Is Software Engineering Really Engineering?

Graham Low
Deputy Director
Centre for Advanced Empirical Software Research
School of Information Systems
The University of New South Wales
Sydney, 2052, Australia
G.Low@unsw.edu.au

There are numerous definitions of software engineering. Sommerville [1] notes “their common factors are that it is concerned with software systems built by teams rather than individuals, uses engineering principles in the development of these systems and includes both technical and non-technical aspects”. The IEEE Software Engineering Technical Council in its web page “encourages the application of engineering methods and principles to the development of computer software”.

The common element in the preceding definitions is the application of engineering principles. However what is the generally accepted view of engineering. The dictionary [2] states that engineering is the “practical application of scientific knowledge in the design and construction of...”. Certainly software engineering involves the design and construction of software but the application of scientific knowledge is questionable. Before we decide that this rules out software engineering as a new branch of engineering, it is constructive to examine the history of engineering.

Classical and medieval engineers did not have a quantified basis for their designs [3]. In fact the scientific approach to technology did not emerge until the late eighteenth century and did not make a significant impact on engineering practice until late last century [4]. In fact up until then most technology was based on the traditional skills developed through experience and passed on by means of an apprenticeship system. In many ways this is where software engineering fits today.

The development of modern engineering is based on the increasing use of theory. While many engineering theories are based on scientific theories, many used in design “have to be content with a less firm foundation” [5]. For instance the design of stirred mixing vessels has no scientific theory as its basis, rather it uses dimensionless analysis to scale a known vessel design to a larger system.

Where does this place software engineering. I would argue that it fits in the craft stage of the engineering evolutionary cycle. What is required is the software engineering theories need to be developed. This raises the question as to whether current research in software engineering is designed to assist in formulating these theories. Our research should place more emphasis on formulating models, based on scientific principles wherever possible, and validating these models. Current research is too often based on statistical analysis of data without sufficient underlying theory. Only when we adopt such an approach will software engineering enter into the twentieth century of engineering.