

## CS4513 Distributed Computing Systems

C-term 2016  
Mark Claypool

## Topics

- Background
- Admin Stuff
- Motivation
- Objectives
- Class material!

## Professor Background (Who am I?)

- Mark Claypool (professor, “Mark”)
  - Professor, Computer Science
  - Professor, Interactive Media and Game Development
  - Systems guy
- Research interests
  - Network games
  - Multimedia performance
  - Congestion control (protocols, AQM)
  - Wireless networking

## Student Background (Who are you?)

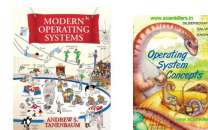
- |                        |                                 |
|------------------------|---------------------------------|
| 1. Year?               | 5. Language of Choice?          |
| 2. Major?              | 6. Expertise (low 1 to 5 high)? |
| 3. Background          | a. C/C++                        |
| a. cs3013?             | b. Java                         |
| b. cs3516?             | c. Unix                         |
| c. cs4516?             | 7. Other?                       |
| 4. Platform of Choice? |                                 |

## Syllabus Stuff

- <http://www.cs.wpi.edu/~cs4513/c16>
- Class: Tu, Fr 2-3:50pm
- TAs: Salah Ahmed
- Office hours:
  - TA: TBA, FLA22
  - Prof: TBA, FLB24b
  - Or by appointment
- Email
  - [claypool@cs.wpi.edu](mailto:claypool@cs.wpi.edu) (me)
  - [cs4513-staff@cs.wpi.edu](mailto:cs4513-staff@cs.wpi.edu) (me + TA)
  - [cs4513-all@cs.wpi.edu](mailto:cs4513-all@cs.wpi.edu) (class)

## Text Book

- OS Book from cs3013
  - File systems
- Research papers
  - Links
- Learned
  - Read by you
  - Presented by me
  - Tested for exam
- May be “recommended” papers/links



### Range of Topics

- File Systems
  - Distributed File Systems
- The Web
- Network Games
- Misc
  - Peer-to-Peer File Sharing
  - Cloud Computing
- Communication
- Architectures
- Synchronization
- Virtualization
- *Performance*

### Course Structure

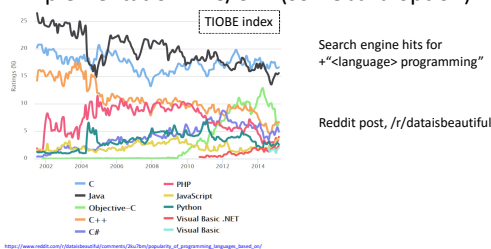
- Recommended background
  - Operating Systems (cs3013)
  - Computer Networks (cs3516)
  - Good programming skills (required)
- Grading
  - Exams (50%)
  - Projects (50%)

### Exams

- 2 exams
- 50% of grade
- Non-cumulative
- In-class
  - Closed-note, Closed-paper, Closed-friend

### Projects (1 of 3)

- Implementation in Linux (or Cygwin on Windows)
- Implementation in C/C++ (some Java option)



### Projects (2 of 3)

Rank	Language	Score
1.	Java	100.0
2.	C	99.9
3.	C++	99.4
4.	Python	96.5
5.	C#	91.3
6.	R	84.8
7.	PHP	84.5
8.	JavaScript	83.0
9.	Ruby	78.2
10.	Matlab	72.4
11.	SQL	71.3
12.	Shell	70.7
13.	Go	67.3

"What are the most popular programming languages? The only honest answer: It depends."

S. Cass "The 2015 Top Ten Programming Languages", *IEEE Spectrum*, July 2015

Weights and app so you can try

<http://spectrum.ieee.org/computing/software/the-2015-top-ten-programming-languages/>

### Projects (3 of 3)

- All are done *solo* (no groups)
- Topics
  - Proj 1 – Files
  - Proj 2 – Distributed shell
  - Proj 3 – File/Media sharing
  - Proj 4 – Network game (text based)
  - (BS/MS) – Cloud computing

## Slides

- On the Web (maybe after class)
- Powerpoint and PDF
- Caution! Don't rely upon the slides alone! Use them as supplementary material
  - (come to class)

## Timeline

<http://www.cs.wpi.edu/~cs4513/c16/timeline.html>

- *Estimate* of assignment dates
- Use it to help plan

## BS/MS Credit

- May be taken for graduate credit
  - Need to be admitted to BS/MS program
- Written permission via [approval form](#)
- Need “B” or better on all projects and exams
- May get 1 more graduate credit
  - Register 1/6<sup>th</sup> ISP
  - Additional programming project (Cloud Game)

## Goals and Objectives

### Goals

- Understand the principles of file systems from the operating system perspective.
- Gain experience writing distributed systems code.
- Understand the issues in distributing a shared, virtual world on multiple computers.
- Realize the implementation of a distributed system.
- Gain knowledge of communication, architectures, synchronization and virtualization as they pertain to distributed systems.

### Objectives

- Implement utilities that make extensive use of system calls pertaining to file systems.
- Design and implement a distributed system from scratch.
- Implement functionality for a networked game engine.
- Implement the distribution of state in a virtual world.
- Demonstrate an understanding of Web, peer-to-peer and file systems as distributed systems.

## Why This Class?

- WPI CS requirements
  - Gotta take something in Systems
  - Gotta take five 4000-level courses
- Can get BS/MS credit
- Distributed systems are the future
  - The Network is the Computer, The Cloud, Mobility and Wireless
- Distributed Systems are cool!
  - Algorithms, Networks, Hardware...
- Programming
  - The more you do, the better a computer scientist/software engineer you become
- Fun!