An Introduction to Software Engineering

Objectives

- To introduce software engineering and to explain its importance
- To set out the answers to key questions about software engineering

Software

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

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.

FAQs about software engineering

- What is software?
- What is engineering?
- What is software engineering?
- What is the difference between software engineering and system engineering?
- Why is so difficult to develop software?
- What do we have for complexity and change?
- What is software engineering in practice?
- What is a Model?
- What is the state of the art of software engineering?

What is software?

Software is: (1) instructions (computer programs) that when executed provide desired features, function, and performance; (2) data structures that enable the programs to adequately manipulate information and (3) documentation that describes the operation and use of the programs.

Kinds of software?

- 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.

Software Applications?

- system software
- application software
- engineering/scientific software
- embedded software
- product-line software
- WebApps (Web applications)
- AI software

Software—New Categories?

- Open world computing—pervasive, distributed computing
- Ubiquitous computing—wireless networks
- Netsourcing—the Web as a computing engine
- Open source—”free” source code open to the computing community (a blessing, but also a potential curse!)
- Also … (see Chapter 31)
  - Data mining
  - Grid computing
  - Cognitive machines
  - Software for nanotechnologies

What is engineering?

- **Engineering is the applied science of acquiring and applying knowledge to design, analysis, and/or construction of works for practical purposes; for solving problems [Bruegge].**

- **The *profession* in which a knowledge of the *mathematical* and *natural sciences* gained by study, experience, and practice is *applied with judgment* to develop ways to *utilize*, economically, the *materials and forces of nature for the benefit of mankind* [Accreditation Board for Engineering and Technology, ABET-1996]**

Origin of Software Engineering?

- Historical Aspects
  - 1968 NATO Conference, Garmisch [Bauer]
  - The Chaos Report: Standish Group
  - Aim: to solve the “Software Crisis”
  - Software is delivered
    - Late
    - Over budget
    - With residual faults

Scope of Software Engineering

- Why cannot bridge-building techniques be used to build operating systems?
  - Attitude to collapse
  - Imperfect engineering
  - Complexity
  - Requirements Change
  - Maintenance
  - Software Development is intensively human activity (Software Houses Yes, Software factories NO)

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.

[Sommerville]

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.* [Pressman]

Definición de Ingeniería de Software

Construcción de Software de Calidad bajo restricciones de tiempo y presupuesto [*1968]*

Ingeniería del Software es el estudio de los principios y metodologías para desarrollo y mantenimiento de sistemas de software

[Zelkovits, 1978].

Ingeniería del Software es la aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadora y la documentación necesaria requerida para desarrollar, operar (funcionar) y mantenerlos.

[Bohem, 1976].

Definición de Ingeniería de Software

Ingeniería del Software trata del establecimiento de los principios y métodos de la Ingeniería a fin de obtener software de modo rentable que sea fiable y trabaje en máquinas reales

[Bauer, 1972].

Software Engineering is a collection of techniques, methodologies and tools that help with the production of a high quality software system, with a given budget, before a given deadline, *while change occurs*.

[Bruegge, 2005].

What is software engineering?

- *Software Engineering: (1) The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software. (2) The study of approaches as in (1).* [IEEE]

What is the difference between software engineering and system engineering?

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

Why is so difficult to develop software?

- Different worlds (*from Analogic to Discrete*)
- Developers lack problem domain knowledge
- Clients/users don't know exactly what they need
- Problem domains are complex
- Software systems are complex creations
- Software development projects are subject to constant change (*difficult to manage*)
- Technologies change fast

Why is so difficult to develop software?

- **Complexity:**
  - The system is so complex that no single programmer can understand it anymore
  - The introduction of one bug fix causes another bug

Why is so difficult to develop software?

- **Change:**
  - The “Entropy” of a software system increases with each change: Each implemented change erodes the structure of the system which makes the next change even more expensive (“Second Law of Software Dynamics”).
  - As time goes on, the cost to implement a change will be too high, and the system will then be unable to support its intended task. This is true of all systems, independent of their application domain or technological base.

What do we have for complexity and change?

Complexity

- Requirements engineering
- Analysis and Design Modelling
- Use of Architecture
- System and Object Design
- Use Object-Oriented Technology
- Testing

What do we have for complexity and change?

Change

- Appropriate Communication
- Effective Management of the Project
- Process Engineering
- Management of Change
- Quality Management
- Appropriate Tools and Methods

What is software engineering in practice?

- Problem solving
- Knowledge acquisition
- Modelling

Problem Solving

- Software engineering is an engineering activity.
- It is not algorithmic.
- It requires experimentation, the reuse of pattern solutions, and the *incremental evolution* of the system toward a solution that is acceptable to the client.

Knowledge acquisition

- Knowledge acquisition is not sequential, as a single piece of additional data can invalidate complete models.

Modeling

- The purpose of science is to describe and understand complex systems, such as a system of atoms, a society of human beings, or a solar system
- There IS another type of system that we call an artificial system [SIMON]
- One of the basic methods of science is modeling

Modeling

- Software engineers deal with complexity through modeling, by focusing at anyone time on only the relevant details and ignoring everything else.
- In the course of development, software engineers build many different models of the system and of the application domain.

What is a Model?

- A model is an abstract representation of a system that enables us to answer questions about the system.
- Models are useful when dealing with systems that are too large, too small, too complicated, or too expensive to experience firsthand. Models also allow us to visualize and understand systems that either no longer exist or that are only claimed to exist.

What is the state of the art of software engineering?

Modelo de componentes de una profesión de ingeniería de software [FRoM]			
Conocimiento documentado	Educación	Profesión	Actividades
Cuerpo de conocimientos de la ingeniería de software (SWEBOK)	Modelos curriculares (ABET)	Certificación (IEEE-CS)	Conferencias y talleres (ACM-SIGSOFT, y IEEE-CS)
Estándares (ISO/IEEE)	Programas educativos	Licenciamiento	
Publicaciones (IEEE/ACM)	Acreditación de programas educativos	Sociedades de profesionales	
Procesos de trabajo (CMMi, ITIL)	Investigación (SEI, ESI)	Desarrollo e Innovación	
Código de ética (ACM/IEEE TaskForce)			