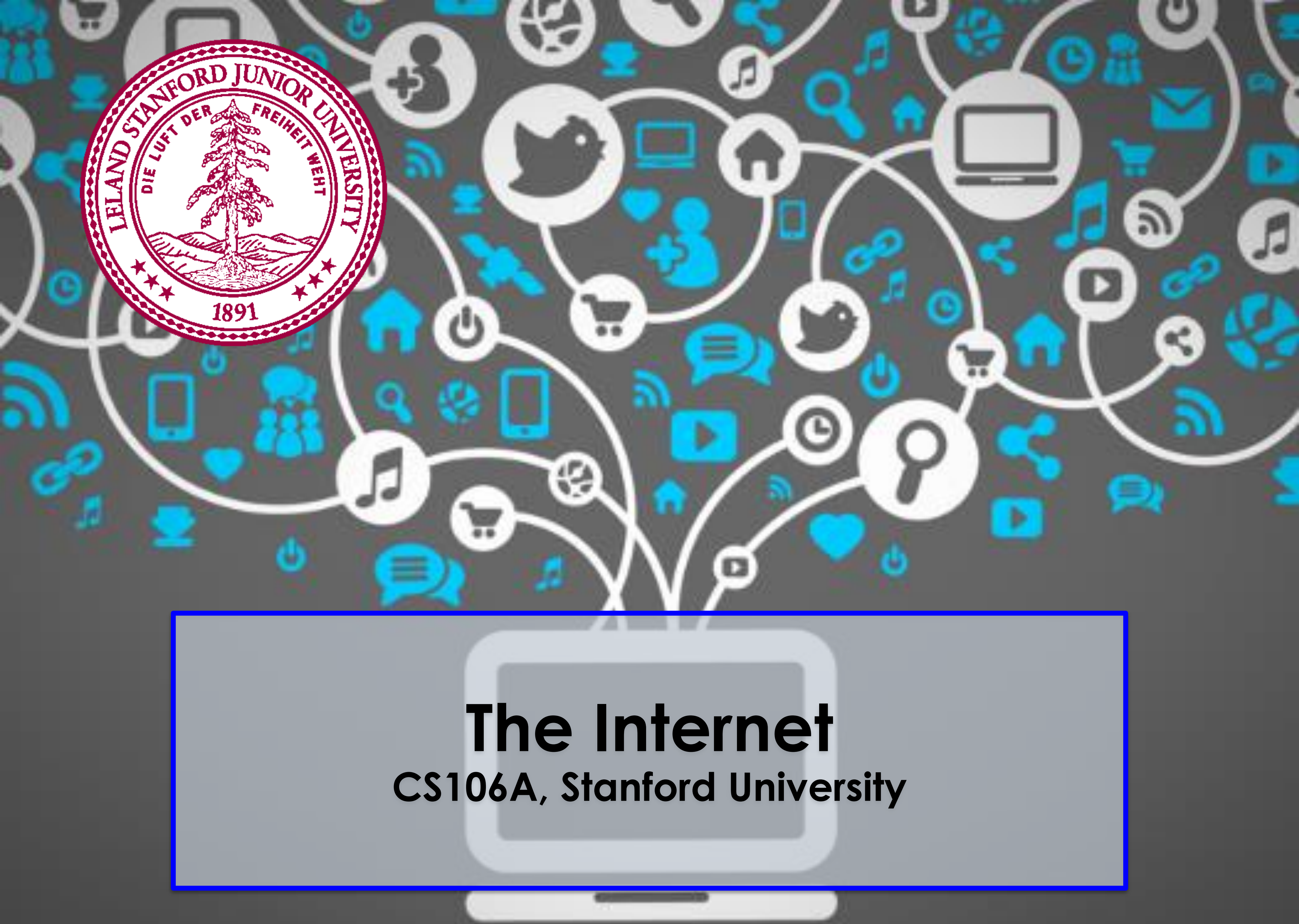
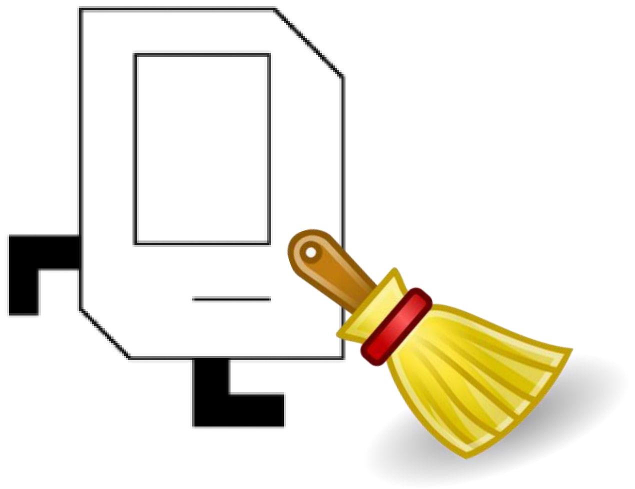


The Internet

CS106A, Stanford University

Housekeeping



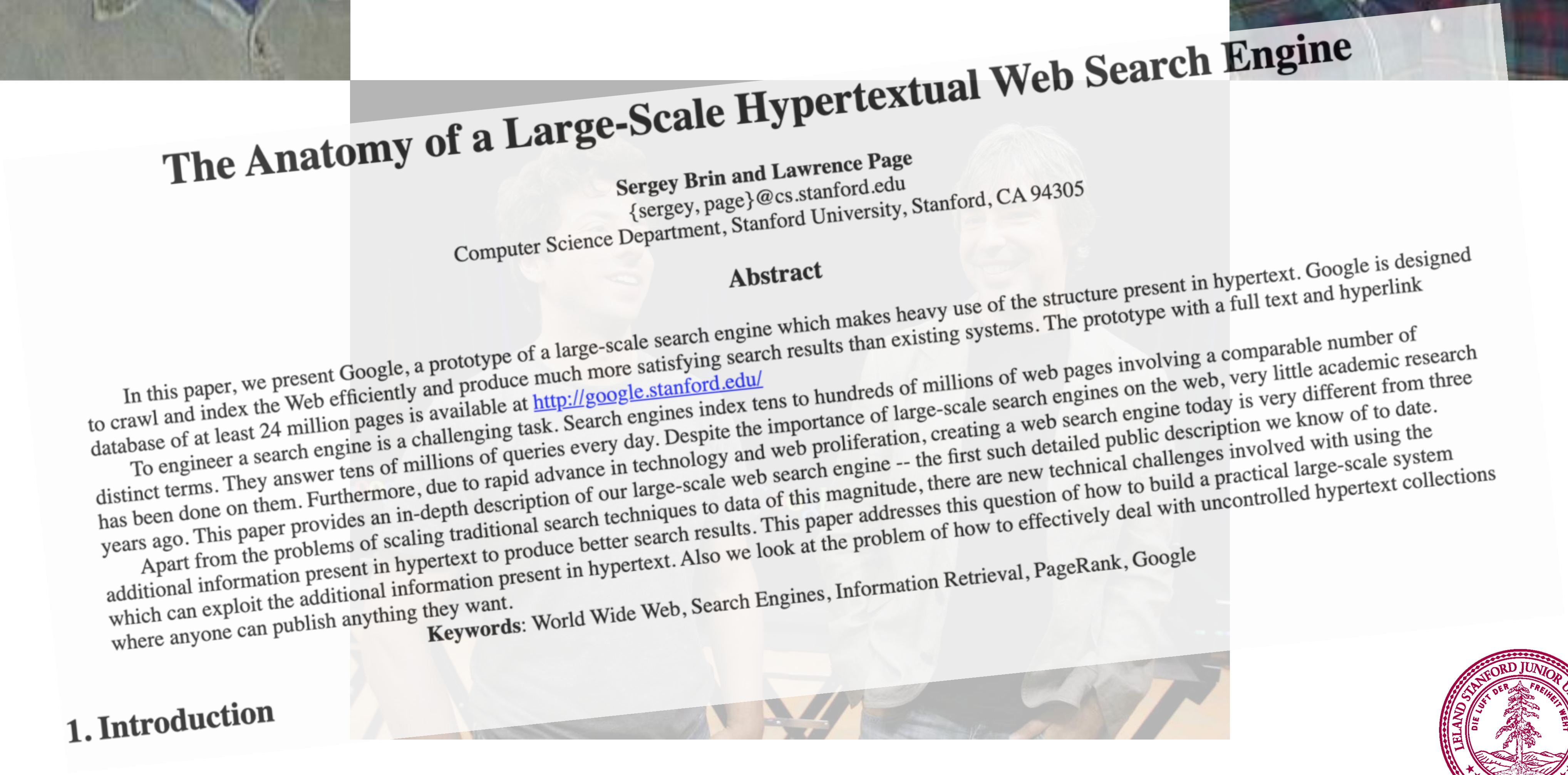
- Assignment #6 due today
  - Poll: https://pollev.com/assignment6
- Assignment #7 goes out today
  - Due Wednesday, June 1st
  - Late days (free or otherwise) cannot be used on Assignment #7
- Acknowledging end-of-quarter stress
  - Take care of yourselves and each other
  - If you need help, please reach out
- Ethics mini-lecture on search engines

Ethics of Search

Ethics of Search

Ethics of Search

Ethics of Search



The Anatomy of a Large-Scale Hypertextual Web Search Engine

Sergey Brin and Lawrence Page
{sergey, page}@cs.stanford.edu
Computer Science Department, Stanford University, Stanford, CA 94305

Abstract

In this paper, we present Google, a prototype of a large-scale search engine which makes heavy use of the structure present in hypertext. Google is designed to crawl and index the Web efficiently and produce much more satisfying search results than existing systems. The prototype with a full text and hyperlink database of at least 24 million pages is available at http://google.stanford.edu/

To engineer a search engine is a challenging task. Search engines index tens to hundreds of millions of web pages involving a comparable number of distinct terms. They answer tens of millions of queries every day. Despite the importance of large-scale search engines on the web, very little academic research has been done on them. Furthermore, due to rapid advance in technology and web proliferation, creating a web search engine today is very different from three years ago. This paper provides an in-depth description of our large-scale web search engine -- the first such detailed public description we know of to date.

Apart from the problems of scaling traditional search techniques to data of this magnitude, there are new technical challenges involved with using the additional information present in hypertext to produce better search results. This paper addresses this question of how to build a practical large-scale system which can exploit the additional information present in hypertext. Also we look at the problem of how to effectively deal with uncontrolled hypertext collections where anyone can publish anything they want.

Keywords: World Wide Web, Search Engines, Information Retrieval, PageRank, Google

1. Introduction

Ethics of Search

"we expect that advertising funded search engines will be inherently biased towards the advertisers and away from the needs of the consumers ...

Since it is very difficult even for experts to evaluate search engines, search engine bias is particularly insidious. ...

a search engine could add a small factor to search results from "friendly" companies, and subtract a factor from results from competitors. ...

[W]e believe the issue of advertising causes enough mixed incentives that it is crucial to have a competitive search engine that is transparent and in the academic realm."

Brin & Page 1998

What is Bias in Search?

Possible Concerns about Bias in Search:

(1) "search-engine technology is not neutral, but instead has embedded features in its design that favor some values over others"?

Possible Concerns about Bias in Search:

(1) "search-engine technology is not neutral, but instead has embedded features in its design that favor some values over others"?

- "relevance" to user
- "quality" of results

Possible Concerns about Bias in Search:

(1) "search-engine technology is not neutral, but instead has embedded features in its design that favor some values over others"?

- "relevance" to user
- "quality" of results

Thick normative terms that encode values!

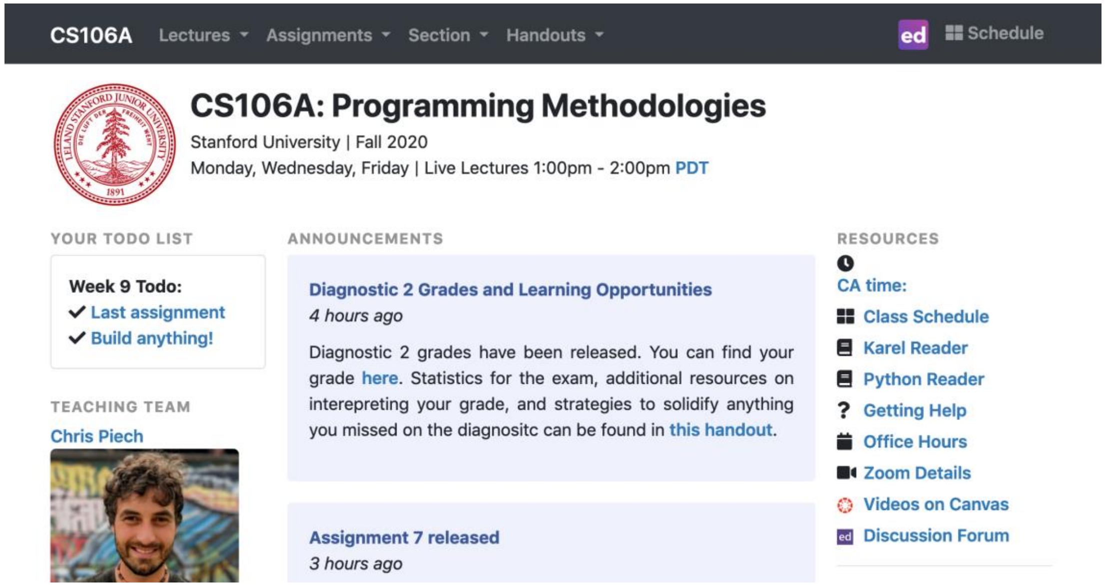
Possible Concerns about Bias in Search:

(2) “major search engines **systematically favor** some sites (and some kinds of sites) over others in the lists of results they return in response to user search queries”?

Concerns about Bias in Search:

(2) "major search engines systematically favor some sites (and some kinds of sites) over others in the lists of results they return in response to user search queries"?

"High quality website"



CS106A: Programming Methodologies

Stanford University | Fall 2020

Monday, Wednesday, Friday | Live Lectures 1:00pm - 2:00pm PDT

YOUR TODO LIST

Week 9 Todo:
- ✓ Last assignment
- ✓ Build anything!

TEACHING TEAM

Chris Piech

ANNOUNCEMENTS

Diagnostic 2 Grades and Learning Opportunities

4 hours ago

Diagnostic 2 grades have been released. You can find your grade here. Statistics for the exam, additional resources on interepreting your grade, and strategies to solidify anything you missed on the diagnositc can be found in this handout.

Assignment 7 released

3 hours ago

RESOURCES

CA time:

Class Schedule

Karel Reader

Python Reader

Getting Help

Office Hours

Zoom Details

Videos on Canvas

Discussion Forum

Concerns about Bias in Search:

(2) "major search engines systematically favor some sites (and some kinds of sites) over others in the lists of results they return in response to user search queries"?

"low quality website"

mehranandjulietteburritos.com

Nothing links to it,

doesn't actually help you find burritos

Concerns about Bias in Search:

(3) "search algorithms do not use objective criteria in generating their lists of results for search queries"?

Our criteria are

- "relevance" to user
- "quality" of results

Relevance is a subjective metric.

It can't be determined without asking, relevant to whom?

Relevance and Advertising

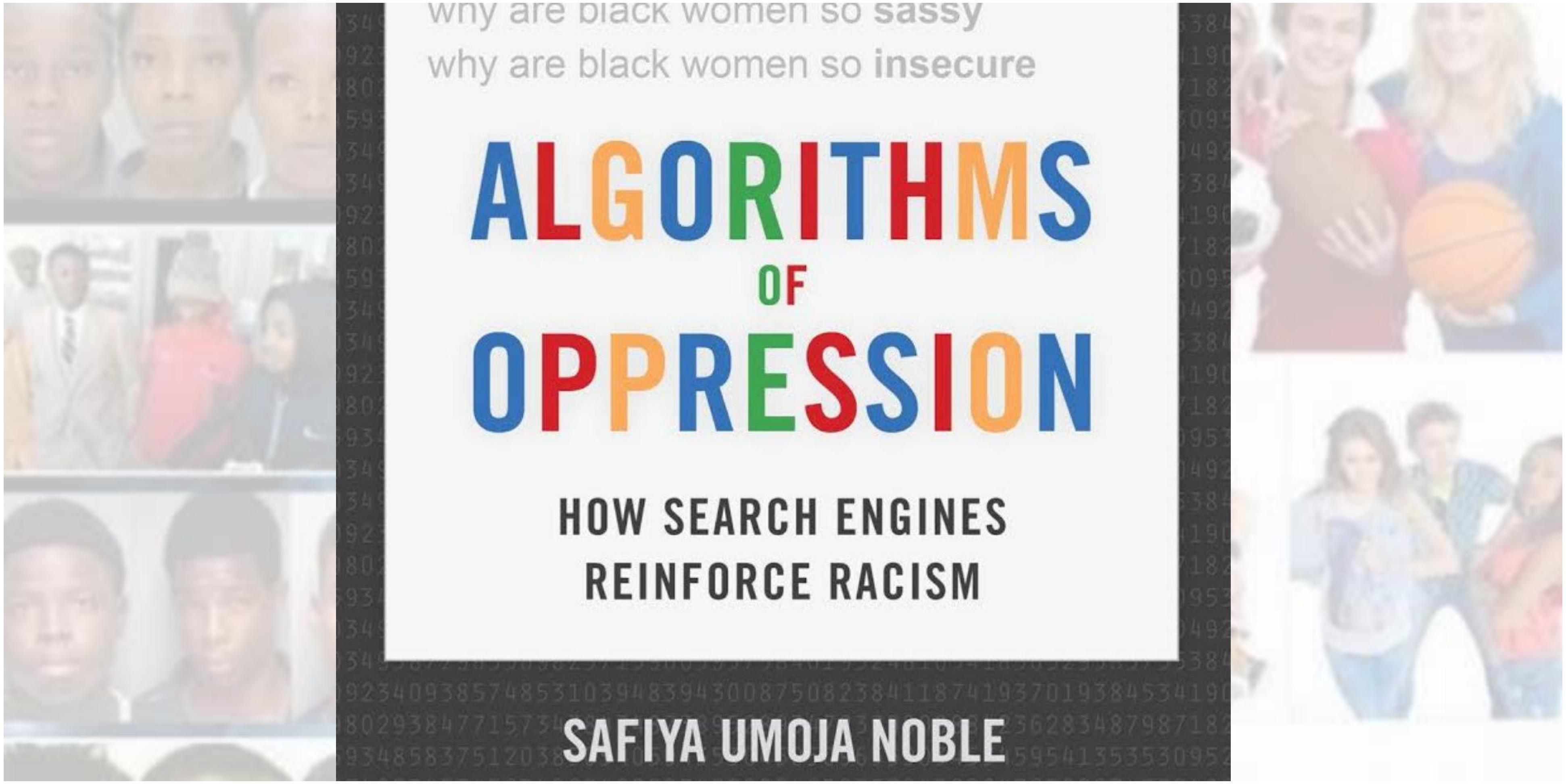
How might advertising affect relevance?

Advertising bias might lead to delivering what the advertiser wants the user to see, rather than what the user considers relevant.

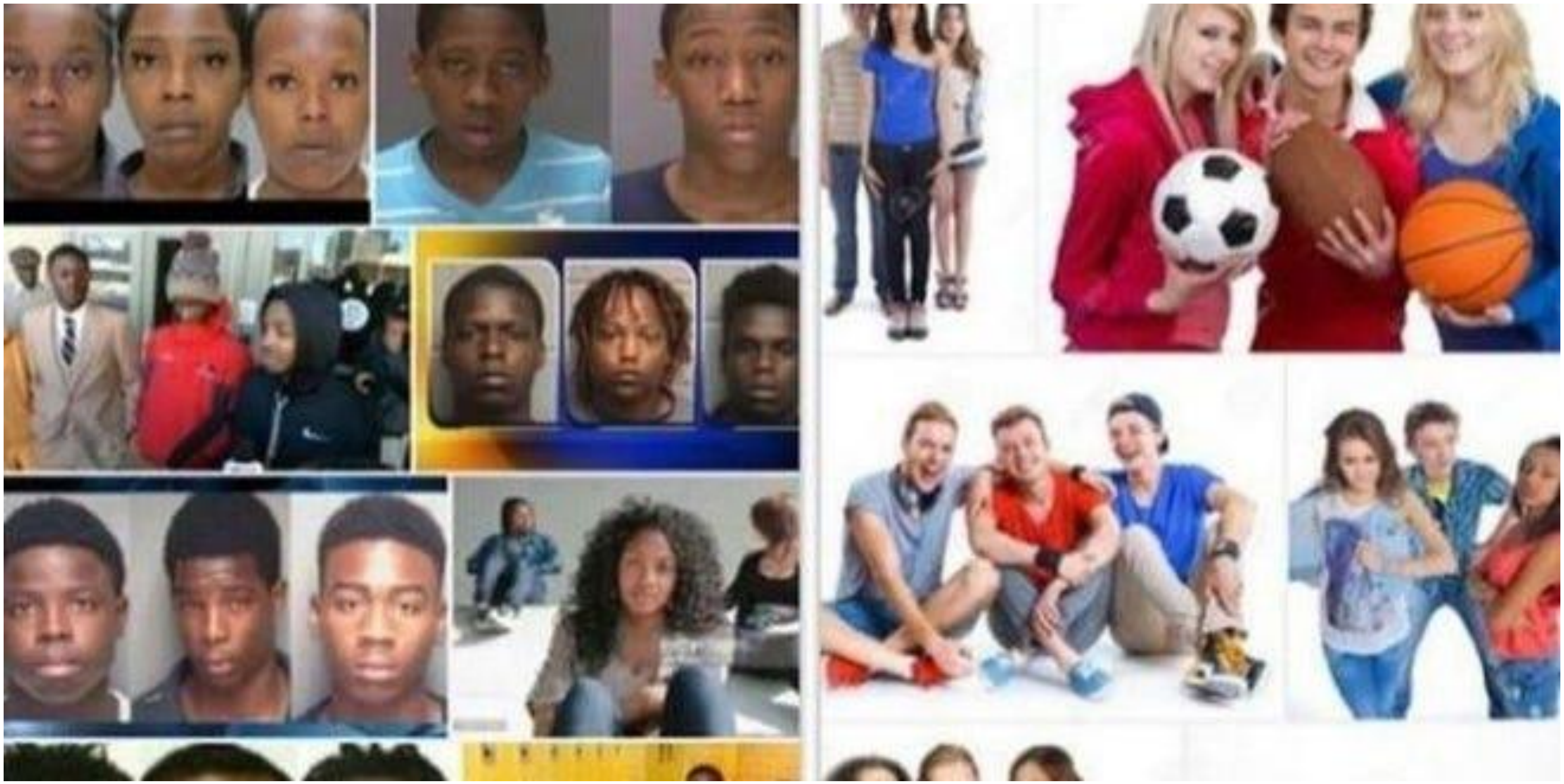
Hence advertising bias would move search results further from the subjective success of delivering the user's desired results.

What about quality? Is that an objective metric?

Bias, Quality, and Lack of Objectivity


why are black women so sassy
why are black women so insecure
ALGORITHMS OF OPPRESSION
HOW SEARCH ENGINES REINFORCE RACISM
SAFIYA UMOJA NOBLE

Bias, Quality, and Lack of Objectivity



Search for "three black teenagers" vs "three white teenagers"

Objectivity: Definitions

Objectivity as freedom from bias?

Objectivity as faithfulness to facts?

Objectivity as absence of normative commitments?

Objectivity as integration of multiple perspectives & reflexivity about values in the methodology?

Objectivity: Definitions

Objectivity as integration of multiple perspectives & reflexivity about values in the methodology

Personalization & Democracy

Search engines have been called the "gatekeepers of the web," tools that "structure knowledge" for internet users & "contribute to the public use of reason." Does this confer obligation?

One reason programming is fun is because of the internet...

Learning Goals

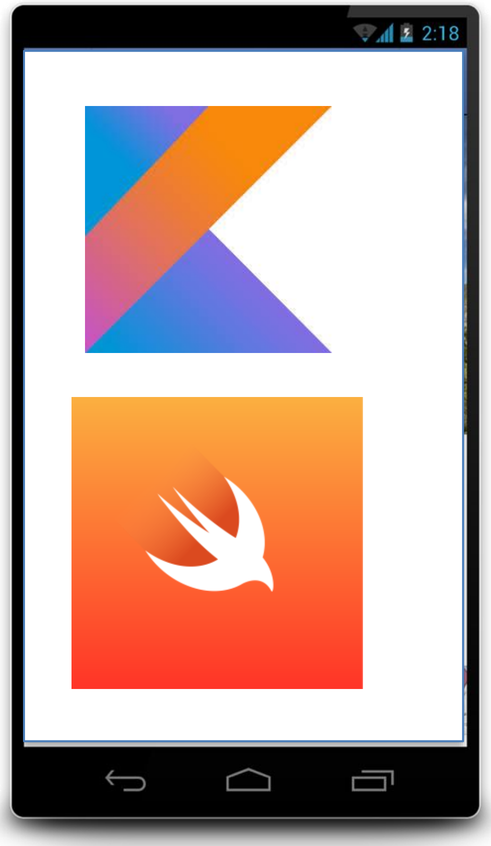
1. Write a program that can respond to internet requests

How does your phone communicate with Facebook?

The program on your **phone** talks to the program at **Facebook**

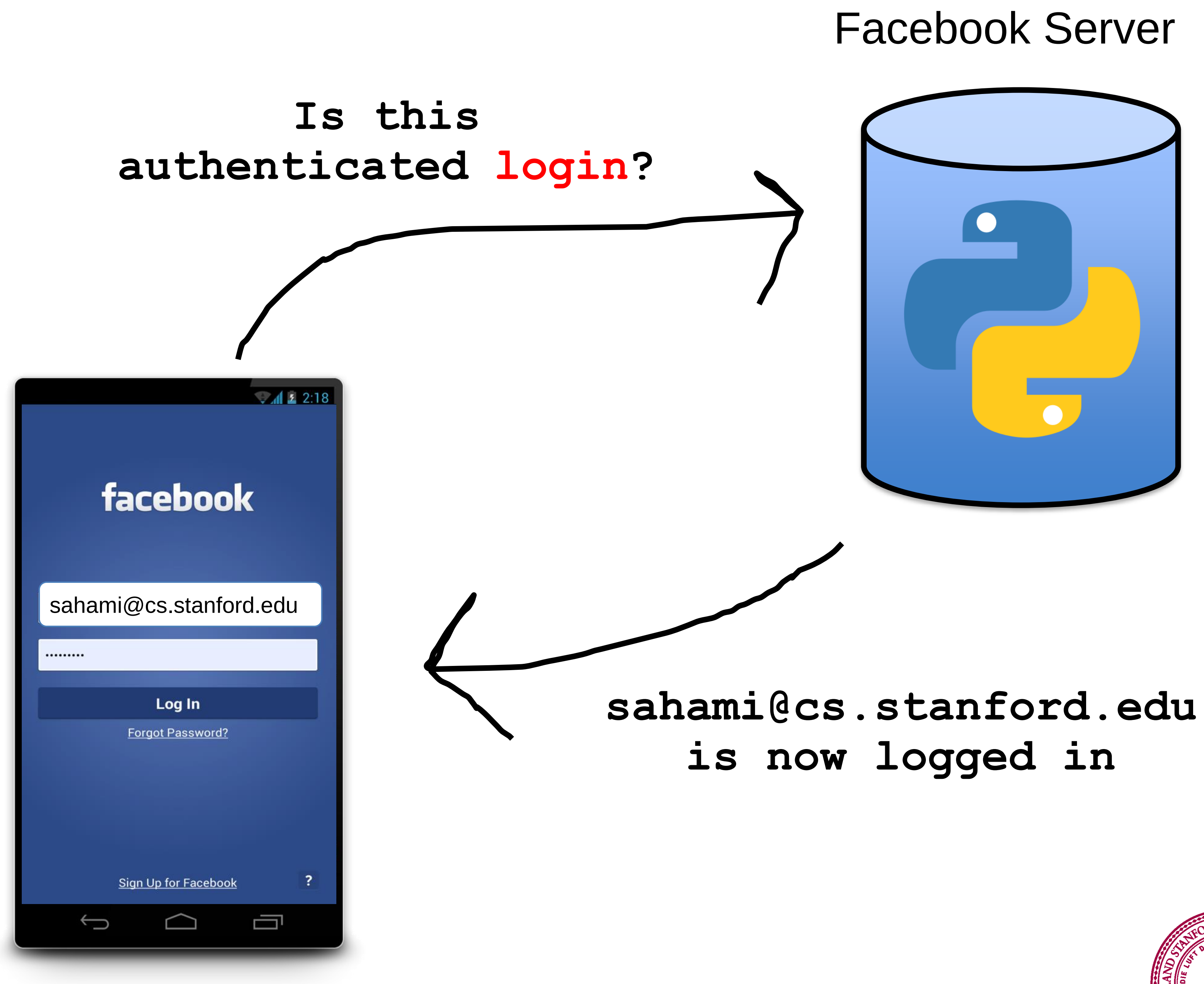
Facebook Server

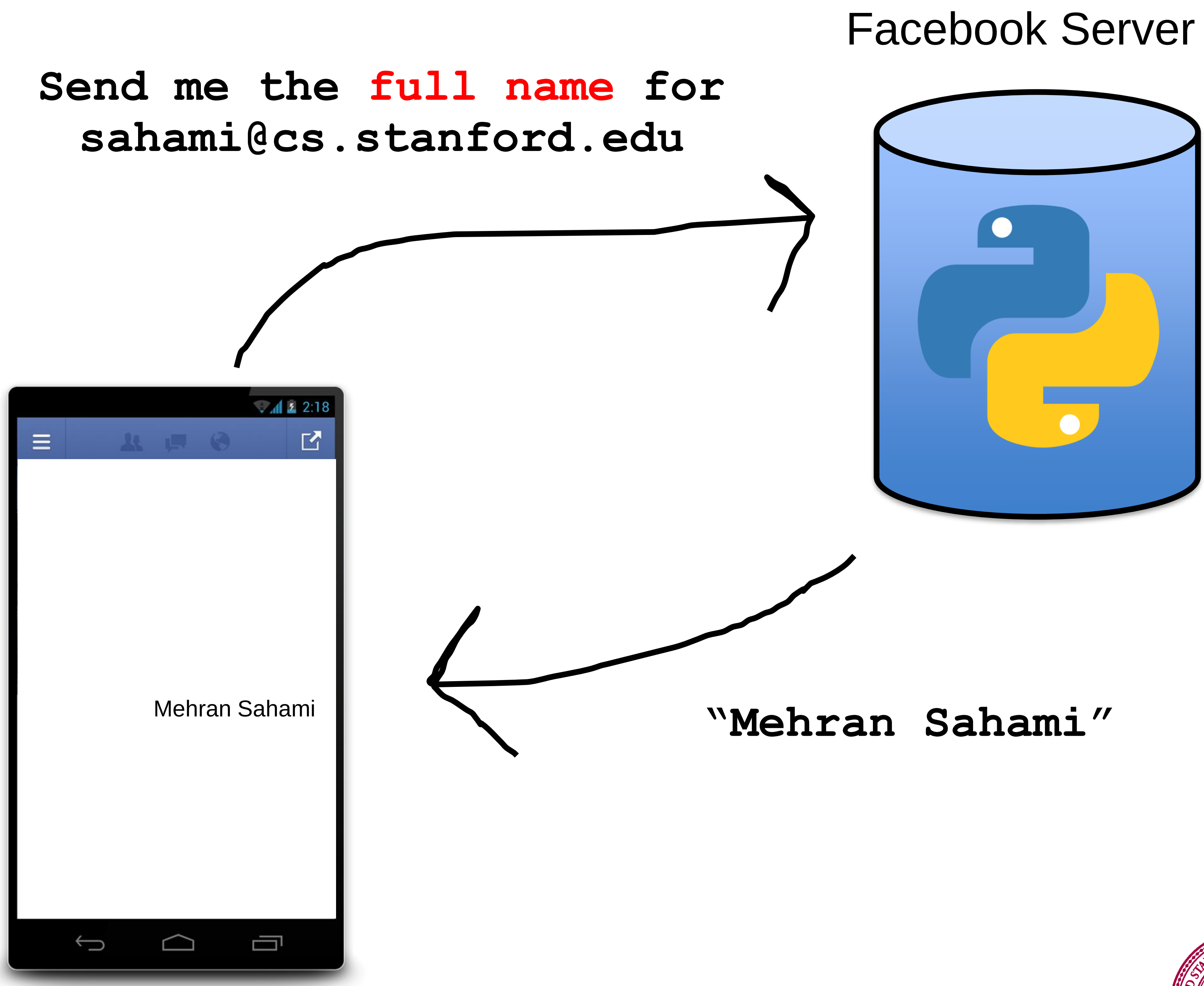
JavaScript with HTML are the languages of websites



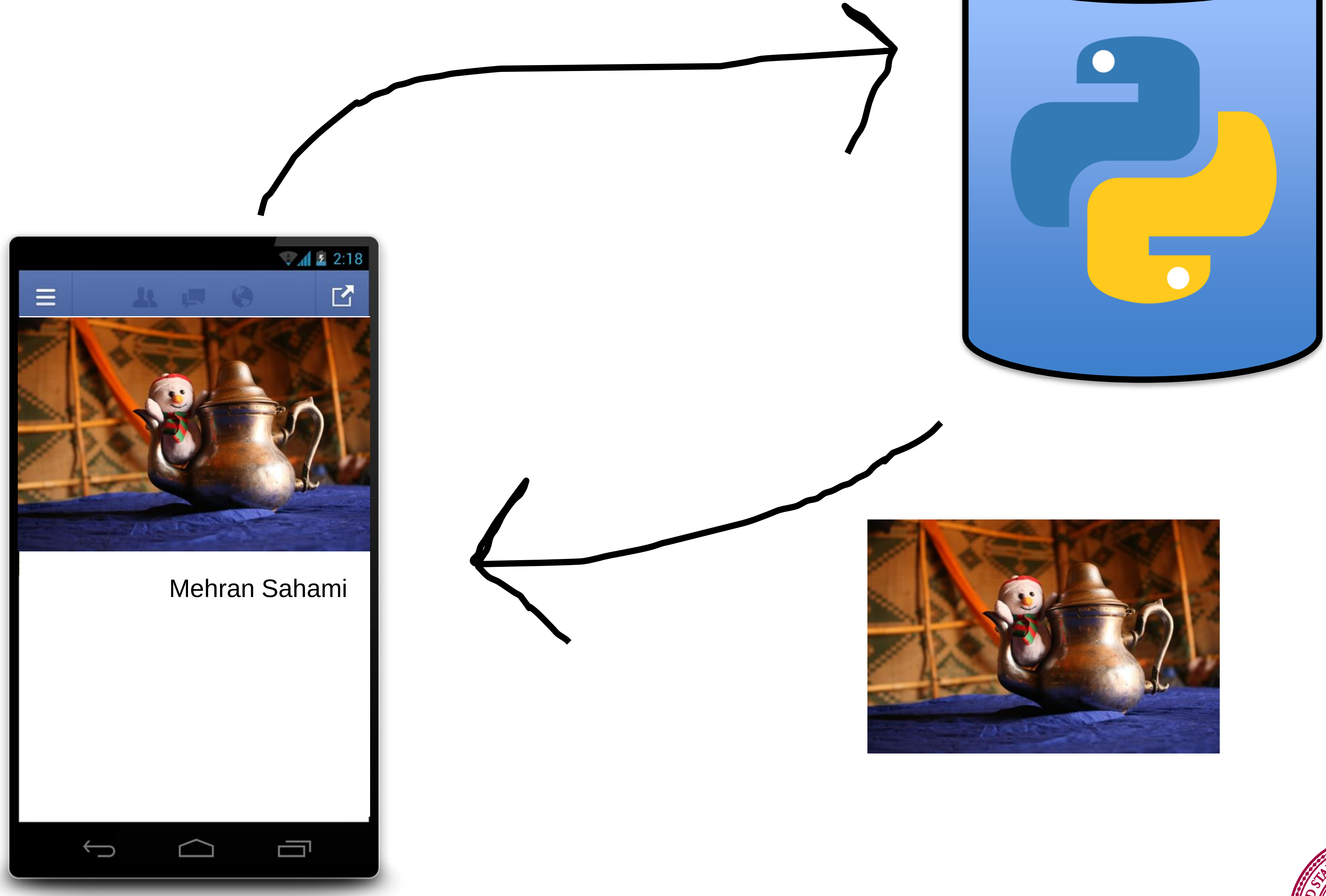
Kotlin is the language of Android phones

Swift is the language of Apple phones


Facebook Server
Is this authenticated login?
facebook
sahami@cs.stanford.edu
Log In
Forgot Password?
Sign Up for Facebook
sahami@cs.stanford.edu is now logged in

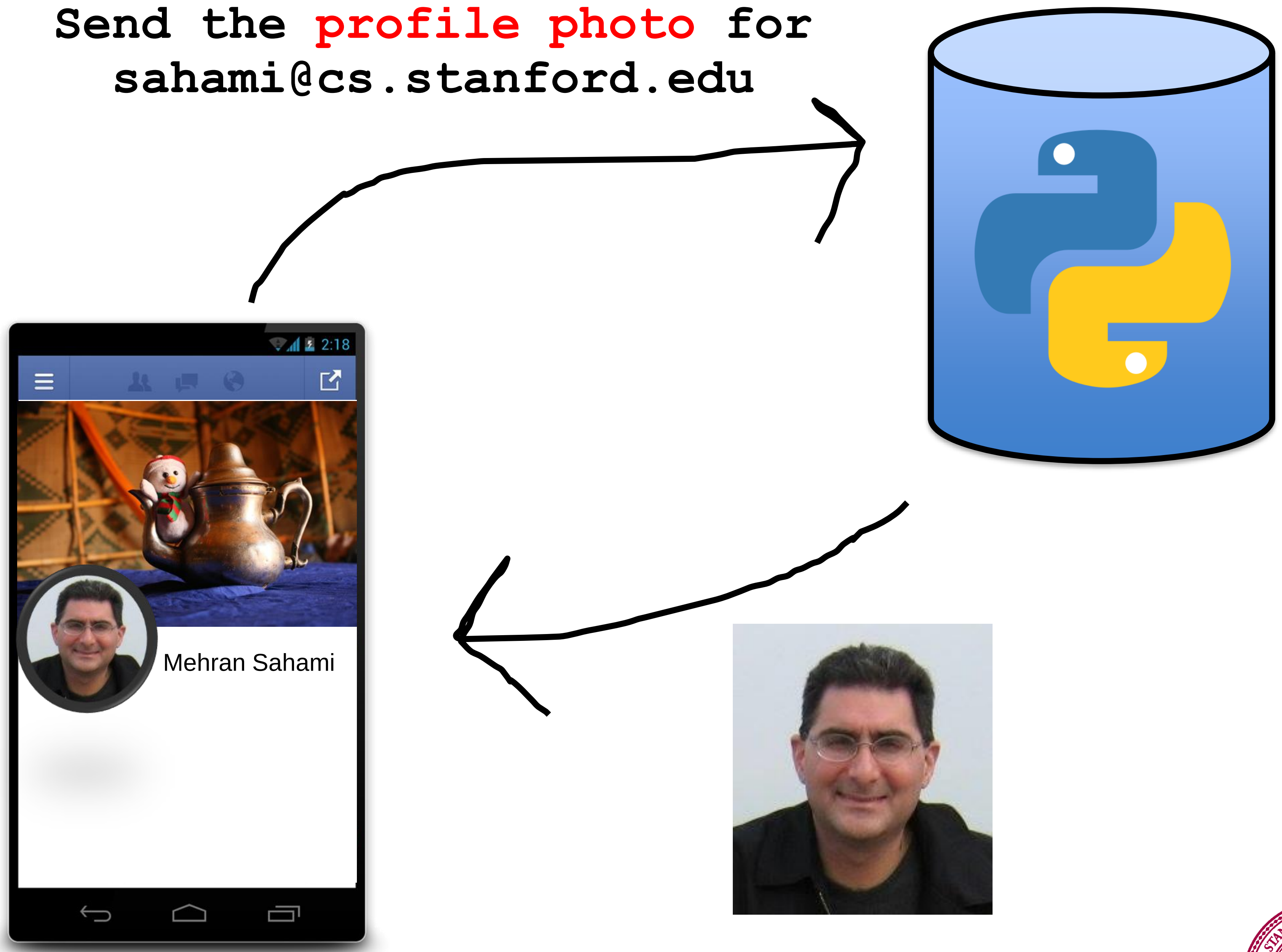

Facebook Server
Send me the full name for sahami@cs.stanford.edu
"Mehran Sahami"
2:18
Mehran Sahami

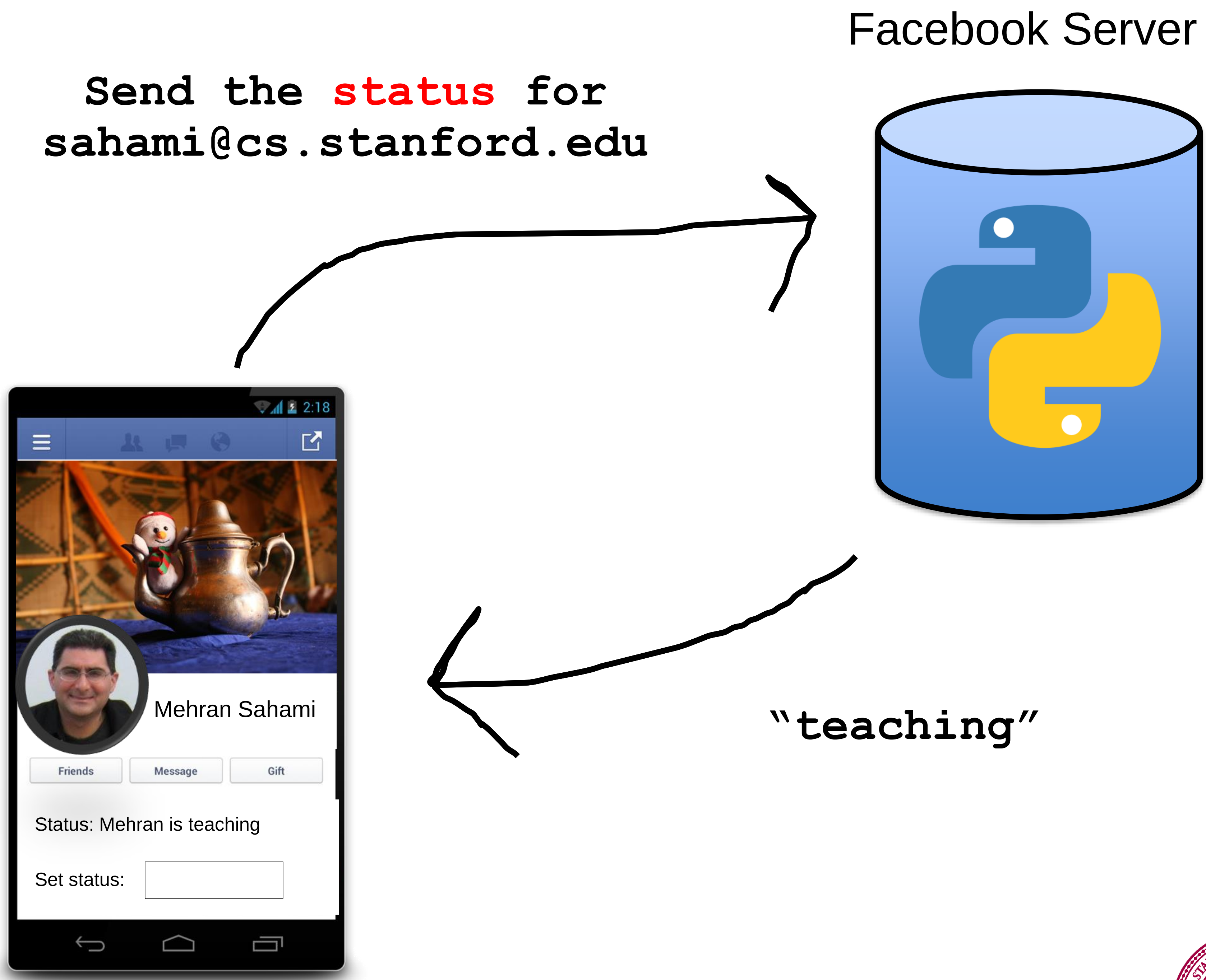
Send me the cover photo for sahami@cs.stanford.edu



Facebook Server

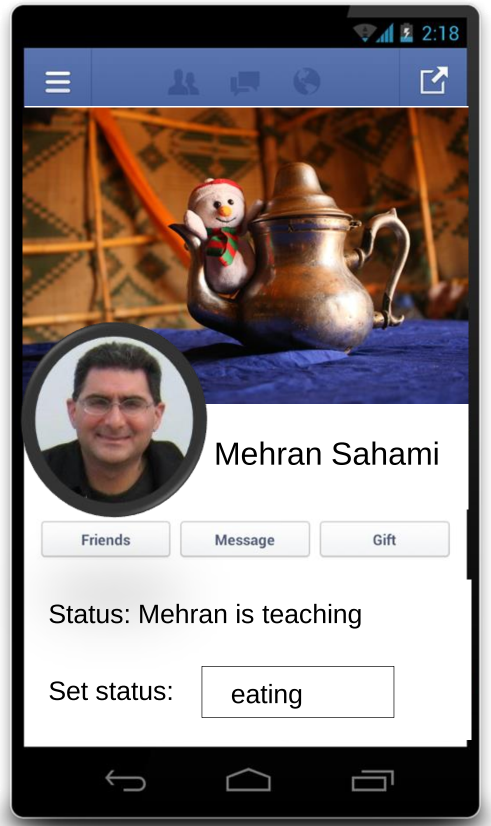
Send the profile photo for sahami@cs.stanford.edu




Facebook Server
Send the status for
sahami@cs.stanford.edu
"teaching"
2:18
Mehran Sahami
Friends
Message
Gift
Status: Mehran is teaching
Set status:

Set the status for
sahami@cs.stanford.edu
to be "eating"

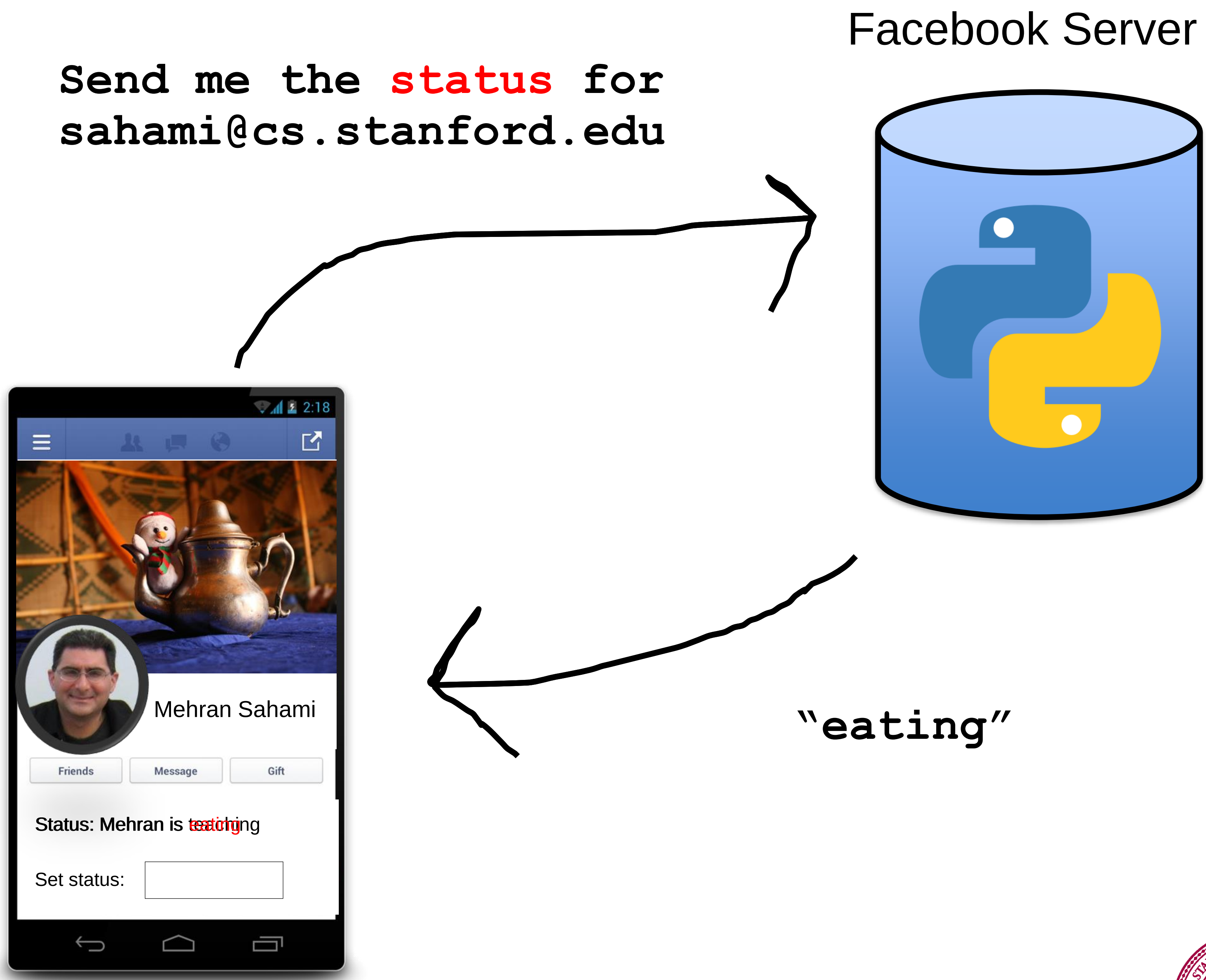
Facebook Server



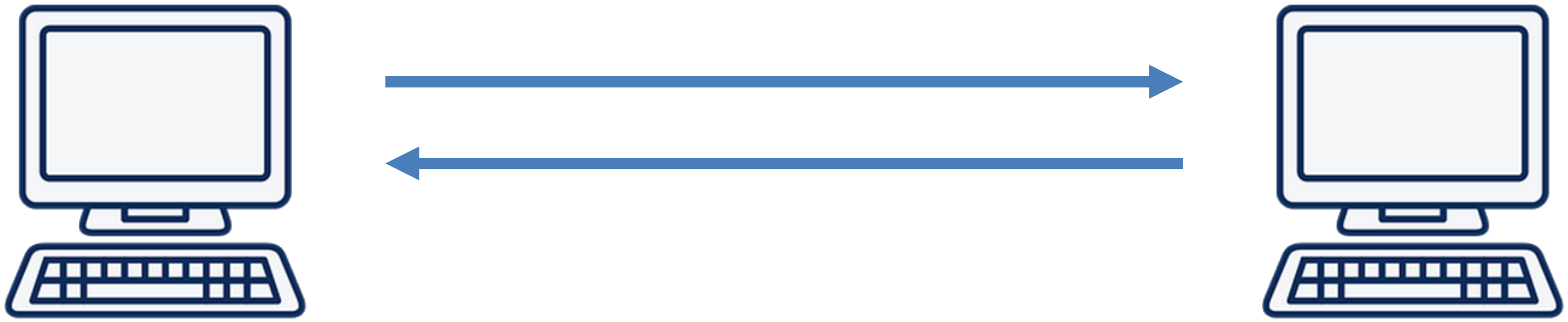
Status: Mehran is teaching

Set status: eating

"success"

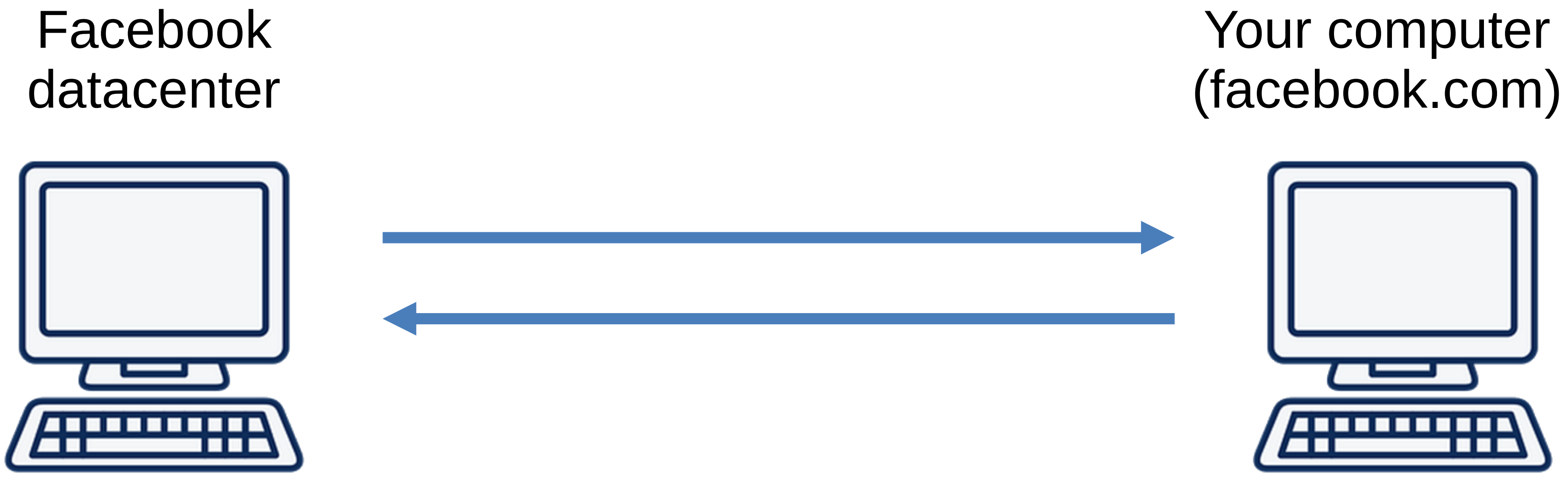

Facebook Server
Send me the status for sahami@cs.stanford.edu
"eating"
2:18
Mehran Sahami
Friends
Message
Gift
Status: Mehran is eating
Set status:

Background: The Internet



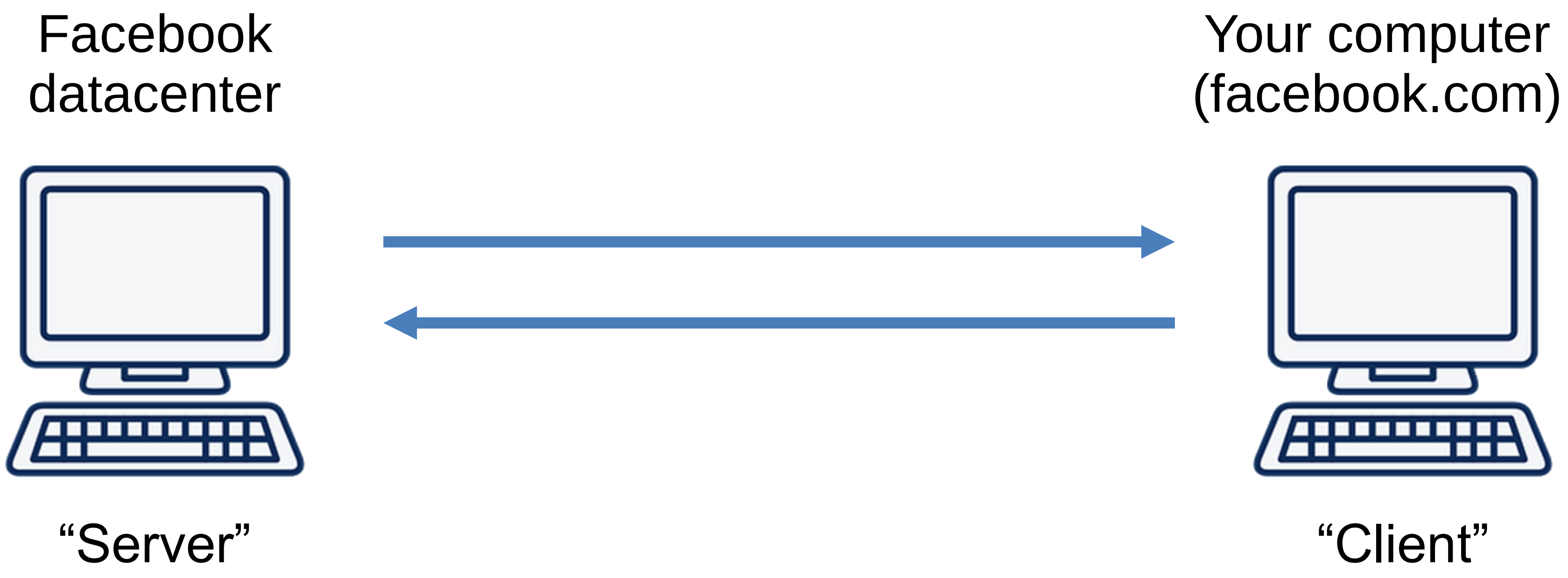
The internet is just many programs sending messages (as *Strings*)

Background: The Internet



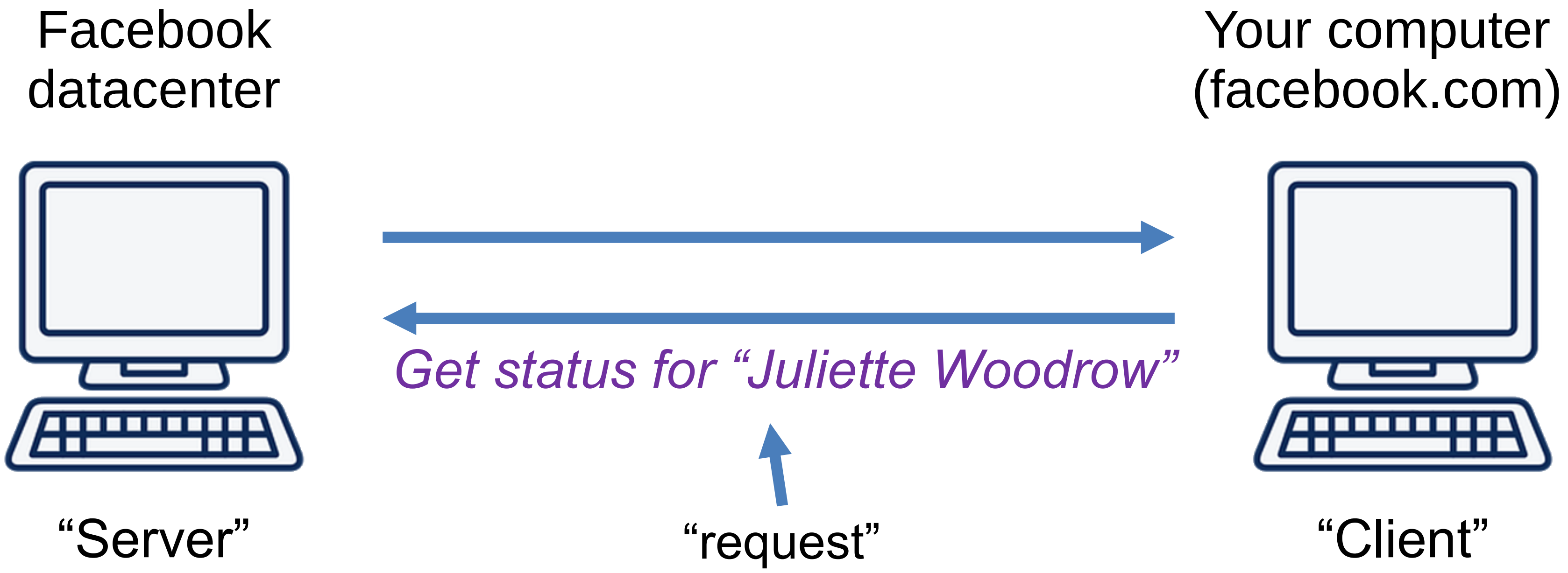
The internet is just many programs sending messages (as *Strings*)

Background: The Internet



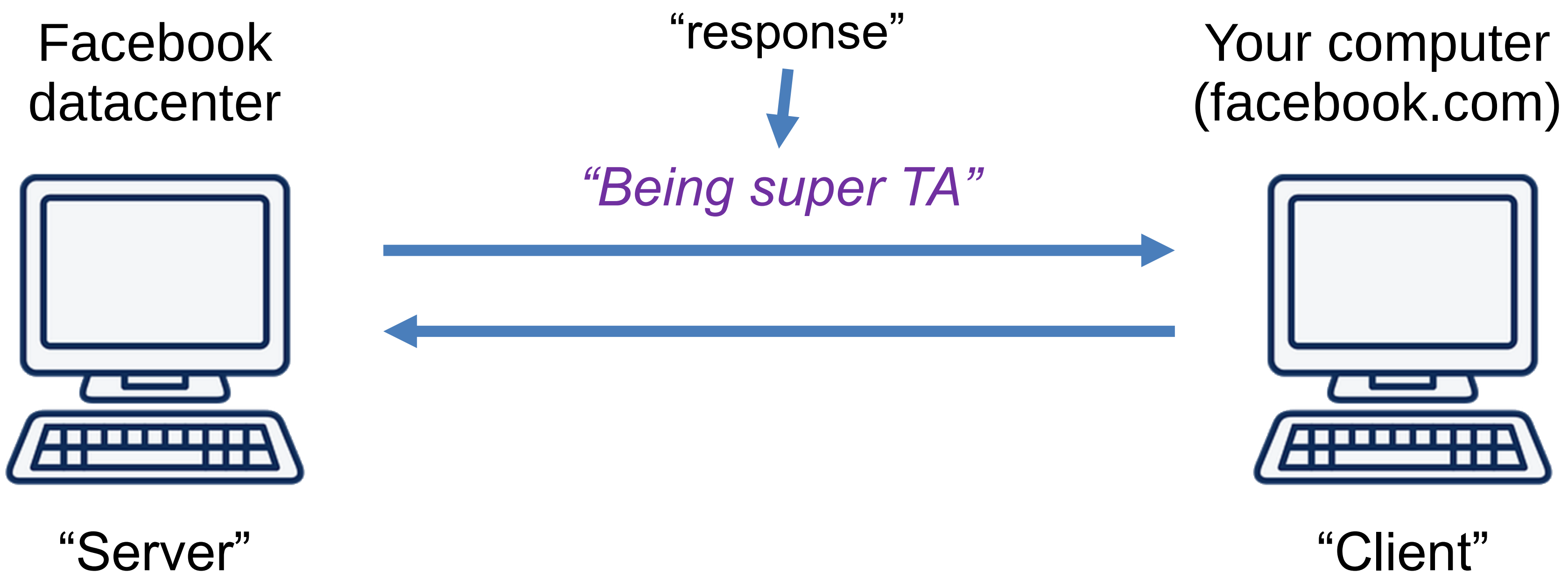
The internet is just many programs sending messages (as *Strings*)

Background: The Internet


Facebook datacenter
Your computer (facebook.com)
Get status for "Juliette Woodrow"
"Server"
"request"
"Client"

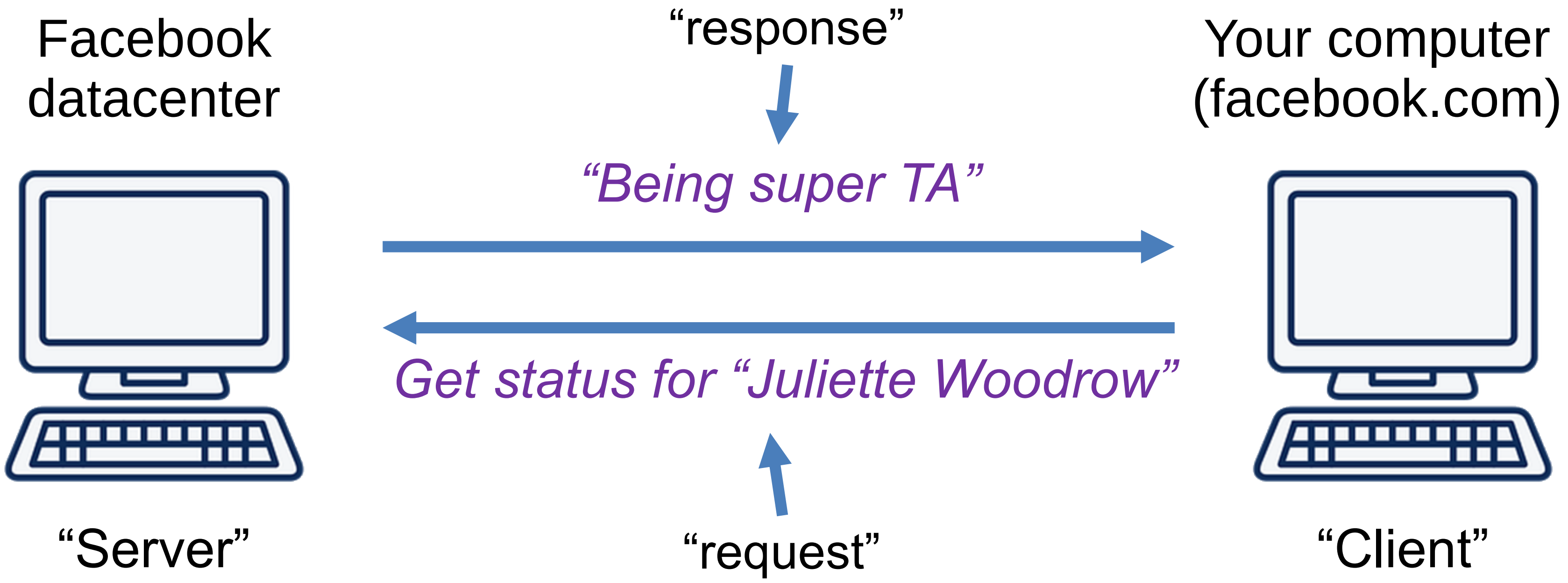
The internet is just many programs sending messages (as *Strings*)

Background: The Internet



The internet is just many programs sending messages (as *Strings*)

Background: The Internet

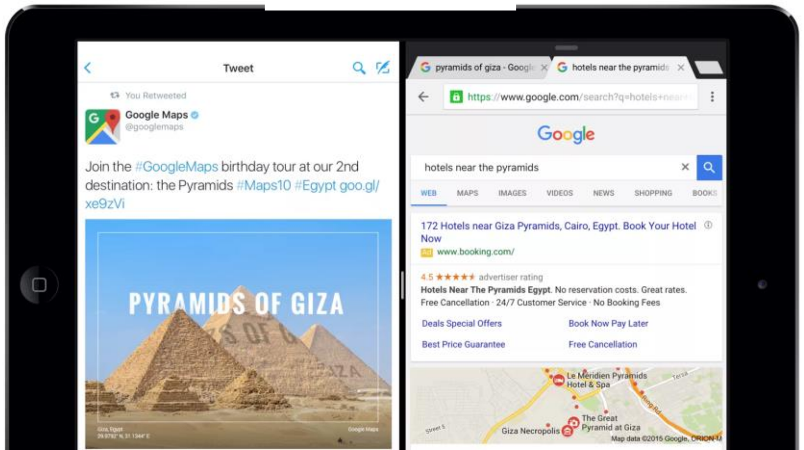


The internet is just many programs sending messages (as *Strings*)

There are (generally) two types of internet programs: Servers and Clients

Internet 101

Computers on the internet

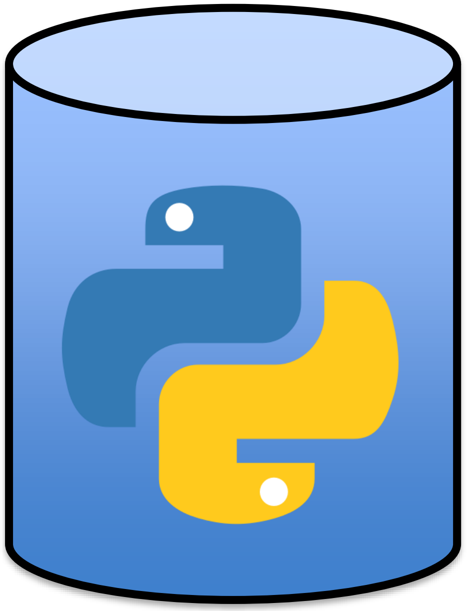


Join the #GoogleMaps birthday tour at our 2nd destination: the Pyramids #Maps10 #Egypt goo.gl/xe9zVi

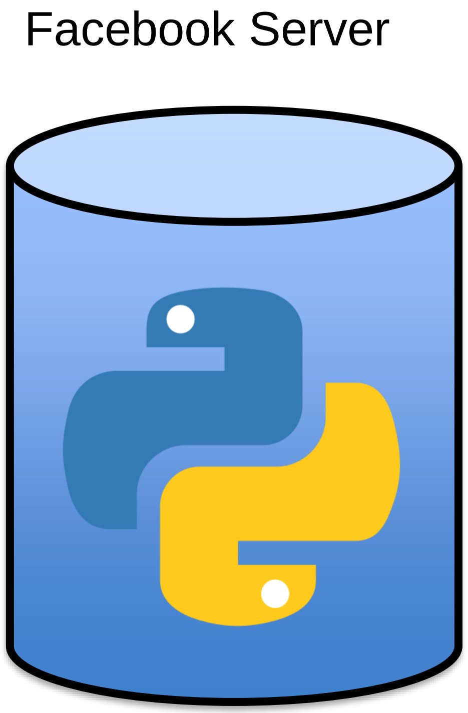
172 Hotels near Giza Pyramids, Cairo, Egypt. Book Your Hotel Now

www.booking.com/

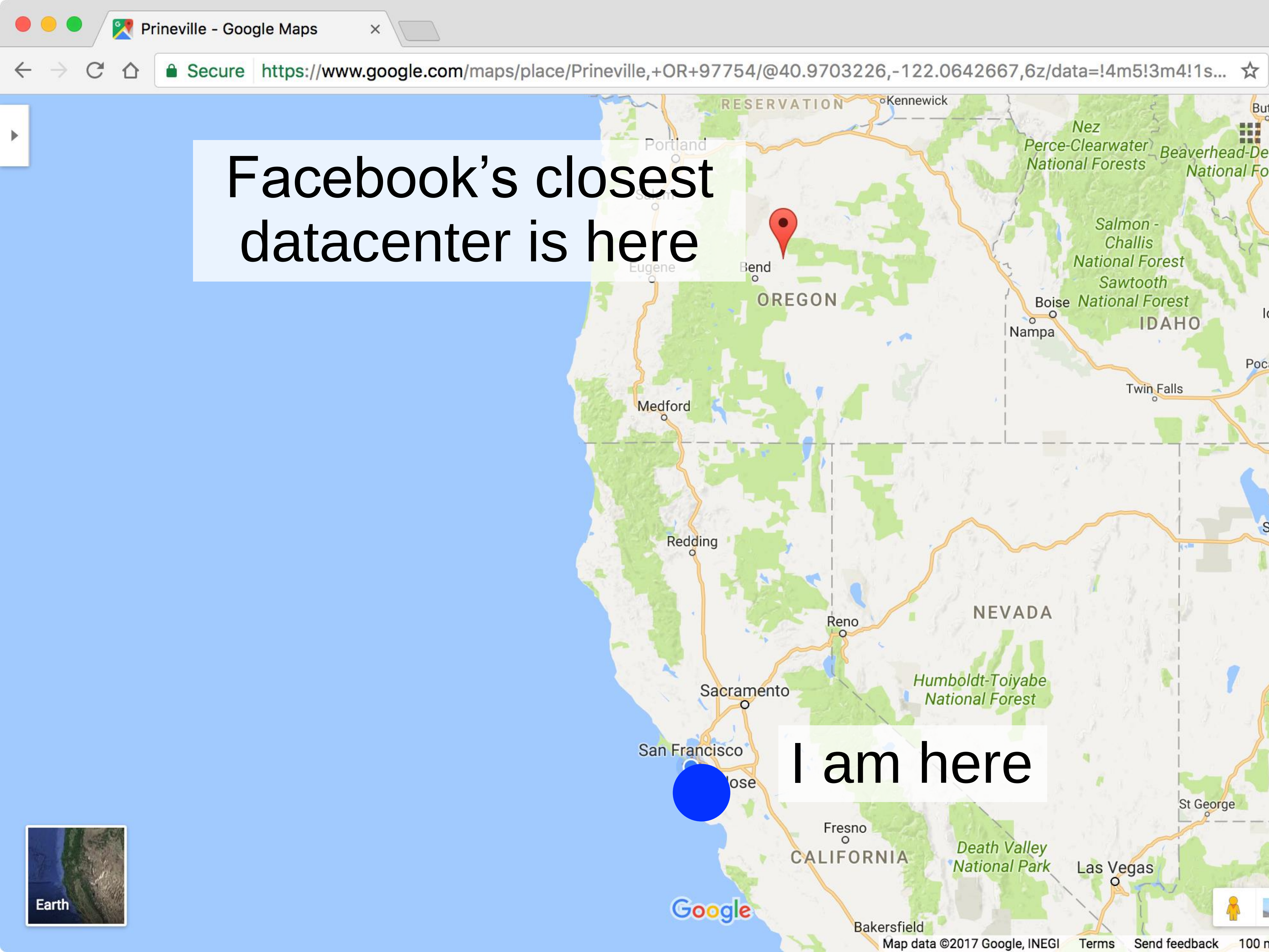
Hotels Near The Pyramids Egypt. No reservation costs. Great rates. Free Cancellation · 24/7 Customer Service · No Booking Fees



Servers are computers (running code)

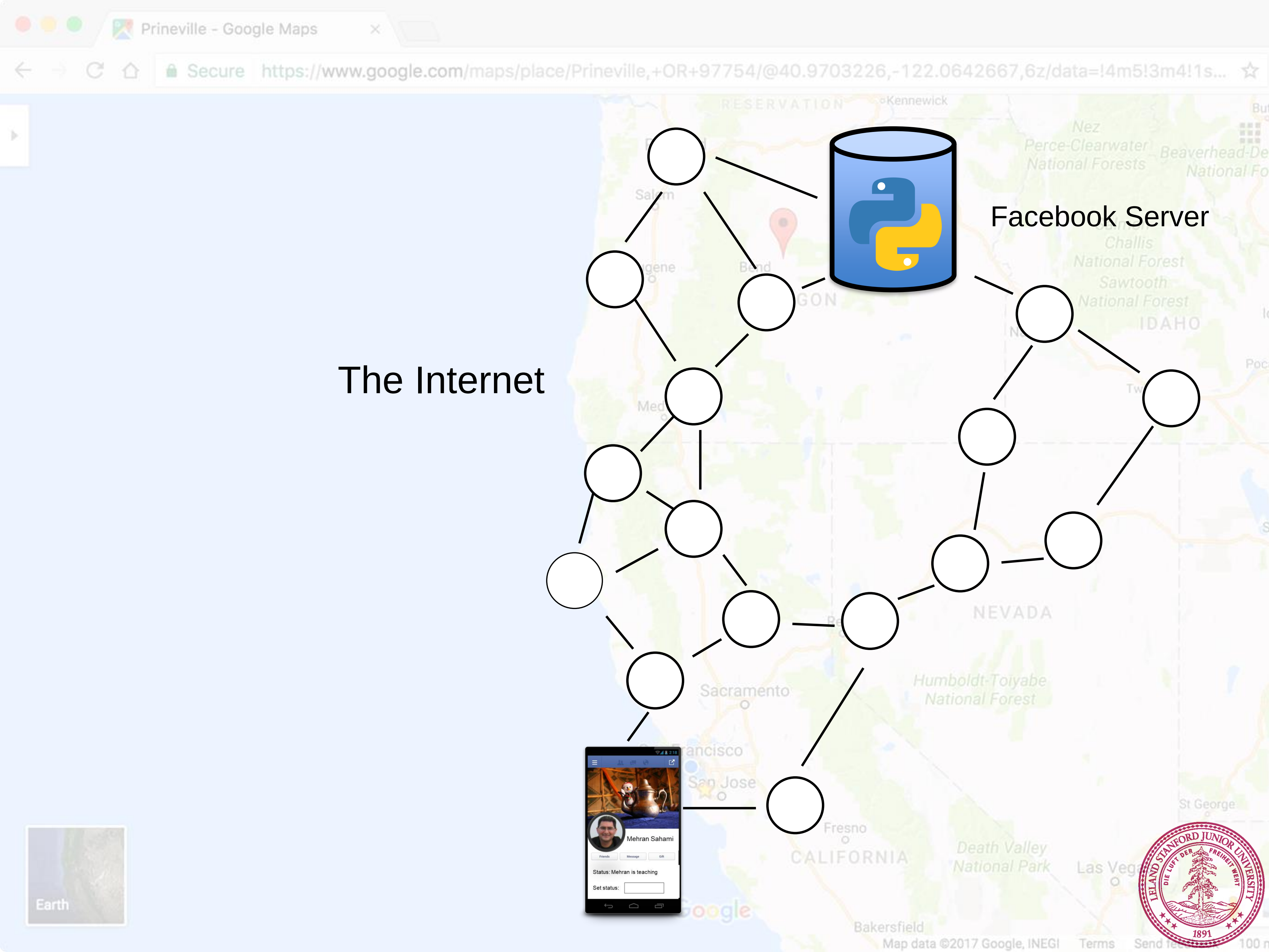


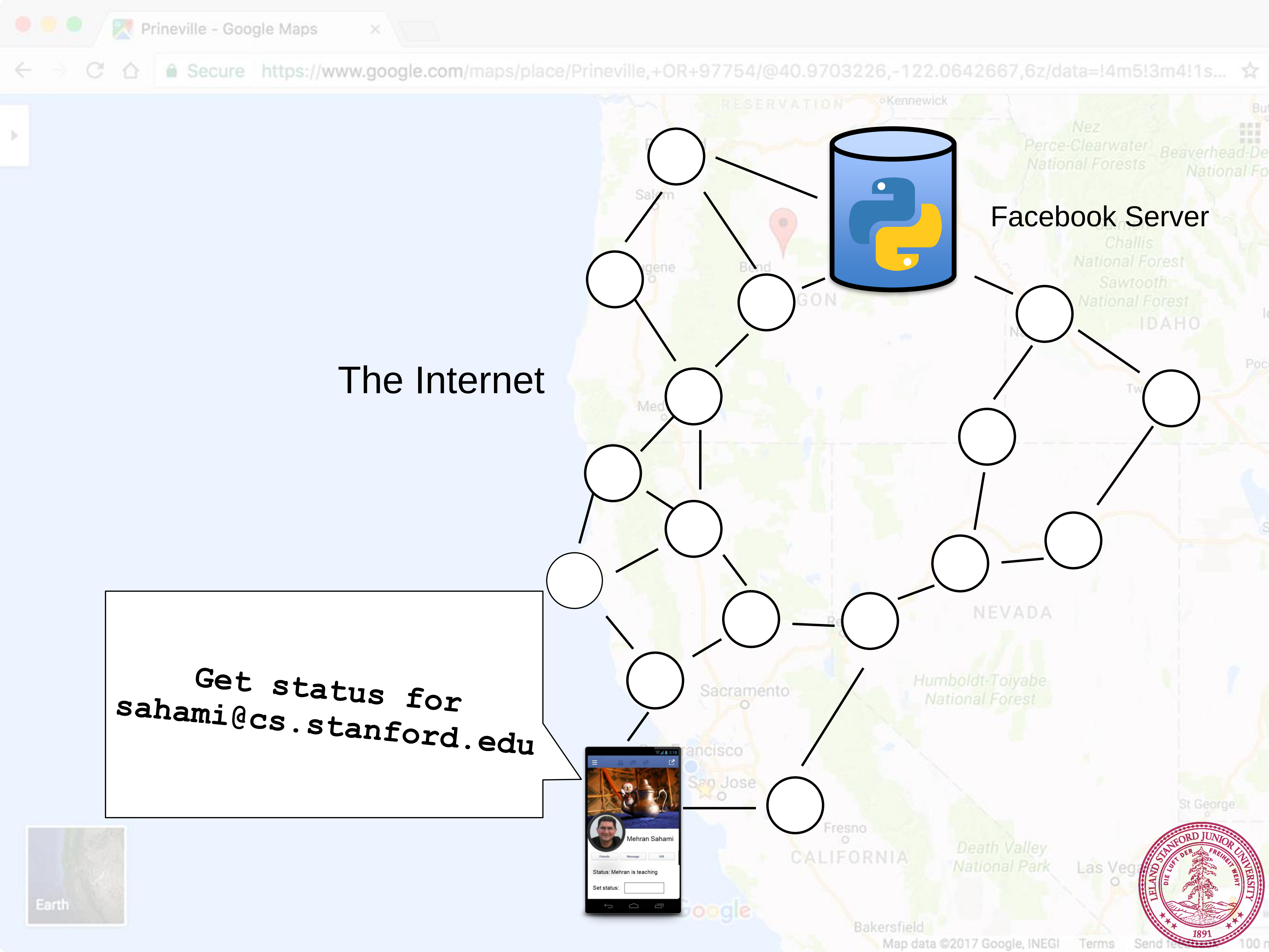
=

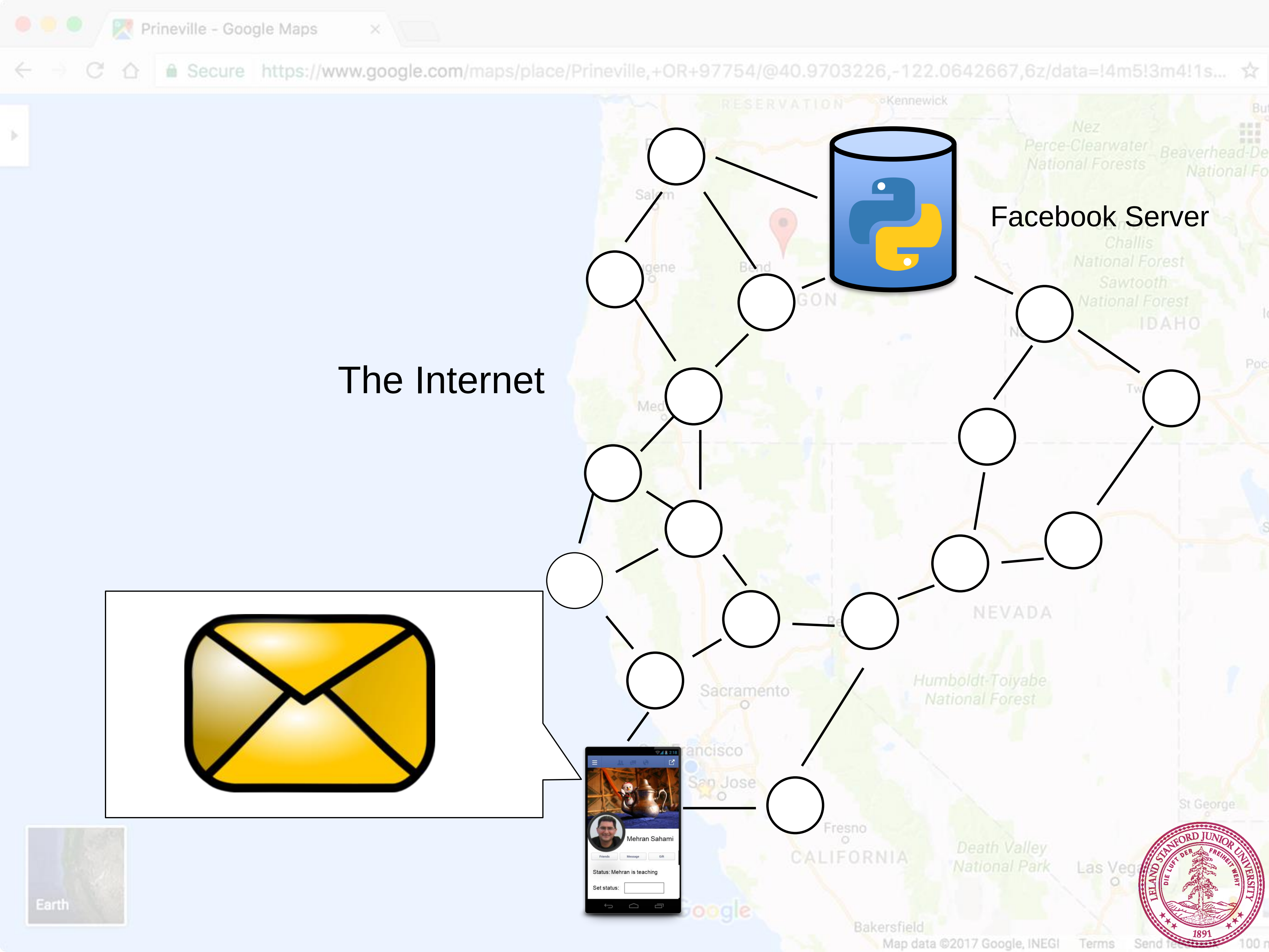


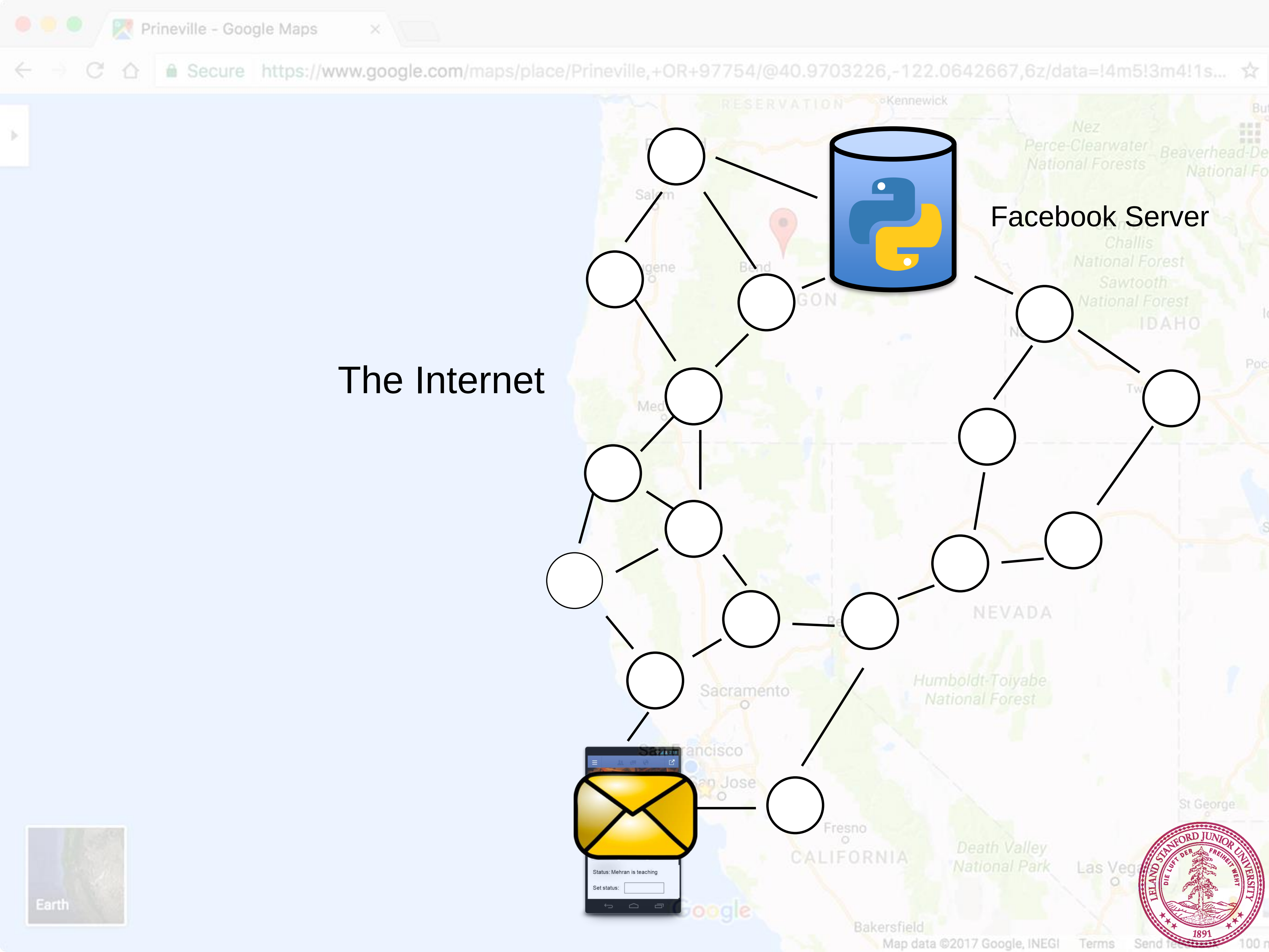
Facebook's closest datacenter is here

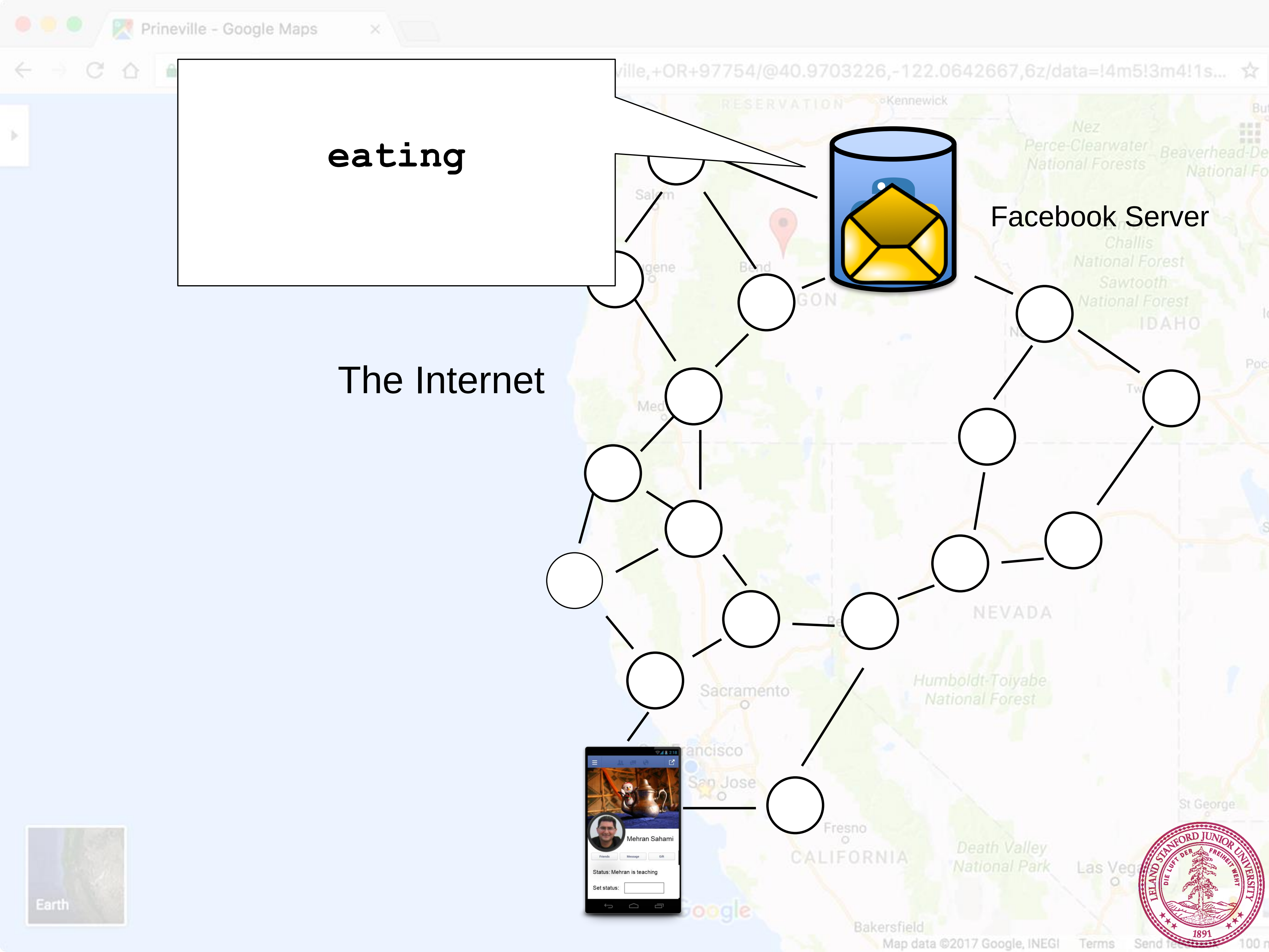
I am here

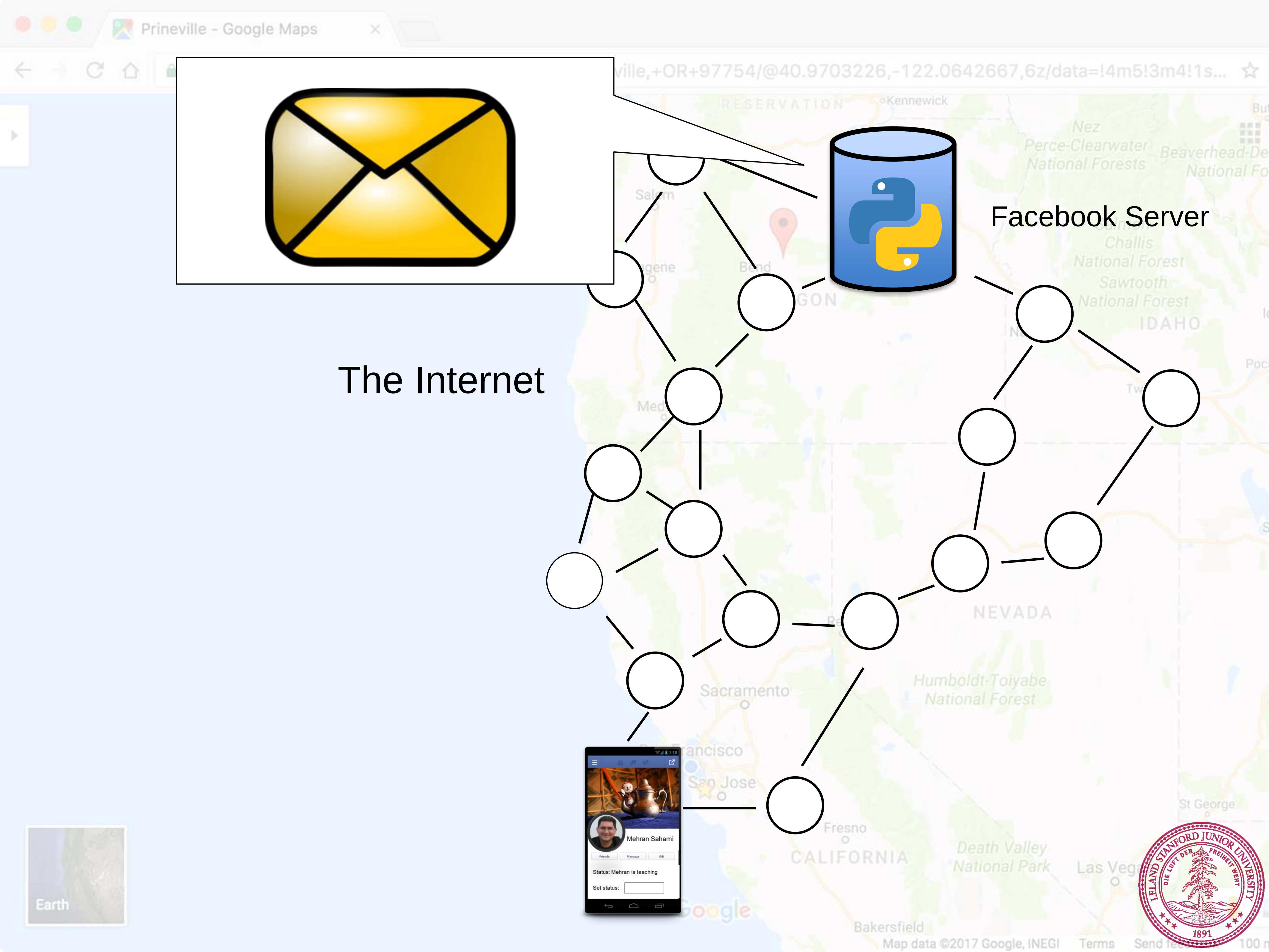

Facebook Server
The Internet
Mehran Sahami
Status: Mehran is teaching
Set status:

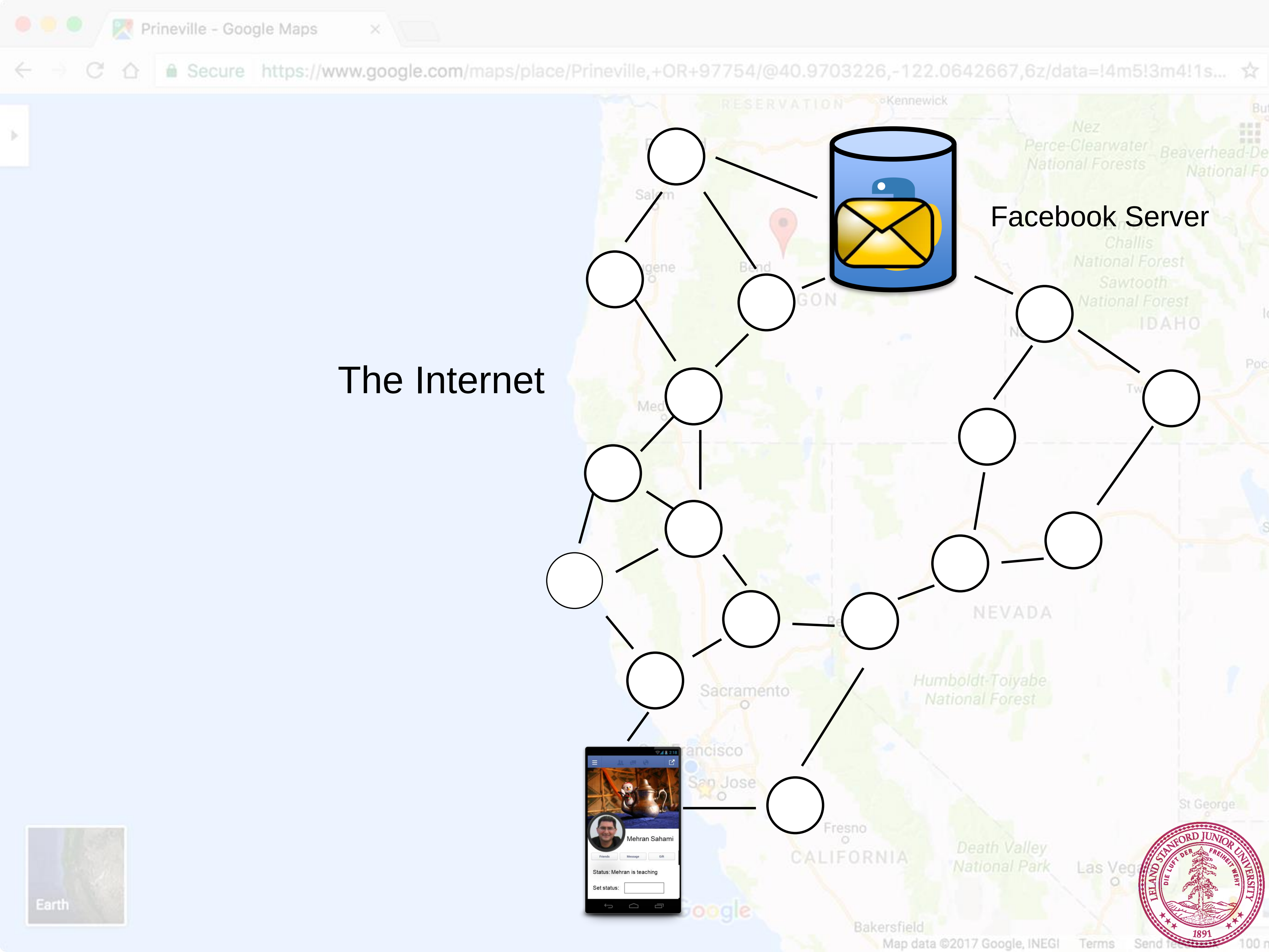

Prineville - Google Maps
https://www.google.com/maps/place/Prineville,+OR+97754/@40.9703226,-122.0642667,6z/data=!4m5!3m4!1s...
Facebook Server
The Internet
Get status for sahami@cs.stanford.edu
Mehran Sahami
Status: Mehran is teaching
Set status:


Facebook Server
The Internet
Mehran Sahami
Status: Mehran is teaching
Set status:


Facebook Server
The Internet
Status: Mehran is teaching
Set status:

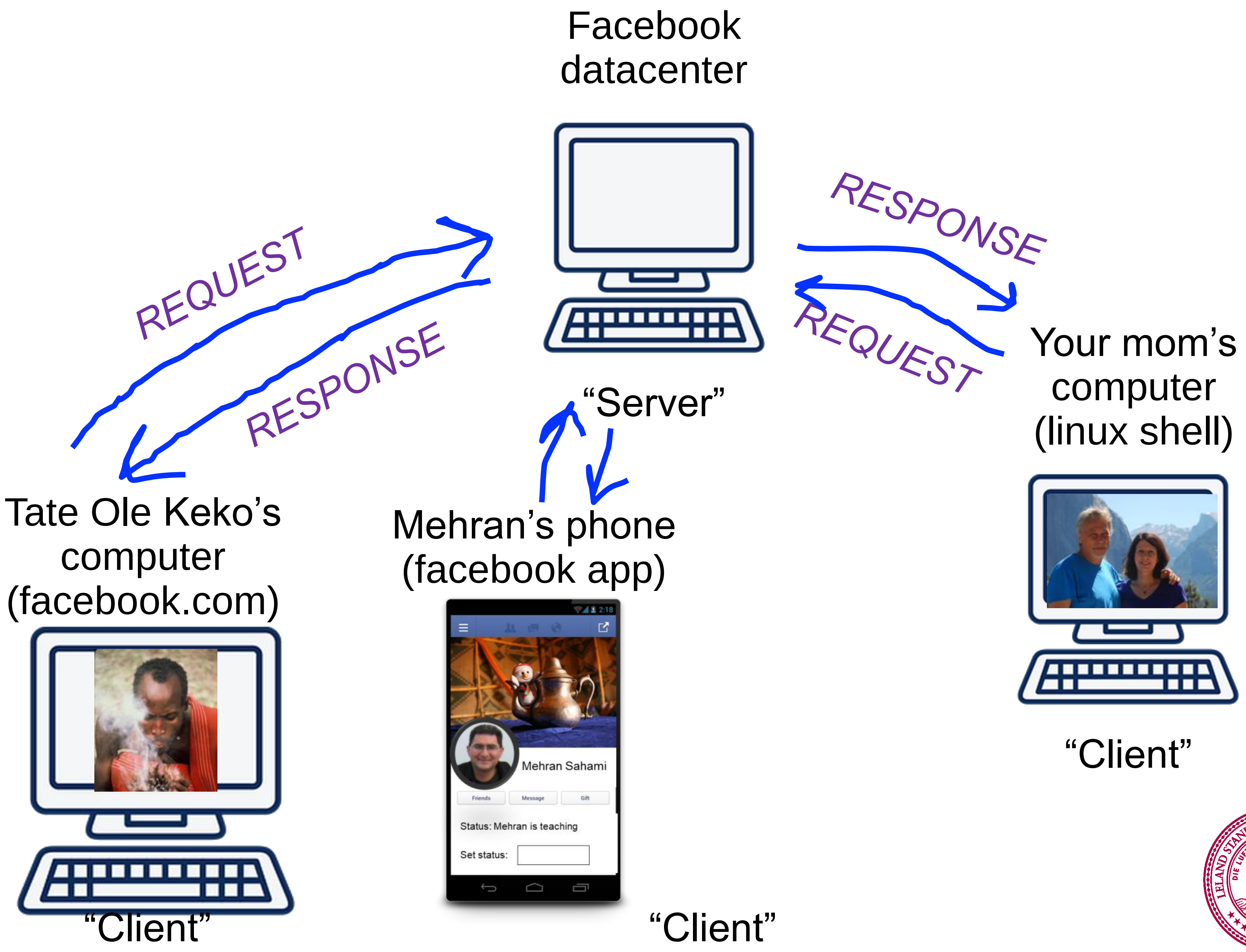

eating
Facebook Server
The Internet
Mehran Sahami
Status: Mehran is teaching
Set status:


Facebook Server
The Internet
Mehran Sahami
Status: Mehran is teaching
Set status:


Facebook Server
The Internet
Mehran Sahami
Status: Mehran is teaching
Set status:
Prineville - Google Maps

Many computers can connect to the same server

The Internet


Facebook datacenter
"Server"
REQUEST
RESPONSE
RESPONSE
REQUEST
Tate Ole Keko's computer (facebook.com)
"Client"
Mehran's phone (facebook app)
Mehran Sahami
Friends
Message
Gift
Status: Mehran is teaching
Set status:
"Client"
Your mom's computer (linux shell)
"Client"

Most of the Internet

Aka “the backend”

Aka “the frontend”

Server / Clients

Aka “the cloud”

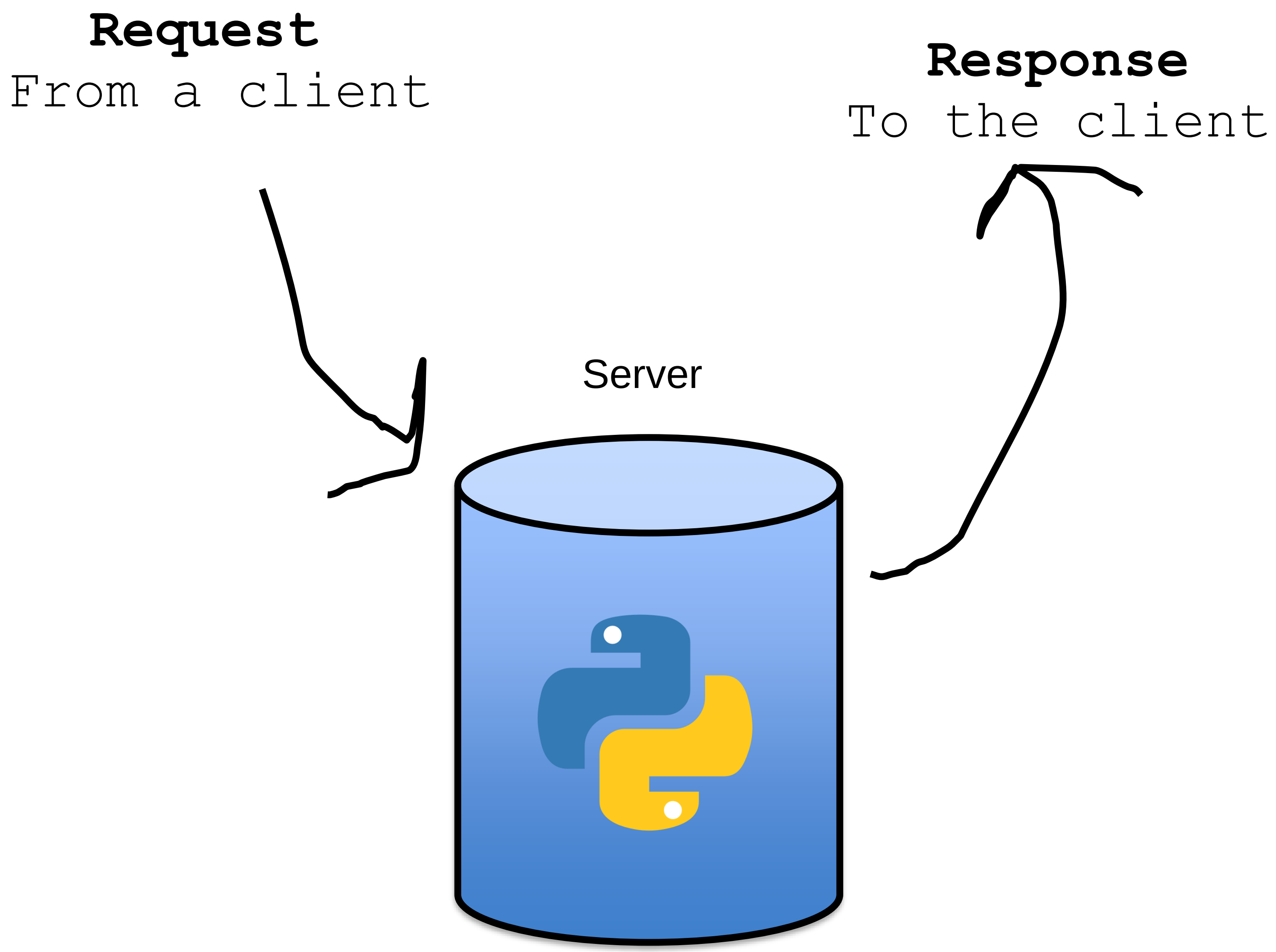
Aka “the brains”

Aka “the GUI”

Today, the server

A server's main job is to respond to requests

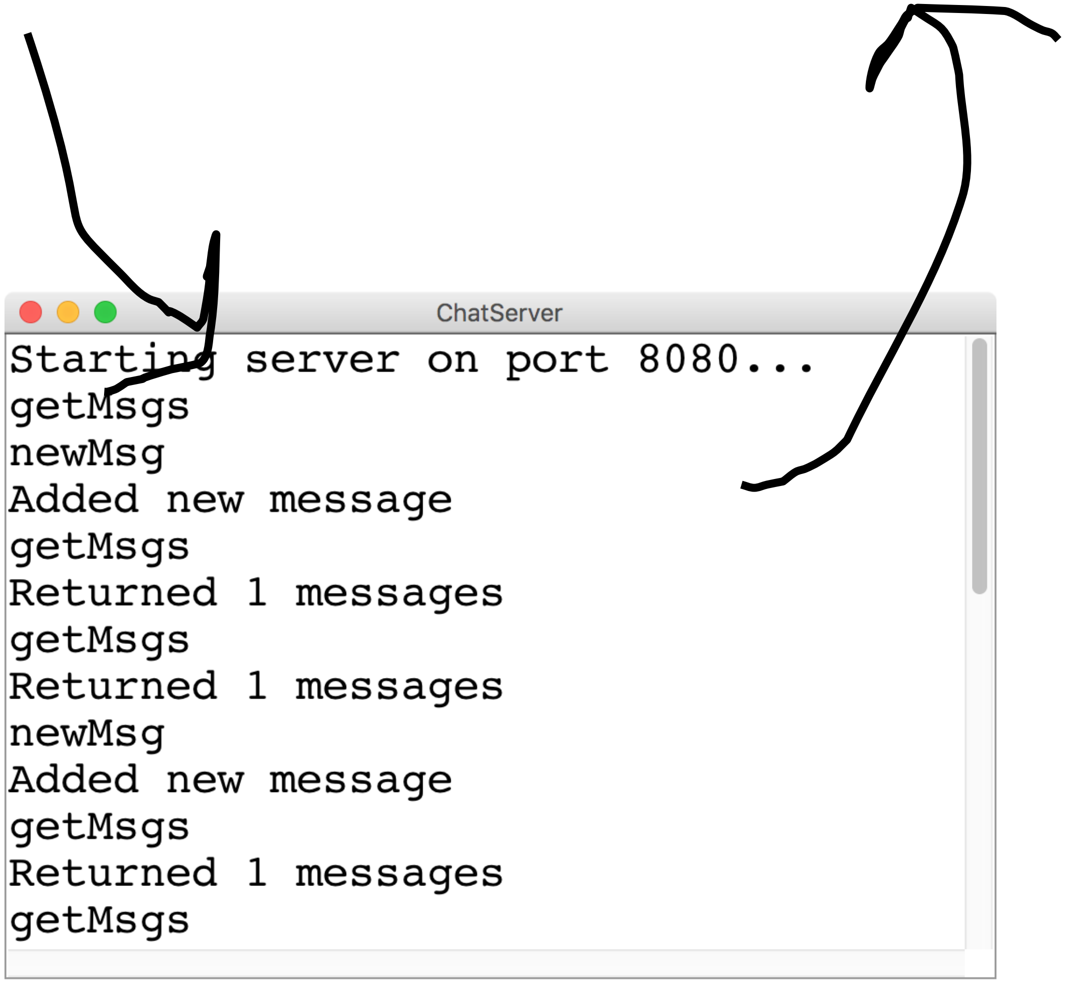
A Server's Simple Purpose



A Server's Simple Purpose

Request
someRequest

String
serverResponse



```
Starting server on port 8080...
getMsgs
newMsg
Added new message
getMsgs
Returned 1 messages
getMsgs
Returned 1 messages
newMsg
Added new message
getMsgs
Returned 1 messages
getMsgs
```

Servers on one slide

1

```
# handle server requests (must be in a class)
def handle_request(self, request):
    # return a string response!
```

2

```
# turn on the server
def main():
    # make an instance of your server class
    handler = MyServer()
    # start the server!
    SimpleServer.run_server(handler, 8000)
```

3

```
# enjoy
```

Servers on one slide

1

```
# handle server requests (must be in a class)
def handle_request(self, request):
    # return a string response!
```

2

```
# turn on the server
def main():
    # make an instance of your server class
    handler = HitServer()
    # start the server!
    run_server(handler, 8000)
```

3

```
# enjoy
```

Servers on one slide

1

```
# handle server requests (must be in a class)
def handle_request(self, request):
    # return a string response!
```

2

```
# turn on the server
def main():
    # make an instance of your server class
    handler = HitServer()
    # start the server!
    run_server(handler, 8000)
```

3

```
# enjoy
```

Servers on one slide

1

```
# handle server requests (must be in a class)
def handle_request(self, request):
    # return a string response!
```

2

```
# turn on the server
def main():
    # make an instance of your server class
    handler = HitServer()
    # start the server!
    run_server(handler, 8000)
```

3

```
# enjoy
```

Servers on one slide

1

```
# handle server requests (must be in a class)
def handle_request(self, request):
    # return a string response!
```

2

```
# turn on the server
def main():
    # make an instance of your server class
    handler = HitServer()
    # start the server!
    run_server(handler, 8000)
```

3

```
# enjoy
```

Servers on one slide

1

```
# handle server requests (must be in a class)
def handle_request(self, request):
    # return a string response!
```

2

```
# turn on the server
def main():
    # make an instance of your server class
    handler = MyServer()
    # start the server!
    run_server(handler, 8000)
```

3

```
# enjoy
```

Servers on one slide

1

```
# handle server requests (must be in a class)
def handle_request(self, request):
    # return a string response!
```

2

```
# turn on the server
def main():
    # make an instance of your server class
    handler = MyServer()
    # start the server!
    SimpleServer.run_server(handler, 8000)
```

3

```
# enjoy
```

Servers on one slide

1

```
# handle server requests (must be in a class)
def handle_request(self, request):
    # return a string response!
```

2

```
# turn on the server
def main():
    # make an instance of your server class
    handler = MyServer()
    # start the server!
    SimpleServer.run_server(handler, 8000)
```

3

```
# enjoy
```

What is a Port?

Servers on one slide

1

```
# handle server requests (must be in a class)
def handle_request(self, request):
    # return a string response!
```

2

```
# turn on the server
def main():
    # make an instance of your server class
    handler = MyServer()
    # start the server!
    SimpleServer.run_server(handler, 8000)
```

3

```
# enjoy
```

Servers on one slide

1

```
# handle server requests (must be in a class)
def handle_request(self, request):
    # return a string response!
```

2

```
# turn on the server
def main():
    # make an instance of your server class
    handler = HitServer()
    # start the server!
    run_server(handler, 8000)
```

3

```
# enjoy
```

Servers on one slide

1

```
# handle server requests (must be in a class)
def handle_request(self, request):
    # return a string response!
```

2

```
# turn on the server
def main():
    # make an instance of your server class
    handler = HitServer()
    # start the server!
    run_server(handler, 8000)
```

3

```
# enjoy
```

What is a Request?

```
/* Request has a command */
command (type is string)

/* Request has parameters */
params (type is dict)
```

```
// methods that the server calls on requests
request.command
request.params
```

```python
class Request:
    """
    Request class packages the key information from an internet request.
    An internet request has both a command and a dictionary of
    parameters.  This class defines a special function __str__ which
    means if you have an instance of a request you can put it in a
    print function.
    """
    def __init__(self, request_command, request_params):
        # Every request has a command (string)
        self.command = request_command
        # Every request has params (dictionary). Can be empty: {}.
        self.params = request_params

    def get_params(self):
        # A 'getter' method to get the params
        return self.params

    def get_command(self):
        # A 'getter' method to get the command
        return self.command

    def __str__(self):
        # A special method which allows you to 'print' a request
        # as a string.
        return str(self.__dict__)
```

First Server Example!

```
import SimpleServer


# We define a class to handle server requests
class MyFirstServer:
    def __init__(self):
        pass

    # This is the server request callback function.
    # You can't change its header.
    def handle_request(self, request):
        print(request)
        return 'Happy Monday, wonderful cs106a!!!'


def main():
    # Make the server handler
    handler = MyFirstServer()
    # Start the server to handle internet requests at specified port
    SimpleServer.run_server(handler, 8000)
```

Who makes requests?

Who makes requests?

Other programs can send requests!

```
response = requests.get('https://xkcd.com/353/')
```

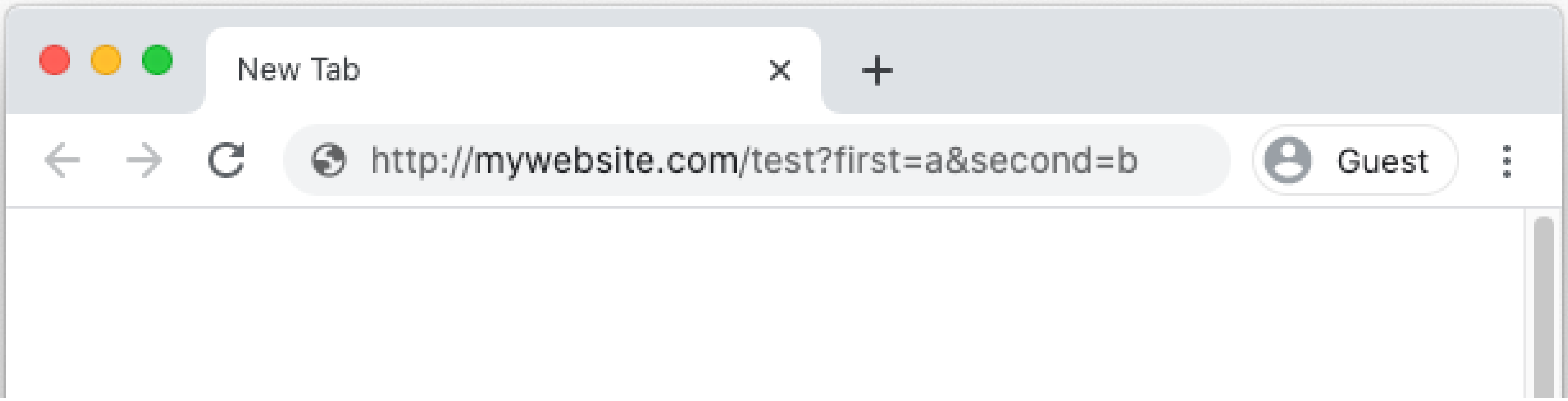
Who makes requests?

Other programs can send requests!

```
response = requests.get('https://xkcd.com/353/')
```

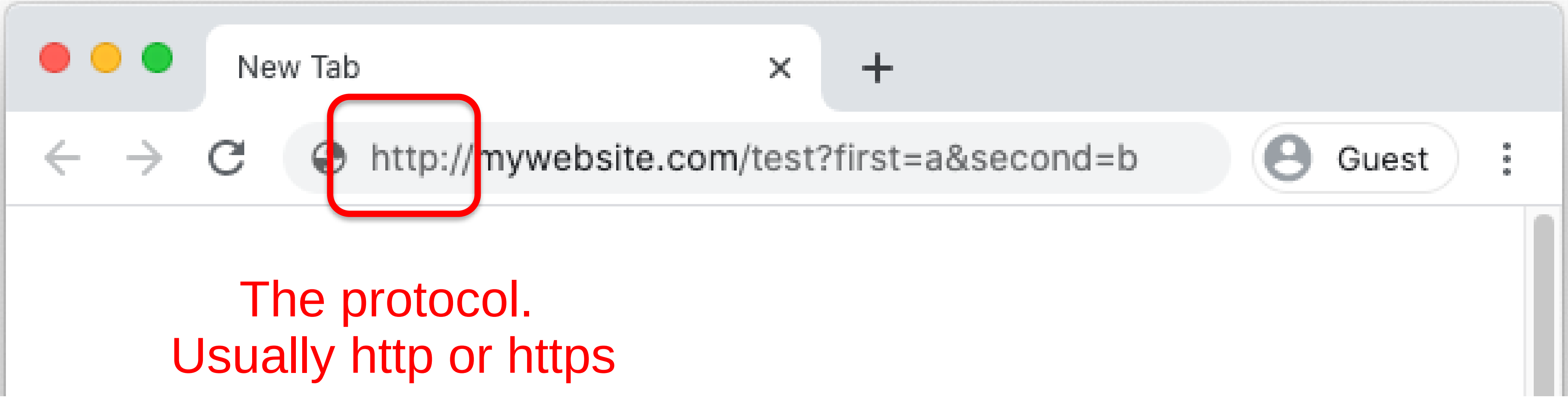
Web browsers can send requests!

Anatomy of a Browser Request



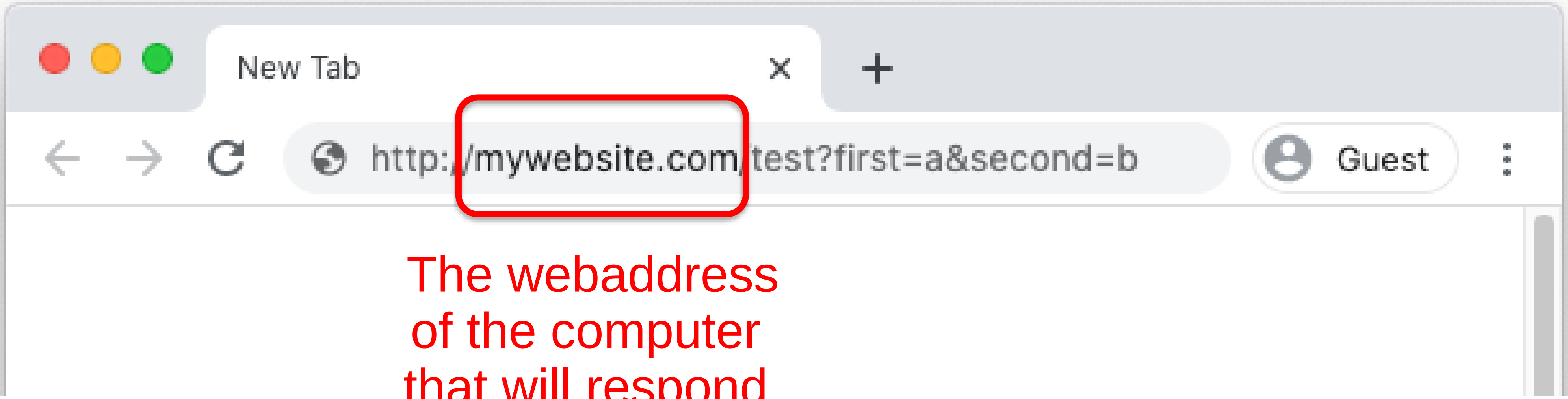
http://mywebsite.com/test?first=a&second=b

Anatomy of a Browser Request



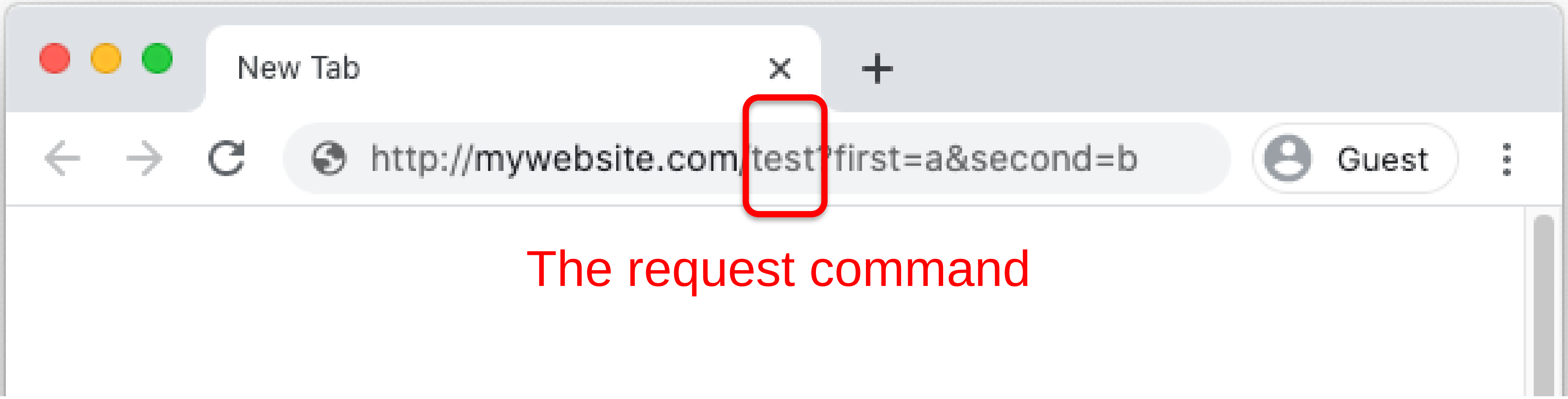
The protocol.
Usually http or https

Anatomy of a Browser Request



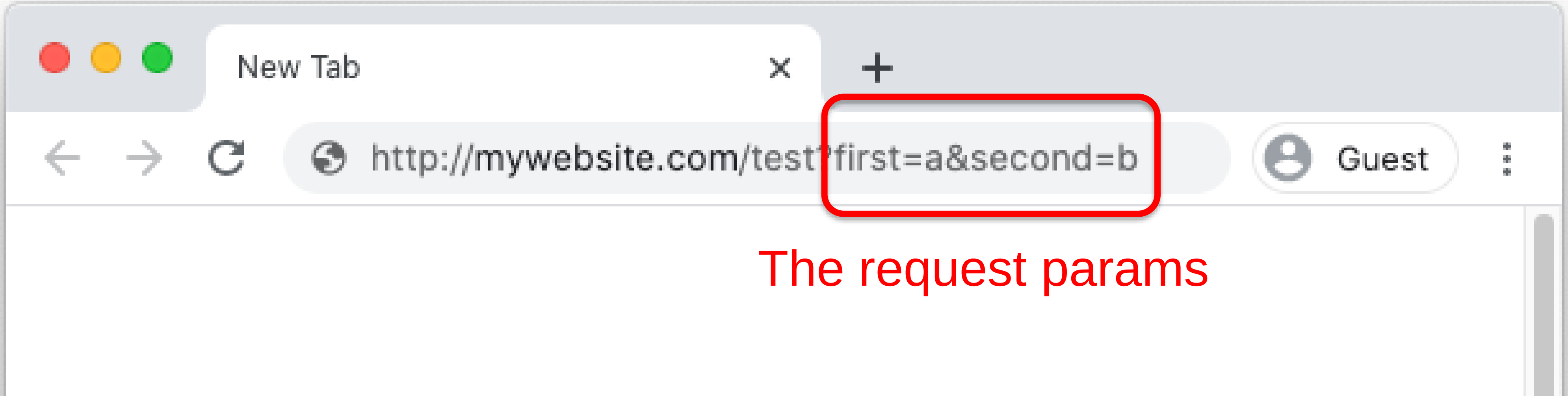
The webaddress
of the computer
that will respond
to the request

Anatomy of a Browser Request



The request command

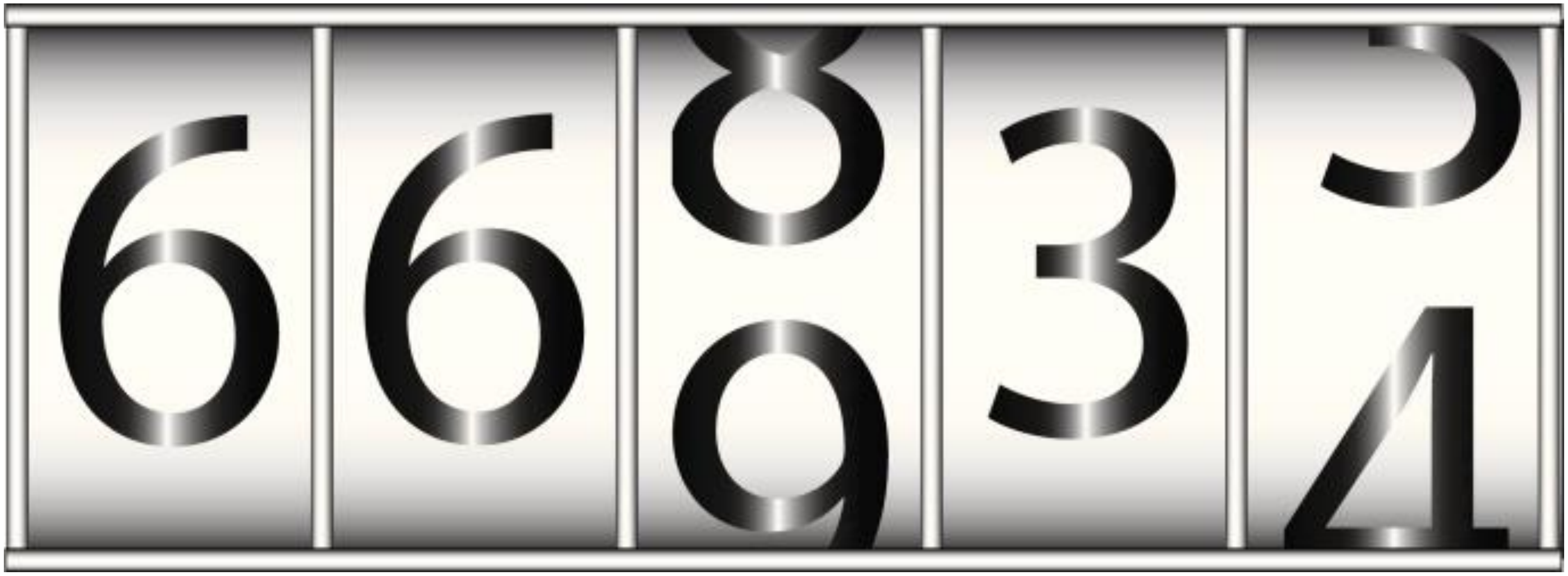
Anatomy of a Browser Request



http://mywebsite.com/test?first=a&second=b

The request params

Hit Counter



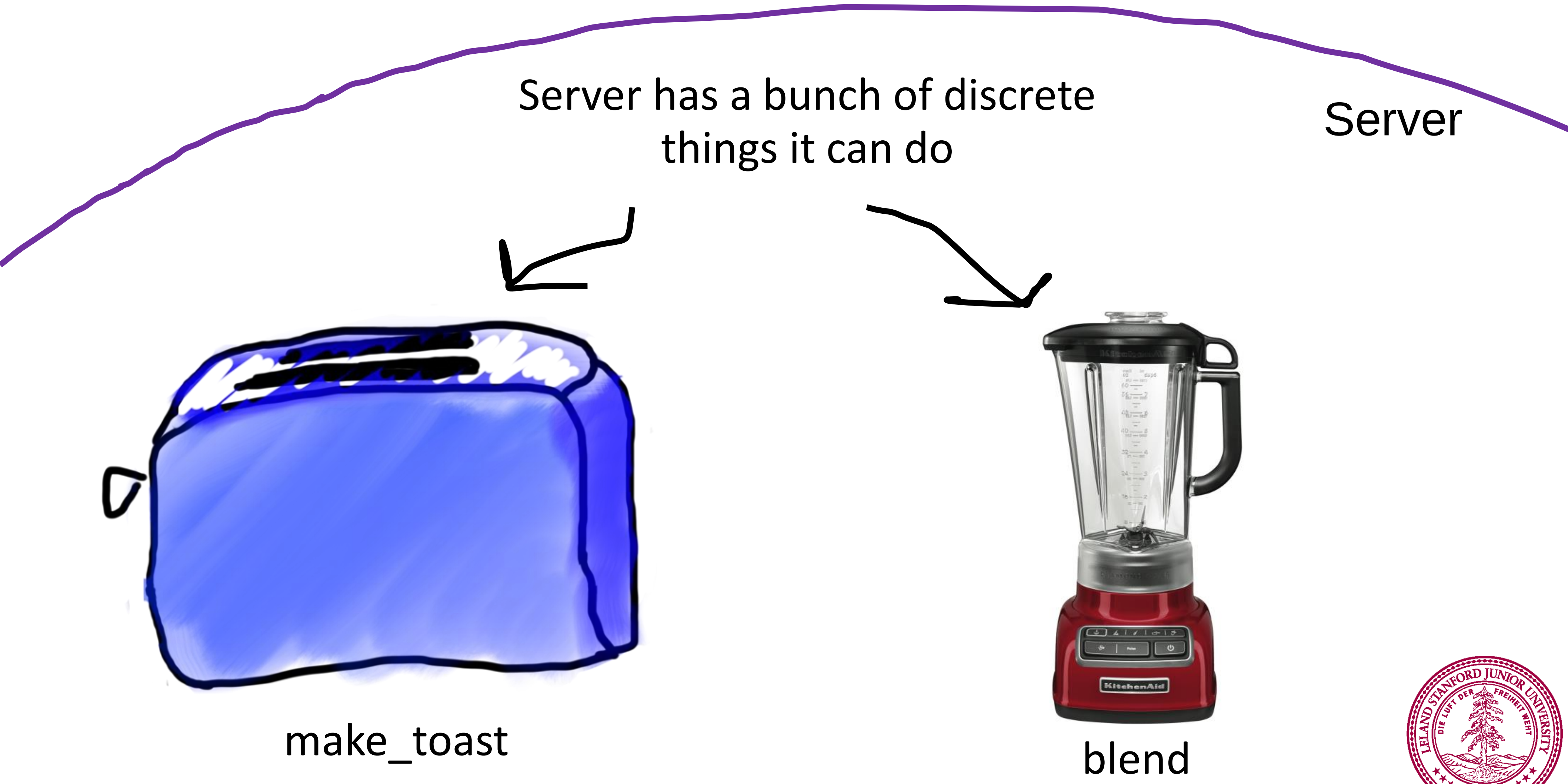
Recall Requests

```
/* Request has a command */
command (string)

/* Request has parameters */
params (dict)
```

```
// methods that the server calls on requests
request.command
request.params
```

Requests are like Remote Method Calls

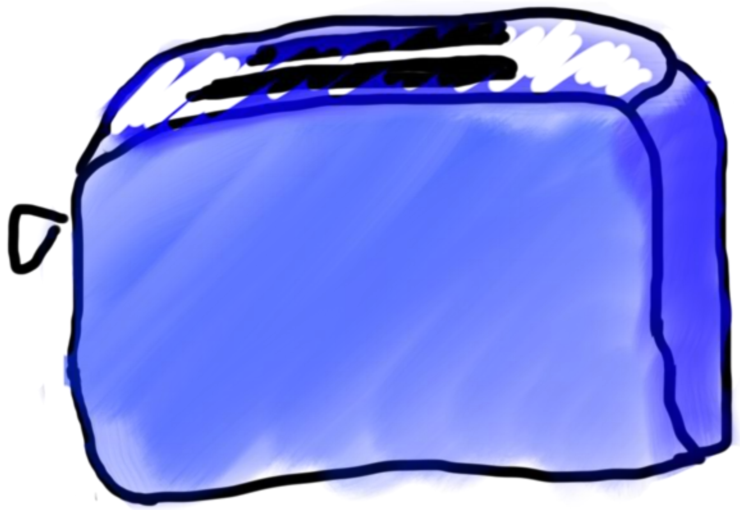


Server has a bunch of discrete things it can do

Requests are like Remote Method Calls

Server has a bunch of discrete things it can do

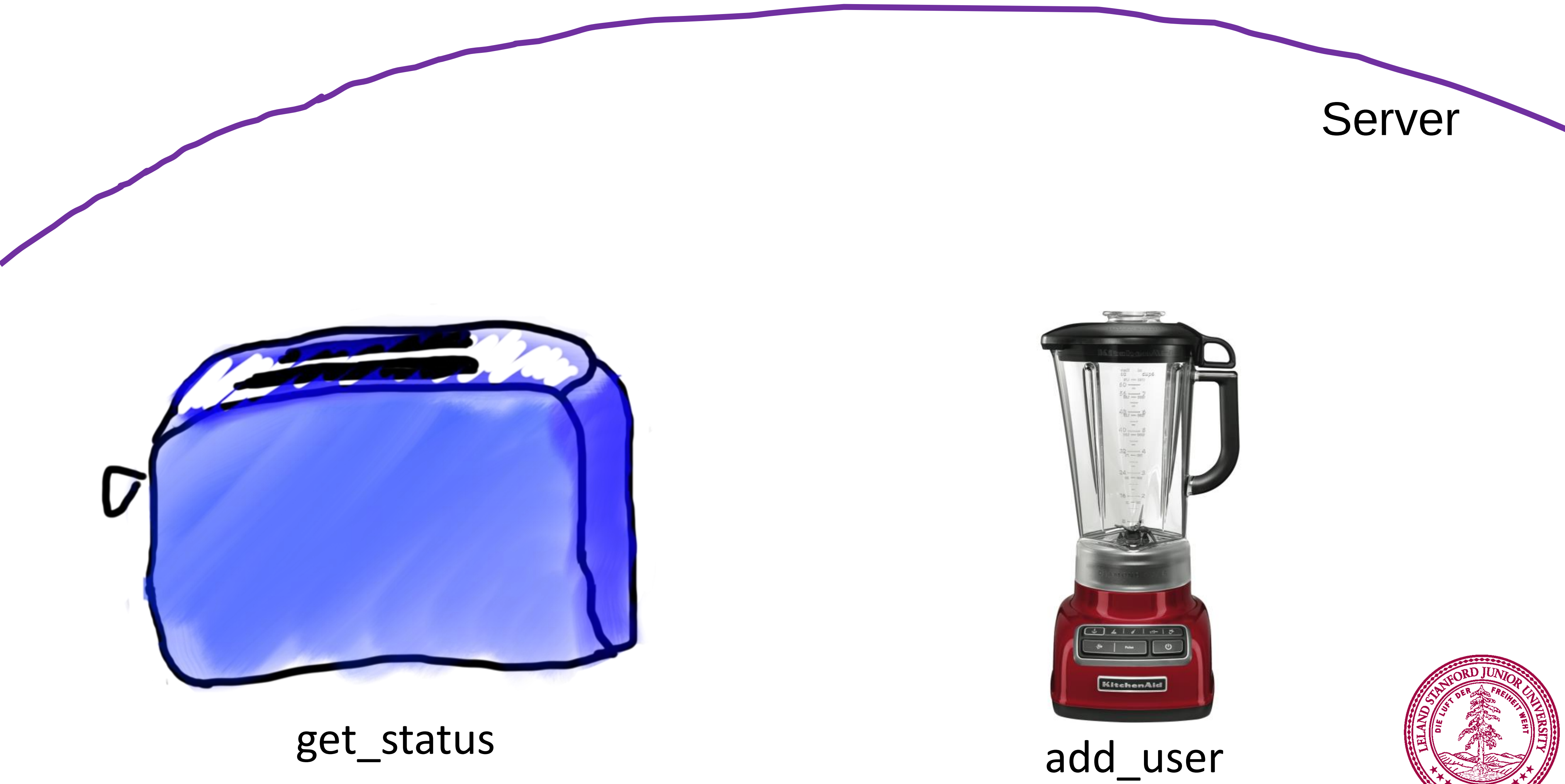
Server



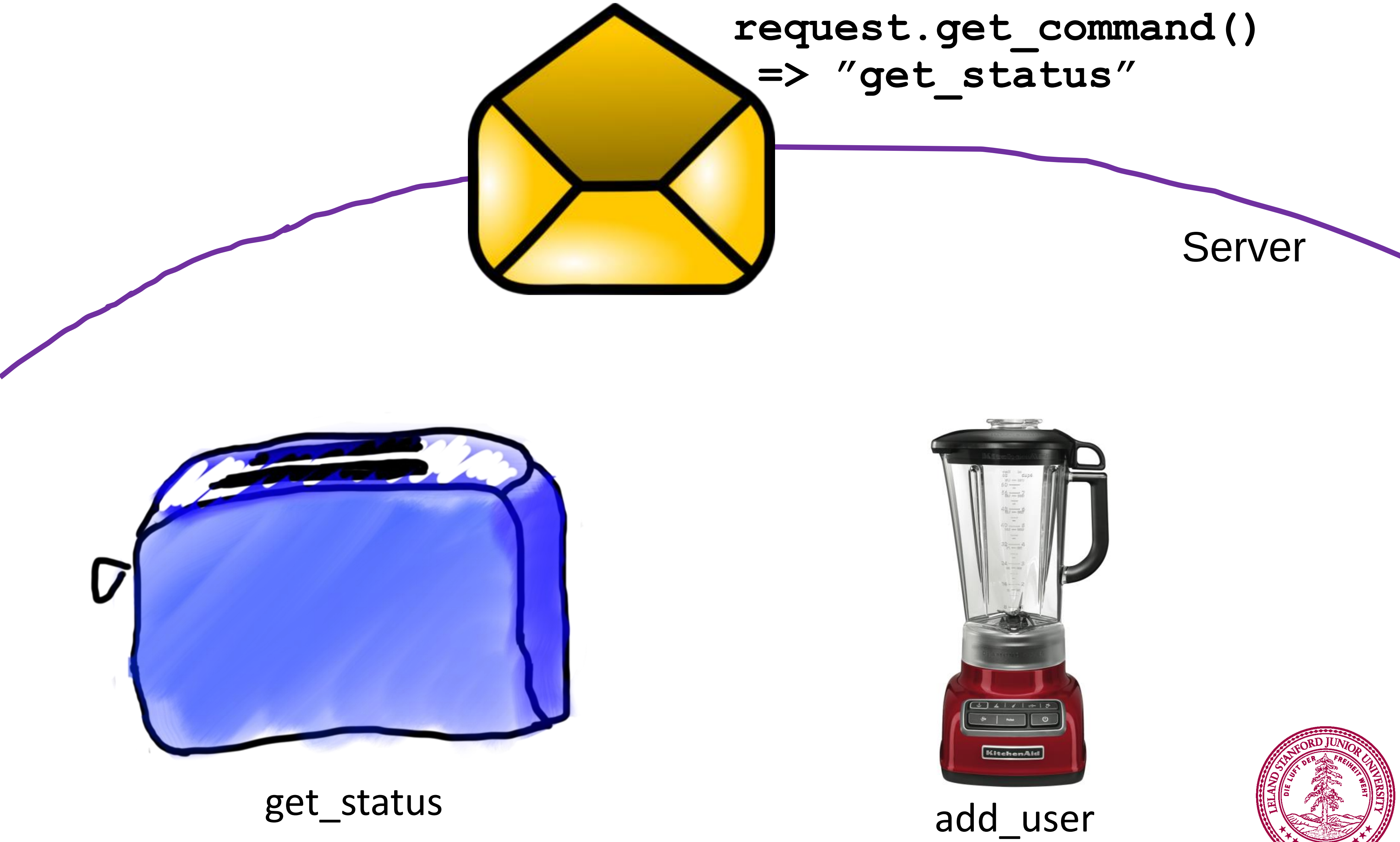
get_status

add_user

Requests are like Remote Method Calls

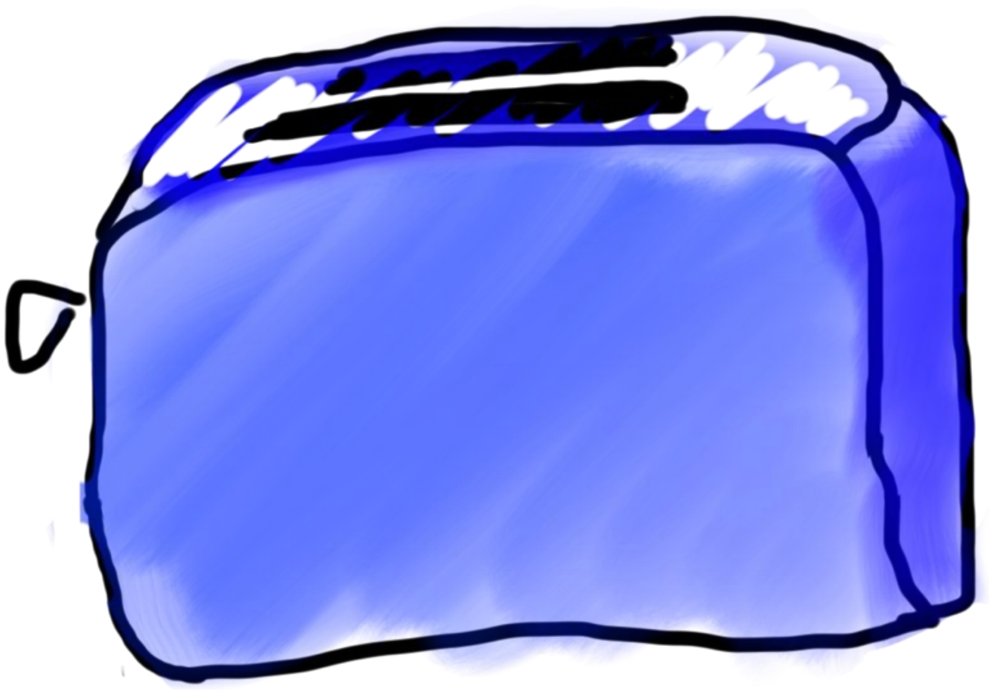


Requests are like Remote Method Calls



```
request.get_command()
 => "get_status"
```

Requests are like Remote Method Calls

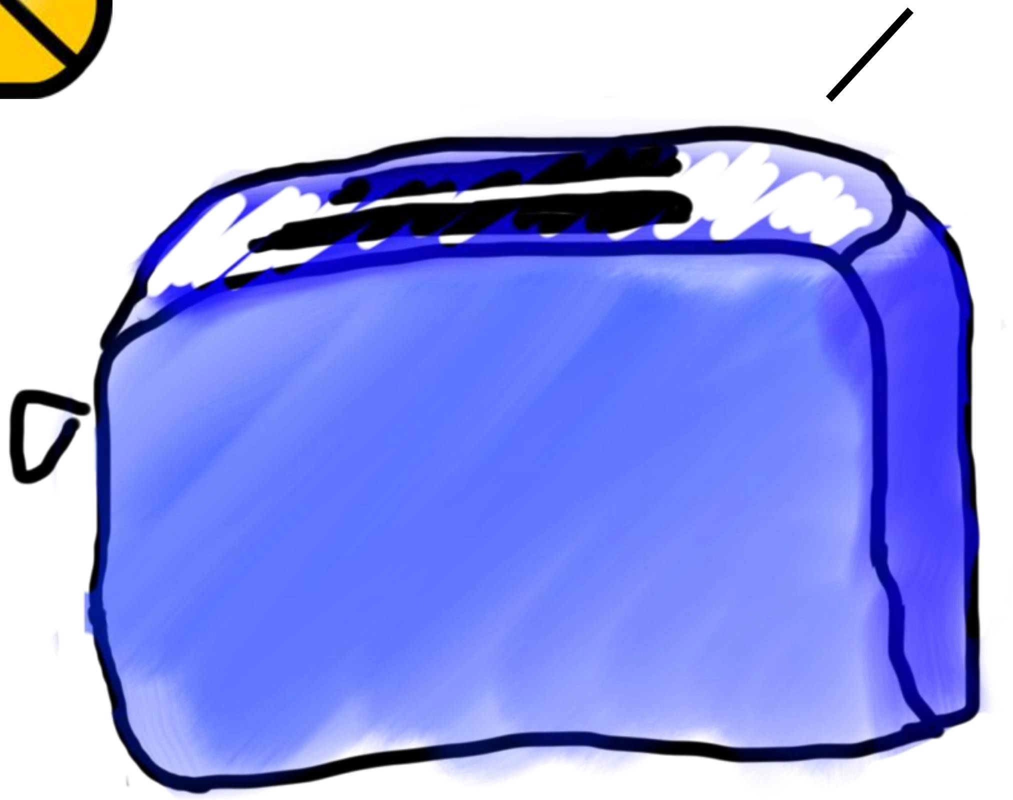

To make toast, I need a parameter which is the kind of bread
get_status

Requests are like Remote Method Calls

I was given a parameter!

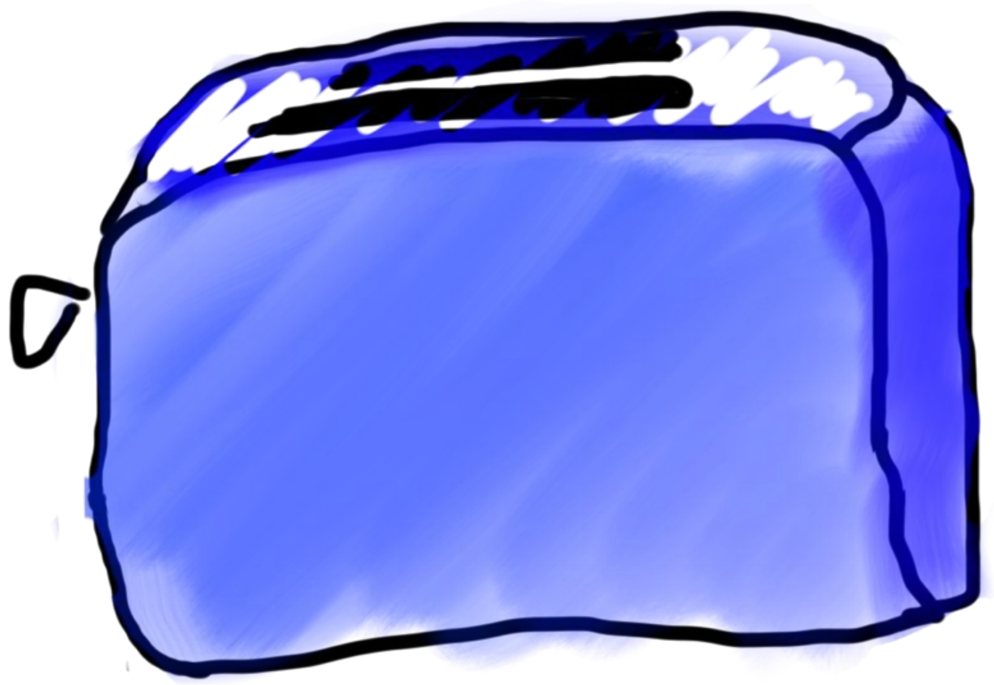
get_status

Requests are like Remote Method Calls



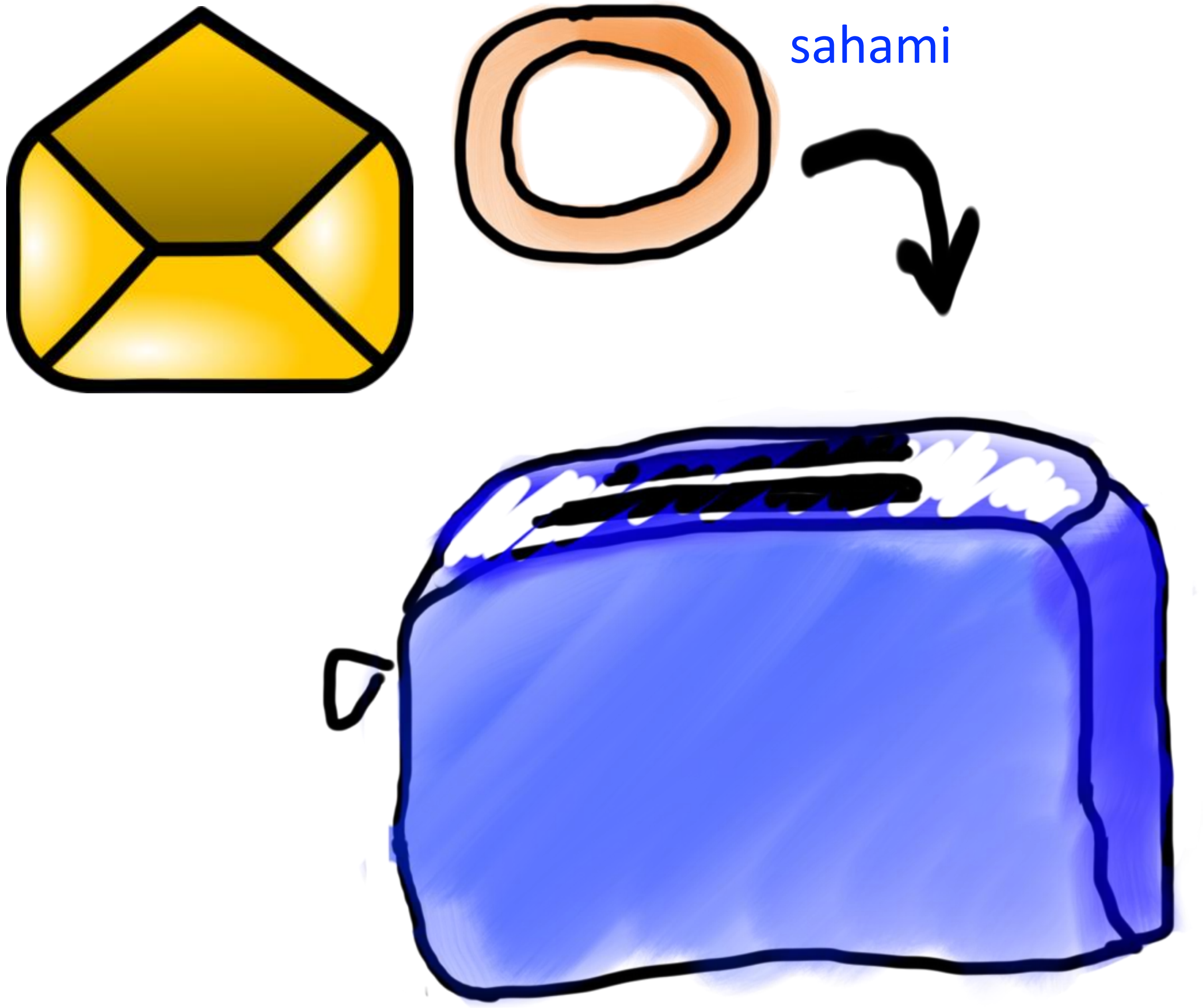
get_status

Requests are like Remote Method Calls



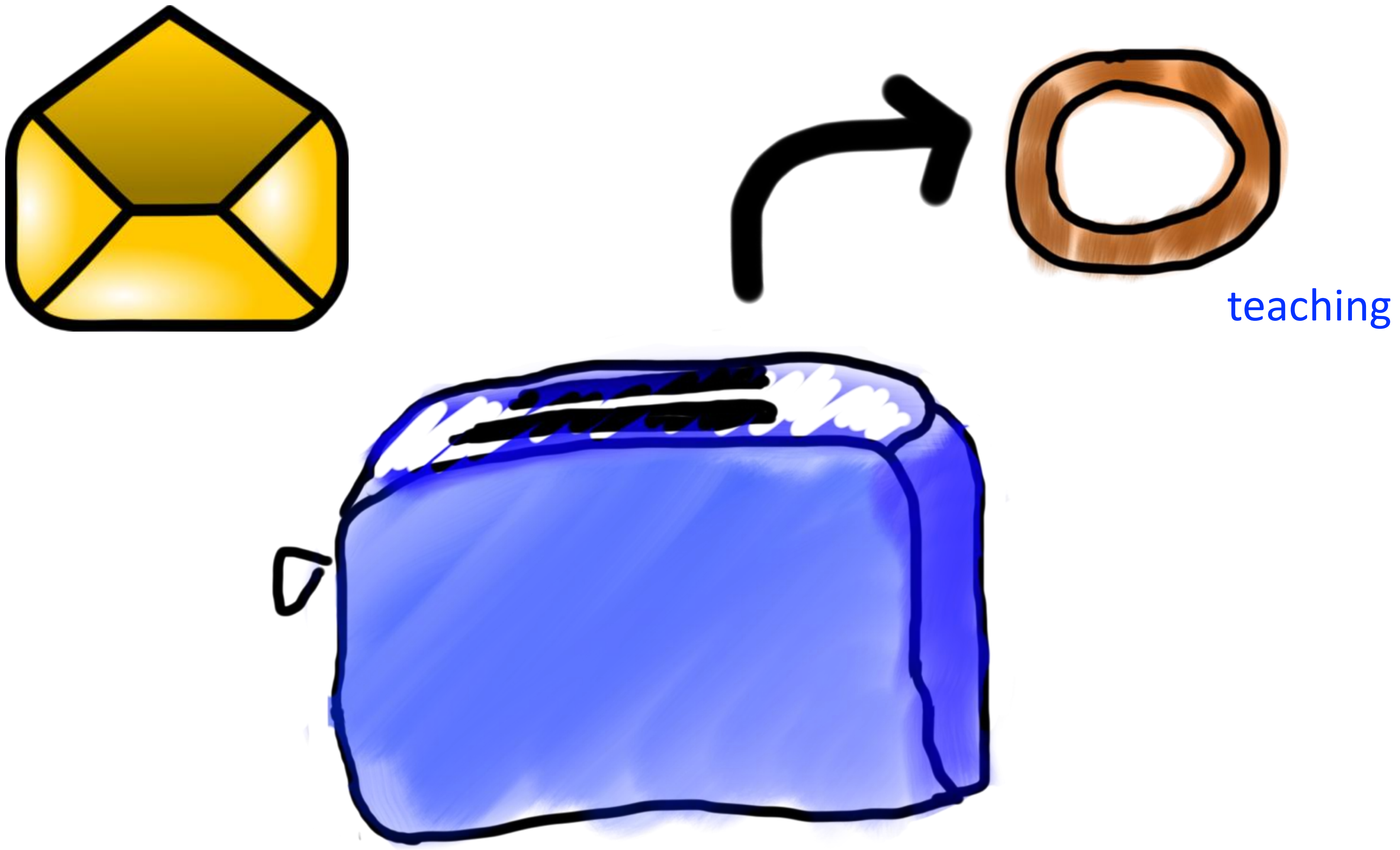
get_status

Requests are like Remote Method Calls

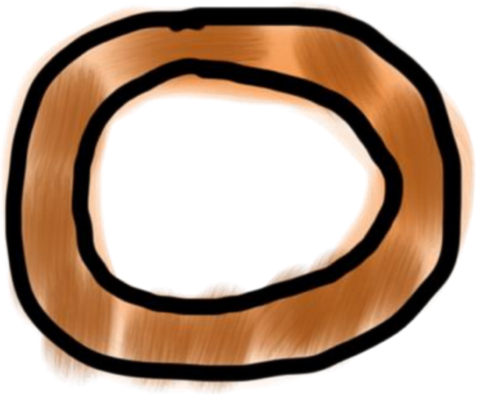


get_status

Requests are like Remote Method Calls

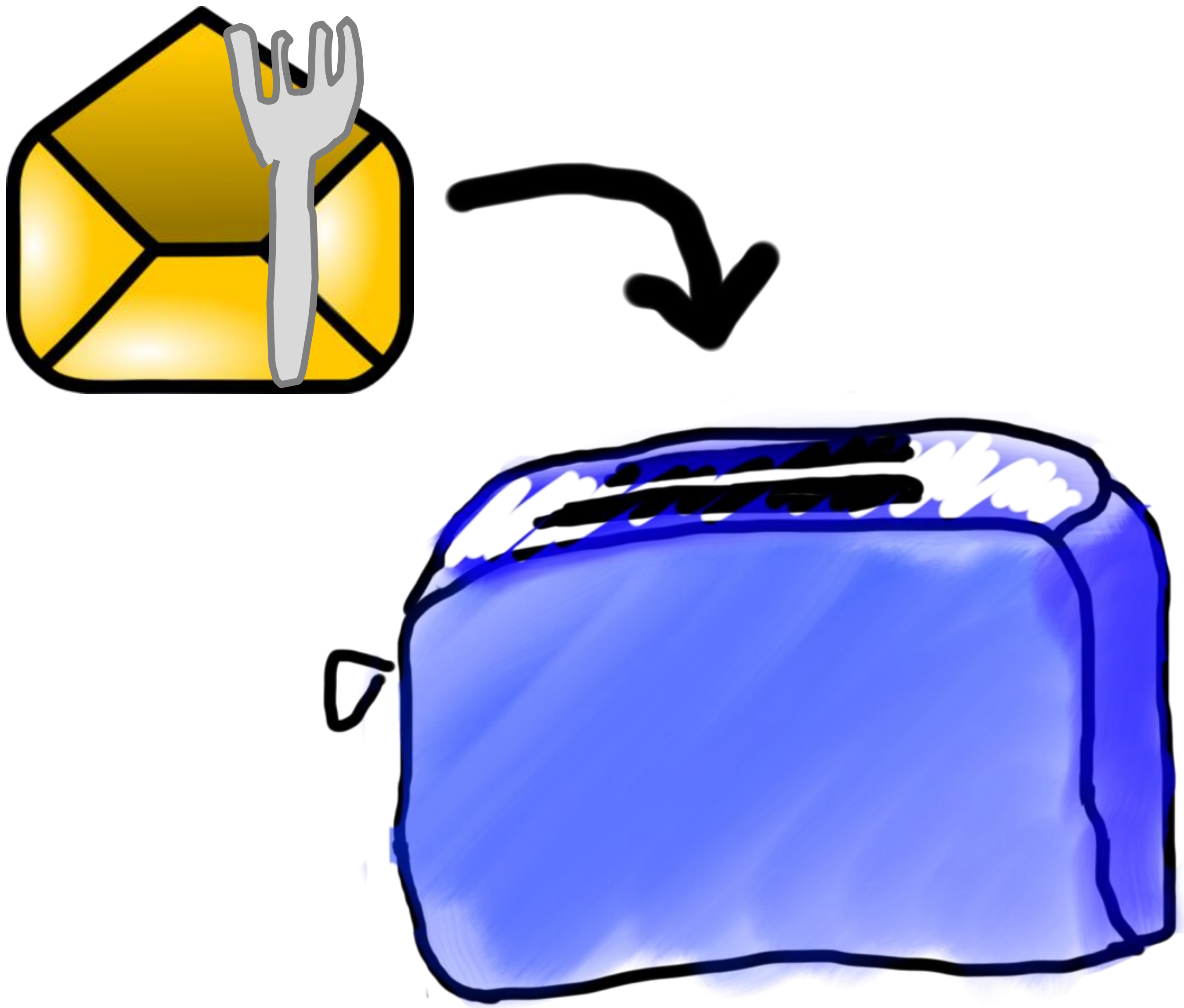


```
def handle_request(self, request):
    cmd = request.command
    if cmd == 'get_status':
        user = request.params['userName']
        status = self.get_status(user)
        return status
```



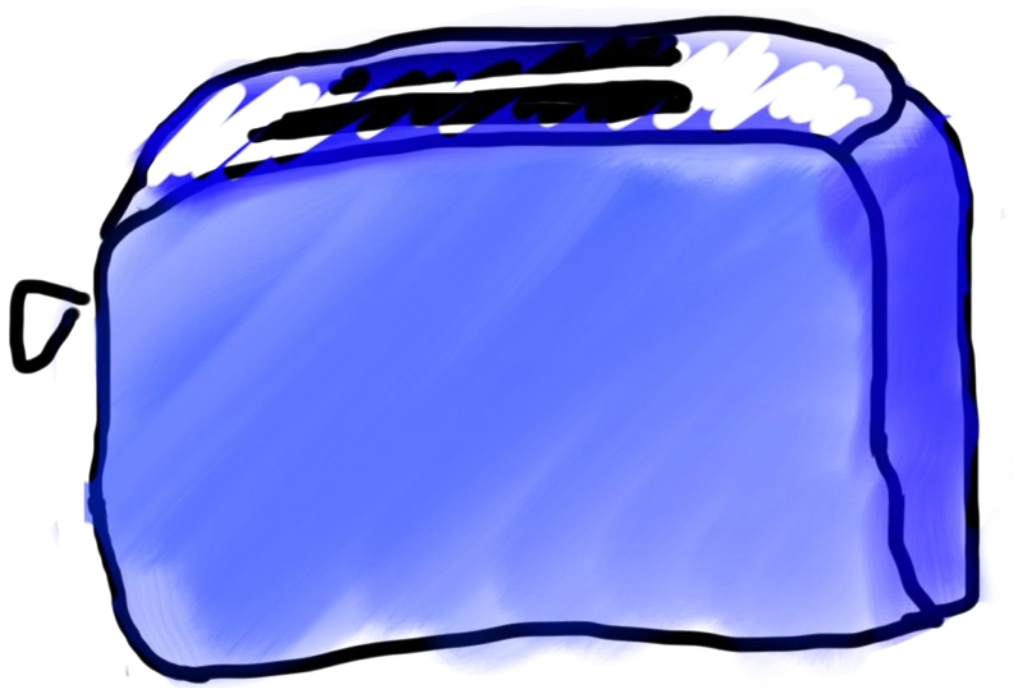
Must be a string!

Requests are like Remote Method Calls



Requests are like Remote Method Calls

Requests are like Remote Method Calls



Requests are like Remote Method Calls