



Introduction

The people can't turn the modern world without software. National infrastructures and utilities are controlled by computer-based systems and most electrical products include a computer and controlling software. Industrial manufacturing and distribution is completely computerized, as is the financial system. Entertainment, including the music industry, computer games, and film and television, is software intensive. Therefore, software engineering is essential for the functioning of national and international societies.

Software systems are theorized and governed by physical laws, or by manufacturing processes. The lack of physical constraints, software systems can quickly become extremely complex, difficult to understand, and expensive to change.

There are many different types of software systems, from simple embedded systems to complex, worldwide information systems. It is pointless to look for universal notations, methods, or techniques for software engineering because different types of software require different approaches for software engineering.

There are still many reports of software projects going wrong and 'software failures'. Software failures are a consequence of two factors:

1. **increasing demands:** Systems have to be built and delivered more quickly; larger, even more complex systems are required; systems must to have new capabilities that were previously thought to be impossible. Existing software engineering methods cannot cope (manage in spite of difficulty) and new software engineering techniques must to be developed to meet these new demands.
2. **Low expectations:** Many companies have adopted software development as their products and services have evolved. They do not use software engineering methods in their everyday work.



Consequently, their software is often more expensive and less reliable than it should be. We need better software engineering education and training to address this problem.

The notion of ‘software engineering’ was first proposed in 1968 at a conference held to discuss what was then called the ‘software crisis’: the individual approaches to program development did not scale up to large and complex software systems. These were unreliable, cost more than expected, and were delivered late. Throughout the 1970s and 1980s, a variety of new software engineering techniques and methods were developed, such as structured programming, information hiding and object-oriented development. Tools and standard notations were developed and are now extensively used.

Professional Software Development

What is software? Computer programs and associated documentation. Software products may be developed for a particular customer or may be developed for a general market.

Many people think that software is simply another word for computer programs. However, software is not just the programs themselves but also all associated documentation and configuration data that is required to make these programs operate correctly. Documentations include system documentation, which describes the structure of the system; user documentation, which explains how to use the system, and websites for users to download recent product information.

Software engineering is intended to support professional software development, rather than individual programming. It includes techniques that support program specification, design, and evolution, none of which are normally relevant for personal software development.



The attributes of good software should deliver the required functionality and performance to the user and should be maintainable, dependable, and usable.

Software engineers are concerned with developing software products (i.e., software which can be sold to a customer). **There are two kinds of software products:**

1. **Generic products:** These are standalone systems that are produced by a development organization and sold on the open market to any customer who is able to buy them. Examples of this type of product include software for PCs such as databases, word processors, drawing packages, and project management tools. It also includes applications designed for some specific purpose such as library information systems, accounting systems, or systems for maintaining dental records.
2. **Customized products:** These are systems that are commissioned (authorized) by a particular customer. A software contractor develops the software especially for that customer. Examples of this type of software include control systems for electronic devices, systems written to support a particular business process, and air traffic control systems.

An important difference between these types of software is that, in generic products, the organization that develops the software controls the software specification. For custom products, the specification is usually developed and controlled by the organization that is buying the software.

Software Engineering

Software engineering is an engineering discipline that is concerned with all aspects of software production from the early stages of system specification through to maintaining the system after it has gone into use.