HealthManagement
Promoting Management and Leadership

Volume 5 / Issue 3-4 / 2010 - Cover

Enterprise Architecture for Healthcare

Author

Jim Quiggle

Chief Architect and Founder,
Altra Concepts Corp.

The emergence of Enterprise Architecture may be a refreshing twist to the standard hype cycles because the Enterprise Architect will be the strongest advocate against hype, product or approach.

What is Enterprise Architecture?

Describing Enterprise Architecture (EA) is difficult outside the context of techno-architectural approaches and mindsets. A recent discussion thread in an Enterprise Architecture Group challenged members to describe EA in 160 words or less. The result, to date, is 1,275 responses ranging from "EA is about knowing where we are, what we have and where we want to be to ensure we build systems and processes which create value and do not paint us into a corner" to "EA is optimal enterprise business process, information and technology asset management tuned to meet customer and shareholder needs in relevant time frames".

Though information technology is following a similar lifecycle as other technologies – invention and then innovation until a consumer commodity is created – this reality is often overlooked, resulting in the seemingly chaotic application of IT within organisations.

Lifecycles and Commonsense

When we fail to understand the position of a technology within its lifecycle, the first victim is commonsense. One example of an IT invention/innovation being deployed as a consumer commodity is in Riverside California. Riverside Community Hospital's new 'ER Wait Time Text Messaging' program urges users to simply text 'ER' to 23000 on a cell phone to display average ER wait time.

Does this IT solution seem to be missing the critical design element, commonsense? If text messaging technology had run its course and become a consumer commodity, hospital decision makers would have interjected their own commonsense, resulting in a different solution, as opposed to being lead by IT enthusiasts and early adopters within the organisation.

At the other end of the scale are technical solutions that get missed because they are tried and true technology commodities. Eric Dishman points out this concept in his recent TED Talk: Take health care off the mainframe (www.ted.com - /talks/eric_dishman_take_health_care_off_the_mainframe.html). His example illustrates how a commodity technology – the telephone – is overlooked. He proposes that simple recording and analysis of the verbal response of a patient allows for passively benchmarking and gauging a patient's awareness over time.

Why has not this common sense technology solution been proposed as opposed to spending resources on a cutting edge technology, such as text messaging, in a questionable manner?

One possibility is that technology solutions, and in particular, healthcare IT solutions, are commonly presented by technologists: the very same people on the edge of invention and innovation, and thus disinterested in non-technical or commodity technology solutions.

Architectural Styles

An understanding of where an IT solution is in its lifecycle and the approach and mindsets of people advocating the technologies is critical in selecting and implementing ‘right’ or ‘best fit’ solutions in healthcare environments.

As illustrated in the diagram below, three types of architectural styles can be defined based on number of common criteria and resulting mindsets. Interestingly, they can also be aligned with the technology to commodity lifecycle. This is likely due to the ‘what you know’ syndrome or “when you know everything about a hammer, the world appears to be full of nails”.

Technical Architects represent the cutting (and often bleeding) edge of information technology. They are commonly subject matter experts in a particular field such as application development or networking and as such, are strong advocates for the deployment of new vendor products within their fields of expertise. Historically, technical architects have been responsible for the selection and implementation of information technology solutions within healthcare organisations. When information technologies are at the invention or innovation stage, Technical Architects are well suited to bring them into production environments. However, the trade-off for their deep understanding often results in multiple technology silos within the organisation.

Solution Architects were the answer for silo’ed organisations. They brought a breath of knowledge related to multiple technical solutions and two primary constructs:

- Service Orientation provided a view into the technology silos by slicing out common components via delivered service paths. This allowed for reusable components that could be shared within the organisation. Because formerly silo’ed components were now shared, Solutions Architects had to perform cross-organisation requirement studies to successfully implement serviceorientated solutions. Many saw this as the long sought after answer for ‘business to IT alignment’ solution. However, Solutions Architects soon discovered that it was nearly impossible to collect organisational requirements without a common understanding of how the organisation operated, in other words, the Framework.
- Frameworks such as ITIL and CobiT became a great complement to Service Orientation. ITIL allows for alignment from the viewpoint of the IT technologist and CobiT from the viewpoint of organisation management.

To date, few organisations have fully developed Service Oriented Architectures with viable Frameworks, largely due to a lack of acceptance. As illustrated in the diagram above, the reasons for organisational buy-in quickly become complex. Finding the sweet spot is not only tough, but even those who have found one have seen true alignment not occurring as expected.

Enterprise Architects may be the long sought after solution to alignment because they are inertly negative, having lived through just one too many hype cycles and the promises of aligned, right fit technologies and roadmaps that just did not come true. “Just implement ITIL and your Service levels will stabilise and costs will go down…” or “after we deploy an SOA, application development will be a snap…”

Enterprise Architecture views people, process and technologies as a ‘systems of systems’. Technical expertise, innovation and interoperable solutions are all systems within incrementally larger and larger systems.

When viewed in such a manner, the application of technology results in some interesting observations. For example, unlike Solutions or Technical Architecture, Enterprise Architecture accepts and embraces chaos as the end-state. This is a bold departure from the mindset of a technical architect where order and predictability is the goal.

Chaos as the end state, from an IT standpoint, is akin to heresy. Interestingly enough, when this concept is presented to business managers the reply is often, ‘you did not know that’. There truly is a chasm between the way technologists and the rest of the world think.

When Frameworks Collide

Today, there seem to be frameworks to fix any kind of challenge: IT Solution Frameworks, Information and Data Management Frameworks, Business and Operational Process Frameworks. The established Frameworks are commonly related to the level of maturity and sponsorship within an organisation but have rarely, if ever, delivered. Why did they not work? First, they were very hard to understand. A common misperception is that unless formally deployed, frameworks do not exist. However, you do not ‘do ITIL’ or ‘install CobiT’.

Frameworks only provide a mechanism and common language for identifying existing components (Functions, Processes and Procedures). For example, every organisation does some kind of event management. It may be a highly complex monitoring system or someone yelling “it is getting hot in here”. Both these examples can be left alone or optimised as needed. One might be over-engineered, another susceptible to human error. Frameworks do a great job of categorising both and presenting them back in a comprehensible manner, to wide audiences.

Where frameworks break down is in terms of what we do with the information. A common criticism regarding frameworks is, “thanks for telling me what I already knew, now what?” with the “now what” part being about the ability to apply logical selections within a complex system of systems. Using the event management example above, technically-focused decision makers would address the human error risk by selecting an automated system while business-focused decision makers may select hiring a second human monitor. While both are viable upgrades, the weighting of trade-offs between the options are nebulous, trapped within a higher level of interrelationships contained at the enterprise level.

Historically, the answer has not been to elevate the viewpoint to enterprise level but to develop specialised frameworks (such as CobiT and ITIL) to enable specific audiences to make IT technology selections.

The result has been yet another layer of silo'ing, this time via process frameworks, and with an unintended consequence: technical architects have never had it so good. Finding a technical requirement that justifies almost any new deployment can now be had from multiple locations. An ITIL Framework will support the development and implementation of complex monitoring and control systems but may not support a rapid development approach. "No problem, requirements for rapid development can be obtained from the CobiT steering committee."

This is not to say that frameworks are not valuable. They are and like Service Orientation, have a place in every organisation. However, they cannot stand alone and do not hold the answer for alignment.

Models, Models, Models – Accepting Chaos

In times past, local television stations in the US often ran selfpromotional campaigns proclaiming their on-staff meteorologist's forecasting accuracy. Over the course of the last decade this claim has faded away, along with apologies for missed forecasts. The shift has been from "sorry we missed that late afternoon rain in the forecast" to "the late afternoon rain yesterday was unexpected".

This shift has been primarily due to a much greater understanding of weather models and the acceptance of chaos. Similar to the Technical and Solution Architectural mindsets, there was a time when meteorologists thought they could figure it all out. However, the more the data they collected, the more did the sheer complexity of weather systems become evident, along with the realisation that the goal of accurate short-term weather forecasting could not be achieved. Attempting to predict afternoon rain, much like predicting the usage of wireless technology, is not achievable when all of the variables are understood.

The enterprise architectural mindset has followed the same course as the meteorologist. It has taken some time. Five years ago, we had no idea that service orientation was not the answer and two years ago we had no idea that frameworks were not going to fill the gaps.

It took failure to identify the true complexity of the enterprise. It also took some time to develop enterprise architects.

At the Enterprise Level, it becomes clear that a state of chaos exists in the form of complex interactions. Several models have been developed to aid in understanding the interactions such as The Open Groups Architectural Framework (TOGAFv9) and the Object Management Group's Business Motivation Model.

These models all address the relationships at the enterprise level and with an enterprise architect available to direct and populate the models, the first step was achieved. However, we were still one step away from understanding the true nature of the enterprise.

Current enterprise models include several foundational assumptions. Many are now beginning to be questioned.

- The first assumption is that the organisation as a whole has a common goal (e.g. a mission statement). This may have been true a number of years ago but it is certainly not how organisations are structured today. In a single healthcare organisation there are multiple influencers, all with their own directives and goals. Often, these are in alignment – more often, they are not. For example a radiology department may be on a rapid upgrade path, expecting IT to keep up in both on-premise storage and data management expertise while the oncology group as a whole may be considering the outsourcing of IT functions.
- The second assumption is that people are willing to change and 'want to do the right thing' or even understood what the right thing is. It took some time for technologists to understand this concept as they have surrounded themselves in change and thrive upon it. However, most people perform within a system of positive feedback loops. Whether these loops are positive or negative for an organisation is largely unknown. What is known is the loops are either positive or negative for the individuals. Changing what has worked for someone over a long course of time is almost impossible unless there is a perceivable upside for them.

Unfinished Business

Identifying and understanding these challenges has led us to the 'Acceptance of Chaos'. This concept has been (and continues to be) very difficult to accept on the part of technical architects, systems engineers and developers. In brief, the concept of leaving something unfinished or broken is a near-insurmountable roadblock for the technically focused mind.

However, the acceptance of chaos within the enterprise does not mean that the enterprise architects have given up. Conversely, they have identified that we must accept that chaos is not going away and that we must learn to design within a chaotic state by modelling and simulating the environment in order to understand possible outcomes.

Swarm Models

It has not been hard to find a working model that addresses design within a chaotic environment, above all the swarm model.

Swarm modelling addresses both challenges listed above in an elegant manner via the following rules. In a swarm there is (are):

Simple Rules: *No Leaders, No Followers, No One Cares - alignment is achieved by design*.

Several recent models have been proposed. One notable model is Nick Malik's Enterprise Business Motivation Model. http://motivationmodel.com

Positive Feedback Looping: *Negative is ignored and dies while positive is reinforced and thrives.*

The most technically elegant design on the planet is of no value if positive feedback loops align against it. Most people like their jobs and how they do them. To ask them to stop or to do them in a different way for the greater good of an organisation just does not happen. A core concept surrounding Centres of Excellence is based on positive feedback looping. However, COEs had a flaw; they were missing the concept of phased transitional emergence.

Phased Transitional Emergence: *Changes just happen by simple 'gap' rules.*

This is the" iPhone or Blackberry Effect". In short, give someone something they like and others will ask for it.

Although this sounds simple, it is contrary to most IT design methodologies in use today. End-users are seldom given the right of selection outside of use cases. After the requirements study stage, technology-focused decision makers select the technology solutions that users must use.

Modelling an enterprise requires the insight of an Enterprise Architect. The role is hard to define and it is even harder to identify those capable of producing an enterprise architecture (no matter their title) because they do not fit into common role buckets. They are (and have been) technical and solutions architects, application developers and systems and process engineers or managers and directors. They may have started in desktop support but have also presented findings, designs and roadmaps to executives and boards. Their experiences have shaped their mindset allowing them to "see" the organisation from the enterprise level. This experience has also created a pragmatic, sometimes almost-fatalistic view due to the hype cycles and unfulfilled promises they have lived through. As such they are uniquely qualified to identify how and when technology is applied, the trade-offs of different approaches and the likelihood of success.

Process and Substance

An interesting characteristic of enterprise architecture and architects is that no one is happy with an enterprise architecture project or initiative until the results begin to emerge. The technical architects, application developers and systems engineers commonly feel that "we're not being asked the right questions" because the technical details relating to how things plug in are not being addressed. Likewise, the solution architects, business decision makers and analysts feel that 'we're off track' because use case and organisational requirements are not being addressed.

Several years ago one of the best descriptions of an enterprise architecture project was "Wax-on, Wax-off". Those familiar with the movie Karate Kid may recall a karate master having a kid waxing his car all summer. The kid finally lost his tolerance for the work and insisted the master teach him karate. However, the master had been teaching him karate; the waxing techniques were training him in both technique and endurance. Similar to enterprise architecture, it is difficult to see the higher level until it is achieved.

Additional reasons for an initial push back to enterprise architecture are the enterprise architects, and much like the karate master, their approaches.

- Enterprise Architects are seasoned professionals. A recent survey in an enterprise architecture forum concluded that an enterprise architect under the age of thirty did not know the definition of enterprise architecture and therefore could not be an enterprise architect. It takes years of experience in multiple roles before the enterprise mindset begins to develop. As such, enterprise architects often know participants' jobs and roles, in addition to a greater understanding of the interrelationships within the enterprise. Their experience and understanding often intimidate individuals within the organisation. It takes a high level of professional maturity to accept the guidance that an enterprise architect can provide.
- Enterprise Architects do not give the answers expected and often ask questions that seem to make no sense.
- On the technical end of the scale, questions relating to software and hardware product selection, refresh or integration appear to be missing. However, from an enterprise viewpoint, these items are simply details that will be derived from higher level selections such as what components are outsourced or provided on premise and the organisation's guiding principles and design statements.

- On the business end of the scale, questions relating to operational process enablement and workflow optimisation appear to be missing. These items are commonly collected in the form of use cases and requirements studies. However, from an enterprise viewpoint, these items will be derived from the organisation technical capacity and capability roadmaps.

Looking back at the definition of Enterprise Architecture:

- EA is about knowing where we are, what we have and where we want to be to ensure we build systems and processes which create value and do not paint us into a corner.
- EA is optimal enterprise business process, information and technology asset management tuned to meet customer and shareholder needs in relevant time frames.

It is clear that both definitions above are correct. However, it can also be said that Enterprise Architecture is the pragmatic mindset of the Enterprise Architect.

�

Published on : Wed, 15 Dec 2010