INSIDE TRACK
Analyst commentary with a real-world edge

FREEFORM DYNAMICS

Demystifying architecture for non-enterprise organisations

Have you got a map?

By Jon Collins, April 2010

Originally published by

The Register®
Biting the hand that feeds IT

This articles was originally published on The Register news and information site as part of an Infrastructure workshop (http://www.theregister.co.uk/hardware/infrastructure/).

Architecture, schmarchitecture... words like "architecture" get banded around in IT like they have a specific meaning, when in reality you'd get ten different definitions from ten different people if you asked. But few would agree that IT should do without some degree of 'structure', be it mere forethought or understanding about what is required or in place.

IT architecture, in principle, is a good thing. Indeed, when we asked respondents just over a year ago about the role of the IT architect, we found that while plenty were dissing them, few had a problem with the idea of IT architecture *per se*.

The trouble is, if you actually decide you want some of it ("we have IT architecture in several colours, sir") you almost immediately hit a brick wall. Documented approaches to IT architecture tend almost immediately to big frameworks and complex, boil-the-ocean approaches that are only really going to be applicable in larger organisations - and even then probably only in moderation.

Of course, I'm saying this in the knowledge that "experts in the field of IT architecture" would push back on the idea that it needs to be hard, onerous, over-complicated or whatever. In my experience, people with 10+ years of experience in any field will be able to pull out the good bits and make it suitable for the need at hand: such people will also be perceived as having sufficient gravitas to be trusted, so that what they say will often be implemented.

However, for the majority of non-enterprise organisations which don't have access to "the knowledge", either in document or guru form, deciding where exactly to start with IT architecture can be difficult: it's a wood-for-the-trees issue.

Where to start with IT architecture?

The good news is, it doesn't have to be so hard, starting with the recognition that IT architecture is a means to an end. Poorly architected IT environments tend to suffer from more downtime than those for which resilience has been built in from the start (as illustrated in one of our reports [1]). They will also be more costly to administer and deliver lower levels of service. IT architecture is not the solution to all these ills – just having a well structured environment will only get you so far. It does however offer the starting point for doing a better job, more efficiently. So, a key first element is to understand:

- What you have in place in terms of IT systems and resources, physical and logical

- What services you are delivering to your business users
- What information repositories you have
- And what dependencies exist between all three – systems, services and information

This much may sound obvious enough, but we are frequently reminded that few organisations actually have a clear understanding of any, never mind all of the above. Attaining this understanding needn't be a very onerous exercise, either, particularly if you start with the Pareto principle in mind: can you identify the 20% of your IT facilities that deliver 80% of the value?

Key to this kind of exercise is to remember the 'means to an end' factor. If you haven't already documented what you have, we're pretty confident that just doing so will turn up some things which merit attention. If so, you're already in a far better place to take forward existing and planned projects.

The next step: Understanding service delivery characteristics

The next level of "architectural capability" – if we can call it that – takes us from the map of what exists today towards an environment which is most appropriate to your own business needs. For example, your systems may need to support high volumes of transactions, or perform to specific availability criteria. The key question here is "What services does your organisation want to provide, and therefore, what services does IT need to support?"

A few years ago I was involved in an exercise extending a software development methodology (RUP, if anyone's interested) to take into account the broader context of business and IT. At one end, this required an understanding of what the business was trying to achieve – in process terms, or quite simply, in terms of the services it wanted to provide its customers. At the other, it required taking into account the types and structures of systems that would be needed to support the resulting applications: would the database need to run on clustered hardware, for example? Would a failover system need to be in place, and/or would load balancing be necessary across multiple systems?

It's answers to questions such as these that can help dictate what kind of IT architecture you may need in place or wish to move towards. Today, many answers exist that didn't before. For example, what we now see as server virtualisation was a mere twinkle in VMware's eye just a few years ago. As was the idea that some services may be better delivered from third-party, hosted or 'cloud' solutions. But even with such innovations, it's important to start with what you are trying to achieve, rather than what is available out there.

Delivering on IT architecture

All said, IT architecture doesn't have to be onerous. Indeed, if you work out all the things you are trying to do and then map them against what you have in place, you may find that you want to keep things as they are. Equally, however, you may spot some incremental improvements you can make to your existing environment, or indeed, you may decide more drastic action is required. In all cases, however, at least you will be starting from a clearer point than, "let's just do it and see what happens" – although most of us will have been guilty of that at some point.

As a means to an end, IT architecture has merit. Indeed, as you build your understanding of what you have, you may turn to well-crafted frameworks such as TOGAF and appreciate their merit.

In the meantime, and especially for non-enterprise organisations, to paraphrase software 'guru' Kent Beck, it's important to recognize that "enough architecture for now is probably enough architecture for now".

[1] Risk and Resilience. The application availability gamble.

(http://www.freeformdynamics.com/fullarticle.asp?aid=373)

About Freeform Dynamics

Freeform Dynamics is a research and analysis firm. We track and report on the business impact of developments in the IT and communications sectors.

As part of this, we use an innovative research methodology to gather feedback directly from those involved in IT strategy, planning, procurement and implementation. Our output is therefore grounded in real-world practicality for use by mainstream business and IT professionals.

For further information or to subscribe to the Freeform Dynamics free research service, please visit www.freeformdynamics.com or contact us via info@freeformdynamics.com.

Terms of Use

This document is provided for your general information and use only. Neither Freeform Dynamics Ltd nor any third parties provide any warranty or guarantee as to the suitability of the information provided within it for any particular purpose.