

Software engineering (2IP25)

Prof.dr. Mark van den Brand



Practical issues

- **Docent:**
 - Prof.dr. Mark van den Brand (m.g.i.v.d.brand@tue.nl), HG5.59
- **Meer informatie over SE (2IP25):**
 - <http://www.win.tue.nl/~mvdbrand/courses/se/1011/>
 - slides
 - opdrachten
 - achtergrond artikelen
- **boek (verplicht):**
 - Software Engineering van Hans van Vliet, 3^{de} editie, Wiley
 - Andere SE boeken kunnen ook gebruikt worden

Practical issues

- **Examen:**
 - **Schriftelijk examen (70%)**
 - gesloten boek
 - materiaal:
 - slides
 - artikelen
 - boek
 - **Praktische opgave (30%):**
 - ???
 - analyseren van URD

Why Software Engineering in general?

- **The nature of software ...**
 - **Software is everywhere**
 - dependable
 - robust
 - **Software is intangible**
 - hard to understand development effort
 - **Software is easy to reproduce**
 - cost is in its *development*
 - in other engineering products, manufacturing is the costly stage
 - **The industry is labor-intensive**
 - hard to automate

Why Software Engineering?

- The nature of software ...
 - untrained people can hack something together
 - quality problems are hard to notice
 - software is easy to modify
 - people make changes without fully understanding it
 - software does not 'wear out'
 - it *deteriorates* by having its design changed:
 - erroneously, or
 - in ways that were not anticipated, thus making it complex

Why Software Engineering?

- The nature of software ...
 - Conclusions
 - software has in general a poor design and it is getting worse
 - demand for software is high and rising
 - we are in an ever lasting 'software crisis'
 - we have to learn to 'engineer' software

Why Software Engineering?

- Types of software ...
 - Custom
 - For a specific customer
 - Generic
 - Sold on open market
 - Often called
 - COTS (Commercial Off The Shelf)
 - Shrink-wrapped
 - Embedded
 - Built into hardware
 - Hard to change

Why Software Engineering?

- Within 30 years the amount of software in cars went from 0 lines of code to more than 10,000,000 lines of code
- More than 2000 functions are controlled by software in high-end cars
- 50/70% of the development costs of hard/software are software costs

Why Software Engineering?

- **Embedded Software as Innovation Driver**
- Software is today the most crucial innovation driver for technical systems, in general
- **By software**
 - we realize innovative functions,
 - we find new ways of implementing known functions with reduced costs, less weight or higher quality,
 - we save energy and, what is, in particular, important,
 - we combine functions and correlate them into multi-functional systems

What is Software Engineering?

- **Software engineering is the establishment and use of sound engineering principles in order to obtain economically software that is reliable and works efficiently on real machines**
- **Other definitions:**
 - The process of solving customers' problems by the systematic development and evolution of large, high-quality software systems within cost, time and other constraints
 - IEEE: (1) the application of a systematic, disciplined, quantifiable approach to the development, operation, maintenance of software; that is, the application of engineering to software. (2) The study of approaches as in (1).

What is Software Engineering?

- **Software engineering concerns the development of large programs**
 - Moving from programming-in-the-small to programming-in-the-large
- **Mastering complexity**
- **Evolution**
- **Efficiency of software development**
- **Cooperation between people is an integrated part of programming-in-the-large**
- **Software has to support its users effectively**
- ...

What is Software Engineering?

- **Solving customers' problems**
 - This is the *goal* of software engineering
 - Sometimes the solution is to *buy, not build*
 - Adding unnecessary features does not help solve the problem
 - Software engineers must *communicate effectively* to identify and understand the problem

What is Software Engineering?

- **Systematic development and evolution**
 - An engineering process involves applying *well understood techniques* in an organized and *disciplined* way
 - Many well-accepted practices have been formally standardized
 - e.g. by the IEEE or ISO
 - Most development work is *evolution*
 - Master course on Software Evolution (2IS55)