Course Overview

Kenneth M. Anderson
University of Colorado, Boulder
CSCI 5828 — Lecture 1 — 01/15/2008

Goals of this Class (I)

- Survey traditional software engineering concepts, terminology, and techniques
- Take an in-depth look at software engineering techniques for dealing with concurrency in software systems
  - Also take a glance at Tim Bray's Wide Finder project and its associated issues
- Learn about Agile Design methods, including Scrum and Extreme Programming
- Review the philosophy and techniques of a successful Web 2.0 company

Goals of this Class (II)

- First topic ensures that a student new to software engineering leaves the course with a grasp of the basic concepts and tenets of the field
- Second topic provides students with an explicit example of how software engineering techniques can be used to tackle complex problem domains
  - Relevant topic: hardly a week goes buy without some mention of multi-core computers and how they are changing the game on developers
    - Single threaded / Single process applications will soon be outshone by multi-threaded / multi-process applications
    - Consider an image processing example
- The latter two topics will provide the student with insight into how the field of software engineering is changing

CAETE Announcements

- In-Class Students
  - CAETE has a busy studio schedule
    - Be sure to exit promptly so next class can begin on time
  - Food and Drink are not technically allowed
    - Drinks are tolerated, as long as you keep the studio clean!
- Distance Students
  - Textbooks can be ordered from the CU Bookstore
    - Call 303-492-6411 or 800-255-9168
    - Or order on-line at <http://cubooks.colorado.edu/>

Due Dates for CAETE Students

- Due dates for assignments and exams for CAETE students are one week behind the due dates for in-class students
    - CAETE students need to make sure that they have a “test proctor” at their company who will be willing to administer the midterm for them
    - If you don’t know who your test proctor is, contact CAETE to find out
        - (Do this during the first week of classes!)

A bit about me…

- Associate Professor
  - At CU since July 1998
- Ph.D. at UC Irvine
- Research Interests
  - Software Engineering
  - Hypermedia / WWW
    - In field since 1991
  - REST-based Web Services

A little bit more…

- 20th semester at CU (!)
- 4th time teaching CSCI 5828
- Software Development Experience
    - Approximately 16 systems, 30K — 100K each
    - Some industry experience with IBM & Unisys
    - Experience with academic/industry collaboration

Office Hours

- ECOT 822
- Monday: 10:30 AM — 11:30 AM
- Thursday: 1:00 PM — 2:00 PM
- … or by appointment

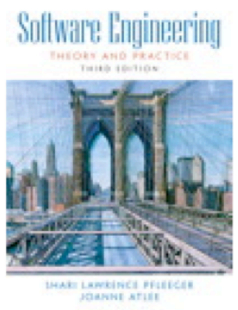
CSCI 5828 — Spring 2008

Foundations of Software Engineering

HOME | WHAT'S NEW | LECTURES | ASSIGNMENTS | EVALUATION | TEXTBOOKS | GRADES | PROJECTS | SYLLABUS STATEMENTS | CONTACT ME

Textbooks

There are two textbooks for 5828 this semester.



Pfleeger and Atlee's book "Software Engineering: Theory and Practice" is in its third edition and will be our guide through the concepts and techniques of traditional software engineering. It contains chapters on all aspects of software development including software process, software architecture, requirements, design (both functional and object-oriented), implementation, and testing. We will be covering a majority of the book this semester and students will have a chance to apply many of the techniques discussed in the book via the semester project and homeworks.

Magee and Kramer's book "Concurrency: State Models & Java Programming" is in its second edition and presents a model-based approach to designing and implementing multi-threaded systems. It nicely complements Jalote's book as it demonstrates the power that can be had by applying traditional software engineering techniques in a disciplined fashion. It also shows students the utility of software models and how such models can be implemented. Finally, it provides software that allows us to construct, analyze, and visualize models of concurrent behavior for multi-threaded systems.

Jan 9, 2008 11:07 AM

Class Info

Time: 2:00 PM — 03:15 PM TR

Location: ECCS 1B12

Useful Links

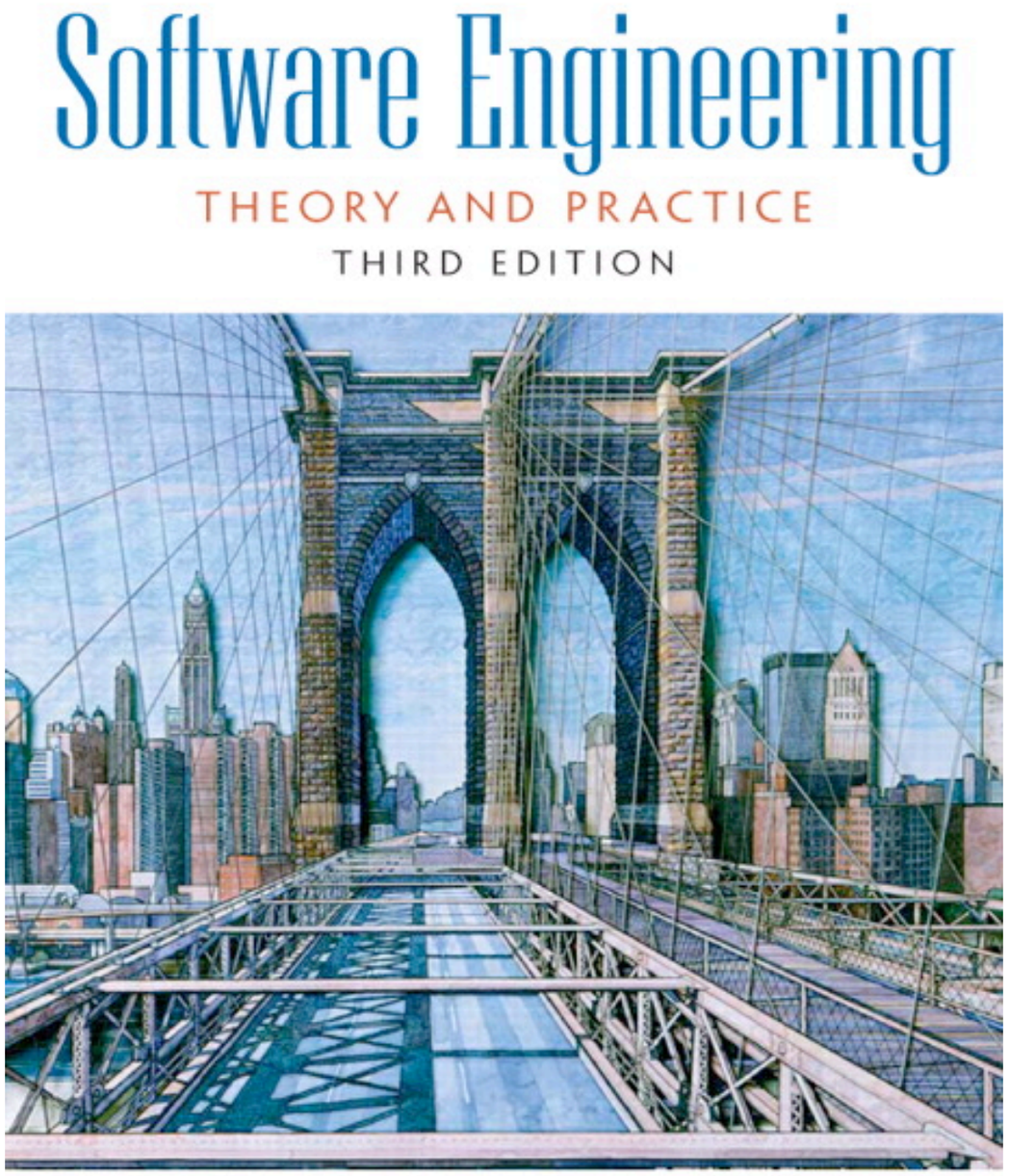
- Professor's Home Page
- Department of Computer Science

Class Website:
<http://www.cs.colorado.edu/~kena/classes/5828/s08/>

About the Class Website

- Check the website every day!
    - An RSS feed of the What's New page makes this an easy task
- The website is your source for
    - class schedule
    - lectures
    - assignments
    - announcements
    - etc.

Textbooks (I)



Pfleeger and Atlee's Software Engineering: Theory and Practice

Our guide to traditional software engineering. Well written and comprehensive.

Addresses "art versus science" debate with respect to developing software-based systems

See for instance "Undergraduate Programming" written recently by Joel Spolsky

Textbooks (II)

Magee and Kramer's Concurrency: State Models & Java Programming

aka "The Two Jeffs"

Every developer needs to understand concurrency (esp. in a multi-core world)

This book does an excellent job covering this topic with a software engineering perspective; indeed it shows the power that lies in the techniques and tools of traditional software engineering

Textbooks (III)

- The course will switch between the two topics
  - presenting an intro to software engineering one day
  - looking at the issues of modeling concurrency the next
- Please start reading the first two chapters of each text book

Class Participation

- I welcome participation by students
  - Feel free to interrupt me during lecture to ask questions!
  - “Stupid Questions” — No such thing!
  - Discussion: “Silent Tomb” — Boring for me and you!
- If I’m speaking too fast, stop me and ask me to slow down

Teaching Philosophy

- “sage-on-stage” vs. “guide-at-your-side”
- Answering questions
    - Sometimes the answer will be “I don’t know!”
- Current Tech
    - I try to incorporate references to current technology in my teaching
        - Ruby, Python, Ruby on Rails, Django, etc.
- Bias?
    - I don’t use Microsoft Windows and so have no experience with .Net, C#, Vista, etc., however I’m not “anti-Microsoft” and I will welcome student’s use of and presentations about Microsoft technology

Course Evaluation

- Your grade will be determined by your work on
  - Reviews (simple questions to make sure you do the reading)
    - One a week (including this week), grades: “+”, “✓”, or “-”
    - Review for this week will be posted tomorrow
      - Will be on chapter 1 of the Pfleeger textbook
  - Homeworks (typically problems from the Concurrency textbook)
  - Midterm (Tuesday, March 4th in-class, by Tuesday March 11th CAETE)
  - Class Project (Due at the last class session; students will work in teams)

- NO FINAL!

Moodle

- We will be using a Web-based system called “The Moodle” to submit homework assignments
- You will need to enroll in the CSCI 5828 course
  - Go to http://moodle.cs.colorado.edu/
  - If you do not have a moodle account, create one
  - Log in
  - Click on “All Classes” button and look for CSCI 5828 for Spring 2008
  - Click on it and follow instructions to enroll
    - You will need an enrollment key: seTaPSpring2008

Honor Code

- I encourage collaboration in this class via the reviews and the class project
- I'd like, however, for the homeworks and midterm to be worked on individually
    - As such, the honor code statement will be printed on each homework assignment and the midterm to remind you that collaboration is not allowed
- The Student Honor Code applies to classes in all CU schools and colleges. You can learn about the honor code at:
    - <http://www.colorado.edu/academics/honorcode/>.

Late Policy

- Assignments handed in late incur a 20% penalty
- Assignments can be handed in up to two weeks after initial due date (except for the final assignment of the class project)
  - after that you are out of luck…
- This does not apply to “reviews”. They will be due each Saturday at 11:55 PM; if you don’t submit a review, it counts as a “-”.

Syllabus Statements

- The University asks that various statements be presented to students at the start of each semester
  - Disability Accomodations
  - Religious Observances
  - Classroom Behavior
  - Discrimination and Harassment
  - Honor Code
- These statements are on the class website at: <http://www.cs.colorado.edu/~kena/classes/5828/s08/syllabus_statements.html>

What is Software Engineering? (I)

- The computer science discipline concerned with developing large applications. Software engineering covers not only the technical aspects of building software systems, but also management issues, such as directing programming teams, scheduling, and budgeting.

- Software
  - Computer programs and their related artifacts
    - e.g. requirements documents, design documents, test cases, specifications, protocol documents, UI guidelines, usability tests, …
- Engineering
  - The application of scientific principles in the context of practical constraints
    - Consider: Chemist versus Chemical Engineer

What is Software Engineering? (II)

- What is Engineering?
  - Engineering is a sequence of well-defined, precisely-stated, sound steps, which follow a method or apply a technique based on some combination of
    - theoretical results derived from a formal model
    - empirical adjustments for unmodeled phenomenon
    - rules of thumb based on experience
- This definition is independent of purpose, i.e., engineering can be applied to many disciplines

What is Software Engineering? (III)

- Software engineering is that form of engineering that applies…
  - a systematic, disciplined, quantifiable approach,
  - the principles of computer science, design, engineering, management, mathematics, psychology, sociology, and other disciplines…
- to creating, developing, operating, and maintaining cost-effective, reliably correct, high-quality solutions to software problems. (Daniel M. Berry)

What is Software Engineering? (IV)

- Issues of Scale
  - Consider this Call for Papers for this SE-related workshop
    - First ICSE Workshop on Software Technologies for Ultra-Large-Scale (ULS) Systems
- Issues of Communication
  - Consider the set of tools provided by sites like Assembla.com
- Issues of Regulation
  - Other engineering disciplines require their members to be certified
    - SE does not; should it?
- Issues of Design: dealing with the increasing integration of software, hardware, “process”, the user’s environment, legacy devices, and the like
  - Consider Apple’s MacBook Air (announced today!)

First Assignment

- Call my office phone at (303) 492-6003
- In your message:
    - State and SPELL your name
    - State that you are enrolled in CSCI 5828
    - Provide me with a nickname for you
    - Spell your nickname for me
- Nicknames used to post grades on the class website
    - As a result, your nickname should not resemble your name or any of your current nicknames!

Coming Up Next

- Lecture 2: Introduction to Software Engineering
    - Chapter 1 of Pfleeger and Atlee
- Lecture 3: Introduction to Concurrency
    - Chapter 1 of Magee and Kramer