World Applied Sciences Journal 6 (Supplement 1): 62-67, 2009
ISSN 1818-4952
© IDOSI Publications, 2009

The Impact of Service-oriented Architecture on the Agility and Coherence of Enterprise

[1]S. Tarverdizadeh and [2]S.A. HoseiniBay

[1]Department of Information Technology,
MS Student, School of Eng, Alzahra University,Tehran, Iran
[2]Department of Management, MS Student, Tehran University, Tehran, Iran

Abstract: In this paper the impacts of Service-Oriented Architecture (SOA) on enabling both agility and Coherence Studied in Enterprises there is an increasing importance of IT and a corresponding increase in the tension between agility and coherence. Both are essential conditions for an efficient and effective IT use, but both conditions must be held in balance. Nowadays there are many new development methods for increasing agility and also architectural methods for increasing coherence. What is missing is a solution that combines both aspects together. In some literatures the service- oriented architecture has introduced as a solution which increases both aspects. However they didn't give any reason about this increasing. In this paper at first we identify the qualities of an agile and coherent enterprise. Then research the impacts of any key-components of SOA on each of these qualities.

Key words: Enterprise Architecture · Service-Oriented Architecture (SOA) · Agility · Coherence · Enterprise

INTRODUCTION

In an ever so rapidly changing world, businesses survive and thrive through the ability to adapt to change quickly and profitably. Agility, the "smooth and dexterous performance" in response to the unexpected, describes this ability. At the same time Coherence is necessary to ensure the correct interaction of the various business processes and to allow the organization to present itself as a uniform entity. However there is an increasing importance of IT and a corresponding increase in the tension between agility and coherence. Both are essential conditions for an efficient and effective IT use, but both conditions must be held in balance. The challenge facing the modern organization is finding the correct balance between coherence and agility [1].

Service oriented architecture is a new technology which has just come about in the past few years and is introduced as a way to increase both agility and coherence of enterprise. It is a new way to design software promoting the ideas of reusability, long life span and easy updating [2].

This paper describes how Service oriented architecture can help businesses achieve agility and become more coherent. In this way with mapping the qualities and aspects of an agile and coherent enterprise with key components and concepts of service oriented architecture we recognize the impact of service oriented architecture on agility and coherence of enterprise.

The Qualities of the Agile and Coherent Enterprise: In this context enterprises are defined as followed: "an organization (or cross organizational entity) supporting a defined business scope and mission [3-4]. This definition supports the description of enterprises as system that is stated in the IEEE standard, namely "A system exists to fulfill one or more missions in its environment" [5].

Agility is "the ability of an organization to adapt proficiently (thrive) in a continuously unpredictable business environment" [6-4]. An agile enterprise minimizes costs and time-scales associated with change in order to take maximum advantage of it [7].

The qualities of an agile and coherent enterprise are shown in the Table 1 [8].

Corresponding Author: S. Tarverdizadeh, Department of Information technology , MS Student, School of Eng,
Alzahra University,Tehran , Iran. E-mail: tarverdi@niordc.ir.

Table 1: Agile and Coherent enterprise qualities

Qualities	Description
Sensing	The ability to perceive environmental conditions; gather useful information from the system and readily detect changes; also the ability to anticipate changes.
Learning	The ability to effectively modify organizational behaviours and beliefs through experience; the ability to use information to improve the organization.
Adaptability	The ability to effect changes in systems and structures in response to (or in anticipation of) changes in environmental conditions.
Quickness	The ability to accomplish objectives in a short period of time; pace at which changes are accomplished; the rate of movement within organizational systems.
Innovation	The ability to generate many solutions to a problem
Flexibility	The ability to do a number of different things with the same systems and structures; having processes, systems and equipment adaptable to a number of different uses.
Concurrency	The ability to effectively perform related activities at the same time, with the same assets.
Efficiency	The ability to use minimal resources to accomplish desired results.
Simplification	The elimination of undesirable variation through adoption of a set of core standards
modularization	The packaging of system functions into components with strong
integration	The matching and assembly of components into an effective system
availability	The readiness to accept and carry out functions under any environmental stresses and threats
scalability	The ability to adapt a solution from the targeted design or implementation to a significantly larger (or smaller) scale
Security	The protection of information and computer resources from unauthorized use or change and the attribution of information and agreements to identified parties
reliability	The ability of a system or component to perform its required functions under stated conditions for a specified period of time

In General, Companies Using Agile Processes Have:

- Lower costs
- Better productivity
- Better quality
- Better business satisfaction [9]

Enterprise Agility Has Three Core Enabling Elements:

- Accurate timely awareness that a change should be made, enabled by focused knowledge management processes,
- Effective value-propositioning skills to prioritize among competing changes and competing response-alternatives to those changes and
- A facilitated ability to change business processes and to customize operational responses in real time, which we call response ability [10].

Methods of Increasing Agility and Coherence in Enterprise: There is an ever-increasing emphasis on thenecessity for agility and coherence in the development of an IT solution. The IT world has created several responses to this necessity. These responses are aimed at accelerating the IT development process or at improving the coordination between individual IT developments.

Acceleration of the development process is being sought in employing new development methods or in implementing standard packages, while improving coordination between developments is being sought in development under architectural guidance.

To increase the speed at which applications are constructed, several new IT development methods have been created such as DSDM (Dynamic Systems Development Method) and XP (extreme Programming).

These new methods set aside the many and often complex principles used by the more traditional approach and replace them with fewer and less-complicated principles [1].

Both DSDM and XP take into account that user requirements may change during the development process. This is supported by “just in-time” planning and by ensuring that parts of the system are not created before they are needed

DSDM and XP are just two examples of new development methods that focus on increasing the speed of the development process. Other methods exist with this focus and, without a doubt, more will follow.

Such new development methods are aimed at quickly producing IT solutions, targeted at a specific business goal. They do not concern themselves with the question as to how the solution will relate to and cope with other events within the organization. They do not give any guarantee in respect of coherence [1].

Many organizations attempt to achieve enhanced coherence in their IT developments by improving the architectural awareness of the organization.

Even those who see the direct benefits of using architecture are confronted with the fact that compliance with architecture costs a great deal of valuable time and, therefore, they often decide that, just this once, architecture will be set aside. Their excuse is that the market demands an immediate response and there is insufficient time to wait for architecture [1].

In brief, compliance with architecture is recognized as an answer to the necessity for coherence; but, at the same time, it is seen as a hindrance in the IT development process.

Methods of Increasing Agility and Coherence in Enterprise: The answer to the desire for more agility is being sought in new development methods and the answer to the need for more coherence is being sought in architectural awareness. However, new development methods, which provide an increase in agility, do not bring any guarantee of coherence and architectural awareness, which should provide for coherence, is perceived as being a hindrance to progress and, moreover, proves difficult to implement.

What is missing is a solution that combines both aspects and the answer is to be found in a combination of both a new development method and architecture. This, in turn, demands a new approach to architecture. Working under architectural guidance must no longer be seen as synonymous with wasting time. Rather, it should become synonymous with gaining time [1].

Architecture has an essential role in creating the ability to react and providing an organization with the capacity to respond to changes in the market, even in situations when such changes cannot be predicted. In doing so, however, architecture itself must undergo a number of changes. What we need is an agile architecture, an architecture that has been specifically designed to facilitate the speed of change.

Industry analysts tout SOA as the best way to protect IT investments and simplify application integration, resulting in reduced costs, shorter development cycles and overall productivity gains for the IT organization [1].

SOA is an implementation of business process as a set of independent but cooperating web services. The main advantage of SOA is that the interface of the services is decoupled from its implementation. This allows the consumers of the services to relay on the well-defined interface of the service that are based on open standards regardless of the software and hardware used to implement the service [12].

Unlike object-oriented architecture, SOA services are designed to interoperate with different development technologies, which make them flexible and reusable [13].

It is important to note that SOA is not an off-the-shelf technology; rather it is a way of architecting and organizing IT infrastructure and business functionality. SOA is a paradigm for designing, developing, deploying and managing discrete units of logic i.e. services, within a computing environment and has the following attributes:

- Functionality is organized as a set of modular, reusable shared services
- Services have well-defined interfaces and encapsulate key business processes
- Customer facing solutions serve as customized views of these services for different segments and can access these shared services as needed.
- The reusable shared services are built without making any assumptions of who (portal or another service) will consume these services [14].

Methods of Increasing Agility and Coherence in Enterprise: The SOA reference model is concerned with seven main concepts: service, visibility, interaction, real-world effect, service description, policies and contracts and execution context. These are grouped into three categories according to the functionality that they bring to SOA. The first one is the category of service which has the stand alone concept of service. The second category is the dynamics of services which contains the concepts that are critical to the operation and use of the service. This contains concepts of visibility, interaction and real-world effect. The last grouping is that of the concepts that support those in the dynamics of services. This contains the ideas of service description, policies and contracts and execution context [2].

Services are provided by service providers and used by service consumers. The consumers discover a provider that can fulfill its needs through a service description and interact with it through the service's interfaces to result in a real-world effect [2].

A service is just a means to bring needs and capabilities together. A service allows its capabilities to be accessed through its service interfaces, but can only be used within the constraints that its policies and contracts, along with the consumer's, have specified.

Visibility is the basic idea that the service provider and service consumer need to be able to “see” each other in order to interact. Within the concept of visibility there are three subcomponents that need to be addressed as they act as preconditions for visibility. These are the notions of awareness, willingness and reachability.

Awareness is the idea that one party needs to know about the existence of the other parties for an interaction to occur. This requires that a service description and set of policies be available for the consumer to know about the existence of and capabilities of the service. Willingness is needed as a consumer and provider may be aware of each other, but may not be willing to interact.

The last aspect of visibility is the somewhat obvious one of reachability, in that the consumer and service provider need an actual means to communicate. Without such means an interaction is impossible as the two have no visibility of each other. Once visibility has been satisfied and the preconditions met, the next step is to actually interact with the service. This is the act of a consumer performing actions against the service. The final step is the real-world effect, or the reason that the consumer is using this service.

The service description lays out what the services capabilities are and is the means to which a consumer can decide if the service is suitable to fulfill its needs. The service description should contain information about the service’s existence and reachability, the set of functions that it provides, the policies and constraints that it runs under and that it will comply, to some extent, with the policies and constraints that consumer imposes, as well as what messages may be exchanged (information model) and how to interact with the service (behavior model).

The second concept that supports the dynamics of a service is that of the policies and contracts that constrain the service. Policies represent some constraint on the use, deployment, or description of some owned entity of a participant. The remaining concept is that of an execution context as briefly described. The execution context, more specifically, is the infrastructure elements, policies and constraints and process of a specific instantiation of a service. This effectively acts as the path between the consumer and provider.

These underlying concepts of SOA are critical in understanding what is required at a minimal level for such a solution [2].

Methods of Increasing Agility and Coherence in Enterprise: Figure 1 shows the agile and coherent enterprise cube which shows 3 demensions of agility nd coherence as we mentioned about them.

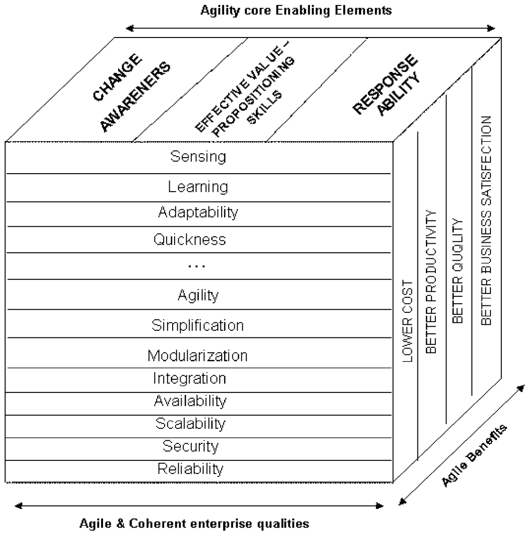

Fig. 1: Agile and Coherent Enterprise cube

The mapping of three faces of the above cube with seven concepts of service oriented architecture we have explained is shown in Table 2. In each cell if we have found any relation between two sides in the literatures or in our view, we have ticked it. It is tried to explain the reason of each relation.

Because services are self describing and runtime concepts, they may be used by consumers without development effort by the service provider. Sometimes the using parties may be anonymous or have no prior relationship with the provider. These characteristics create the potential for services to be used on a much broader basis, perhaps by unconventional consumers and markets to change business models in terms of cost, scale and scope [15].

- Services are a runtime concept- they are available to authorized users in real time. This enables real time business intelligence, together with real time response to events. [15]
- With real time, heterogeneous access services can handle optimization of information ownership and improved data quality replacing replicating critical data with real time access [15].
- Properly designed services create looser coupling between business models and technologies, which reduce dependency on specific products and technologies and break down barriers between application silos.

Table 2: Mapping the qualities and aspects of agile enterprise with SOA concepts

		Dynamics of services`			Supporting issues		
Agile and Coherent Enterprise qualities	Services	Visibility	Interaction	Real world effect	Service description	Policies and contracts	Execution context
sensing			✓				
learning							
adaptability	✓					✓	
quickness	✓						
innovation			✓				
flexibility	✓					✓	
concurrency			✓				
efficiency	✓					✓	
simplification	✓						
modularization	✓						
integration	✓						
availability	✓					✓	
scalability	✓						
security	✓					✓	
reliability	✓					✓	
Agility core enabling elements							
Change awareness			✓				
Effective value propositioning skills							
Response ability	✓		✓				
Agility Benefits							
Lower costs	✓		✓	✓			
Better productivity	✓		✓				
Better quality	✓		✓				
Better business satisfaction	✓		✓			✓	

- A service oriented architecture that properly reflects real word business services creates convergence of the business and IT perspective which enables much greater efficiency, adaptability and cost control in business relationships and structures.
- The service interfaces allow new components to be introduced and existing ones to be easily replaced or upgraded, so the SOA will almost certainly be more adaptable and responsive to change in business requirements.
- For the service consumer, the process must be organized such that only the service matters and there is no dependence upon knowledge of the implementation. So the service designers have to provide formal descriptions and contracts within which consumers can then use the service in whatever way they see fit. This causes considerable benefits of flexibility accrue.
- A service-oriented architecture Promote the reuse of application components that are key to increased productivity and rapid application deployment
- SOA allows multiple layers to be developed in parallel and to be plugged together on completion because connection points are abstracted by standard APIs (concurrency) [16].
- SOA facilitates the continuous improvement of processes without changing application code, which leads to competitive advantage and increased ROI [16].
- Services have well-defined interfaces and encapsulate key business processes. (security)
- The reusable shared services are built without making any assumptions of who (portal or another service) will consume these services. It causes more efficiency [17].
- SOA increases application quality through reuse of proven and tested components [17].
- SOA causes reduction of IT costs by leveraging existing investments and enabling applications to work together better heterogeneously [17].
- SOA promotes a modular enterprise, promising a high degree of reusability of business services, ensuring consistency [14].

- SOA promotes the ability to rapidly reconfigure the business process by selecting from the available set of services, thereby providing the agility to adapt to the business requirements introduced by stakeholders and business partners [14].
- Services reusability feature reduces the time required for building new applications and in turn reduces the total cost of ownership (TCO) for such applications [18].

CONCLUSION

In this paper a new approach to architecture, Service Oriented Architecture, was presented that is explicitly aimed at achieving business goals quickly in a constantly changing environment In this regards, we had researched the impact of service oriented architecture on agility and coherence of enterprise with mapping the qualities and aspects of an agile and coherent enterprise with key components and concepts of service oriented architecture. As a result we had recognized that SOA has effect on improving most of the qualities of an agile and coherent enterprise and also the core enabling elements of agility. SOA highlights the key benefits of agile enterprise. However we didn't find any tight relation between SOA and "Effective value propositioning skills" and the quality of "learning" in an agile enterprise.

REFERENCES

1. Bieberstein, N., S. Bose, L. Walker and A. Lynch, 2005. Impact of service-oriented architecture on enterprise systems, organizational structures and individuals. Ibm Systems J., 44: 4.
2. Strom, M., K. Estes, K. Kesavarapu and B. Shipe, 2006. Service Oriented Architecture: Final Paper: Paper Available Online.
3. FAWG F.A.W.G., 2001. A Practical Guide to Federal Enterprise Architecture.
4. Veltman, E. and V. Reekum 2006. Determining the Quality of Enterprise Architecture Products. Master's Thesis Business Informatics.
5. IEEE, 2000. IEEE Recommended Practice for Architectural Description of Software-Intensive Systems. New York. The Institute of Electrical and Electronics Engineers Inc.,
6. Dove, R., S. Hartman and S. Benson, 1996. An Agile Enterprise Reference Model with a Case Study of Remmele Engineering. Agility Forum: Report available Online.,
7. Boden, T., 2004. The Grid Enterprise Structuring the Agile Business of the Future. BT Technology J., 22(1): 107-117.
8. Stamos, E. and E. Galanou, 2006. How to evaluate the agility of your organization-Practical guidelines for SMEs.
9. Shine Technologies, 2002. Agile methodologies Survey Results. www. Shine Tech. com, a passion for Excellence.
10. Dove, R., 2004. Enterprise Agility-What Is It and What Fuels It?. First published as Utility Agility - What Is It and What Fuels It? - Part 2. Issue Alert® 10/29/04. UtiliPoint International.
11. EGlobal CIO Inc. 2006. Agile Enterprise IT–Oxymoron to Reality. Accessible on www. eGlobalCIO. com
12. Ramaswamy S., S.R. Member and M. Malarvannan, 2006. Service Oriented Architectures for Grid Computing Environments: Opportunities and Challenges. Retrieved from www. cybelink. com
13. Donge, T., 2006. Building an Agile Enterprise Based on Cordys' Process-Centric, Event-Driven, Service Oriented Architecture. Whitepaper Available Online.
14. Mansukhan, M., 2005. Service Oriented Architecture White Paper. Accessible on www. hp. com/go/SOA.
15. Sprott, D., 2004. SOA: An introduction for managers. CBDI Forum, www.cbdiforum.com
16. FAMIS Software Inc. 2005. Introduction to service oriented architecture. Accessible on www. famis. com
17. Borland Software Corporation, 2004. Building an agile enterprise with Service-Oriented Architecture. Accessible on www.borland.com.
18. Sumanta, D.E.B., Infosys technology Ltd, 2003. Building IT architectures for Enterprise Agility. http://www.infosys.com