The great software nightmare
By Bruce McCabe

Published in *The Australian* 20 June 2006

It's the IT Manager's biggest nightmare. The big software project that costs millions more than budgeted, falls years behind schedule, and never ends up doing what it is supposed to do.

The nightmare occurs again and again inside large organisations. Like groundhog day, we seem doomed to repeat the mistakes. Occasionally, when a large or strategic project fails, it makes headlines, compounding the manager's misery. Just last week, Booz Allen Hamilton released a scathing report on the Integrated Cargo System (ICS) software developed by the Australian Customs Service (this system was at the centre of a crisis when containers were not being cleared fast enough from Australian ports last October) criticising the testing procedures and the "big bang" implementation approach. But ICS is only the latest in a long line of big software projects that have run into big problems. We can be certain there will be many more.

In the consulting community there is no shortage of people willing to pretend they know how to do it better. Unfortunately, what they peddle is, for the most part, worthless. Long on PowerPoint slides and fluffy advice about "what" needs to happen (and nine tenths of this guidance consists of recycled flow charts restating the bleeding obvious) but short on evidence and detailed information on "how" to really do those things in practice, which is of course the hard part. Consultants from the most prestigious organisations are among the worst culprits. The proof is in the pudding – armies of them fail, time and again, to avoid the problems that derail software projects.

Which is why it was so refreshing to chat recently with two scientists that are (a) focussed on producing practical, actionable tools that managers can apply in the very early stages of projects, (b) determined to use substantive research to ground their understanding of why software development projects succeed or fail, and (c) are prepared to back their findings and challenge some of the accumulated, accepted wisdom in this space.

The two scientists were Professor June Verner and Dr Steven Bleistein, members of the software engineering group at the National ICT Australia (NICTA) institution. We spent a morning at the Australian Technology Park discussing their conclusions after reviewing 400 software projects in Australia, the US and Latin America.

More than a few of those conclusions challenge conventional thinking. They found that UML, for example, did not help. This was fascinating to me as in the past few years I have heard managers talk up UML absolutely everywhere. UML, or Unified Modelling Language, provides a standard way of describing software objects and is associated with standardised methods for software design. Plenty of people think of it as a magic bullet. Verner and Bleistein, however, found that projects where it was used were just as likely to fail, and were possibly even more likely to fail, because managers tended to use UML as a surrogate for other factors.

They also found that the most experienced project managers were not the most successful: the personal qualities of the project manager far outweighed the benefits of experience. All decision-makers know that having the right project manager is critical, but it appears that priorities are often wrongly applied when they choose one.

A recurring theme in our conversation was how critical failure points in a software project are frequently outside the direct control of the project manager. Critical cost and timetable estimates are often completed before a project manager is even appointed. Many initiatives are doomed from the start because project managers work to a vision quite different to that held by senior executives in the business.

Verner, Bleistein and their team at NICTA are focused on turning their work into practical, commercialisable tools. I don't think commercialisation will be a problem. The business value of their work is self-evident and, more importantly, Japanese, Indian and a few US companies are already lining up and asking this group to bring its expertise to their internal software projects. NICTA is also placing a special emphasis on building insights into projects where software development is partially outsourced, which is incredibly important right now.

Verner and her team are currently looking to work directly with more Australian companies and government agencies. IT Managers should take advantage of that -- it could stop some of their nightmares coming back.

Bruce McCabe has researched technology trends since 1995