SETLabs Briefings
ENTERPRISE ARCHITECTURE & BUSINESS COMPETITIVENESS VOL 2 NO 4
Oct – Dec 2004

Over-Engineering Enterprise Architecture and Business Competitiveness

By Shireesh Jayashetty, Pradeep Kumar Manjunatha & Harish Kashyap

Over-engineering of Enterprise Architecture, often ignored by technology as well as the business community, is a risk and a waste of resources.

Evolving models such as outsourcing and collaboration, and an increasing number of mergers and acquisitions have made planning the corporate IT landscape more challenging and competitive. While technology is helping businesses become more competitive, businesses are now more demanding than ever before. Enterprises that lack an agile IT infrastructure may well bid farewell to global markets.

One of the ways to stay competitive would be to get technology to enable business and business to drive technology innovation, which can be achieved by robust enterprise architecture (EA). While attaining agility and robustness the tendency among managers is to ignore this interdependency between technology and business and to over-engineer the enterprise architecture. Some of the factors that may contribute to over-engineering are market pressures, architectural style and preference, over-enthusiasm and so on. This may result in a negative impact on business competitiveness.

OVER-ENGINEERING DEFINED

Enterprise architecture can be considered over-engineered if its deliverables far exceed the business requirements, thereby making it complex, more expensive, and difficult to maintain. Whether a given architecture is over-engineered or not is mostly contextual and depends on the problem it is intended to solve. For example, the pyramids of Egypt may seem over-engineered to some people. They may be correct if they perceive the pyramids as mere tombs.

IMPACT OF OVER-ENGINEERING ON BUSINESS COMPETITIVENESS

As mentioned earlier, technology innovation

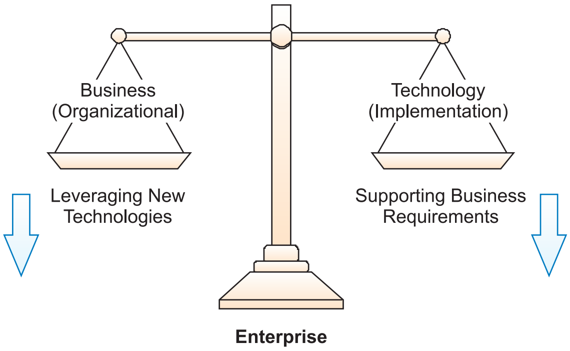

Figure 1: *Business-IT alignment leads to a balanced ecosystem in an enterprise.*

Source: *Infosys Experience*

evolves because of business demands and business evolves with technology innovation. This forms a balanced ecosystem between the business needs of the enterprise and technological innovation (Figure 1).

Enterprise architecture influences the ecosystem that in turn influences the enterprise architecture. The cyclic nature of influence impacts the evolution of future enterprise capability. Any discrepancy, such as over-engineering, is likely to disturb the ecosystem and result in a direct or indirect negative implication.

Typically, over-engineered enterprise architectures are a result of technology trying to deliver more than what the business really needs. Consider the following example.

A coalition loyalty management business has two key processes and three players in their business. The processes are:

- Points accrual process (customer who collects points for purchase) and
- Points redemption process (customer who uses points to purchase)

The players are partners, customers, and the loyalty business itself. A partner may participate in one or both of the processes. From a technology perspective, one could assume that providing both accrual and redemption in real time would add value to customers and the loyalty business. In the case of point redemption, it would add value as it provides customers with an easy way to use their loyalty points enabling better cycling of rewards. However, in the case of accruals, this benefit may not be worth the investment in infrastructure and solution. The customer can afford to see his point balance updated at the end of the day. The core concept of loyalty is to reward customers for coming back and as mentioned earlier, a partner may allow only accruals and no redemptions at all.

As is evident from the above example, being pragmatic is a good approach to define solutions that suit business needs, without trying to overdo it just because technology can do it. There are several other implications of over-engineering on an enterprise and its business competitiveness.

Business-IT Alignment: In quest of perfection, the architecture team may not synchronize with business objectives and needs. Much time and effort may be wasted by focusing on aspects that

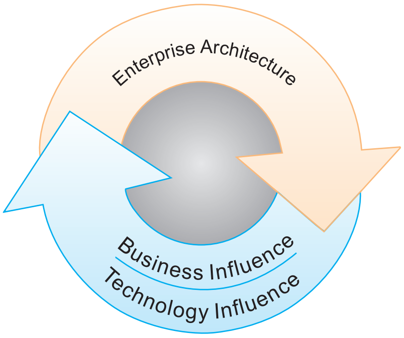

Figure 2: *Enterprise Architecture influences the Enterprise eco system and vice versa*
Source: *Infosys Experience*

are meant to be simple. For example, many enterprises, especially investment banking companies, invested considerable effort and money in building their own database transaction engine or an application server equivalent infrastructure, while their focus was to address the business trading issues and problems.

Agility: A complex architecture does not offer businesses and systems the agility to evolve and adapt to new environments. It may require extra effort (and investments) to make the systems flexible and adaptable. Some complex legacy systems may have to be supported and carried forward as unnecessary baggage because the enterprise lacks the knowledge to modify them. For instance, while system decomposition is one of the industry best practices for good architecture, mindless decomposition will result in a spaghetti of systems making them inflexible and difficult to modify.

Redundancy: Due to over enthusiasm and/or lack of knowledge, many enterprises, especially startup companies, demand redundant systems and solutions. For example, Building a 24x7 real time fail-over system for administration functionalities that are internally used in business hours is redundant. Similarly real time synchronization of on-line transaction processing (OLTP) and marketing databases is redundant in a scenario where the impact of marketing decisions is not immediate or real time.

Complexity: Generalization leads to complexity. While attempting to make the system generic so that it addresses various needs or achieve extreme flexibility in terms of quality attributes or business needs, it may in fact end up being very complex. For example, presentation components of an intranet, Web-based application rendering XML which later gets converted to HTML using XSLT leads to complexity. This approach befits an application that is accessed by various presentation channels like the Web, mobile phones, etc., but it is complex and over-engineered for an internal application.

Operations cost: An IT system that is more complex than necessary will need more money to build and also to keep it running. The obvious fall out of a complex system is increased time-to-market and budget overshoots.

In addition to the above, other factors such as global sourcing, regulatory requirements, shrinking budgets, and increased competition need to be tackled for sheer survival reasons.

WHAT LEADS TO OVER-ENGINEERING?

A brief look at the various factors that lead to over-engineering will help avoid the same, as well as identify the affected areas. The factors – covering all the constituents of an enterprise ecosystem – can be broadly classified into four categories: People, Investment, Management, and Market.

The People Factor

People determine the nature and culture of an organization. The choices made by various people who are involved in the decision-making process influence the business and technical roadmap. Since enterprise architecture is part of the business roadmap, the people factor influences EA as well. Personal preferences, breadth of knowledge, tendency to over-generalize, over-enthusiasm, unrealistic end-user demands, and scarcity of required skills, all play a role in how enterprise architecture evolves.

The Investment Factor

It is blindingly obvious that businesses exist to make money. All decisions made in a business set-up, including investments in information technology, are driven by the expected returns. Hence factors such as existing investments in hardware/software or surplus IT budgets may result in over investment or compromised architecture that could lead to over-engineering. For example enterprises with a successful and robust EAI backbone demand that any intra- or inter-application communications happen using the existing backbone. It is appropriate to use it for inter-application communication but for an intra-application communication (client to server where a synchronous call is appropriate), it is an overkill.

Any enterprise driving an Enterprise Architecture initiative must have a fully functional Architecture Review Board to periodically examine the business-IT alignment

The Management Factor

Enterprise architecture is as much about management as it is about technology. According to Jeff Schulman, technology comprises only one-third of the architecture effort[1]. Other factors such as organizational politics, lack of leadership, non-conformance to decisions, stakeholder non-participation, absence of good processes, bad communication channels, and unnecessary focus on managing hypothetical risks result in lost and/ or misplaced focus while also influencing the quality of enterprise architecture.

The Market Factor

Most technology vendors market their wares as a solution that can address all the needs of an organization. This simply cannot be true because each business has a unique selling point that enables it to compete and survive. Hence, a generic product cannot fulfill all the needs of an organization.

Many enterprises expect technology per se to give them the competitive advantage irrespective of their business needs. This invariably leads to over-engineering.

HOW TO ASSESS AND AVOID OVER-ENGINEERING

In order to avoid over-engineering, the soft spots that could have been over engineered or have the potential to be over- engineered in an enterprise architecture need to identified.

Though there are no definitive guidelines, a few thumb-rules can be formulated to identify potential over engineering by keeping the influencing factors in mind.

Assessing People Factor

The architect's background or his/her prior experience and environment greatly influences the design of the enterprise architecture. This assessment will bring out issues such as zeal, biases, personal choices, skills, and other factors as defined earlier. The problem can be avoided by constituting a team of architects with multiple skill sets instead of an individual, thereby institutionalizing a scientific decision making process.

Assessing the Investment Factor

Many enterprises may not have a financial plan for spending and utilization. Being either too conservative or too liberal during key decision-making would result in under utilization of budgets. This can perhaps be avoided if there is effective planning, tracking, and utilization of resources.

Assessing the Management Factor

Any enterprise that cares to implement enterprise architecture should have a fully functional Architecture Review Board (ARB). The absence of an ARB or the presence of an ineffective one will lead to an imbalance in the enterprise eco system hence over-engineering. Assessment of the ARB can bring out factors like effective participation, the presence of right processes, alignment with strategic business goals, etc.

An ARB should be responsible for defining processes and guidelines, defining roles and responsibilities, involving subject matter experts and business users in decision-making and validation processes.

Assessing the Market factor

All innovations made by an enterprise may not be relevant to its business environments. Therefore, assessing the focus of the architecture group is important. Innovation is necessary and the enterprise can focus on driving that innovation and not necessarily focus on applying the innovation in multiple contexts. This can perhaps be avoided by sourcing the non-focus areas to a partner or vendor who has the expertise.

CONCLUSION

Over engineering an enterprise architecture is an area that does not receive the amount of attention it really deserves. Enterprise architects need to realize that over engineering is a risk and may result in wasted resources. Organizations can avoid over engineering if due attention is given to the following areas early in the enterprise architecture lifecycle:

- Assess the architect's background for relevant experience
- Create and periodically assess the Architecture Review Board
- Assess the current utilization and allocation of budget
- Assess the Architectural focus.

Organizations, without doubt, benefit from keeping an eye on this subtle aspect and taking a pragmatic view while defining the enterprise architecture to achieve and sustain business competitiveness.

REFERENCES

1. Management issues dominate enterprise architecture, Letter From the Editor, Jeff Schulman, Gartner, January 2003
2. IT Doesn't Matter, Nicholas Carr, Harvard Business Review, 2003
3. Strategy of Acceleration: Time to Change

Culture and Architecture, Jorge Lopez, Gartner G2 research, July 2002

4. Enterprise Architecture: Far Too Important to Be Left to the IT Team, Alexander Drobik, Gartner G2 research July 2002
5. Architecture as business competency, Ruth Malan and Dana Bredemeyer, 2004, Bredemeyer Consulting
6. Setting some realistic enterprise architecture goals, Paul Ramsay, Equinox Limited, November 2003

Authors featured in this issue

HARISH KASHYAP

Harish Kashyap is a Senior Architect with Infosys Europe Practice. He has several years of experience in implementing large-scale internet projects. He can be contacted at harishk@infosys.com

PRADEEP KUMAR MANJUNATHA

Pradeep Kumar Manjunatha is a Senior Architect with Infosys System Integration Services Practice. He has several years of experience in implementing large-scale projects that involve client / server and distributed technologies. He can be contacted at kmpradeep@infosys.com

SHIREESH JAYASHETTY

Shireesh Jayashetty is a Senior Architect with the Infosys System Integration Services Practice. He has several years of experience in implementing large-scale projects that involve Mainframe and open technologies. He can be contacted at shireeshj@infosys.com

For information on obtaining additional copies, reprinting or translating articles, and all other correspondence, please contact:

Telephone : 91-80-51173878

Email: SetlabsBriefings@infosys.com

Infosys acknowledges the proprietary rights of the trademarks and product names of the other companies mentioned in this issue of SETLabs Briefings. The information provided in this document is intended for the sole use of the recipient and for educational purposes only. Infosys makes no express or implied warranties relating to the information contained in this document or to any derived results obtained by the recipient from the use of the information in the document. Infosys further does not guarantee the sequence, timeliness, accuracy or completeness of the information and will not be liable in any way to the recipient for any delays, inaccuracies, errors in, or omissions of, any of the information or in the transmission thereof, or for any damages arising there from. Opinions and forecasts constitute our judgment at the time of release and are subject to change without notice. This document does not contain information provided to us in confidence by our clients.

Infosys®

POWERED BY INTELLECT
DRIVEN BY VALUES