

MSE 551 - Strategic Software Engineering Fall 2005

Instructor: Stuart Faulk (faulk@cs.uoregon.edu)

This syllabus is subject to change depending on the needs of the class. Students should refer to the class web page for up-to-date information on assignments and topics.

Catalog Description: Where traditional software engineering focuses on the development and maintenance of individual systems strategic software engineering addresses the development of multiple systems over time. Significant gains in productivity, cost, and schedule can result from systematic improvement of the software development process and systematic reuse of life-cycle products over multiple developments. This course covers the principles, methods, and tools for strategic software development including process modeling and improvement, developing programs as of families of systems, and systematic approaches to code generation and the reuse of non-code products including requirements, documentation, and design.

Prerequisites:

MSE Foundation and Context courses

Rationale:

Long-term process improvement and increased productivity requires process and product optimization across multiple development cycles. Significant savings can result from reusing the conceptual structures of software development including requirements, software architecture and detail design. This courses focuses on principles and methods for optimizing process and product over multiple developments.

Objectives:

Students will understand the principles of strategic software engineering and be able to apply them to adapt their software development processes and technologies to improve quality, reduce costs, and meet changing business needs. Students shall:

- Understand business case models for strategic software development and how to analyze a particular case.
- Understand principles and techniques for formal process modeling and continuous process improvement.
- Understand basic economics of developing a program family and be able to construct a simple business case.
- Demonstrate the ability to apply a domain analysis and engineering process to develop families of software systems.
- Demonstrate the ability to adapt a formal process model to achieve long term technical and business goals.

Required Text(s):

Software Product-Line Engineering: a Family-Based Software Development Process, David M. Weiss, Chi Tau Robert Lai, Addison Wesley, 1999, ISBN 0-201-69438-7

Software Product Lines: Practices and Patterns, Paul C. Clements and Linda M Northrop, Addison-Wesley, 2001, ISBN 0201703327

Readings:

Copies of the readings that are not in the course texts will be supplied on the class website for download in advance of the required class.

Outline of Topics:

Week 1: Introduction to Strategic Software Engineering

Overview of MSE program, course content, and grading. Introduction to the Strategic Software Engineering view of software development. Root causes of the ongoing “software crisis.” Inefficiencies inherent in a conventional (sequential) software development paradigm. Viewing a company's overall software development process: key concepts and objectives of process modeling and continuous process improvement. What it means to take a "product-line" view of software development. Overview of product line development strategies and the resulting efficiencies of scale.

- ◆ The “software crisis”
- ◆ Inefficiencies of sequential development
- ◆ A strategic view of software development
- ◆ The Celsius Tech case study

Reading:

Re-read [Brooks 87] Brooks, F. “No Silver Bullet”

(Re-)read the chapter on Celsius Tech in *Software Architecture in Practice* (the text from the Software Architecture course). This is Chapter 16 in the 1st edition and Chapter 15 in the 2nd.

[Gibbs 94] Gibbs, W. T., "Software's Chronic Crisis"

[Paulk 98] Paulk, et al, "The Capability Maturity Model for Software"

Assignment: On-line discussion of how the following things changed (and what changed about them) for Celsius Tech to take a product-line approach. Compose and post your own views, then discuss the posts with classmates:

- ◆ The development process
- ◆ The major products of development
- ◆ The company organization

Week 2: Process Modeling and Process Improvement I

Review of concepts of software engineering processes and process improvement. The process-improvement process. Overview of process specification: why we specify processes, qualities of a good process spec, contents of a good process spec. Process abstraction and modeling: discussion of modeling approaches. Concepts of process modeling

- ◆ The process modeling process
- ◆ Process improvement vs. product improvement
- ◆ Discussion of CMM
- ◆ Discussion of assignment
- ◆ Process specification and modeling

Readings:

[Kirby 90] "A Formalization of a Design Process, Process"

[Parnas 85b] "A Rational Design Process: How and Why to Fake It,"

[Song 98] "Engineering Software Design Processes to Guide Process Execution"

Assignment: On-line discussion questions

Week 3: Process Modeling and Process Improvement II

Process specification and modeling. Techniques for modeling software processes. Using software process models: enactment, analysis, testing, baselining, and maintaining. Discussion of modeling methods in the literature. Introduction to a process modeling technique. Viewing software process models as a product of software engineering.

- ◆ Use of process models
- ◆ Discussion of Song & Osterweil's modeling method
- ◆ Overview of the PASTA modeling tool

Readings:

Text Chapter 6-7,

A Process Modeling Approach and Notation (Documentation for PASTA process modeling tool)

Assignment: Use a process modeling tool to build a model of the "ideal" software development process described in [Parnas 85b]. For this exercise, we use the PASTA modeling tool that is also used to model the process in the textbook.

Week 4: Product Line Development

Overview of product line development processes. Software engineering foundations for developing program families including key principles – role of modularization, information hiding, and abstraction.. Understanding and using the FAST PAST process model. An example development including work products.

- ◆ In class presentation of PASTA process models and discussion of lessons learned (1 hour)
- ◆ Foundations for product-line development: key assumptions and principles

Readings:

Weiss text Chapter 1, 2, 3

Clements text 1-3

[Parnas 76] "On the Design and Development of Program Families"

Week 5: The FAST Product Line Development Process

Overview of the FAST process: Economic Modeling, Application Engineering and Domain Engineering. Economics of product-line development using FAST. Understanding and using the FAST PASTA formal process model provided in the text.

- ◆ Domain analysis
- ◆ Domain engineering
- ◆ Application Engineering
- ◆ The FAST/PASTA model (in class demonstration)

Readings:

Weiss text Chapter 4-5
Clements text 4
Articles

Week 6: Domain Analysis

Products, methods and principles of Domain Analysis. The Commonalty Analysis process and products: how to determine the scope of a family of systems, how to determine and define the common requirements of the family, how to determine and define the variabilities that distinguish family members. Economic modeling: assessing and modeling the return on investment from developing software as product lines.

- ♦ Discussion of [DeBaud 99], concepts of value and product-line scope

Readings:

Weiss Chapter 8 and relevant parts of the process model
Clements text 5

Assignment:

- 1) FWSOS project. Do revised Commonalty Analysis document for next week.
- 2) Project proposals: submit proposal for term project

Week 7: Domain and Application Engineering

Formal model of the domain engineering process. Development of the application engineering environment. Designing and developing product-line architectures. Generation technologies for code, documentation and other work products. Introduction to composition-based and compiler-based approaches to product generation.

- ♦ Walkthrough of the FWS example generation technology
- ♦ Presentation/Discussion of Commonalty Analysis part of FWSOS

Readings:

[Faulk 00] "Value Based Software Engineering"

Assignment: Complete Part 2 of the FWSOS project (revision of the module guide and uses hierarchy).

Week 8: The Business Context

Organizational and business issues in applying strategic software engineering. Structuring an organization to develop software processes and product lines – key roles and responsibilities. How functional units in the organization must relate and interact to ensure alignment of strategic business goals and technical development decisions.

Readings: Clements/Northrop Chapter 10

Assignment: On-line discussion

Week 9: Projects

No class. Work on and discuss class projects

Week 10: Project Reviews

Presentation and critique of student projects (results of applying the FAST process to a small system product line). Review of course material.

Summary of recommended assignments:

- ♦ CelsiusTech Summary: This assignment has students consider carefully a case study in product-line engineering with the goal of achieving a better understanding of how the process, product and organizational structure of a business change to develop software as product lines.
- ♦ Process Modeling: The students should develop a process model for a subset of a software development process, preferably using a modeling tool. For the first version of the course, students developed a model of Parnas' "rational design process" (top level) using the PASTA tool.
- ♦ FWSOS Project: Apply the FAST Domain Engineering process to develop a set of products for a small product line. Students do a small-scale, hands-on application of product-line development.

Project/Term Paper

Every student is required to do a project or term paper and present findings and results to the class. The goal of the project is to gain hands-on experience applying some of the concepts and material from the course to a real problem, preferably from your own work experience. You should scope the project so that it adequately demonstrates the ability to apply class material but can be completed by Week 10. The project consists of three parts:

1. Proposal: develop a written proposal for your project. The proposal should describe a) what you plan to do, b) which aspect from the course you will apply and how, and c) what product you plan to product. A page is sufficient. The instructor will provide examples from previous classes.
2. Written report or other products: a written report and possibly other products (e.g., adaptable code, a commonality analysis document, an economic analysis, etc.). The usual scope of these is roughly ten pages or so depending on the subject.
3. Presentation: prepare a presentation (e.g. PowerPoint) on your project for class. The goal of the presentation should be to convey what you learned from your project to your classmates. Your presentation should not run longer than 20 minutes so everyone will have a chance.

Due dates: Proposal: Week 7

Project/Term Paper: Week 10

Presentation: Week 10 (continues to week 11 if needed)