

What is Software Engineering?

- A disciplined approach to the production of software systems covering all aspects of the *software lifecycle*.
- The aim is to produce fault-free software that satisfies the user's needs and is delivered on time and within budget.

The term “software engineering” was coined at a NATO software development conference to address the perceived “software crisis”.

- It was stipulated that software should be built with as rigorous engineering techniques as bridges and buildings.
- However, software comes with a special and unique set of problems and challenges; consider, when was a civil engineer asked to port his bridge to a different river? Something that is routinely asked of software engineers!

To address these unique challenges software engineering draws from many different fields to create a disciplined approach to all aspects of software development.

Software Engineering and Architecture

Software engineering is not unlike the activity of architects and contractors.

- Start with a high-level description of what the *customer needs*.
- The architect will then build successively refined models in order to *understand* and *validate* the customer's wishes.
- Once the architect is sure what is needed, plans are drawn up and the contractor moves forward with the actual *implementation* – that is the construction.

Actually, this analogy is much deeper than it appears to be – recently the software engineering community has adopted “Design Patterns” as basic building blocks during software development. Design patterns were developed by architects in the mid 1970's in order to simplify design.

We will encounter design patterns later on in this course.