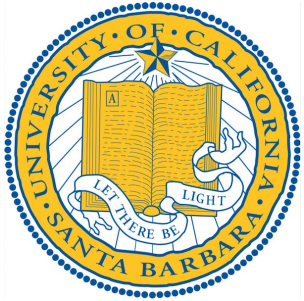
Welcome!

- Please grab an index card and write:
  - Your name
  - Your email address
  - Where you are at in your education (ex: 2nd year MS)
  - Previous experience with web technology
  - What do you hope to get from this class?

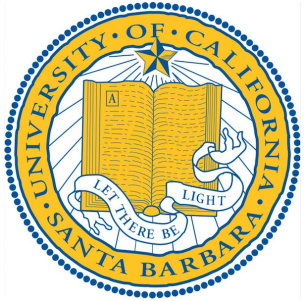


CS 290B

Scalable Internet Services

Andrew Mutz

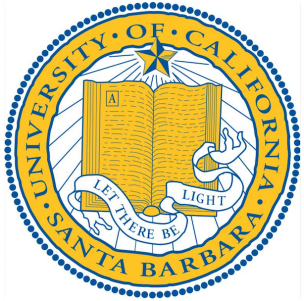
October 2, 2014



Today's Agenda

- Course Introduction
  - Motivation
  - Course Structure

- The Lifecycle of a Web Request



Course Introduction

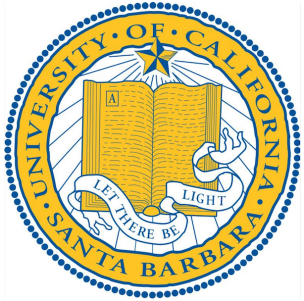
Let's say...

...I want to find a home to live in.

...I am lost in a foreign city.

...I want to go on a date.

... what do I do?

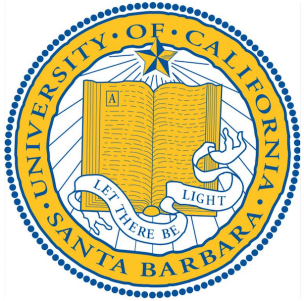


Course Introduction

Every day, billions of people use the same suite of technologies to solve these problems: internet services.

As these services get increasingly popular, they need to continue to function.

Scaling even relatively simple web applications (Twitter) can be very complex.



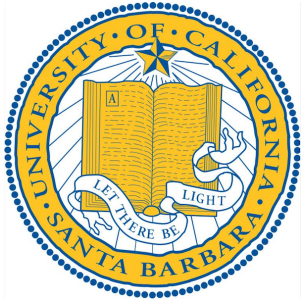
Course Introduction

How do we build internet services that continue to function as the number of users repeatedly doubles?

...As the size of the data set repeatedly doubles?

Buying faster hardware only delays the inevitable.

At the end of this course, you will understand the state of the art in building scalable internet services.



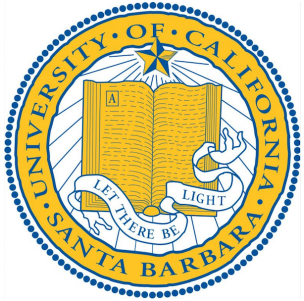
Course Introduction

This course is **project intensive**. We will be learning and building end-to-end scalable internet services.

This is **not a deep-dive** in Relational Databases,

Networking, Distributed Systems, Network Security, or Cloud Computing, but will touch on each.

- For those, see 274, 276, 271, 279, 290B-Wolski

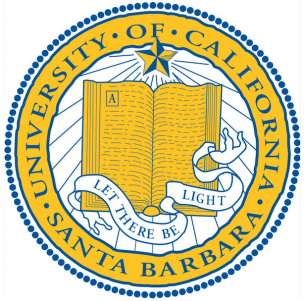


Course Introduction

Course structure

- Lectures Tues & Thurs at 1pm
  - The how and why of scalable internet services
  - Guest lectures from industry

- Labs each Wed at 6pm
  - Code-specific, project relevant
  - Project demonstration at labs



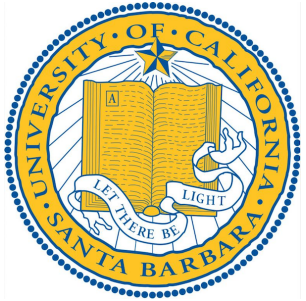
Course Introduction

How you will be evaluated

- No final, no midterm
- All evaluation based on project & other coding

Course Texts

- Agile Web Development with Rails
- High Performance Browser Networking

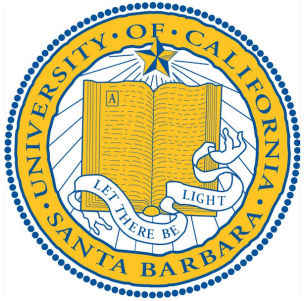


Course Introduction

Course Project

- Goal: Gain hands-on experience deploying a scalable web service

- Ideal characteristics:
  - Use Ruby/Rails/EC2
  - Have an interesting and large data set
  - Have some element of transactional updates

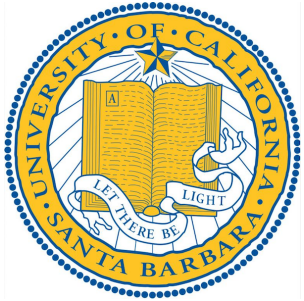


Course Introduction

Course Project

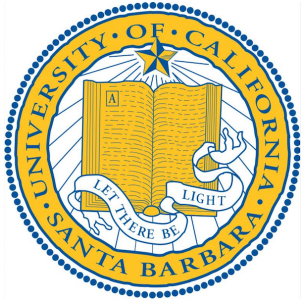
- Process
  - Project teams of 4 students
  - Use weekly sprints to make progress
  - Use modern software engineering techniques:
    - Scrum, TDD, Pair programming

- Look to http://cs290.com/projects for inspiration



Course Introduction

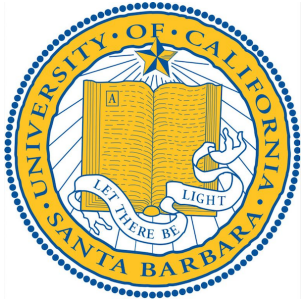
- Course websites
  - http://cs290.com
  - https://github.com/scalableinternetservices/
  - https://piazza.com/
    - Email notifications are a good idea
    - Students helping others is strongly encouraged



System Desiderata

desideratum (noun), plural desiderata

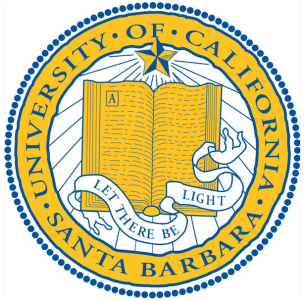
1. something wanted or needed.



System Desiderata

What do we want or need from an internet service?

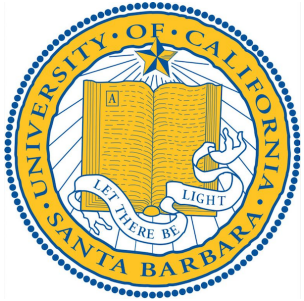
1. It should be always available
2. It should be fast
3. It should scale
    a. What do we mean by scale?
4. It should be secure
5. It should be correct and bug-free
6. It should be quick to build and easy to modify



System Desiderata

These needs form the organization of the course:

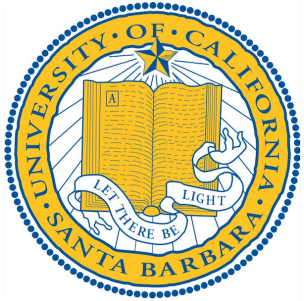
1.	High Availability	October 16
2.	Low Latency	October 21
3.	Horizontal Scaling	October 14, 23, 28
4.	Security Basics	October 30
5.	Stable Data Layer	October 23
6.	Working Software Quickly	October 9



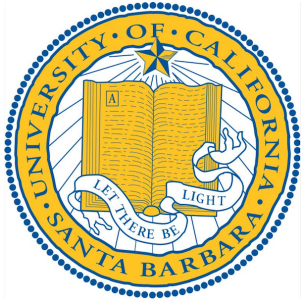
System Desiderata

After these topics, lectures will be composed of industry guest speakers and forward-looking topics in internet services:

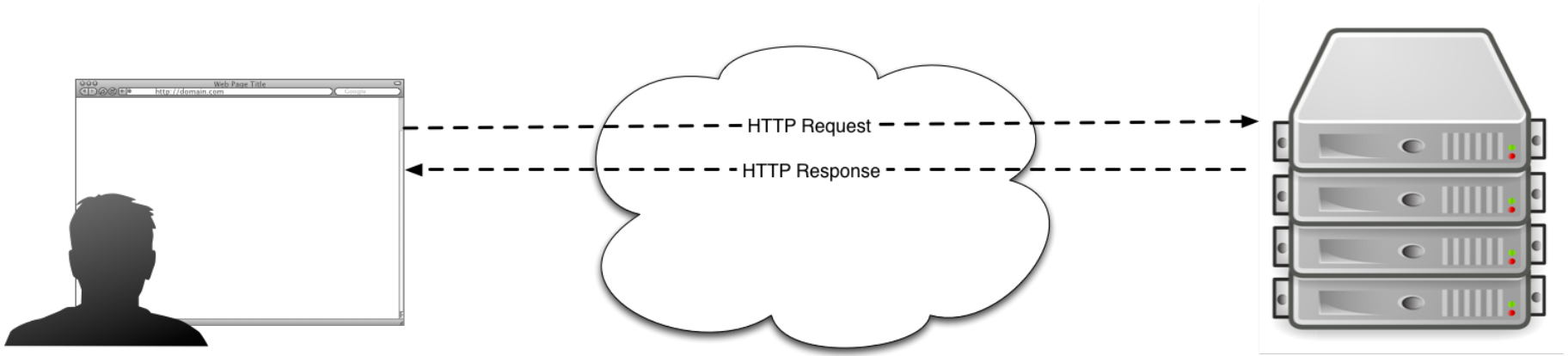
REST & Web Services	November 18
Optimizing for Mobile	November 18
HTTP 2.0 & SPDY	November 20
DevOps & Docker	November 20
asm.js & Thicker Clients	December 2
WebRTC & WebSockets	December 2



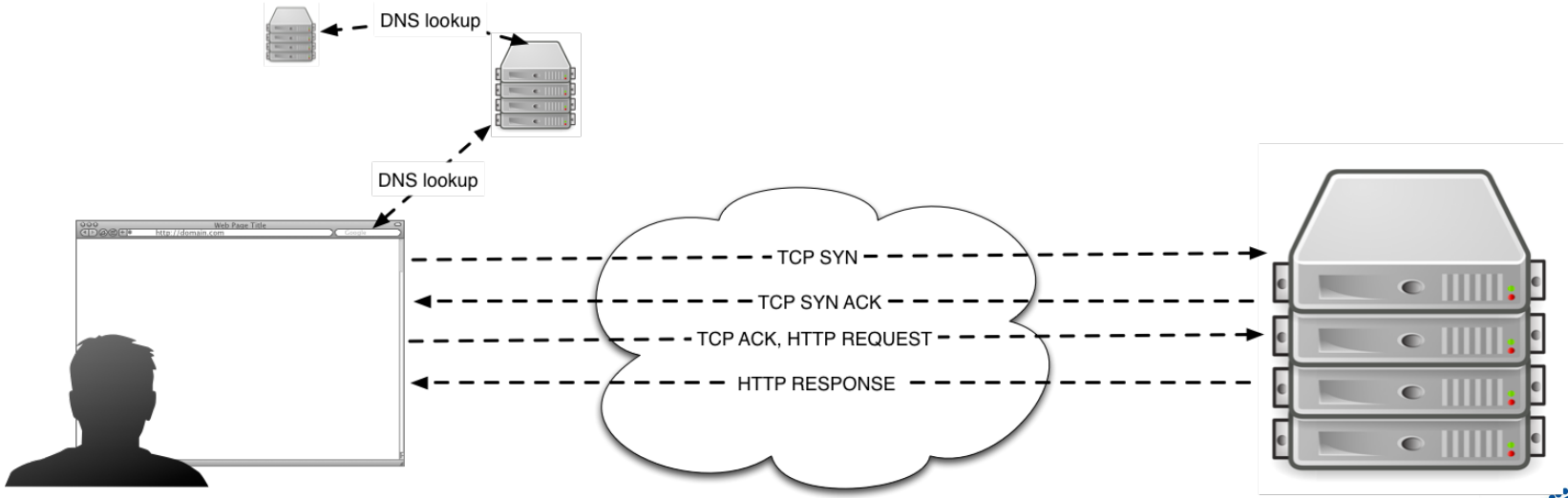
The Lifecycle of a Request

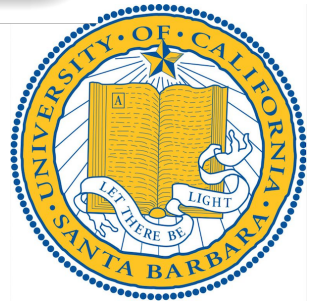


The Lifecycle of a Request

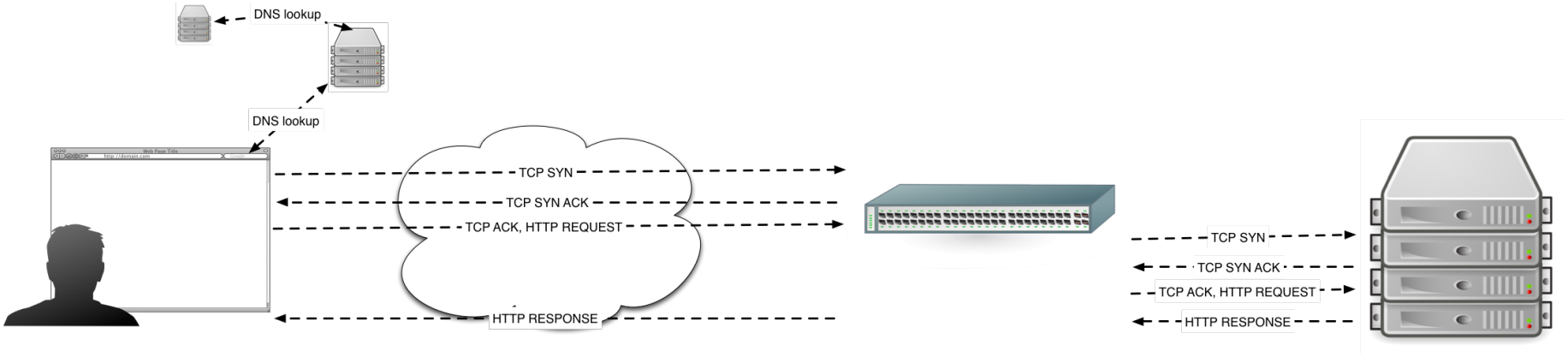


The Lifecycle of a Request

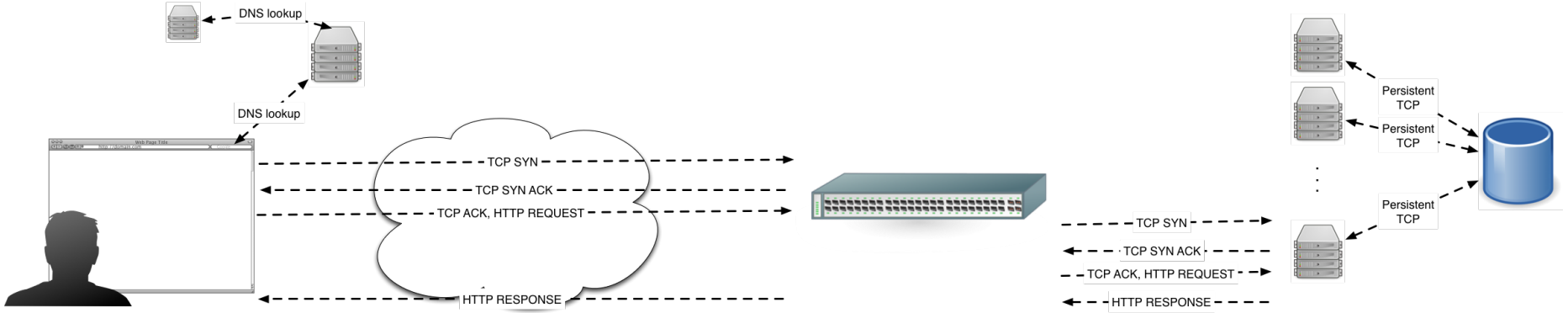


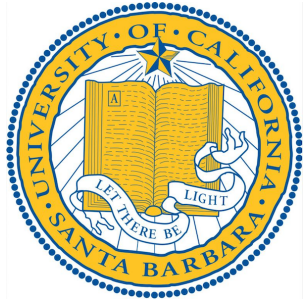


The Lifecycle of a Request

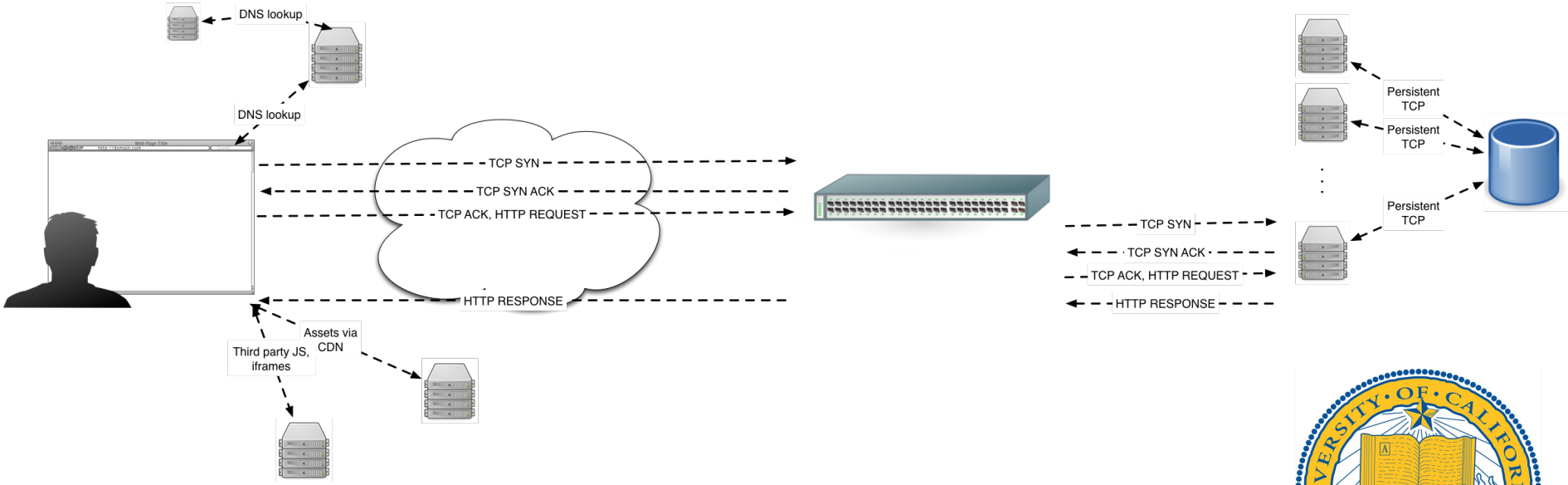


The Lifecycle of a Request

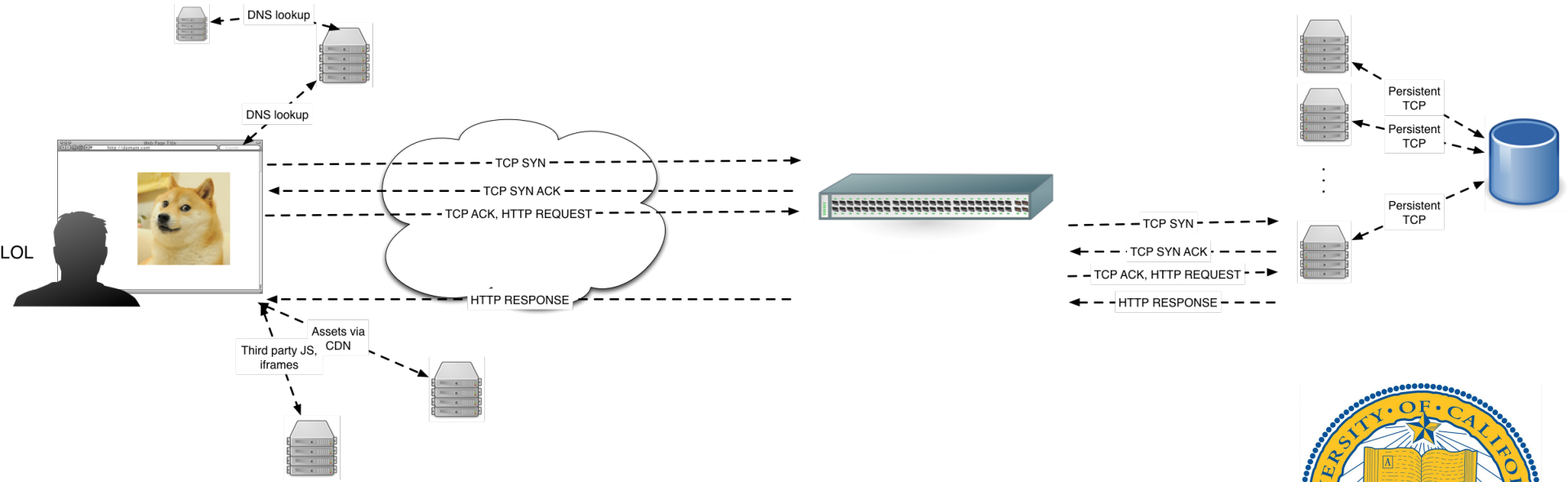




The Lifecycle of a Request



The Lifecycle of a Request



For Next Time...

- Read Chapters 1 & 2 of HPBN
- Read http://cs290.com/projects/
  - Start thinking about projects and groups
  - Post to Piazza with project ideas
- Start working on Ruby CodeAcademy
  - http://www.codecademy.com/en/tracks/ruby/