe.



Sockets

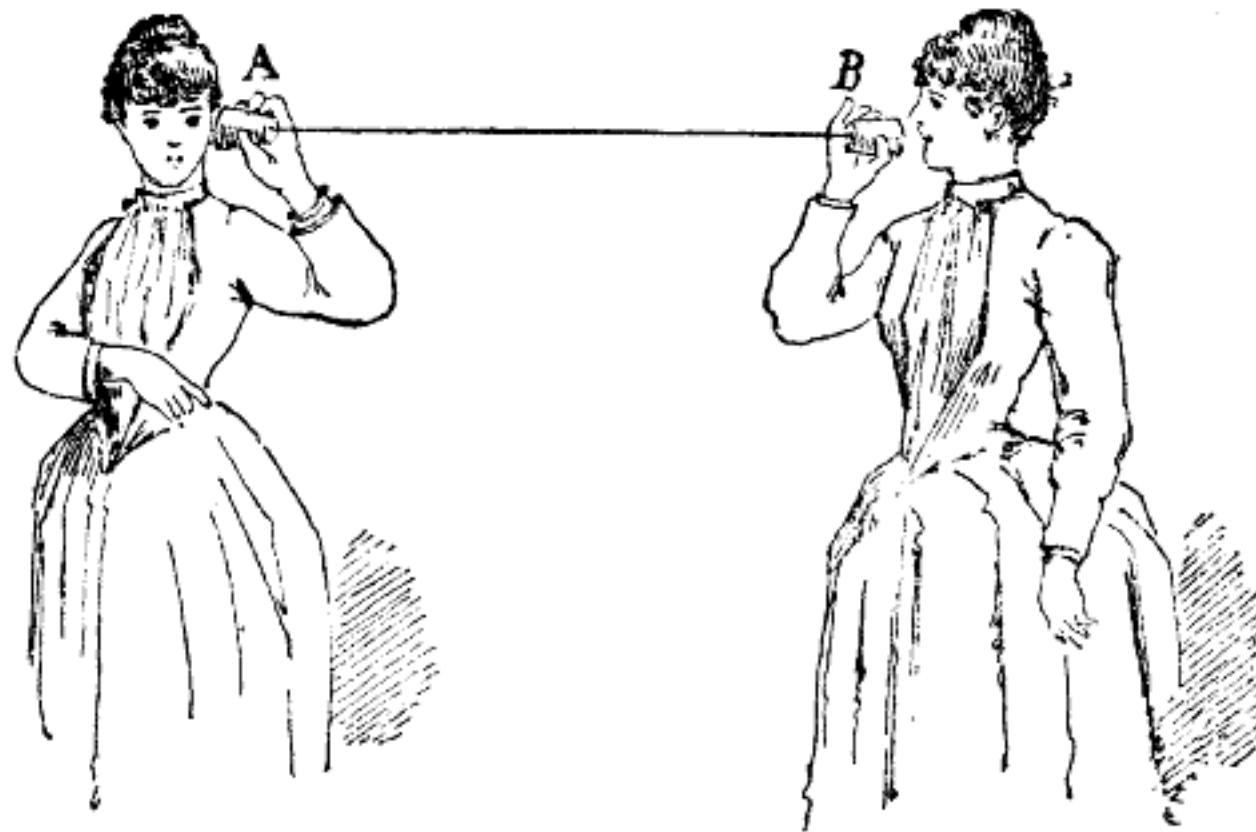
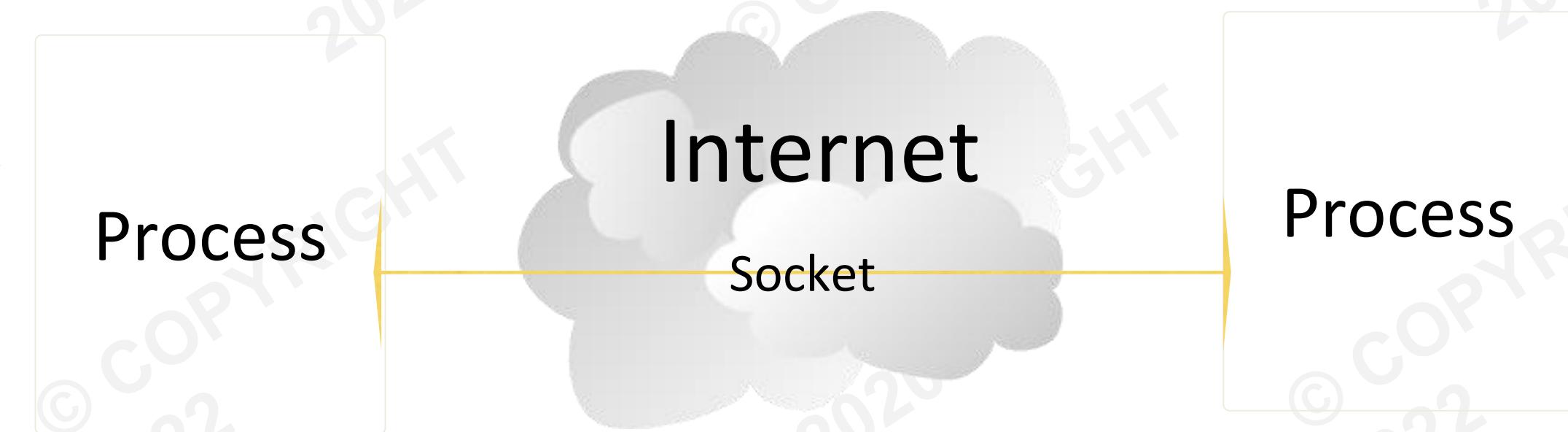


FIG. 76. Trådtelefon.



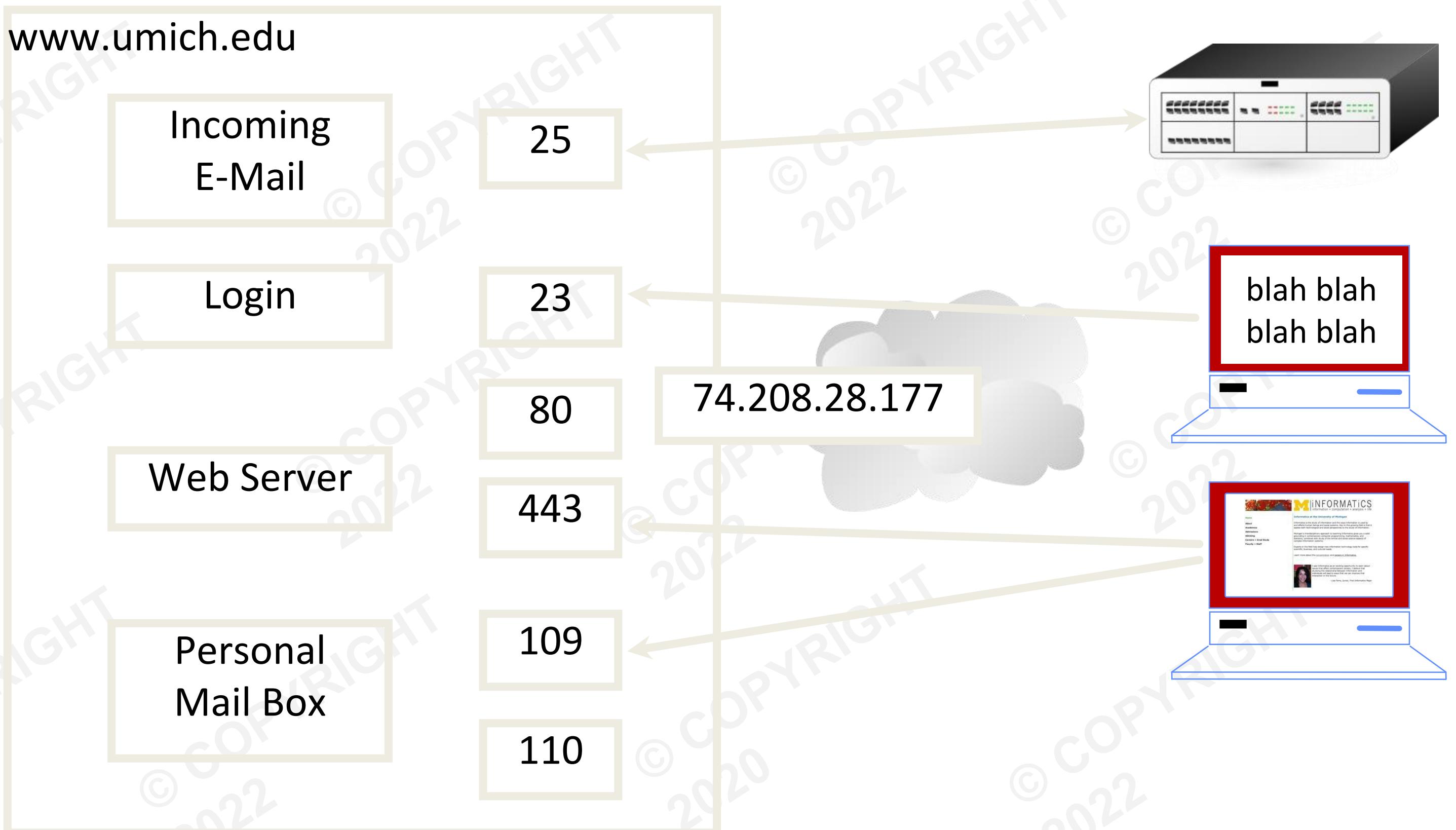
TCP Connections/ Sockets

“In computer networking, an internet socket or network socket is an endpoint of a bidirectional inter-process communication flow across an Internet Protocol-based computer network, such as the Internet.”



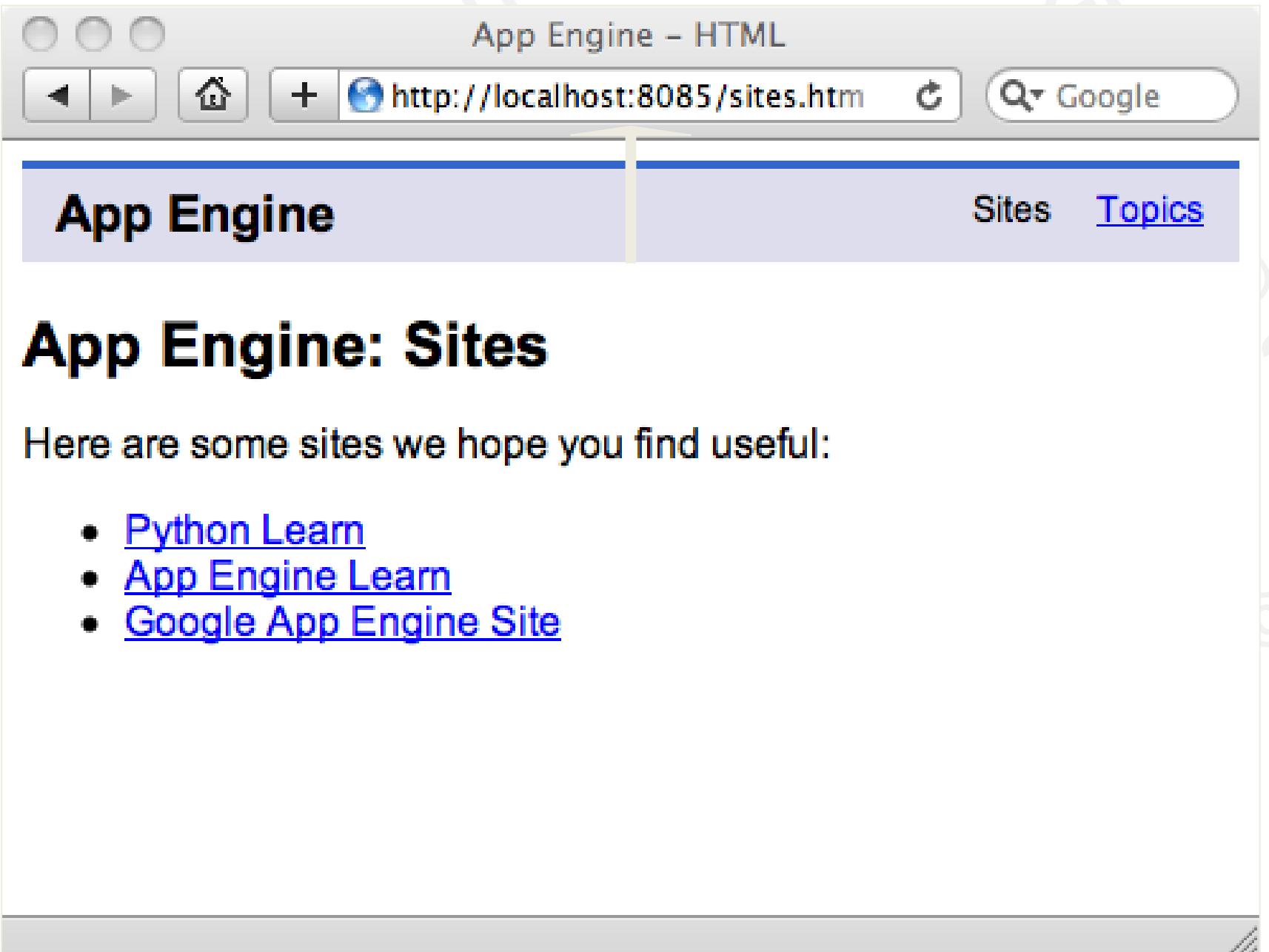
TCP Port Numbers

- A port is an application-specific or process-specific software communications endpoint.
- It allows multiple networked applications to coexist on the same server.
- There is a list of well-known TCP port numbers.



Common TCP Ports

- Telnet (23) - Login
- SSH (22) - Secure Login
- HTTP (80)
- HTTPS (443) - Secure
- SMTP (25) (Mail)
- IMAP (143/220/993) - Mail Retrieval
- POP (109/110) - Mail Retrieval
- DNS (53) - Domain Name
- FTP (21) - File Transfer



Sometimes we see the port number in the URL if the web server is running on a “non-standard” port.

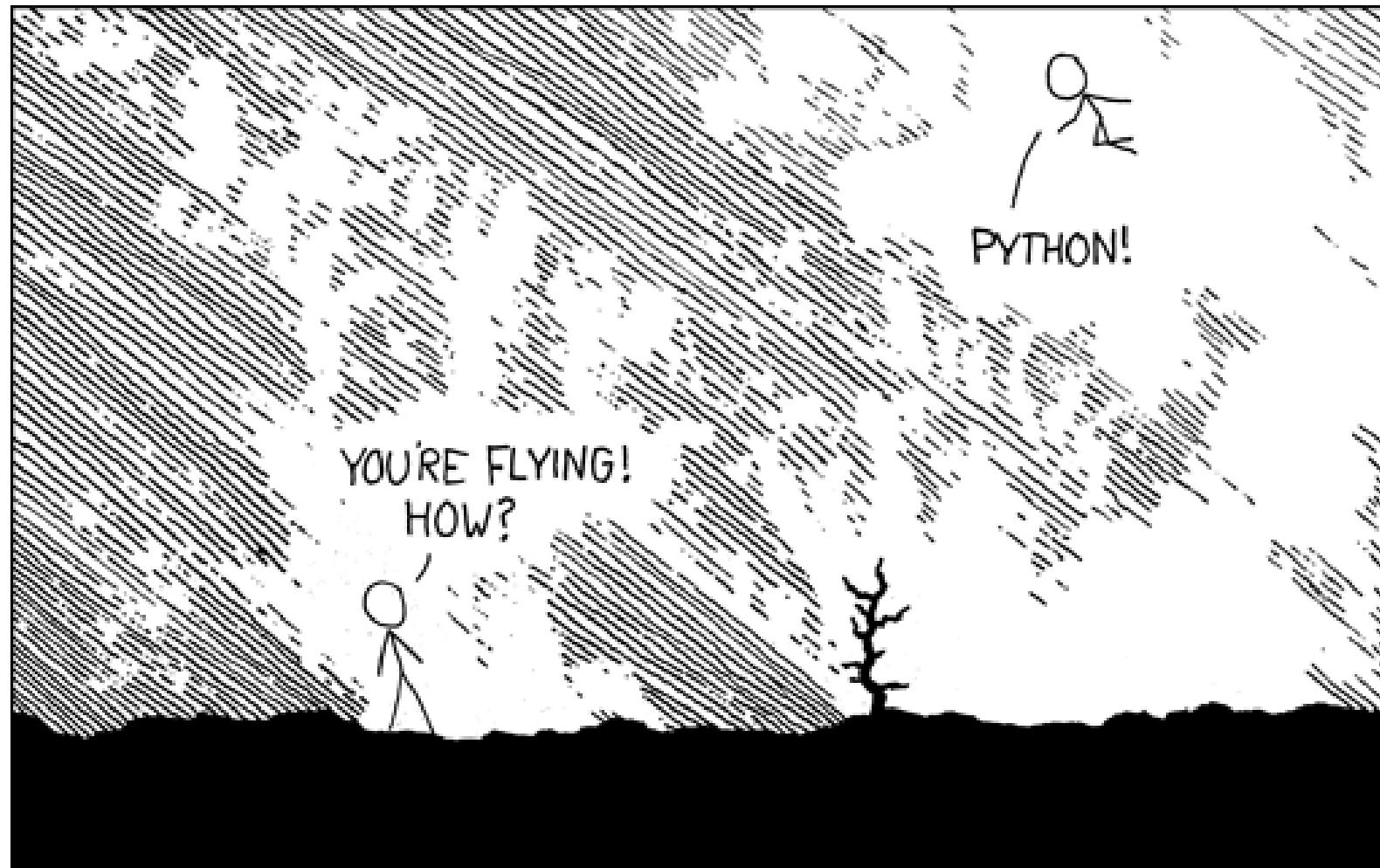
Sockets in Python

Python has built-in support for TCP Sockets.

```
import socket  
mysock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)  
mysock.connect( 'www.py4inf.com', 80 )
```

Host

Port



I LEARNED IT LAST NIGHT! EVERYTHING IS SO SIMPLE!
/ HELLO WORLD IS JUST
print "Hello, world!"

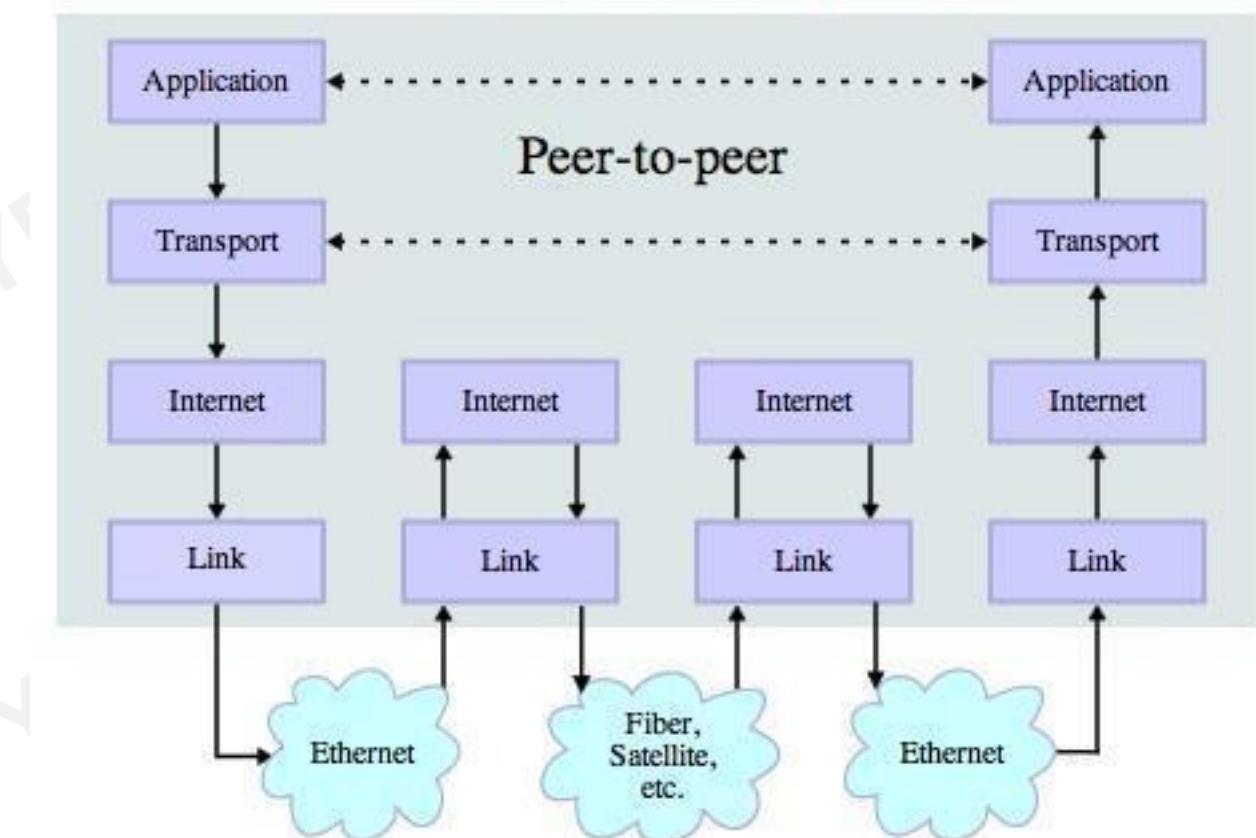
I DUNNO...
DYNAMIC TYPING?
WHITE SPACE?
COME JOIN US!
PROGRAMMING IS FUN AGAIN!
IT'S A WHOLE NEW WORLD UP HERE!
BUT HOW ARE YOU FLYING?

I JUST TYPED
import antigravity
THAT'S IT?
/ ... I ALSO SAMPLED
EVERYTHING IN THE
MEDICINE CABINET
FOR COMPARISON.
/ BUT I THINK THIS
IS THE PYTHON.

Application Protocol

- Since TCP (and Python) gives us a reliable socket, what do we want to do with the socket? What problem do we want to solve?
- Application Protocols
 - > Mail
 - > World Wide Web

Stack Connections



HTTP - Hypertext Transport Protocol

- The dominant Application Layer Protocol on the Internet.
- Invented for the Web - to retrieve HTML, images, documents, etc.
- Extended to be data in addition to documents - RSS, Web Services, etc..
- Basic Concept - Make a Connection - Request a document - Retrieve the Document - Close the Connection.

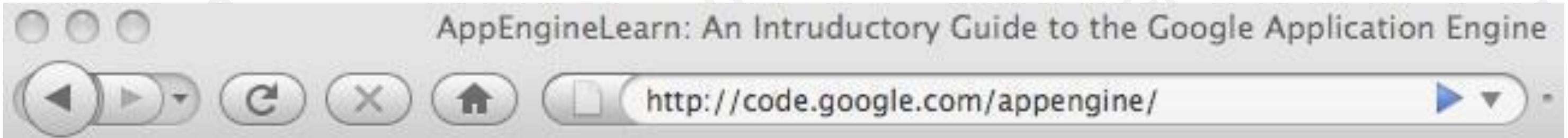
HTTP

The HyperText Transport Protocol is the set of rules designed to enable browsers to retrieve web documents from servers over the internet.

What Is A Protocol?

- A set of rules that all parties must follow, so that we may predict each other's behavior...
...and not bump into each other.
- On two-way roads in USA, drive on the right-hand side of the road
- On two-way roads in the UK, drive on the left-hand side of the road





`http://www.dr-chuck.com/page1.htm`

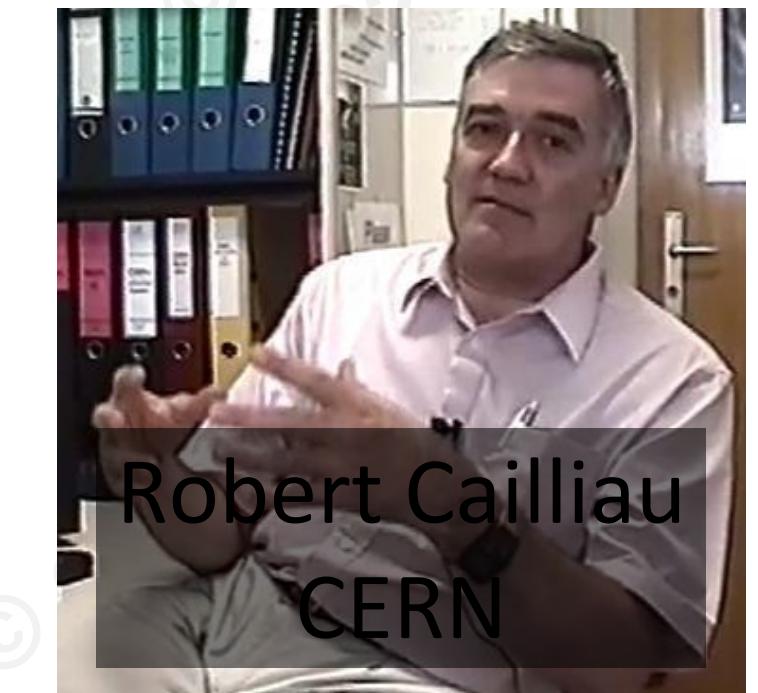
protocol

host

document

<http://www.youtube.com/watch?v=x2GylLq59rl>

1:17 - 2:19



Getting Data From The Server

- Each time the user clicks on an anchor tag with an “href=” value to switch to a new page, the browser makes a connection to the web server and issues a “GET” request - to GET the content of the page at the specified URL.
- The server returns the HTML document to the browser, which formats and displays the document to the user.

Making An HTTP Request

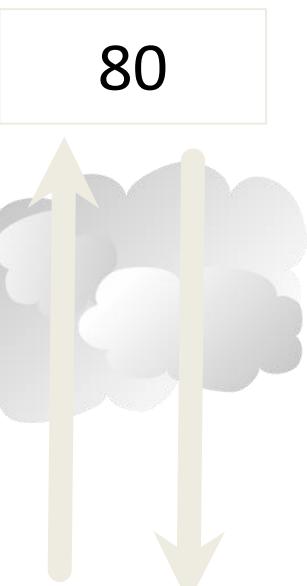
- Connecting to the web server. Example: <http://www.dr-chuck.com/>
- Request a document (or the default document)

GET <http://www.dr-chuck.com/page1.htm>

GET <http://www.mlive.com/ann-arbor/>

GET <http://www.facebook.com>

Web Server



Browser



The First Page

If you like, you can switch to the [Second Page](#).

<h1>The Second Page</h1>
<p>If you like, you can switch
back to the First
Page.</p>



The Second Page

If you like, you can switch back to the [First Page](#).

Go to "http://www.dr-chuck.com/page2.htm"

Internet Standards

- The standards for all of the Internet protocols (inner workings) are developed by one organization.
- Internet Engineering Task Force (IETF).
- www.ietf.org
- Standards are called Request For Comments (RFC's).

INTERNET PROTOCOL

DARPA INTERNET PROGRAM

PROTOCOL SPECIFICATION

September 1981

The internet protocol treats each internet datagram as an independent entity unrelated to any other internet datagram. There are no connections or logical circuits (virtual or otherwise).

The internet protocol uses four key mechanisms in providing its service: Type of Service, Time to Live, Options, and Header Checksum.

<http://www.w3.org/Protocols/rfc2616/rfc2616.txt>

Network Working Group
Request for Comments: 2616
Obsoletes: 2068
Category: Standards Track

R. Fielding
UC Irvine
J. Gettys
Compaq/W3C
J. Mogul
Compaq
H. Frystyk
W3C/MIT
L. Masinter
Xerox
P. Leach
Microsoft
T. Berners-Lee
W3C/MIT
June 1999

Hypertext Transfer Protocol -- HTTP/1.1

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (1999). All Rights Reserved.

Abstract

The Hypertext Transfer Protocol (HTTP) is an application-level protocol for distributed, collaborative, hypermedia information

5 Request

A request message from a client to a server includes, within the first line of that message, the method to be applied to the resource, the identifier of the resource, and the protocol version in use.

```
Request      = Request-Line ; Section 5.1  
           *(( general-header ; Section 4.5  
             | request-header ; Section 5.3  
             | entity-header ) CRLF) ; Section 7.1  
           CRLF  
           [ message-body ] ; Section 4.3
```

5.1 Request-Line

The Request-Line begins with a method token, followed by the Request-URI and the protocol version, and ending with CRLF. The elements are separated by SP characters. No CR or LF is allowed except in the final CRLF sequence.

```
Request-Line = Method SP Request-URI SP HTTP-Version CRLF
```

“Hacking” HTTP

```
$ telnet www.dr-chuck.com 80
```

```
Trying 74.208.28.177...
```

```
Connected to www.dr-chuck.com.
```

```
Escape character is '^]'.  
GET http://www.dr-chuck.com/page1.htm
```

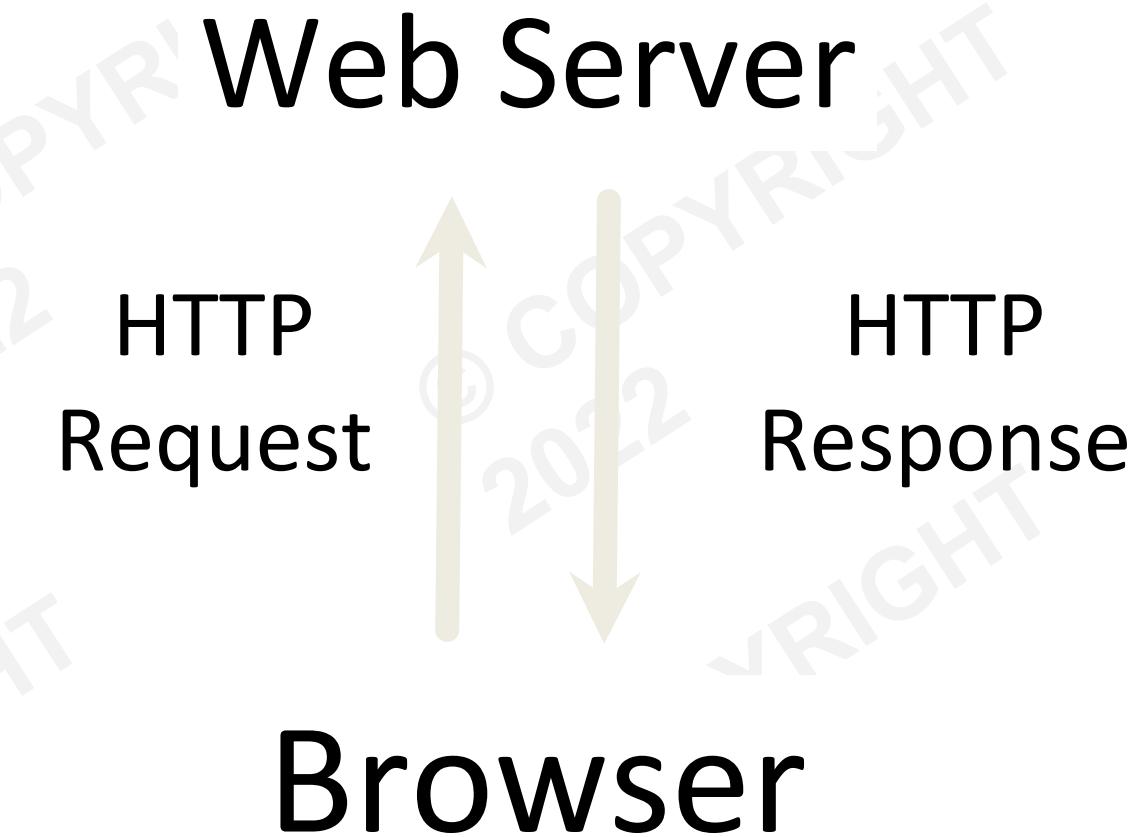
```
<h1>The First Page</h1>
```

```
<p>If you like, you can switch to the
```

```
<a href="http://www.dr-chuck.com/page2.htm">Second Page</a>.
```

```
</p>
```

Port 80 is the non-encrypted HTTP port



Accurate Hacking in the Movies

- Matrix Reloaded
- Bourne Ultimatum
- Die Hard 4
- ...
...



```
80/tcp open http  
81/tcp open  
10/tcp open [mobile]  
# nmap -v -sS -O 10.2.2.2  
Starting nmap 0.2.54BETA25  
Insufficient responses for TCP sequencing (3), OS detection  
accurate  
Interesting ports on 10.2.2.2:  
(The 1539 ports scanned but not shown below are in state: closed)  
Port      State    Service  
22/tcp    open     ssh  
No exact OS matches for host  
Nmap run completed -- 1 IP address (1 host up) scanned  
# sshnuke 10.2.2.2 -rootpw="Z10H0101"  
Connecting to 10.2.2.2:ssh ... successful.  
Re-Attempting to exploit SSHv1 CRC32 ... successful.  
IP Resetting root password to "Z10H0101".  
System open: Access Level <9>  
# ssh 10.2.2.2 -l root  
root@10.2.2.2's password: [REDACTED]  
[REDACTED] CONTROL  
ACCESS GRANTED
```

A screenshot of a terminal window showing a network scan and exploit success. The terminal output includes a port scan for host 10.2.2.2, the use of nmap and sshnuke tools, and a successful exploit of an SSHv1 vulnerability. A separate window titled 'CONTROL' shows the message 'ACCESS GRANTED'.

Telnet Connection

```
$ telnet www.dr-chuck.com 80
Trying 74.208.28.177...
Connected to www.dr-chuck.com. Escape character is '^].
GET http://www.dr-chuck.com/page1.htm
```

```
<h1>The First Page</h1>
<p>If you like, you can switch to the
<a href="http://www.dr-chuck.com/page2.htm">Second
Page</a>.</p>
Connection closed by foreign host.
```



[PORTAL EN ESPAÑOL](#)

[HOME](#)

[PROSPECTIVE STUDENTS](#)

[CURRENT STUDENTS](#)

[FACULTY & STAFF](#)

[ALUMNI, DONORS, & PARENTS](#)



[About U-M](#)

[Academics & Research](#)

[Administration](#)

[Athletics & Recreation](#)

[Employment](#)

[Giving to U-M](#)

[Global Michigan](#)

[Health & Medical Resources](#)

[Libraries & Archives](#)

[Museums & Cultural Attractions](#)

[News & Events](#)

[Schools & Colleges](#)

[State & Community Partnerships](#)

web directory

Search

IN THE NEWS::



[Scientists harness the power of electricity in the brain](#)



[Friends with cognitive benefits: Mental function improves after socializing](#)

- ➡ [Scary chupacabras monster is as much victim as villain](#)
- ➡ [Video: Fashion, power and politics; Washington Post writer at U-M](#)

FEATURED SITES



THE
THOMAS
FRANCIS, JR.
MEDAL IN
GLOBAL
PUBLIC
HEALTH

 [U-M SPEAKS OUT](#)



[Exposing voter system flaws](#)

[Give online](#) 

si-csev-mbp:tex csev\$ telnet www.umich.edu 80

Trying 141.211.144.190...

Connected to www.umich.edu.Escape character is '^]'.

GET /

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"><html
xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
lang="en"><head><title>University of Michigan</title><meta
name="description" content="University of Michigan is one of the top
universities of the world, a diverse public institution of higher learning,
fostering excellence in research. U-M provides outstanding undergraduate,
graduate and professional education, serving the local, regional, national and
international communities." />
```

...

```
<link rel="alternate stylesheet" type="text/css" href="/CSS/accessible.css"
media="screen" title="accessible" /><link rel="stylesheet"
href="/CSS/print.css" media="print,projection" /><link rel="stylesheet"
href="/CSS/other.css" media="handheld,tty,tv,braille,embossed,speech,aural"
/>... <dl><dt><a
href="http://ns.umich.edu/htdocs/releases/story.php?id=8077">
</a><span class="verbose">:</span></dt><dd><a
href="http://ns.umich.edu/htdocs/releases/story.php?id=8077">Scientists
harness the power of electricity in the brain</a></dd></dl>
```



As the browser reads the document, it finds other URLs
that must be retrieved to produce the document.

The big picture...

A screenshot of the University of Michigan's website homepage. The header includes links for text-only, mobile, español, accessibility, disability resources, and contact us. It features a yellow 'M' logo and navigation tabs for HOME, PROSPECTIVE STUDENTS, CURRENT STUDENTS, FACULTY & STAFF, and ALUMNI, DONORS, & PARENTS. A 'PORTAL EN ESPAÑOL' button is also present. The main content area has a blue banner with a photo of a building and a search bar. To the left is a sidebar with links to About U-M, Academics & Research, Administration, Athletics & Recreation, Employment, Giving to U-M, Global Michigan, Health & Medical Resources, Libraries & Archives, Museums & Cultural Attractions, News & Events, Schools & Colleges, and State & Community Partnerships. A 'RECORD UPDATE' section is at the bottom.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">  
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">  
<head>  
<title>University of Michigan</title>
```

....

```
@import "/CSS/graphical.css"/**/;  
p.text strong, .verbose, .verbose p, .verbose h2{text-indent:-  
876em;position:absolute}  
p.text strong a{text-decoration:none}  
p.text em{font-weight:bold;font-style:normal}  
div.alert{background:#eee;border:1px solid  
red;padding:.5em;margin:0 25%}  
a img{border:none}  
.hot br, .quick br, dl.feature2 img{display:none}  
div#main label, legend{font-weight:bold}
```



Firebug reveals the details...

- If you haven't already installed the Firebug FireFox extension, then you need to now.
- It can help explore the HTTP request-response cycle.
- Some simple-looking pages involve lots of requests:
 - HTML page(s)
 - Image files
 - CSS Style Sheets
 - JavaScript files

AppEngineLearn: An Introductory Guide to the Google Application Engine

http://www.appspotenginelearn.com/ Google

Disable Cookies CSS Forms Images Information Miscellaneous Outline Resize Tools View Source

AppEngineLearn

This site provides materials to help learn the Google Application Engine. Before you start to learn the Google Application Engine you should be basically familiar with the Python programming language.

New: You can take a look at the [draft book chapters](#) for my upcoming O'Reilly AppEngine book titled, "Building Cloud Applications with Google AppEngine".

- [Installing Python and JEdit](#) - We recommend using JEdit as your programmer editor and it will be used throughout the Podcasts.
- Installing the Application Engine and writing your first Application.
 - Macintosh: ([Handout](#), [Source Code](#), [Screencast](#), [YouTube](#))
 - Windows Vista: ([Handout](#), [Source Code](#), [High Quality Screencast](#), [YouTube](#))



Inspect Clear Profile

Console ▾ HTML CSS Script DOM Net Options

Console panel is disabled

Use this page to enable or disable following panels. Enabling these panels will reduce performance and will cause a page reload.

<input type="checkbox"/> Console	Support for Console logging.	Disabled Always
<input checked="" type="checkbox"/> Script	Support for JavaScript debugging.	Enabled for www.appspotenginelearn.com
<input checked="" type="checkbox"/> Net	Support for Network monitoring.	Enabled for www.appspotenginelearn.com

Apply settings for www.appspotenginelearn.com

Transferring data from i2.ytimg.com...

ALL RIGHTS RESERVED

TECHNION Azrieli Continuing Education and External Studies Division

AppEngineLearn: An Introductory Guide to the Google Application Engine

http://www.appspotenginelearn.com/ Google

Disable Cookies CSS Forms Images Information Miscellaneous Outline Resize Tools View Source

AppEngineLearn

Book Instructor Python App Engine

This site provides materials to help learn the Google Application Engine. Before you start to learn the Google Application Engine you should be basically familiar with the Python programming language.

New: You can take a look at the [draft book chapters](#) for my upcoming O'Reilly AppEngine book titled, "Building Cloud Applications with Google AppEngine".

- [Installing Python and JEdit](#) - We recommend using JEdit as your programmer editor and it will be used throughout the Podcasts.
- Installing the Application Engine and writing your first Application.
 - Macintosh: ([Handout](#), [Source Code](#), [Screencast](#), [YouTube](#))
 - Windows Vista: ([Handout](#), [Source Code](#), [High Quality Screencast](#), [YouTube](#))



© COPYRIG
2022

ALL RIGHTS
WRITTEN PERMISSION

Inspect Clear : All HTML CSS JS XHR Images Flash

Console HTML CSS Script DOM Net Options

Request	Status	Size	Time
▶ GET www.appspotenginelearn.com	200 OK	appenginelearn.com 7 KB	222ms
▶ GET glike.css	200 OK	appenginelearn.com 3 KB	112ms
▶ GET csev.jpg	200 OK	appenginelearn.com 15 KB	144ms
▶ GET ile-main.js	200 OK	cloudsocial.org 88 B	181ms
▶ GET 93HjHU25low&h	303 See Other	youtube.com ?	258ms
▶ GET l.swf?swf=http%	200 OK	youtube.com 724 B	76ms
▶ GET cps-vfl78303.swf	200 OK	s.ytimg.com 120 KB	1.02s
▶ GET crossdomain.xml	200 OK	i2.ytimg.com 97 B	71ms
▶ GET hqdefault.jpg	200 OK	i2.ytimg.com 24 KB	82ms
9 requests	167 KB	2.04s	

Transferring data from i2.ytimg.com...

© COPYRIG
2022

ALL RIGHTS
WRITTEN PERMISSION

TECHNION

Azrieli Continuing Education and External Studies Division

Lets Create a Socket Connection!

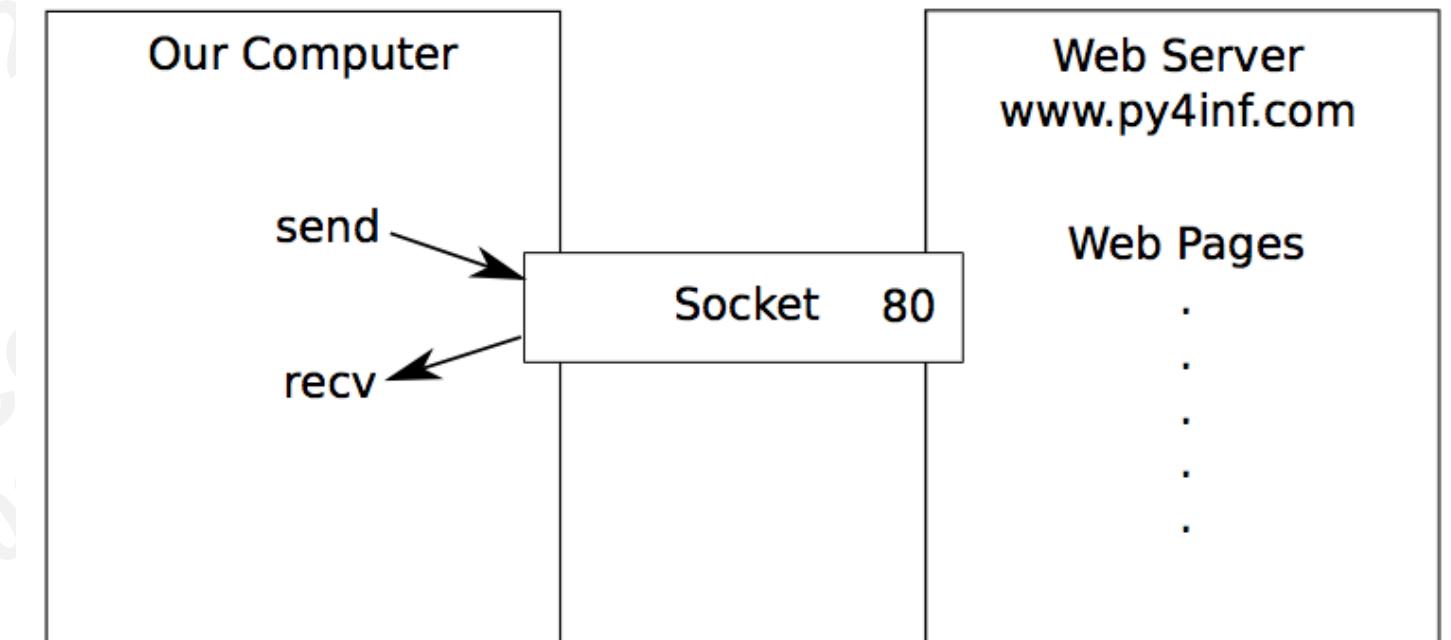
ALL RIGHTS RESERVED © COPYRIGHT 2022 | DO NOT DISTRIBUTE WITHOUT
WRITTEN PERMISSION

An HTTP Request in Python

```
import socket  
mysock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)  
mysock.connect(('www.py4inf.com', 80))
```

```
mysock.send('GET http://www.py4inf.com/code/romeo.txt HTTP/1.0\n\n')
```

```
while True:  
    data = mysock.recv(512)  
    if ( len(data) < 1 ) :  
        break  
    print data  
mysock.close()
```



HTTP/1.1 200 OK

Date: Sun, 14 Mar 2010 23:52:41 GMT

Server: Apache

Last-Modified: Tue, 29 Dec 2009 01:31:22 GMT

ETag: "143c1b33-a7-4b395bea"

Accept-Ranges: bytes

Content-Length: 167

Connection: close

Content-Type: text/plain

But soft what light through yonder window breaks

It is the east and Juliet is the sun

Arise fair sun and kill the envious moon

Who is already sick and pale with grief

HTTP Header

while True:

 data = mysock.recv(512)

 if (len(data) < 1) :

 break

 print data

HTTP Body

Making HTTP Easier With “urllib”

ALL RIGHTS RESERVED © COPYRIGHT 2022 | DO NOT DISTRIBUTE WITHOUT
WRITTEN PERMISSION

Using “urllib” in Python

Since HTTP is so common, we have a library that does all the socket work for us and makes web pages appear as a file.

```
import urllib  
fhand = urllib.urlopen('http://www.py4inf.com/code/romeo.txt')  
  
for line in fhand:  
    print line.strip()
```

```
import urllib  
fhand = urllib.urlopen('http://www.py4inf.com/code/romeo.txt')  
for line in fhand:  
    print line.strip()
```

But soft what light through yonder window breaks
It is the east and Juliet is the sun
Arise fair sun and kill the envious moon
Who is already sick and pale with grief

Like A File...

```
import urllib
fhand = urllib.urlopen('http://www.py4inf.com/code/romeo.txt')

counts = dict()
for line in fhand:
    words = line.split()
    for word in words:
        counts[word] = counts.get(word, 0) + 1
print counts
```

Reading Web Pages

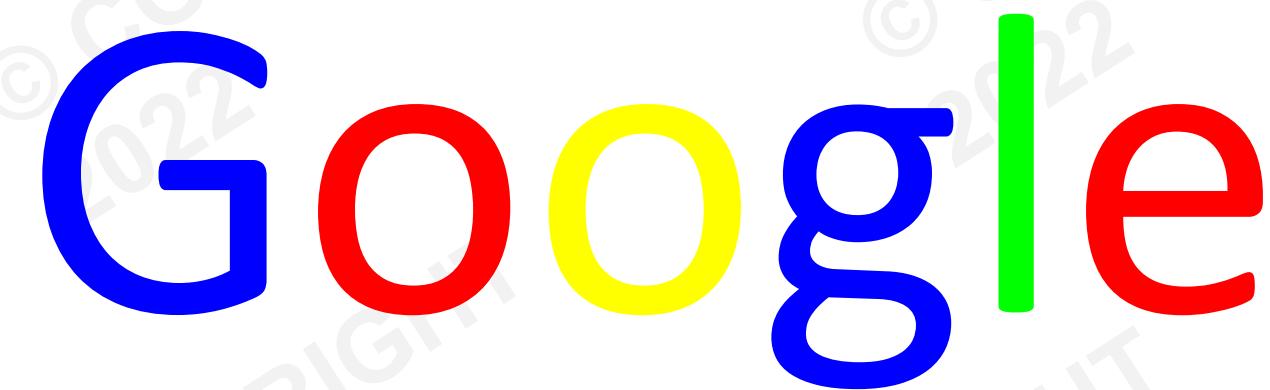
```
import urllib  
fhand = urllib.urlopen('http://www.dr-chuck.com/page1.htm')  
for line in fhand:  
    print line.strip()
```

<h1>The First Page</h1><p>If you like, you can
switch to the<a href="http://www.dr-
chuck.com/page2.htm">Second Page.</p>

Going From One Page to The Next...

```
import urllib
fhand = urllib.urlopen('http://www.dr-chuck.com/page1.htm')
for line in fhand:
    print line.strip()
```

<h1>The First Page</h1><p>If you like, you can
switch to the<a href="http://www.dr-
chuck.com/page2.htm">Second Page.</p>



```
import urllib
fhand = urllib.urlopen('http://www.dr-chuck.com/page1.htm')
for line in fhand:
    print line.strip()
```

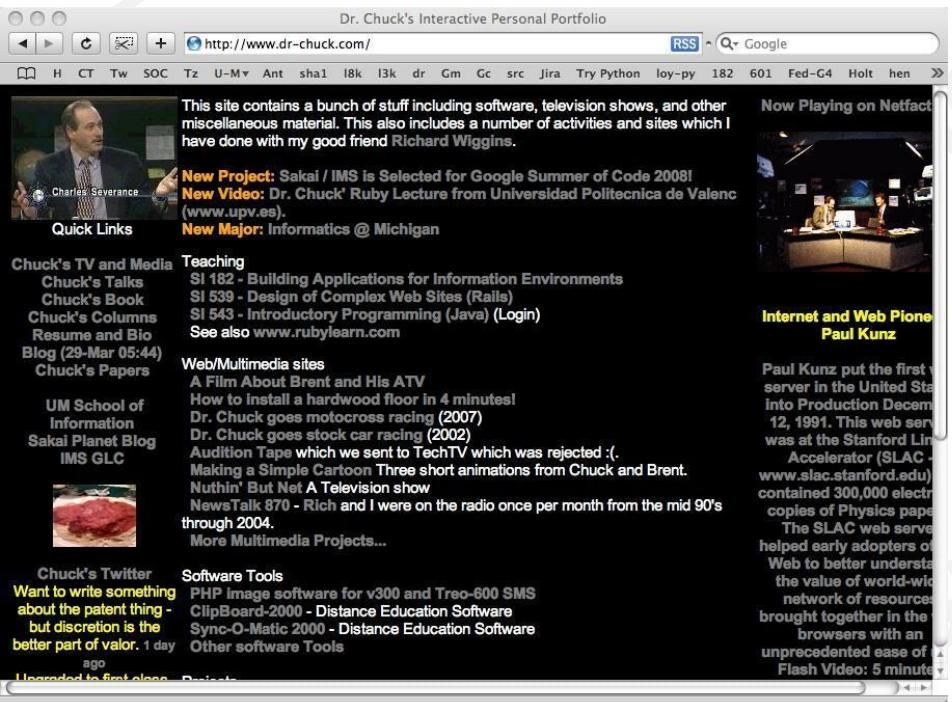
Parsing HTML (a.k.a. Web Scraping)

ALL RIGHTS RESERVED © COPYRIGHT 2022 | DO NOT DISTRIBUTE WITHOUT
WRITTEN PERMISSION

What is Web Scraping?

- When a program or script pretends to be a browser and retrieves web pages, looks at those web pages, extracts information, and then looks at more web pages.
- Search engines scrape web pages - we call this “spidering the web” or “web crawling”.

Server



GET

HTML

GET

HTML

```
charles-severances-macbook-air:Scraping csev$ python
Python 2.5 (r25:51918, Sep 19 2006, 08:49:13)
[GCC 4.0.1 (Apple Computer, Inc. build 5341)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import urllib
>>> f = urllib.urlopen("http://www.dr-chuck.com/")
>>> contents = f.read()
>>> f.close()
>>> print len(contents)
95328
>>> print contents[0:30]
<html>
<head>
  <title>Dr. C
>>> []
```

Why Scrape?

- Pull Data - particularly social data - who links to who?
- Get your own data back out of some system that has no “export capability”
- Monitor a site for new information
- Spider the web to make a database for a search engine

Scraping Web Pages

- There is some controversy about web page scraping and some sites are a bit snippy about it.
 - Google: facebook scraping block
- Republishing copyrighted information is not allowed.
- Violating terms of service is not allowed.

<http://www.facebook.com/terms.php>

User Conduct

You understand that except for advertising programs offered by us on the Site (e.g., Facebook Flyers, Facebook Marketplace), the Service and the Site are available for your personal, non-commercial use only. You represent, warrant and agree that no materials of any kind submitted through your account or otherwise posted, transmitted, or shared by you on or through the Service will violate or infringe upon the rights of any third party, including copyright, trademark, privacy, publicity or other personal or proprietary rights; or contain libelous, defamatory or otherwise unlawful material.

In addition, you agree not to use the Service or the Site to:

- harvest or collect email addresses or other contact information of other users from the Service or the Site by electronic or other means for the purposes of sending unsolicited emails or other unsolicited communications;
- use the Service or the Site in any unlawful manner or in any other manner that could damage, disable, overburden or impair the Site;
- use automated scripts to collect information from or otherwise interact with the Service or the Site;

The Easy Way - Beautiful Soup

You could do string searches the hard way...

...or use the free software called BeautifulSoup from
www.crummy.com

<http://www.crummy.com/software/BeautifulSoup/>

Place the BeautifulSoup.py file in the same folder as your Python code...

BeautifulSoup Script To Scrap The Page

```
import urllib
from BeautifulSoup import *

url = raw_input('Enter - ')

html = urllib.urlopen(url).read()
soup = BeautifulSoup(html)

# Retrieve a list of the anchor tags
# Each tag is like a dictionary of HTML attributes

tags = soup('a')

for tag in tags:
    print tag.get('href', None)
```

<h1>The First Page</h1><p>If you like, you
can switch to the<a href="http://www.dr-
chuck.com/page2.htm">Second
Page.</p>

```
html = urllib.urlopen(url).read()  
soup = BeautifulSoup(html)  
  
tags = soup('a')  
for tag in tags:  
    print tag.get('href', None)
```

python urllinks.py
Enter - http://www.dr-
chuck.com/page1.htm
http://www.dr-chuck.com/page2.htm

Summary

- The TCP/IP gives us pipes / sockets between applications.
- We designed application protocols to make use of these pipes.
- HyperText Transport Protocol (HTTP) is a simple yet powerful protocol.
- Python has good support for sockets, HTTP, and HTML parsing.