

A Brief Overview of Software Engineering

Objectives

- To introduce software engineering and to explain its importance
- To set out the answers to key questions about software engineering
- To introduce ethical and professional issues and to explain why they are of concern to software engineers

Topics covered

- FAQs about software engineering
- Professional and ethical responsibility

FAQs about software engineering

- What is software?
- What is software engineering?
- What is a software process?
- What is a software process model?
- What are the costs of software engineering?
- What are the attributes of good software?
- What are the key challenges facing software engineering?
- What are the professional and ethical responsibilities of software engineering practitioners?

What is software?

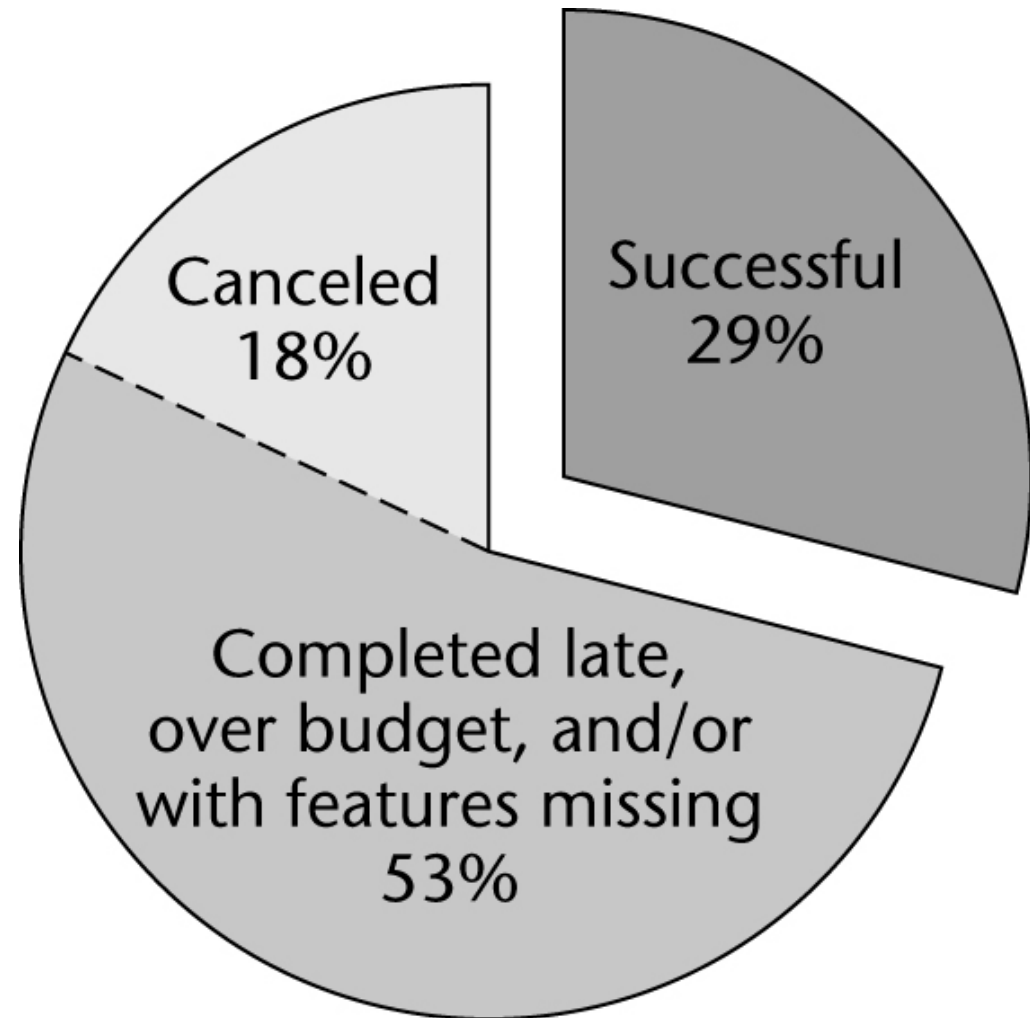
- Computer programs and associated documentation such as requirements, design models and user manuals.
- Software products may be developed for a particular customer or may be developed for a general market.
- Software products may be
 - **Generic** - developed to be sold to a range of different customers e.g. PC software such as Excel or Word.
 - Bespoke (**custom**) - developed for a single customer according to their specification.
- New software can be created by developing new programs, configuring generic software systems or reusing existing software.

What is software engineering?*

- 1968 NATO Conference, Garmisch, Germany
- Aim: To solve the software crisis
- Software is delivered
 - Late
 - Over budget
 - With residual faults

Standish Group Data*

- Data on 9236 projects completed in 2004



Cutter Consortium Data*

- 2002 survey of information technology organizations
 - 78% have been involved in disputes ending in litigation
- For the organizations that entered into litigation:
 - In 67% of the disputes, the functionality of the information system as delivered did not meet up to the claims of the developers
 - In 56% of the disputes, the promised delivery date slipped several times
 - In 45% of the disputes, the defects were so severe that the information system was unusable

What is software engineering?

- Software engineering is an engineering discipline that is concerned with all aspects of software production.
- Software engineers should adopt a **systematic** and organised approach to their work and use appropriate tools and techniques depending on the problem to be solved, the development **constraints** and the **resources** available.

Software engineering

- The economies of all developed nations are dependent on software.
- More and more systems are software controlled
- Expenditure on software represents a significant fraction of GNP in all developed countries.
- Software engineering is concerned with theories, methods and tools for professional software development.

Economic Aspects of Software Engineering*

- Coding method CM_{new} is 10% faster than currently used method CM_{old} . Should it be used?
- Common sense answer
 - Of course!
- Software Engineering answer
 - Consider the cost of training
 - Consider the effect of CM_{new} on maintenance

What is a software process?

- A set of activities whose goal is the **development or evolution of software**.
- **Generic activities** in all software processes are:
 - **Specification** - what the system should do and its development constraints
 - **Development** - production of the software system
 - **Validation** - checking that the software is what the customer wants
 - Verification:
 - **Evolution** - changing the software in response to changing demands.

What is a software process model?

- A simplified, abstract representation of a software process
- **Generic process models**
 - Waterfall;
 - Iterative development;
 - Component-based software engineering
 - Many more

What are the costs of software engineering?

- Costs vary depending on the type of system being developed and the requirements of system attributes such as performance and system reliability.
- **Distribution of costs** depends on the development model that is used.

Software costs

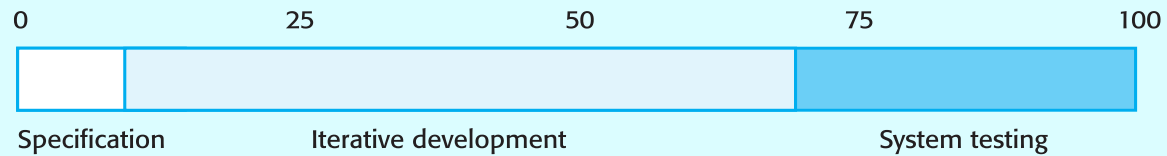
- Software costs often dominate computer system costs. The costs of software on a PC are often greater than the hardware cost.
- Software **costs more to maintain than it does to develop**. For systems with a long life, maintenance costs may be several times development costs.
- Software engineering is concerned with cost-effective software development.

Activity cost distribution for customized software

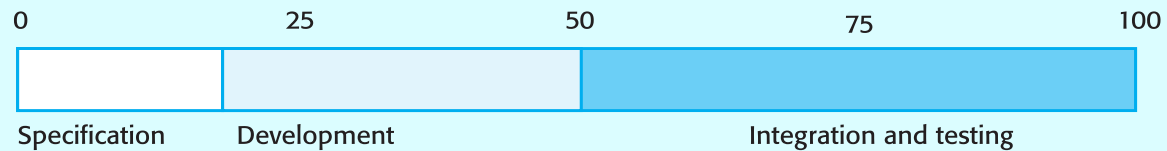
Waterfall model



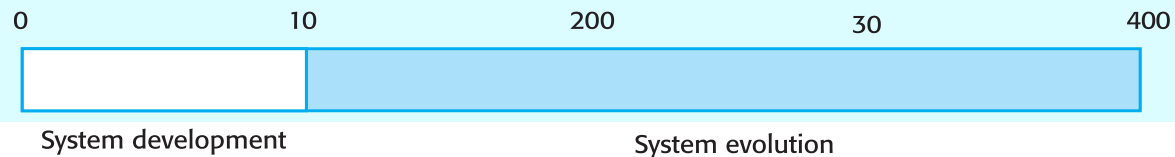
Iterative development



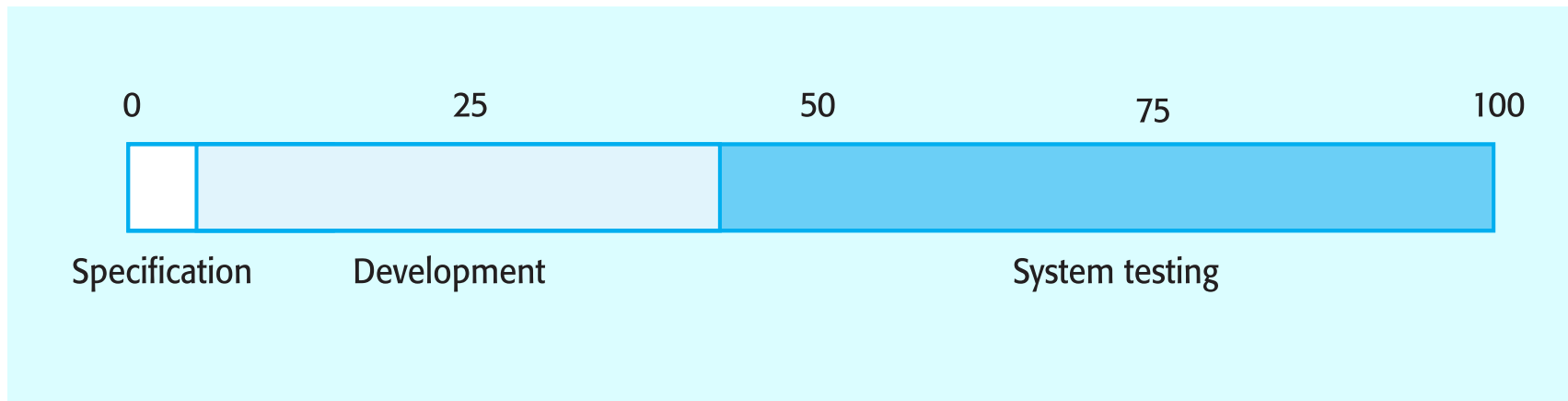
Component-based software engineering



Development and evolution costs for long-lifetime systems



Product development costs for (generic) PC based software



What are the attributes of good software?

- The software should deliver the required functionality and performance to the user and should be maintainable, dependable and acceptable.
- **Maintainability**
 - Software must evolve to meet changing needs;
- **Dependability**
 - Software must be trustworthy;
- **Efficiency**
 - Software should not make wasteful use of system resources;
- **Acceptability**
 - Software must be accepted by the users for whom it was designed. This means it must be understandable, usable and compatible with other systems.

What are the key challenges facing software engineering?

- Heterogeneity, delivery and trust.
- Heterogeneity
 - Developing techniques for building software that can cope with heterogeneous platforms and execution environments;
- Delivery
 - Developing techniques that lead to faster delivery of software;
- Trust
 - Developing techniques that demonstrate that software can be trusted by its users.

Professional and ethical responsibility

- Software engineering involves wider responsibilities than simply the application of technical skills.
- Software engineers must behave in an honest and ethically responsible way if they are to be respected as professionals.
- Ethical behaviour is more than simply upholding the law.

Issues of professional responsibility

- **Confidentiality**

- Engineers should respect the confidentiality of their employers or clients irrespective of whether or not a formal confidentiality agreement has been signed.

- **Competence**

- Engineers should not misrepresent their level of competence. They should not knowingly accept work which is out of proportion with their competence.

Issues of professional responsibility

- **Intellectual property rights**
 - Engineers should be aware of laws governing the use of intellectual property such as patents, copyright, etc. They should be careful to ensure that the intellectual property of employers and clients is protected.
- **Computer misuse**
 - Software engineers should not use their technical skills to misuse other people's computers. Computer misuse ranges from relatively trivial (game playing on an employer's machine) to extremely serious (dissemination of viruses).

Key points

- Software engineering is an engineering discipline that is concerned with all aspects of software production.
- Software products consist of developed programs and associated documentation. Essential product attributes are maintainability, dependability, efficiency and usability.
- The software process consists of activities that are involved in developing software products. Basic activities are software specification, development, validation and evolution.
- Software engineers have responsibilities to the engineering profession and society. They should not simply be concerned with technical issues.

References

- Slide with asterisks in the title are from “Object-oriented and classical software engineering”, authored by Stephen R. Schach, published by McGraw Hill