

Opinion: Enterprise architecture - dogmatic and over-ambitious?

Disdain for legacy systems, lack of business focus... Peter Kemp and Dr John McManus MBCS were disappointed at the disorganised, unstandardised and disintegrated IT landscape that emerged from the move away from mainframe to client/server and web-based systems. Has anything changed?

When we first learned about the concept of enterprise architecture, it appeared to be just what the IT world had been waiting for but, we were disappointed. Tackling a business's IT portfolio from a strategic, centralised perspective appeared to be the only way forward. A decade later, however, is it time to reappraise the direction enterprise architecture is leading and to re-evaluate the benefits that it has realised?

Our disquiet over where enterprise architecture (EA) was taking us began in 2004 when we were attending a presentation on enterprise security by the EA team for a major government department. When it dawned on us what was being proposed we asked 'does this mean that we can make no progress on security until everything has been rewritten in Java?' To which the stunning answer came: 'we can't make progress on anything until we've rewritten everything in Java!'

On the same project, in another presentation, the legacy application estate was described as 'the slime'. We felt very uneasy about this for a number of reasons, not least that we don't share the common EA disdain for legacy systems.

The same year we saw the EA for another major public sector body that showed everything across the enterprise composed of sea of Java components. The thing that struck us was that it would have been impossible to tell to which sort of organisation the proposed EA applied. There was simply no business-focused element to it at all.

As we had been working for that body for over a year, we had formed some ideas of what an EA ought to look like and that it ought to reflect at least five significant business areas, with very different business, technical and operational characteristics: from real-time, mission-critical, safety-critical systems to standard HR, Payroll and Finance systems, to public-facing informational websites. If EA was proposing that these things were all homogeneous, then surely something was badly wrong.

In the main, we put these early observations down to the EA practitioners in question: they had not understood EA properly, at the very least from the point of view of proposing an EA that was practically achievable. But, we have to say, that were not sure we've yet seen an EA strategy that is anything other than impractical, unachievable and, even if it could be achieved, unsustainable. Yet, EA seemed to have gained ground as a hugely beneficial concept and has perhaps reached an apotheosis in the much debated NPfIT (National Programme

for IT) in the NHS.

So, perhaps unfairly, how would we characterise EA (given that EA strategists usually have nothing but contempt for legacy systems and anything that falls outside their view of what applications are acceptable):

- Technology-led, with standardisation of applications, systems and technologies used as driver for enforced business change;
- Dogmatic, in the sense that nominal standardisation at an enterprise level is seen as a more important goal than meeting end users' real requirements;
- Over-ambitious, in the sense that few EA strategies seem to be able to stop short of a idealised, perfect scenario;
- Unverified, in that no one has properly analysed the achievability or sustainability of the proposed EA;
- Divorced from the current state, in the sense that although the current state is usually shown, there is no analysis of how the first steps can be taken from the current to the idealised future state;
- Futuristic, in the sense that the EA strategy plans so far in advance that it doesn't sensibly guide the immediate next IT strategy steps;
- Politicised, in the sense that things are reduced to sound-bites and perceptions and not judged on hard analysis of benefits and weaknesses.

In summary, we would characterise the EA strategist as a chess player who has a long-term strategic view of the perfect position, but who disregards his current position and dismisses with contempt (as a short-term tactical necessity) each and every move that can practically be made.

In other words, there is simply no way to reach the desired position given the starting point and the possible next moves at one's disposal. And, perhaps to go further, the EA strategist blames the current position and the moves available without considering that their strategy itself might be wrong. With the same given starting position and using the available moves as a guide, one could construct a viable and beneficial EA strategy.

In a sense, however, the problems engendered by EA go much deeper than this. The question must be asked of how practical it is for one EA team to think, work and operate at an enterprise level and to make decisions that are beneficial to the business without working on specific business problems.

Not only do analogies with the centrally planned economy come to mind, but also questions about whether the scope of the problem is just too broad. How is it possible to judge the viability and suitability of an EA when it is by its nature abstracted to a very high level? It's tempting to say that the devil is in the detail and EA strategies are prone to proposing an architecture that proves impractical.

One example of this can be seen in the NHS, where the strategy for sharing digital medical images (X-rays, MRI scans and so on) across the NHS assumed that if all the images could be stored in a central image database, then the problem would be solved. The EA strategists did not think any deeper than this because they didn't have to.

But, once the problem is tackled from a practical 'how can digital medical images actually be shared effectively', it does not take long for the architect to realise that more than just the raw images are needed and that, in fact, a range of related patient information must be coordinated along with the images.

One could argue that the above is simply an example of a mistake and that it applies equally to all technical design: a high-level design can always be scuppered by lower level detail. But, it seems to us that EA has deliberately avoided any contact with the real world, partly though the extreme high level at which EA work and partly though the de facto principles that govern EA.

Technical architects at the project level do, in my experience, concern themselves with the next level of detail and, as far as possible, propose an architecture that will survive the detailed design process. On the other hand, enterprise architects do not do this, in our experience. Instead, the onus is placed 100 per cent on the project architects to deliver within an EA that may be flawed, impractical, constantly changing or even internally inconsistent.

Moreover, an EA can be generated relatively quickly. A five-year IT strategy can be generated by an EA team in 3-6 months, perhaps. What happens then? Do the EA's start to test their EA by monitoring its adoption by various projects? In our experience, they do not. Instead, they begin to think further ahead or in increasingly blue-sky terms. The five-year strategy becomes a ten-year strategy and the dogma tightens even further.

Another problem

Another issue we see with EA as it is practised today is the sustainability of the target architecture, which often involves a massive and unrealistic amount of consolidation and homogenisation. We have started to ask of EA strategies: if that were achieved, how could it be developed or upgraded? And often the answer is that it would be problematic, to say the least. Often business functions and applications are consolidated into multi-faceted integrated systems, directly used by many thousands of disparate end users.

Even to consider how an upgrade might be approached is not an easy question. Increasingly, we can see these integrated, enterprise-wide components becoming inert, stuck at their initial implementation and just too big and too all-encompassing to be upgraded.

Finally, there may be a case that diversity in IT systems (even within an enterprise) is a good thing. Standardisation assumes, to some extent, that functional richness and specificity can be achieved within a pre-defined standard framework. But is this true?

In the same way that a large organisation is staffed by diverse human beings, all with their own skills, strengths and weaknesses, why shouldn't the IT systems display a degree of diversity? In terms of IT systems, we are not talking here about commodity products (such as PC's, word processor or email clients), we are talking about applications that work hand-in-glove with the business users who carry out diverse business processes. This would imply, in our view, the EA

recognising diversity across business or departmental boundaries much more than any EA we have seen to date.

If one business process involves a real-time, safety-critical activity where lives are at stake and another involves the backroom processing of overtime payments, then why should these applications be constrained by the arbitrary platform, technology and user interface standards invented by an EA without any real knowledge of the business requirements and any specific analysis of the business benefits that could be gained from deployment of this or that IT system?

On this point, it seems to us, that EA has extended far beyond interoperability standards and specific standards towards a detrimentally generic standardisation of everything that can be standardised. Perhaps, in fact, an enterprise architect should be more concerned with understanding the genuine differences in business and IT requirements across an enterprise, rather than shoe-horning every corner of the business into a one-size-fits-all EA.

In conclusion, we don't have answer to how beneficial an EA approach can be to a business or government department. In many ways, having an enterprise-wide view of things can only be an advantage. We are, however, increasingly concerned that EA concepts are being pushed beyond what can be justified by a business-wide cost-benefit analysis and that EA taken too far can adversely affect the ability to deliver useful business functionality to end users.

In particular, we believe that enterprise architects should be asked to justify their strategies in terms of overall concrete business benefit much more and to provide far more assurance that the proposed EA is viable and does not unnecessarily constrain the ability of the business to benefit from IT systems and technology innovation.

January 2009

[Legal and Privacy Notices](#) © Copyright BCS 2009 [Systems Status](#)

The BCS is a registered charity: No 292786 Patron: [HRH The Duke of Kent KG](#)