

“If IT is from Mars, Business is from Venus, where does that leave PMO?”

Mervin Chiang, Manager Product and Service Development, Leonardo Consulting
Edward Goh, Consultant, Leonardo Consulting

“We’ve just bought a business process management suite, also known as a BPMS. The vendor promises that now businesses can model their processes to tell us what they want and, with a push of a button, we’ll get the blueprints for delivery! *Voila!* No more misunderstanding or scope creep. We’ll be aligned!” Well, yes, the technology is aligned. There is this mythical button. But, just like a marriage, is it really a silver bullet in this cosmic relationship? Or is there something deeper?

Those of us in the IT industry would have experienced this encounter between Business and IT. Business, based on customer needs, asks for additions or changes to scope when the development project is nearing completion. IT looks at the change requests, thinks about all the re-work that must be done to accommodate the changes (and all those wasted earlier efforts), and thinks to itself, “This is so out of the initial agreed scope, why didn’t Business say so in the first place?”

What we need to recognise here is that the nature of the two sides are inherently different (which brings us to the analogy in our chosen title and its familiarity to some bickering comments from each party). Business needs to respond quickly to market demands and changes. IT needs a defined scope of work, in the form of fixed requirements and proper service contracts, to ‘box up’ the efforts they are going to put in. In that regard, IT does not like change. IT thinks Business is fickle-minded and naive to think change is so easy. Business thinks IT is too rigid and wonders how it will win further sales with a support like that.

Yet, in today’s fast-moving business environment, companies need to be agile and responsive to stay ahead of the competition. They need to constantly adapt their products and processes to meet the customers’ changing needs. Software development and system implementation are integral parts that many companies use to support their business activities – ‘necessary evils’, if you will. However, these companies are finding that one of the biggest obstacles to their business agility is very ‘evil’: that the development or implementation processes used by their IT departments or vendors are too slow and inflexible.

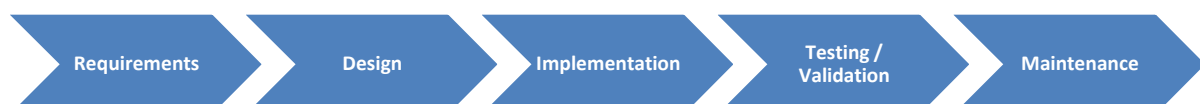


Figure 1: Typical Waterfall Software Development Model

Traditionally, software development is based on the waterfall model, which typically goes through a strictly sequential or cascading flow, as presented in Figure 1. The idea behind the waterfall model is to prevent re-work by getting things right upfront; hence, it discourages revising any work done in previously completed phases (a source of criticism by some for its ‘inflexibility’ and ‘barrier to innovation’). Unfortunately, this seldom happens in reality. Business usually struggles to specify the requirements fully and correctly at the beginning, as many of the requirements only surface through discovery as the project progresses – when IT starts.

Why does that happen? Well, first, many requirements may not be anticipated in the early planning stages of the project. Business planners may not be fully aware of the capabilities of the technology being implemented until they see it several times; they may not know exactly what requirements are truly needed before reviewing a working prototype. The result is that change requests or re-scopes happen at the most expensive time: nearer the end of project. Project timelines get dragged, more resources are pulled in than planned for, and the cost of the project exceeds the original budget.

Interestingly, for the projects that finally do get implemented, 45% of application features and functions are never used (*Standish Group Study Reported at XP2002, Jim Johnson*).

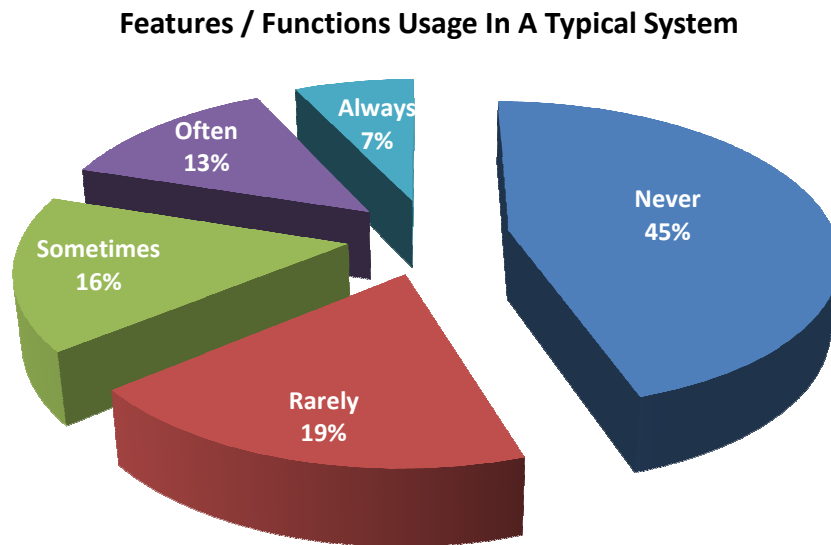


Figure 2: Standish Group Study Reported at XP2002, Jim Johnson¹

As a way around this conundrum, IT tries to use agile methodologies (e.g. Scrum, Extreme Programming), based around iterative and incremental delivery, which aims to solve the issue of inflexibility around the traditional waterfall development methods. Instead of following a regimented plan, agile methods recognise that software development is inherently a learning process; it adopts a more people-centric viewpoint that uses iterative feedback through regular tests and releases of the evolving software to deliver the project. By adopting these agile approaches, companies can respond more swiftly to customer needs and adapt more effectively to change.

The Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Figure 3: Manifesto for Agile Software Development²

You'd think, "Aha! Agile would solve the problem! Give Business what it wants in short iterative cycles. Keep evolving, keep adding ..." Well, along comes the Project Management Office (PMO) that is now charged with the task of consolidating the IT projects' portfolio, managing them in a central office, and standardising project management practices. Companies that recognise the need for a governance structure will soon realise that PMOs find it increasingly hard to govern changing scope or continuous adjustments (as opposed to phases with defined endings) within the agile approaches – and get confused as all the requirements 'ping pong' between the stakeholders. They work better in waterfall. We just went one full circle!

Fundamentally, it entails a mindset change of the people involved for the agile approach to succeed. The methodology or team structure might have changed many times, but the mindset of the people has not. Developers still fall back to the old thinking, expecting fixed requirements, and working according to that expectation. Customers still hold the traditional concept of 'I hire you to do this for me' rather than 'I work with you to complete this'. The divide still remains between the various parties; they are political or cultural – in this case, IT technology cannot solve the problem!

It is said that knowledge is the first step to recovery. Many times, in our experience, the only way to tackle the political and cultural issues and to achieve this symbiosis of governance around all parties would have to be from the top – it would take the endorsement of all top-level management for this to work. BPMS, or any process-related system implementation, has to be jointly owned and a collaborative effort for all parties. All need to embrace the fact that this is an adaptive process, and engagement practices between the stakeholders will have to be different. In fact, we believe that PMO practices need to be reviewed to accommodate a hybrid model between waterfall and agile to support this initiative: “WaterCrumb” – a cross between waterfall and scrum? Clear governance needs to be defined for the roles and hand-offs – trust, alignment and commitment from all sides are critical to success. These are the essential ingredients. If these are absent, then it becomes notoriously hard to govern any agile process automation for continuous process improvement through BPMS.

About the Authors

Mervin Chiang

Mervin’s experience in software integration, customisation, service and process design enables him to approach business process management from an information technology perspective. This includes tool customisation, training, business analysis, integration and implementations using ARIS methodologies and service-oriented architecture concepts. Mervin can be contacted on m.chiang@leonardo.com.au.

Edward Goh

Edward has more than 10 years of IT industry experience in various roles, ranging from pre-sales, support and operations to business planning. Before joining Leonardo Consulting, he was a senior business analyst responsible for strategic planning, change management, driving new business initiatives, and process improvements. Edward can be contacted at e.goh@leonardo.com.au.

References

¹ *Standish Group Study Reported at XP2002*, Jim Johnson.

² *Manifesto for Agile Software Development*, Agile Alliance.