Software Engineering
Prof. Dr. Bertrand Meyer
Dr. Manuel Oriol
Dr. Bernd Schoeller

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
Manuel Oriol

E-mail: Manuel.Oriol@inf.ethz.ch
Office: RZ J9
Appointments: on request, by e-mail
Bernd Schoeller

E-mail: Bernd.Schoeller@inf.ethz.ch
Office: RZ J5
Appointments: on request, by e-mail
Exercise sessions

All groups have one session a week:

- **Monday 10-12**
  - IFW A34 Beat Herlig
- **Thursday 8-10**
  - IFW A34 Hermann Lehner
  - CAB H57 Roman Mitin
  - IFW B42 Wolfgang Schwedler
- **Friday 13-15**
  - IFW B42 Joseph Ruskiewicz
  - IFW C42 Arsenii Rudich

Registration: today and tomorrow
# The assistants

<table>
<thead>
<tr>
<th>Name</th>
<th>Language</th>
<th>Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beat Herlig</td>
<td>German</td>
<td>Monday</td>
</tr>
<tr>
<td>Herman Lehner</td>
<td>German</td>
<td>Thursday</td>
</tr>
<tr>
<td>Wolfgang Schwedler</td>
<td>German</td>
<td>Thursday</td>
</tr>
<tr>
<td>Roman Mitin</td>
<td>English</td>
<td>Thursday</td>
</tr>
<tr>
<td>Joseph Ruskiewicz</td>
<td>English</td>
<td>Friday</td>
</tr>
<tr>
<td>Arsenii Rudich</td>
<td>English</td>
<td>Friday</td>
</tr>
</tbody>
</table>
Course material

Course page:  
http://se.inf.ethz.ch/teaching/2008-S/se-0204/  
→ 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 Solution or in-class corrections
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**
  - 19 February: Introduction
  - 20 February: Development Models
- **Week 2: Requirements**
  - 26 February: Requirements (1)
  - 27 February: Requirements (2)
- **Week 3: From Requirements to Design**
  - 4 March: From Requirements to Design (1)
  - 5 March: From Requirements to Design (2)
- **Week 4: Project Management, Standards**
  - 11 March: Project Management
  - 12 March: Standards (1)
- **Week 5: Standards, Agile and XP**
  - 18 March: Standards (2)
  - 19 March: Development Models
- **Week 6: Easter**
- **Week 7: Testing and QA**
  - 1 April: Introduction to QA and testing (1)
  - 2 April: Introduction to QA and testing (2)
- **Week 8: Testing, System design**
  - 8 April: Introduction to QA and testing (3)
  - 9 April: System Design
- **Week 9: Architecture, Modeling**
  - 15 April: Architecture and Design
  - 16 April: Modeling in UML (1)
- **Week 10: Modeling, Metrics**
  - 22 April: Modeling in UML (2)
  - 23 April: Estimation Techniques and Metrics (1)
- **Week 11: Metrics, Profiling**
  - 29 April: Estimation Techniques and Metrics (2)
  - 30 April: Profiling
- **Week 12: Guest lectures**
  - 6 May: Guest Lecture
  - 7 May: Guest Lecture
- **Week 13: Students Subjects**
  - 13 May: TBA
  - 14 May: TBA
- **Week 14: Documentation, Legacy**
  - 20 May: Documentation
  - 21 May: Legacy Software and Migration
- **Week 15: Legal Issues, Exam**
  - 27 May: Social, legal and ethical issues
  - 28 May: Exam
Special lectures

We will have two guest lectures and will put them in the timetable when best fit.

We will pick subjects from what you wrote to us on the card...

Feedback!
Grading

1/3 Project

2/3 Exam

Project: a software engineering development

Exam:
  review of all the concepts in the course
  + questions about the project
The project

Purpose:

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

This year’s project has a strong emphasis on testing (especially the last part). The reason is that this constitutes techniques underused in industry and for which the technology is here.
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 fill in the green card and pick a group