# An Introduction to Software Engineering

# What is software?

- Computer programs and associated documentation (e.g. requirements, design models and user manuals).
- 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?

- An engineering discipline that is concerned with all aspects of software production.
- Soft. 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.

# What is the difference between software engineering and computer science?

- Computer science is concerned with
  - theory and fundamentals;
- Software engineering is concerned with
  - the practicalities of developing and delivering useful and cost-effective software.

# What is the difference between software engineering and system engineering?

- System engineering is concerned with all aspects of computer-based systems development:
  - hardware, software and process engineering.
- Software engineering is part of this process
  - regards the software infrastructure, control, applications and databases in the system.
- System engineers are involved in:
  - system specification,
  - architectural design,
  - integration and deployment.

# 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
  - Evolution - changing the software in response to changing demands.

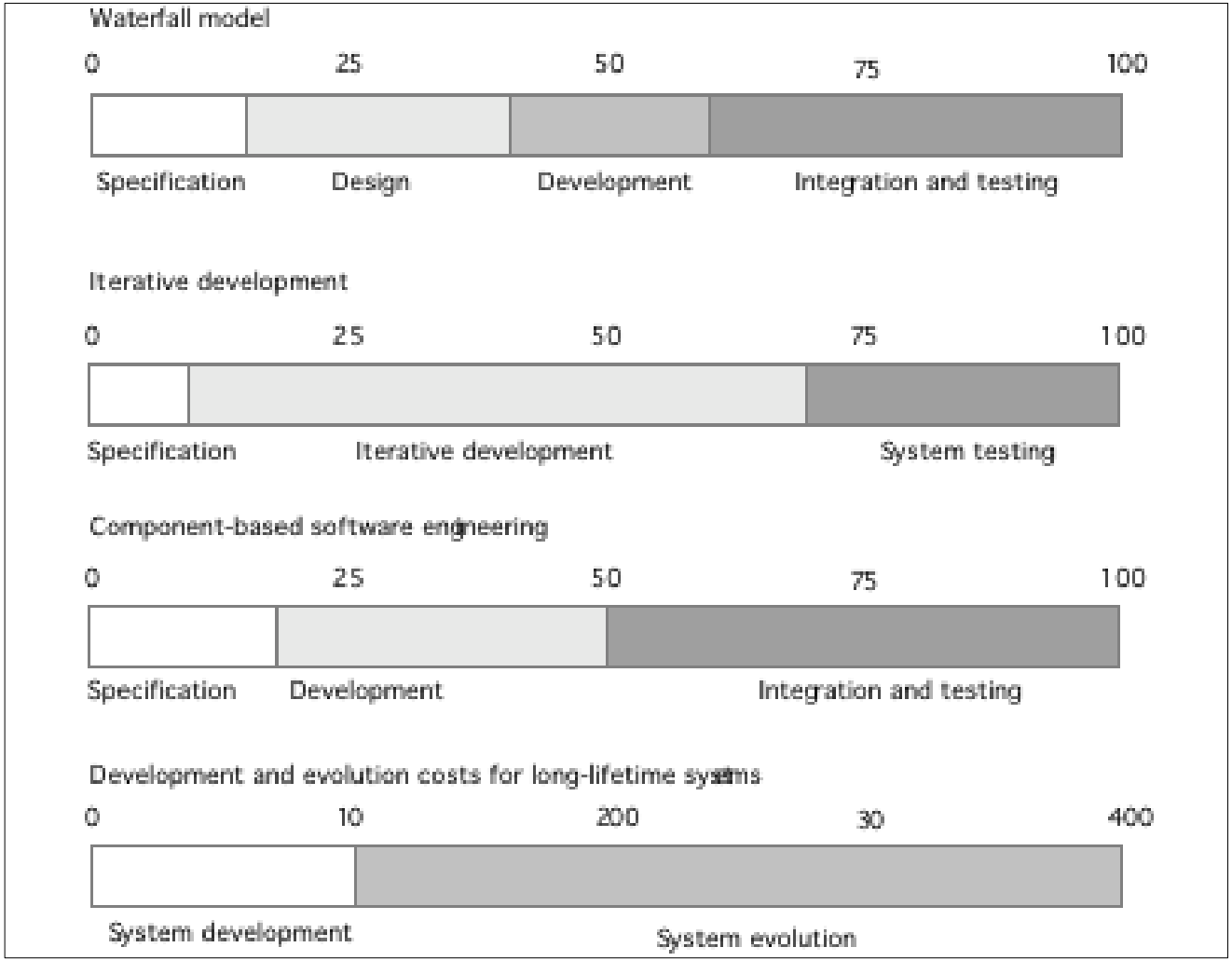
# What is a software process model?

- A simplified representation of a software process, presented from a specific perspective.
- Examples of process perspectives are
  - Workflow perspective - sequence of activities;
  - Data-flow perspective - information flow;
  - Role/action perspective - who does what.
- Generic process models
  - Waterfall;
  - Iterative development;
  - Component-based software engineering.

# What are the costs of software engineering?

- Roughly:
  - 60% of costs are development costs,
  - 40% are testing costs.
- For custom software, evolution costs often exceed development costs.
- Costs vary depending on:
  - the type of system being developed
  - performance and system reliability.
- Distribution of costs depends on the development model that is used.

# Activity cost distribution



# What are software engineering methods?

- Structured approaches to software development
  - Model descriptions and Notations
    - Descriptions of graphical models which should be produced;
  - Rules
    - Constraints applied to system models;
  - Recommendations
    - Advice on good design practice;
  - Process guidance
    - What activities to follow.

# What is CASE (Computer-Aided Software Engineering)

- Software systems that are intended to provide automated support for software process activities.
- CASE systems are often used for method support.
- Upper-CASE
  - Tools to support the early process activities of requirements and design;
- Lower-CASE
  - Tools to support later activities such as programming, debugging and testing.

# 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 accepted by the users for which it was designed. This means it must be understandable, usable and compatible with other systems.

# What are the key challenges facing software engineering?

- 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

- Soft. Enginnering is more than the application of technical skills.
- Engineers must behave in an honest and ethically responsible way
- Ethical behaviour is more than simply upholding the law.

# Issues of professional responsibility

- Confidentiality
  - Engineers should normally 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 outwith their competence.

# Issues of professional responsibility

- Intellectual property rights
  - be aware of local laws about intellectual property such as patents, copyright, etc.
  - be careful to ensure that the intellectual property of employers and clients is protected.
- Computer misuse
  - Eng. should not use their technical skills to misuse other people's computers.
    - From game playing on an employer's machine,
    - To extremely serious (dissemination of viruses).

# ACM/IEEE Code of Ethics

- The professional societies in the US have cooperated to produce a code of ethical practice.
- The Code contains eight Principles related to
  - The behaviour
  - The decisions
- …including practitioners, educators, managers, supervisors and policy makers, as well as trainees and students of the profession.

# Ordine degli Ingegneri

- Regolamentazione e Controllo della Professione
- Codice Deontologico
- …
- Tariffazione
- …. anche gli informatici possono iscriversi!
  - Settore: Ingegneria dell'Informazione

- … valutate bene!!!!

# Code of ethics - principles

- PUBLIC
  - Software engineers shall act consistently with the public interest.
- CLIENT AND EMPLOYER
  - Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.
- PRODUCT
  - Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.

# Code of ethics - principles

- JUDGMENT
  - Software engineers shall maintain integrity and independence in their professional judgment.
- MANAGEMENT
  - Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
- PROFESSION
  - Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.

# Code of ethics - principles

- COLLEAGUES
  - Software engineers shall be fair to and supportive of their colleagues.
- SELF
  - Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.

# Ethical dilemmas

- Disagreement in principle with the policies of senior management.
- Your employer acts in an unethical way and releases a safety-critical system without finishing the testing of the system.
- Participation in the development of military weapons systems or nuclear systems.