

---

# Learning to Innovate? Begin at the End

RAF CAMMARANO

**“Good judgment comes from experience, and  
experience comes from bad judgment”**

Unknown



» The innovation lifecycle consists of five steps: ideation, selection, development, implementation and refinement.

» Most IT Department's begin their innovation efforts at ideation or development. But it is possible to begin at implementation and refinement.

» 'Launch and learn' is a central tenet of innovation. It covers the last two phases of the innovation process: implementation (launch) and refinement (learn).

» Implementing industry best practices and delivering business projects are two areas where IT can launch and learn.

» There's no easy way of becoming innovative. Mistakes will be made along the way. But ultimately the only way to develop good judgement is to make those mistakes and learn from them.

CIOs who lament the growing irrelevance of their IT Departments and their descent into a commoditised, order-taking role must understand that a myopic focus on operational excellence will only accelerate this demise. In business it's often said that the race to "lowest-cost" can only have one winner. That winner won't be your IT Department. External specialists will always trump internal generalists.

To remain relevant, IT Departments must learn to innovate.

To become proficient in any skill, including innovation, you must combine a sound theoretical foundation with practical experience. There is plenty of theory available on



innovation but the only way to become innovative is to actually do it. You need the theory and the practice, and neither is an effective substitute for the other.

Developing innovation capabilities is often a lengthy, risky and costly exercise but luckily, IT Departments are in the enviable position of being able to begin developing this capability almost risk-free.

## The Innovation Lifecycle

To be an effective innovator you must be proficient across the entire innovation lifecycle<sup>1</sup>:

- **Ideation:** generating and capturing ideas
- **Selection:** selecting the best ideas to pursue
- **Development:** transforming nebulous ideas to tangible solutions
- **Implementation:** market testing and full-scale deployments of innovations
- **Refinement:** continuous improvement

Successful innovation is hard because critical decisions need to be made at each stage of the lifecycle, and often they must be made with minimal information. Since most IT Departments' risk aversion manifests itself in 'analysis paralysis', it's not surprising that IT Departments aren't great innovators.

## Where to Begin

Most IT Departments begin their innovation journey in one of two ways.

The first, and most obvious, place to start is at the beginning: ideation. Such Departments kick off their innovation efforts with great fanfare by creating mechanisms for capturing good ideas (websites for submitting suggestions, creativity competitions, roadshows, brainstorming sessions and collaboration events).

The second place many IT Departments start is as the development phase. Technologists are comfortable with research and development and there aren't many people sitting in IT Departments who wouldn't jump at the chance to sit in a corner and tinker with new technology.

---

<sup>1</sup> This linear product innovation process is slightly different from the iterative service innovation cycle but for the purpose of this article the distinction is irrelevant.



Each of these approaches has its merits, but also limitations.

Starting with ideation has the advantage of generating many ideas, some of which will be worthwhile. The problem is that the IT Department typically doesn't have the capabilities or the resources to convert those good ideas into useful innovations. The initial innovation fanfare and fervour slowly morphs into disillusionment and cynicism as nothing tangible comes from all those ideas.

Starting with development means the IT Department can quickly create working solutions. Eliminating technical risk is a good thing, and nothing gives the innovation effort a shot in the arm like working solutions, even basic prototypes. The main drawback to starting at the development phase is that innovation becomes technology-led, with IT generating solutions in search of problems.

So if you can't start at the beginning of the innovation process (ideation) or in the middle (development) why not start at the end?

### **Starting at the End**

Starting at the end of the innovation pipeline would be nonsensical if you didn't have developed ideas coming through the innovation pipeline. Luckily those ideas exist – and they have been developed to the point that they are ready to be implemented. Two sources of such gift-wrapped, ready-to-implement solutions are:

- Industry best practices
- Business projects

Mention innovation to most IT staff and it conjures up images of garage-based start-ups going from rags to riches with a 'cool' idea. Starting with industry best practices and business projects isn't as 'sexy', it does put everyone on notice that innovation is hard work. To paraphrase Thomas Edison, innovation is 1% inspiration and 99% perspiration.

### **Launch and Learn**

"Launch and learn" is a central tenet of innovation. It covers the last two phases of the innovation process: implementation (launch) and refinement (learn).





Launch and learn consists of setting up mechanisms for capturing feedback, feeding it back to the relevant parties, making decisions about the feedback (prioritising, managing conflicts and tradeoffs), feeding the decisions into the product process and then launching the amended offering.

Launch and learn is not a substitute for “analysis and planning” but rather it recognises that you cannot know everything upfront. That doesn’t absolve you of the responsibility to know everything that *is* possible to know. The more work you do upfront, the less time consuming and costly the ‘learn’ phase will be.

Launch and learn is also not the same thing as “trial and error”. Trial and error is about guessing and hoping for the best. Launch and learn is about controlled experimentation.

The entire innovation process is obviously important but an organisation’s ability to launch and learn is the key determinant of success. Very few Silicon Valley start-ups make money from their original idea (Google for example, is making money from ads not search). No matter how thorough you are, there’s no way to be sure how your product will be received by the market. The lesson is clear: if you want to be a successful innovator you must launch, learn, refine, adapt.

IT Departments are lucky because they can practice these two critical phases of the innovation lifecycle relatively risk free – by implementing best-practices and delivering business projects.

### **Implementing Industry Best Practices**

Best-practices are common knowledge, widely distributed and often freely available. They may not be new to the universe but they’re new to your IT Department – and hence qualify as innovation (although that’s disputed by some purists).

With best practices you don’t need to spend time on idea creation, selection or development because best-practices are nothing more than good ideas which have been ‘productised’. Think about all the templates, books, training courses and implementation guides available for such things as TOGAF, ITIL, Prince2 and the various flavours of agile development. All IT Departments need to do is implement and refine.



Unfortunately implementation is not easy. Just as having the best ideas and most innovative product is no guarantee of market success, so too, the good ideas incorporated into best-practices don't guarantee improvements. In both cases it's about 'launch and learn'.

In another article, *Why IT Must Overturn Its Obsession With Best Practice*, I explain three methods of implementing best-practices (adopt, adapt then adopt, adopt then adapt) and point out that 'adopt and adapt' is the best approach. Unfortunately this approach is rarely taken because most IT Departments do not have launch and learn capabilities. This shortcoming leads CIOs to conclude that whatever is launched is what the Department will be stuck with (i.e. the suboptimal 'adapt then adopt' approach).

There is a missing link between the explicit knowledge embedded in best-practices and actual results. The missing link is the tacit knowledge that isn't written down in any TOGAF book, Prince2 manual or Agile website. You can bring in experts who have this tacit knowledge in their heads but that doesn't fully solve the problem because there is still a disconnect: the expert has both the explicit and tacit knowledge about the best-practice but knows little about your organisation, whereas you know a lot about your organisation but comparatively little about the best-practice. Each side has half the picture – a situation reminiscent of the classic 1980s movie *Hear No Evil, See No Evil* starring Richard Pryor and Gene Wilder, where one character is blind and the other deaf. Launch and learn is an effective way of overcoming this disconnect.

What better way to develop the implementation and refinement steps in the innovation process than by practicing them on our own internal improvement initiatives?

### **Delivering Business Projects**

The whole point of delivering projects is to realise business benefits. Once a project is delivered the real work of adoption and benefits realisation begins.

IT Departments are notorious for completing projects and then leaving the benefits realisation to the business. From IT's perspective, a project is deemed complete when the users are trained and the solution is deployed. Left to their own devices, the organisation *will* achieve a certain level of business benefits from the project but if IT becomes an active participant, rather than passive observer, those benefits can be far greater. The reason for this is simple: the business unit sponsoring the project will be involved in



projects sporadically, whereas the IT Department works on projects all the time. The far more experienced IT Department is far better placed to develop the capabilities needed to drive adoption and deliver benefits.

The concept of 'benefits realisation' is bandied around in many IT Departments but very few are actively developing capabilities in this area. Benefits realisation is really a "launch and learn" process. If an IT project is not delivering the anticipated business benefits, IT Departments are often at a loss to figure out what to do. Imagine if a start-up company launched a product that wasn't getting the traction they expected but said "we're right, the market is wrong". This ludicrous situation is similar to the attitude most IT Departments take: our solution is excellent but the poor uptake is the 'markets' fault (the market being the organisation).

IT must become more actively involved in the process of diffusing solutions throughout the business and realising benefits. What better way to practice those skills than by maximising the business benefits of projects the organisation has already invested in?

## **Conclusion**

Whenever you learn a new skill, you break it down into its constituent parts. When you learn to play tennis for example, you learn grip, stance, swing, positioning, footwork, serve, forehand, backhand, lob, drop shot and volley. The best players don't learn by playing – they learn by repetitively honing each individual skill, and then combining all of them into a complete game.

Similarly, you can't just go out and 'innovate'. You need to develop each individual skill and then combine them into an effective innovation process. IT Departments can develop two such skills (implement and refine) with minimal risk by implementing IT industry best practices and delivering business projects.

IT Departments who start in these two areas can gain the experience required to develop good judgement in other areas of the innovation lifecycle. □

---

*For more articles please visit [www.rafcammarano.com](http://www.rafcammarano.com)*

