Software Engineering

Prof. Dr. Bertrand Meyer

March 2007 - June 2007

Lecture 1: Introduction
Goal of the course

Introduce you, in both theory and practice, to:

The challenges and techniques of building production software in an industrial environment
What is software engineering?
The world of software

Software now controls much of the modern world:

- Business operations (administrative etc.)
- Government
- Factories
- Transportation
- Defense, finance, health...
- Many traditional processes, e.g. publishing
- New technologies now mainstream, e.g. the Web, which have transformed our life

Achille’s heel of this revolution: cost, schedule, quality
Business and political concern, not just technical
The need for quality software

Many lives and businesses now depend on software.

We now need larger, more complex, and safer software systems on predictable schedules.

Without different software practices, this will not happen.

The **Team Software Process (TSP)** addresses this need.

The **PSP** provides the knowledge and skill that developers need to work on **TSP** teams.
Many projects fail

... See Info I slides
Why projects fail

Five major reasons:

- Unrealistic commitments
- Inadequate leadership and management
- Lack of control (no personal plans by developers, insufficient knowledge by management)
- Insufficient quality (driven by quality of worst part)
- Insufficient technology (methods, tools, languages)
So, what is software engineering?

The production of operational software satisfying defined standards of quality
Software engineering

... includes programming, but is more than programming

As von Clausewitz did not write: “the continuation of programming through other means”.
The five components of software engineering

- **Describe**: Requirements, design specification documentation
- **Implement**: Design, programming
- **Assess**: Testing and other V&V* techniques
- **Manage**: Plans, schedules, communication, reviews...
- **Operate**: Deployment, installation, *Validation & Verification*
In this course...

... Every one of these five aspects gets approximately equal weight
Practical information
The Teaching Staff
Bertrand Meyer

E-mail: Bertrand.Meyer@inf.ethz.ch
Office: RZ J6
Appointments: on request
Secretary: Claudia Günthart, (01) 632 83 46
Exercise sessions

All groups have one session a week:

- **Monday 10-12**
  - IFW A34 Werner Dietl

- **Thursday 8-10**
  - IFW A34 Hermann Lehner
  - CAB H57 Jörg Derungs
  - IFW B42 Marco Piccioni

- **Friday 13-15**
  - IFW B42 Mitra Purandare
  - IFW C42 Adam Darvas

Registration: today and tomorrow
# The assistants

<table>
<thead>
<tr>
<th>Name</th>
<th>Language</th>
<th>Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adam Darvas</td>
<td>English</td>
<td>Friday</td>
</tr>
<tr>
<td>Jörg Derungs (from April)</td>
<td>German</td>
<td>Thursday</td>
</tr>
<tr>
<td>Werner Dietl</td>
<td>German</td>
<td>Monday</td>
</tr>
<tr>
<td>Hermann Lehner</td>
<td>German</td>
<td>Thursday</td>
</tr>
<tr>
<td>Marco Piccioni</td>
<td>English</td>
<td>Thursday</td>
</tr>
<tr>
<td>Mitra Purandare</td>
<td>English</td>
<td>Friday</td>
</tr>
<tr>
<td>Arsenii Rudich</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(no exercise group)</td>
<td></td>
</tr>
</tbody>
</table>
Exercise sessions

All groups have one session a week:

- **Monday 10-12**
  - IFW A34 Werner Dietl

- **Thursday 8-10**
  - IFW A34 Hermann Lehner
  - CAB H57 Jörg Derungs
  - IFW B42 Marco Piccioni

- **Friday 13-15**
  - IFW B42 Mitra Purandare
  - IFW C42 Adam Darvas

Registration: today and tomorrow

Note: most groups will miss a session in week 2 or 3 - check with assistant
Course material

Course page:
http://se.inf.ethz.ch/teaching/ss2007/252-0204-00/
→ Check it at least twice a week

Lecture material:
- Lecture slides
- No required textbook, but we will recommend books as we go; see bibliography on course page

Exercise material:
- Exercise sheets
- Master solutions
Electronic forums

Discussion forums:
http://forum.vis.ethz.ch

Mailing list for each group

The usual advice and rules:

- Use the forums and mailing lists! Take advantage of every help you can get.
- Don’t be shy. There are no stupid questions.
- Criticism welcome, but always be polite to every participant and observe the etiquette.

To email the whole teaching team (professor and assistants):

softeng@se.inf.ethz.ch (soon operational)
Exercise sessions and project

Make sure to attend all sessions

Exercise sheets will be distributed by your assistant during the exercise session

Do all exercises and the project
Lecture plan

Week 1: Basics
- 20 March Introduction
- 21 March Basics of the software process, PSP (1)

Week 2: Requirements
- 27 March Requirements (1)
- 28 March Requirements (2)

Week 3: Management
- 3 April Project management (tools)
- 4 April Project management (principles)

Week 4: Process
- 10 April PSP (2)
- 11 April Software engineering process, formal/agile

Week 5: From requirements to design
- 17 April From Requirements to Design
- 18 April System Design

Week 6: UML
- 24 April Modeling in UML
- 25 April From requirements to design using UML

Week 7: Introduction to V&V
- 02 May Introduction to the V&V lifecycle

Week 8: Design
- 8 May Detailed Design
- 9 May Refactoring

Week 9: Implementation
- 15 May Implementation
- 16 May Implementation

Week 10: Testing
- 22 May Testing basics
- 23 May More on testing

Week 11: V&V beyond testing
- 29 May Debugging
- 30 May Other V&V techniques

Week 12: Deployment
- 05 June Deploying a software system
- 06 June Social, legal and ethical issues:

Week 13: CMMI
- 12 June Introduction to CMMI
- 13 June More on CMMI

Week 14: Conclusion
- 19 June HCI and usability
- 20 June Final project presentation
Special lecture

Tomorrow (21 March), second hour

Guest lecture by Mark Howard, Axa Rosenberg (Orinda, California)

An object-oriented software process in a large investment management organization
Grading

50% Project

50% Exam

Project: a software engineering development (see next...)

Exam: review of all the concepts in the course
The project

*Computer Science Academic & Research Daily Advertising Service (CSÁRDÁS)*

**Purpose:**

Introduce you to problems and techniques of software construction in industry, with constraints mimicking those of actual projects.

It is not just a programming project but includes the five key aspects of software engineering as taught in this course (DIAMO):

- **Description**
- **Implementation**
- **Assessment**
- **Management**
- **Operation**
More on CSÁRDÁS

Model: CSEL (the Computer Science Events List):
  [http://events.informatics-europe.org](http://events.informatics-europe.org)
written by Marco Piccioni
Source code available for guidance

CSÁRDÁS is not an artificial example but reflects a real need
We expect that the best student project will actually be deployed at
the end of the semester

**Group size:** three students

**Technologies imposed by the customer:**
- Web server: Apache
- Operating system: Linux
- Programming language: Eiffel
- Web framework: EiffelWeb
- Database technology: mySQL
Project deliverables

**Week 1, due week 5:**
- Requirements document

**Week 5, due week 7:**
- Test plan for another group’s requirements
- Not released to that group

**Week 7, due week 12:**
- Design and implementation of your requirements

**Week 12, due week 14:**
- Test other group’s system
About the project

There will be “stakeholders” to help you define the requirements

Informatics Europe can benefit from CSÁRDÁS. We hope to release the best student project (with credit) on the Web page of the association

The first project document, describing the first assignment, is on the Web page. Note: it will be revised, please check again by the end of the week.
The production of operational software satisfying defined standards of quality
Software engineering today

Three cultures:

- Process
- Agile
- Object

The first two are usually seen as exclusive, but all have major contributions to make.
Process

Emphasize:

- Plans
- Schedules
- Documents
- Requirements
- Specifications
- Order of tasks
- Commitments

Examples: Rational Unified Process, CMMI, Waterfall...
Agile

Emphasize:

- Short iterations
- Testing (over specifications); “Test-Driven Development”
- Constant customer involvement
- Refusal to commit to both functionality and deadlines
- Specific practices, e.g. Pair Programming

Examples: Extreme Programming (XP), lean programming
Object-oriented (the Eiffel variant)

Emphasize:

- Seamless development
- Reversibility
- Single Product Principle
- Design by Contract
More on this course

Challenging but should be a rewarding experience.

Expect to learn a lot and have your views challenged

The project will be demanding; situation similar to what is found in industry. Reserve enough time for it throughout the semester.

Observe the rules (in particular confidentiality)
What to do now

Please choose a group