PUTTING ENGINEERING BACK INTO SOFTWARE

TAKING AN ENGINEERING APPROACH TO SOFTWARE DEVELOPMENT COULD HELP IMPROVE ICT PROJECT SUCCESS RATES.

By Bill Malkin and Peter Hitchiner

There is a general perception that engineers are technologists/scientists who are just involved with building things. While engineers do build things, engineering involves so much more in essentially meeting the needs of the community in the broadest sense.

An engineer not only needs to deliver what the customer says is required but it needs to be fit for purpose and serve the needs of the community for its foreseeable lifetime. That requires resilience, scalability, adaptability and cost-effectiveness. The engineer needs to understand not only what the customer defines as the current need but also to understand the lifecycle of the requirement and the options for delivery and optimisation in the solutions, not necessarily (and typically not) what is at first defined as the requirement.

Engineering draws together architecture, planning (requirements), design, systems, implementation, verification, validation, commissioning, maintenance, upgrading and adaptation to changing requirements and changing environments; indeed, the entire lifecycle and including the economics of that lifecycle. It may include the means of marketing and selling the requirement (addressing the need and fitness for purpose) and it will certainly include attention to legal issues, whether contractual, OH&S or other regulations, and may also include dispute resolution and all forms of risk management – technical, legal and programmatic.

Engineering embraces the entire ecosystem, including interdependencies, of the product and its lifecycle: the successful engineer needs to be aware of this and act accordingly. Wikipedia defines engineering as "the discipline, art, skill and profession of acquiring and applying scientific, mathematical, economic, social, and practical knowledge, in order to design and build structures, machines, devices, systems, materials and processes that safely realise improvements to the lives of people" (http://en.wikipedia.org/wiki/Engineering). Engineering is all about the creation and delivery of products and services that are fit for purpose and meet user needs.

In response to questions at a recent meeting of persons of influence, a senior engineer of a leading Australian "engineering company" was reported as saying that he does not see the company's, or by inference his, job as one of "marketing or spin" but of simply creating what the customer has requested. Now, we are relying on a media report so will regard this as apocryphal; nevertheless, we were disturbed by the implication that, as so often occurs, engineers think their role is contained in a space which has boundaries of technology and building things. This seems to be symptomatic of a malaise which affects many projects: that of insufficient attention or understanding of the holistic role of engineering in delivering successful outcomes.

We have no doubt that the leaders, such as John Bradfield, behind successful iconic projects such as the Sydney Harbour Bridge, and those who created the Snowy Mountains scheme, had more than technology and building things on their minds when they delivered those projects. They were engineers and visionaries: they may not have foreseen their impact today but something must have been in their minds which allowed those projects to continue to enrich our lives.

'Scientists investigate that which already is; Engineers create that which has never been.' –Albert Einstein

Successful engineers such as Sir John Monash, the extraordinary military leader and engineer after whom the university is named, were not just scientists, technologists and builders – they were the best engineers and leaders in a very broad sense.

Furthermore, this goes to the core of professional engineering competency standards. Engineers Australia is in the process of reviewing its Stage 2 Competency Standards (CPEng). One of the items is "responsibility for decisions", which describes a professional engineer's element of competency to "provide technical and commercial leadership and be responsible for making decisions on part of or the whole of the engineering activities". There is more of course. When projects do not adopt a comprehensive and holistic engineering approach, we should not be surprised if projects frequently deliver surprises and enter the failure category.

In his Charles Todd Oration in 2011, Telstra's CEO David Thodey lamented the lack of leadership in the ICT industry. He might well have targeted the engineering profession for not providing current leadership in the ICT industry.

So what has this to do with software engineering? Software engineering is, or needs to be, booming in Australia. However, on the face of it at least, this is not immediately apparent. Very few people currently have the job title of software engineer. For some very strange reason, *software engineers* prefer to call themselves *technical architects*, *Java architects*, *.NET architects, developers*, etc – anything in fact, as long as it has the word "*architect" or "developer"* in it. However, this seems to represent a seemingly small part of what a software engineer should be doing.

Twenty five years ago (when the Australian Software Engineering Conferences, ASWEC, were established) software engineers all demanded the word "engineering" to be in their job title; now "architect" is de rigueur. Despite the fact that many of us are (or should be!) following a software engineering process in order to perform our work well, many still much prefer, like George Costanza in *Seinfeld*, to be called an architect.

Why is this? If we look at the history of architecture and engineering in the building industry, we see that the very same confusion existed for quite some time in that industry also.

Although the distinction between architects and engineers seems obvious now, this distinction has only arisen relatively recently.

The division between the two roles began to occur in the 18th century. First of all, to distinguish the engineer who carried out similar tasks to a military engineer but in a civilian context, the two professions of civil engineering and military engineering came into existence (Engineers' Council for Professional Development definition, *Encyclopaedia Britannica*). Then in 1818 the Institution of Civil Engineers was founded in London. The institution received a Royal Charter in 1828, formally recognising civil engineering as a profession.

Civil/structural engineering covers a very wide area of construction. Civil/structural engineers who specialise in working with architects in the building industry are sometimes referred to as architectural engineers. This creates some confusion as we now have the word architect associated with an engineer, although the architectural engineer does not perform any architectural duties and, indeed, is not permitted to by law.

The question that now needs to be answered is, "what were the differences between the two professions that caused them to split and form two, very separate, professional organisations?"

This question has a very simple answer: architects are required to answer the *what* question (appearance and functionality), and engineers are required to answer the *how* (fitness and functionality). It is the responsibility of the architect to ascertain and document what the client's functional requirements are and what the client wants its form to be. It is the responsibility of the engineer to determine how those functional requirements and form are to be implemented and ensure fitness for functional purpose as currently envisaged and to enable resilience in use and scalability and adaptability in the future.

'Those who cannot remember the past are condemned to repeat it.' –George Santayana

Software engineering, the *how*, is the application of a systematic, disciplined, quantifiable approach to the development, operation and maintenance of software, and the study of these approaches; that is, the application of engineering principles and practice to software.

If ICT projects are to be delivered successfully, that is safely, effectively, efficiently and sustainably, it is essential that the rigours of engineering discipline be applied; not just architecture but also the holistic structure required by an engineering approach.

The poor support for the organisers of the Australasian Software Engineering Conference (ASWEC) 2012 may be a symptom of the current low level of appreciation of the important role of software engineering for rigorous, reliable and affordable ICT systems and at the same time the basis of a thriving ICT industry. ASWEC has been an almost annual event sponsored by ACS and Engineers Australia for over 20 years.

Software engineering research, practice and education has never been more important than it is today. ASWEC needs your support: please contact the authors if you agree with these philosophies and would be happy to become involved to further their adoption. ■

Authors Peter Hitchiner FIEAust, CPEng (Member ACS Telecommunications Board and Chair ITEE College Board, Engineers Australia) and Bill Malkin MACS (Snr), MIEAust (Qualified Civil, Systems and Software Engineer; Convenor, Software Engineering and Architecture JTP SA; and Consultant with DWS Advanced Business Solutions) can be contacted at peter@insighteleconsult.com.au and bill@malkin.com, respectively.

The authors acknowledge with thanks feedback from all especially that from Dr Jocelyn Amarego and Chris Skinner.