orbus software

White Paper
Enterprise Architecture: Bridging Entrepreneurs and Hard Problems

WP0168 | November 2014

Karl Schulmeisters

Karl Schulmeisters is Technology Advisor for the Carver Global Health Group where he provides expert leadership in Business Process Architecture, Enterprise Architecture and Cloud Computing. Karl's current emphasis is on the impact and integration of disruptive technologies into traditional enterprise IT organizations: Cloud, Mobility, Consumerized IT, Machine Learning/Big Data and Social Media.

Karl Schulmeisters is an internationally recognized speaker – his most recent speaking engagement was at the Congress on the Future of Engineering Software in St. Petersburg Russia. He welcomes your comments at *karl.schulmeisters@cg-hg.com*

This white paper is a bit of a change from my previous papers. In my previous white papers I've talked about the concrete aspects of Enterprise Architecture: best practices for deployment, ways to succeed against resistance, integrating it with Agile methodologies etc.

In this paper I will look at how Enterprise Architecture can help that most wily of creatures, the technology entrepreneur. One view of Entrepreneurs is that they will do anything to get their business off the ground. They will try lots of options to find a way to profitability.

But there is some interesting research out there that suggests that successful entrepreneurs come up with an idea and stick to their guns about the idea. Allen Lee of Cowboy Ventures writing in Tech Crunch http://techcrunch.com/2013/11/02/welcome-to-the-unicorn-club/ writes about ultra-successful startups – which he dubs "Unicorns" – startups with billion dollar exits – identifying that amongst other things, these companies

- Avoided a Big Pivot in their product vision. 90% of the Unicorns are working on their original product vision
- Took 7+ years to payout.
- Most were consumer oriented solutions

So what does this have to do with Enterprise Architecture?

Architecture is Structure - Structure Imposes Limits

Enterprise Architecture at its fundamentals is about imposing a structure – and architecture, that integrates both the business goals and the underlying technology roadmaps and capabilities. As with all structures it imposes limits. And at first glance it would seem that structured limits are the antithesis of what a startup needs. Yet each of the three above bullet points requires structure to be reached (more on why consumer oriented solutions require structure below).

There is also the old adage that boundaries are important to the creative process. Scott McDowell, founder of the consulting firm CHM Partners told Fast Company[1]

> *"Whether or not they're created by an outside client or you yourself, a set of limitations is often the catalyst that sets creativity free".*

So while the popular view of entrepreneurial work is that it is creative and breaking limits – the reality is that the creativity necessary for successful startups needs to have some structure within which to create. And there really are two forms of that structure

Business Structure

Without a business structure to deliver a new technology or solution to the customer, be it consumer or business, any new venture is going to fail. Industry after industry is littered with good ideas that went nowhere simply because the business structure behind them was inadequate. This is not to say that the business structure needs to necessarily be created within existing forms. A great example of this is Amazon.

In a recent interview with Charlie Rose, Steve Ballmer speaking about Amazon.com opined:

> *"In my world you're not a real business until you make some money."*[2]

And yet many would argue that it is precisely Ballmer's inability to see beyond the traditional business structure that lead to many of Microsoft's difficulties in bringing innovation to market.

Systems Structure

Without structure in your systems, it is not only difficult to understand what capabilities you can deliver, but also what it costs you to deliver your capabilities. Without structure to your data, it is very difficult to understand what you know and do not know or what value you have. And without understanding your interfaces, you don't know what systems you can integrate with and how difficult that integration can be.

[1] http://www.fastcompany.com/3024458/how-to-be-a-success-at-everything/why-your-creativity-needs-boundaries-to-thrive

[2] https://www.youtube.com/watch?v=PSqb5s3xTlc

Entrepreneurs Fail Fast

One of the hallmarks of modern entrepreneurism is the mantra of "Fail Fast". As I discussed in my whitepaper on Integrating Agile into Enterprise Architecture[3]. Agile and Enterprise Architecture while at first incompatible, can be made to work together. One of the most critical areas of this integration is the facilitation of Failing Fast.

Failing Fast Means Trying Ideas and Testing

Failing Fast means trying ideas quickly, testing them and then using the results to improve your solution or your ideas. Writing on Richard Branson's Virgin blog as a guest contributor Richard Kastelein[4] cites that

> *"Fail Early, Fail Often is a motto born from Silicon Valley"*

And the reason for this is that by failing "early" we identify where the weak spots or the bad ideas are and reduce the time chasing down those blind alleys.

The simplest way to do this is with "thought problems": IE "What Ifs". This is where we start to be able to leverage the techniques of Enterprise Architecture.

Using Enterprise Architecture Techniques to Fail Fast

Enterprise Architecture helps identify both the business side and the systems side of your ideas. By formalizing your business goals and business roadmaps and tying them to underlying systems architecture, even if that architecture is only on paper, you have a structure against which to run your earliest ideas and see if they succeed or fail.

One example from an early part of my career was a low cost networking project I worked on. At the time I had an intimate background in the network stacks and protocols that were in popular use. This was the era of Prodigy and AOL and before the HTTP protocol had been invented. Network hardware and network adapters were expensive and finicky to use. And I was working for a company that owned IP on using some of the control lines for serial ports to increase the data speeds that could get sent across a serial RS-232 cable.

The initial project mandate I received was:

> *"Can you build a network that will use our patented high speed serial connection to run a network using this RS-232 switch our partner has built?"*

[3] http://www.orbussoftware.com/resources/downloads/enterprise-architecture-and-agile-development,-contradiction-or-synergy/

[4] http://www.virgin.com/entrepreneur/fail-early-fail-often

From a purely technical perspective the answer was “yes”. But fortunately I was lucky enough to also ask the questions of:

- What is an acceptable lag for transferring files that you expect our customers to accept?
- How much slower than Novell Netware can we be and still be viable?
- Do we have a hard commitment to use this particular RS-232 hardware on this project or can we build something more appropriate?

These are all business questions that are part of an Enterprise Architecture Business Requirements and Business Value layer, though of course they were not called out as such at the time. The answers to those three questions gave me a framework with which to run some quick switching experiments with the serial switch that we had, because the answer to the third question was “we told our partner we would help them sell the switches they have already made”.

By running some very simple packet delivery and switching tests, - that took me 2 days to code, test and validate – I determined that although we could conceivably build out such a network – the performance would be so inferior to the existing networks (Novell and others in use at the time) and the time to transfer even small files would be so unbearably long, that we could never really meet the performance goals the business side had for the project.

This gave the business leadership in this small startup the information with which to go back to our partner and ask: “Can this switch be made faster”? And when the answer was “No” – we were able to walk away from the project.

The net time invested was 3 days of a senior developer/systems architect (me) and a few hours from the business leadership.

A prime example of ‘failing early” in the project.

It is also an example of a failure that led to walking away from a project, though at the time the understanding that Enterprise Architecture needed to integrate business goals and systems capabilities in a formal manner was not broadly understood and my asking those questions was more of just trying to understand my requirements.

Structure Enables Crisp Thinking About Capabilities

What this tells us is that the structure that Enterprise Architecture brings to a project enables crisper thinking about the capabilities that one has in the systems one is working with as well as often helping to clarify the business goals and objectives. I firmly believe that until I asked the questions specifically about what the acceptable performance goals were – there had not been a concrete analysis of what was and was not acceptable.

In most cases, Entrepreneurs are starting from a Business Goals premise. Identifying a customer need or a perceived customer need and moving to fill it. Facebook, by remaining focused on its customer goals and vision, avoided the pitfalls of over-monetization in the early days, and thus was able to survive where MySpace failed by losing sight of the business value being delivered to customers. Thus for most Entrepreneurs the guidelines I presented in my whitepaper "Enterprise Architecture: Outside In"[5] is the most appropriate approach to take, and I recommend downloading it for some best practices.

Enterprise Architecture Inside-Out

The Enterprise Architecture in many ways is particularly appropriate when the idea comes from the technology or systems side. In large IT environments Enterprise Architecture was initially targeted at bringing alignment between IT projects and business goals and eliminating IT project that are driven primarily by a chasing technological change. Similarly in many startups clever uses of technology become the justification for that technology.

The technology then becomes "a solution in search of a problem". By applying the Enterprise Architecture principles of identifying the corresponding Business Goals and Customer needs early on a much better case can be made for validating or rejecting those business goals. This is particularly important in the identification of the "minimum viable product"

Minimum Viable Product

The concept of a Minimum Viable Product comes out of work by Alexander Ostwarlder and his 2010 PhD Thesis on the Business Model Ontology.[6] It has further lead to the notion of a Business Model Canvas[7] that is documented in the online book: Value Proposition Design[8]

[5] http://www.orbussoftware.com/resources/downloads/enterprise-architecture-inside-out-or-outside-in/
[6] http://www.hec.unil.ch/aosterwa/PhD/Osterwalder_PhD_BM_Ontology.pdf
[7] http://www.businessmodelgeneration.com/canvas/bmc
[8] https://strategyzer.com/value-proposition-design

Essentially it breaks the process of defining a product – the Minimal Viable Product – into 9 categories

1. The Value Proposition – What is the unique value your solution brings to each customer segment
2. The Customer Segments – Who are your customers and what are their characteristics
3. The Channels – the channels to deliver your value proposition to the customers
4. The Customer Relationships that the channels establish between your organization and the customers
5. The Revenue Streams these channels generate
6. The Key Resources you require to deliver the Value Proposition(s)
7. The Key Activities you need to engage in to deliver the Value Propositions)
8. The Key Partners to help deliver this value
9. The Cost Structure of your solution.

From an Inside Out perspective. The technology itself will address part of your Key Resources, contribute to your Cost Structure analysis, and impact your channels. But as is clear the rest of these are business drivers.

So this takes us back to the importance of mapping the technologies and the business goals together. Precisely what Enterprise Architecture tools and techniques are designed to do.

Do Entrepreneurs Need Formal Enterprise Architecture Tools

Ok so does this mean that Entrepreneurs need full blown Enterprise Architecture tools? The answer as it so often is, is that “It Depends”. Clearly during the early stages of a technology idea and the working out of the early value propositions and other aspects of the idea things like notebooks, note taking and organizing apps, e-mail and basic Office applications are clearly going to be sufficient.

Intrapreneurs Also

But as the project grows, or if you in an Intrapreneurial Role – namely running an entrepreneurial project within a large organization, being able to migrate these documents into a more formal process becomes important. Frankly this is one of the strengths of a tool like Orbus Software’s iServer. As it is designed to take standard Microsoft Office documents and map them into standard frameworks like TOGAF thus providing the sort of formals structure Intrapreneurs need.

To give an idea of how quickly such tools become important, the ClearRoadmap project that I recently lead was a 12 man-month project. But with a near-shored development team and a target market aiming to support the mHealth and Medical Device Approvals process industry we quickly had a design specification in excess of 25mBytes in size, and a project backlog of over 1100 items.

This simply is too much complexity to manage just through emails and spreadsheets. While for Version 1.0 managing these feature sets in a tool like Microsoft's Visual Studio is adequate – for future versions we will be bringing in a formal tool like Orbus Software's iServer. Just as Visual Studio provides us with a code repository for our source code, iServer will provide us with a tool that manages mapping our Business requirements and roadmaps into our technology

Persistence and The Dark and Lonely Place

A recent article in Entrepreneur Blog[9] was titled:

> ***The Entrepreneur's Secret Weapon: Persistence.***

And Richard Branson writing also for the Entrepreneur blog identified "Determination" and "hard work" as two critical factors in success[10].

Persistence then is a critical factor in the success of any entrepreneur. The biggest stumbling block to persistence is fear. Fear of the unknown factors, the "gotchas".   And one of the things that the Enterprise Architecture approach to entrepreneurship provides is a level of confidence that you have a set of proven processes and techniques in place that connect your business goals and your technical capabilities.

So when that "Dark and Lonely Place" of setbacks invariably comes along, you have a structure and a process to fall back on to move through it.

Therefore I believe that Enterprise Architecture is the structure that enables Entrepreneurial Creativity rather than stifling it.

[9] http://www.entrepreneur.com/article/233630
[10] http://www.entrepreneur.com/article/225827

Orbus Software
3rd Floor
111 Buckingham Palace Road
London
SW1W 0SR
United Kingdom

+44 (0) 870 991 1851
enquiries@orbussoftware.com
www.orbussoftware.com

orbus software