Interpreting the Solution Architect's Role at a leading IT and Consultancy Company

Pedro Miguel Faria Magalhães, Information Systems Department, University of Minho, Guimarães, Portugal, pmfmagalhaes@gmail.com

Rui Dinis Sousa, Information Systems Department, Algoritmi R&D Centre, University of Minho, Guimarães, Portugal, rds@dsi.uminho.pt

Abstract

At the end of the twentieth century, an individual or a small company could design and develop an application or even some bigger and scalable systems. However, due to their increasing complexity, project size and required higher levels of integration, the need for new roles and processes associated with software development drastically increased, calling for skills to assure the integrity and effectiveness of solutions.

A new role, called "Solution Architect", emerged over time and has been formally used in leading companies such as Cisco, IBM, Microsoft, Oracle and SAP, but it is still lacking agreement on its scope and responsibilities.

Having the retail sector as the context in which a leading IT and consultancy company operates, this research builds upon the experience of seven Solution Architects from a pool of fourteen inquiring them on their perceptions about their roles in order to better understand the required skills and boundaries of this new role against more established architectural roles.

At the end of this research, it was clear that the interpretations of this role's span across a project's lifecycle as well as the boundaries to other roles such as the Enterprise Architect's one were not the same. Nevertheless, the consultancy side of the Solution Architect role gathered the most consensuses as opposed to the more traditional technical side, which wasn't selected by any Solution Architect. Anyway, the scope dilemma still persists within a project leading to a role overlap between Enterprise and Solution Architect.

Keywords: Solution Architect, Domain Architect, Enterprise Architect

Introduction

The solution architecture profession may have something to learn from more established counterparts in other areas such as the building architecture profession.

At first, building was just an integrated craft with the builder taking care of all the details from designing structures to managing the building process. The industrial revolution allowed for the specialization of the work and details became matters for specialists. Designing structures while managing the relationship between the client and the builder became the role of a building architect (Lewis, 1998).

Something similar is happening in the IT domain where the growing complexity of solutions is calling for more attention to the importance of the role of an architect at several levels, be it at the enterprise or software level (Bredemeyer and Malan, 2006).

Software development projects still present lower rates of success as reported from surveys performed over the last decade by the Standish Group showing just a slight improvement from 29% in 2004 to only 32% in 2009 of software projects delivering acceptable results on time and budget (Kaur and Sengupta, 2011). Two mains causes have been pointed out at the root of high failures rates for software development projects (Wiersem, 2009):

- Globalization of software development - ideally, people work close to each other and any problems are quickly solved through personal contact. The problems begin when distributed development enters the equation (Herbsleb, 2007).
- Exponential increase in software complexity due to service orientation - service orientation is known for supporting richer interdependencies in the business. For every 25% increase of complexity in the business, there is an increase of 100% in the software complexity for the systems that need to support that business (Glass, 2002).

Delivering a given solution within time and budget while aligned with the understanding and expectations of the business is quite a challenge that attends to several tasks such as identifying business needs (IIBA, 2006), applying engineering principles on the IT solution through a systematic and disciplined approach to software development (IEEE, 2004) and managing the solution delivery (PMI, 2008).

While project managers stay responsible for managing solution delivery, architects should play an increasing role in the projects ensuring the quality, usability and time to market of the solution, just like the building architect when working together with the builder.

To better understand the roles of architects such as the Solution Architect and the Enterprise Architect, it is of paramount importance to understand their scope: the first one focuses on something called “Release” while the second one focuses on the bigger picture, the Project, with each project having one or more Releases. These Releases can happen simultaneously or sequentially, depending on the constraints and dependencies between each Release.

While integrating the capabilities of several specialists, the Solution Architect is responsible for the quality and feasibility of the solution to be delivered on time and meet stakeholders’ expectations. Regular planning is carried out with customers. Cross-functional and regular planning with customers is required in order to map the provider's resources and capabilities against the customer's situation and set joint goals (Brady et al., 2005; Storbacka et al., 2011). It is important to focus on untapped needs, or to proactively sense barely explicit customer specifications (Matthyssens and Vandenbempt, 1998; Storbacka, 2011).

In order to do it successfully, and form a well-balanced team with the project manager, the Solution Architect should have a leading role on the Business Analysis and System Engineering domains, making sure that requirements, design and development stay aligned and feasible. Additionally he should have a supporting role in the project management domain, making sure that planning, resources, commitments and risk recognition stay in line with the solution under development.

For a typical IT project, the architect’s span of control will have to travel across a great number of knowledge areas in order to be able to take responsibility for the solution’s architecture, quality and feasibility (Wiersem, 2009). Service providers, like the company involved in this research, which engage in high-value integrated solutions, must also radically extend the time span of their focus towards life-cycle care (Helander and Möller, 2008). As solutions, by definition, are longitudinal, providers need to

secure that they are integrated into the customer's process so that the providers are able to support value creation during usage of the solution (Storbacka, 2011).

A simplistic vision of the Solution Architect role is that he or she "just" creates architecture blueprints, and his/her responsibilities encompass all the activities of doing so. This includes articulating an architectural vision, conceptualizing and trying alternative approaches and validating the resultant architecture against business requirements. However, every experienced architect knows that the role is not only a technical one, but also a more political and strategic, on one hand, and a consultant role, on the other.

For many developers, the most requested role is the Solution Architect. This type of architect is the one that manages the development effort and is responsible for the baseline vision and its execution in order to create the solution itself.

The core task of a Solution Architect is to convert requirements into architecture and design, which later become the blueprints of the solution to be created (Bogue, 2005). While doing that, the Solution Architect has to be a technophile, a business strategist, an organizational politician, a consultant and a leader (Bredemeyer and Malan, 2009).

Dealing with a variety of role definitions for architects, a leading Information Technology Consulting company with customers all over the world had just decided to create an Architecture Pool of knowledge and resources as a competency center available for every project. This ended up setting a perfect timing for this research to contribute for the clarification of roles, namely, the Solution Architecture role.

The following sections describe how this challenge was tackled, the results obtained and some conclusions that can be drawn from this research.

Methodological Approach

Having the opportunity to analyze the Solution Architecture's definition and instantiation in a leading Information Technology and Consultancy company where architects are assigned to several projects around the world, we positively answered to the challenge of looking into the panoply of architectural roles with a special focus on the Solution Architect role.

The company was functionally divided in "pools" or competency centers, and people where distributed over those centers. These structures work to answer project's needs in terms of resources, research and proposals.

Until the Architecture Pool's creation, all professionals were Subject Matter Experts (SME), but the company started to work with larger customers, with growing needs and mainly, the requests for standards were perceived as essential to compete at high levels. Thus, the definition of the Solution Architect role within the company, as well as the other one now called "Domain Architect" was mandatory, not only to distinguish the professional's field of expertise, but to create a role that once applied to a selected professional, he or she could be well aware of what it takes to deal with methodology's phases, component hierarchy, what affects those components, legacy systems migration, etc.

The study was conducted within the Architecture Pool and started with a meeting with the Architecture Pool Manager, who explained the view of how the architecture roles were used across different projects, giving an enterprise-practical sight. A survey was then deemed useful to capture the Solution Architects' points of view and experience providing not only a practical view of how the company's architects operate, but also to have a glimpse of the misalignment there was, if any. This survey was jointly

designed with the Architecture Pool Manager to be administered to the fourteen Solution Architects that were part of the Architecture Pool.

The aforementioned survey was released to the Solution Architects and through role instantiation, a set of scenarios were described and the participants were urged to choose, with no limitations, those which they thought were correct. Most of the scenarios had overlapped ideas and the boundaries were not clearly defined so that the architects felt the need to provide a bit of their experience in the corresponding open field available for additional information.

Results

The first meeting with the Architecture Pool Manager, posed a few challenges. The first one was to understand that even having professionals from the same company, just the fact that they have worked in such different environments, gave them a very distinctive view and opinion about their role's boundaries and interpretation.

Beginning with the experience of an architect that worked in a project with a top UK retailer, the roles implemented at this level were the Enterprise Architect, Solution Architect and Domain Architect for each area of expertise. The Solution Architect role was specifically requested by this customer, due to its wider knowledge regarding project phases, components, what would interact and affect those components. A huge factor to request this role was to concentrate that broader knowledge and apply it to the switch from legacy systems. The customer didn't want a *big-bang* like model because there was just too much data sensitivity. This specific architecture role had to know every step within the software solution provided, and know the mapping of which modules to be installed in order to fulfill the established requirements.

At this UK retailer, the Solution Architect was involved right from the beginning, by designing the solution's roadmap. This roadmap acknowledged which set of requirements would be addressed with the functionality available from that specific module and a pooling document was generated to gather the scope and the roadmap for the next release.

This pooling document was very detail-centric regarding the definition of each involved module, whereas at this UK retailer a document existed with a higher level of abstraction, an alpha document. This alpha document was created for the Enterprise Architect to interact with the stakeholders, because it was a summarized version of the components. However, if more details were requested, the alpha document would have to be jointly analyzed with the pooling document to see the integration level between each involved component and the blueprint document, as the name suggests a lower level architecture document which describes the level of customization that each involved module would have.

As already mentioned in the methodological approach section, a survey was administered to collect the multiplicity of perceptions associated with distinct work experiences in Solution Architect roles.

The survey was sent to fourteen architects from the Architecture Pool, having answered half of them. The questions in the survey were:

1. A multiple choice question asking "How would you define the Solution Architecture and the Enterprise Architect roles?"
2. An open-ended question asking "What would you change, if you could, during past projects course?"
3. An open-ended question asking "In a project scope, how do you assure or think should be assured that everyone is aligned, strategically and technically wise?"

The questions made available on the survey were put together after the meeting with the Architecture Pool Manager and the commonly accepted instances of the role were also integrated, as well as some controversial definitions, in order to urge the Solution Architects to assume a critical stance.

The first question of the survey had the following choices:

a) A Solution Architect is often, but not always, focused on technical architecture and the meeting of non-functional requirements, often in the context of deploying specific applications.

b) The Solution Architect is deeply involved during the first phases such as consultancy, Planning & Scoping and Solution Definition but his/her involvement decreases until, ideally, it reaches zero at the system's development, having a quality assurance role only.

c) A Solution Architect acts as a domain architect in a project scope. Depending on the size of the project, we will have, for example, a Business Solution Architect, a Technology Solution Architect and a Data Solution Architect.

d) A Solution Architect acts as the accountable entity for the several architectures developed under the several domain architects guidance.

e) A Solution Architect steps in when an application becomes so vast and complex that dealing with the overall technical vision and planning, and translating business needs into technical requirements becomes a full-time job.

f) An Enterprise Architect is a planning role that is responsible for identifying the future state of an organization's IT environment and engages wherever and whoever necessary to help guide project teams to deliver towards it.

g) The Solution Architect is a member of the Enterprise Architecture team but becomes at a later stage also a member of the Development team. His role is mixed; he is the bridge between concepts and implementation. However, the Solution Architect does not operate at the Strategic Architecture level (at Enterprise level).

The results were as follows in Figure 1:

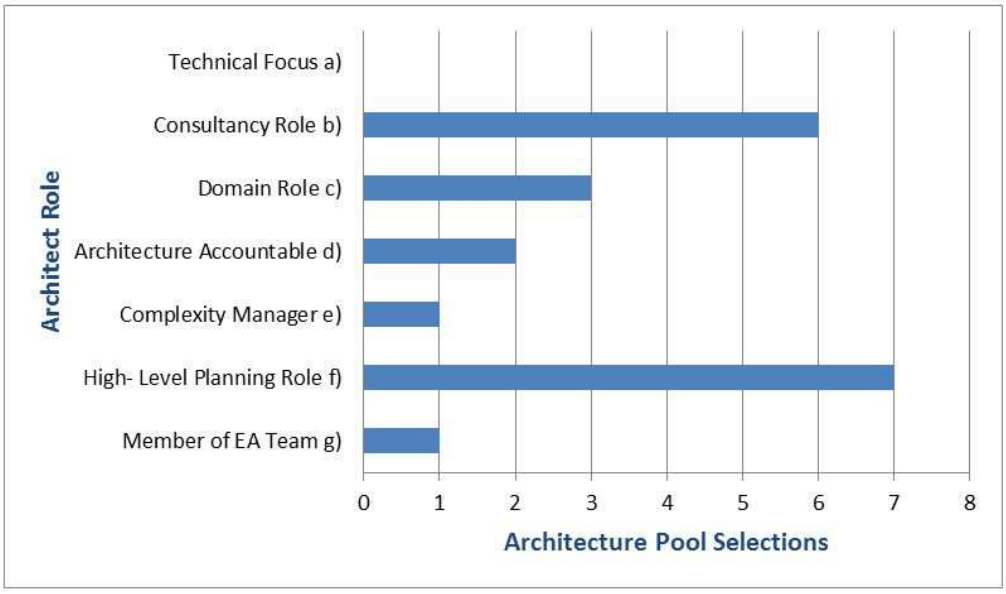

Figure 1 - Solution and Enterprise Architect roles

This question, with a multiple choice nature, addressed several scenarios for the Solution Architect role and the possible overlap between this role, the Domain Architect and the Enterprise Architect roles.

Regarding the second question, more than one path was observed. A first respondent stood for the case that during Conference Room Pilot phase, which is the first one to present the company's solution portfolio and define, with the customer, the project's scope and requirements, a Solution Architect must be present, so that he/she can start designing the solution from the project's beginning. This architect shouldn't, however, continue to carry on his shoulder the responsibility of discussing module-specific requirements, which is the Domain Architect's responsibility. The Solution Architect must be focused on the overall logic and integration between modules, thus building a solid solution. Other topic approached by a second respondent was the proximity there was between the company and customer's teams, in which is stated that customer's resources should be involved earlier and tighter in the process, ideally from the beginning. A third respondent suggested that in previous projects, the solution architect role should have been included on the overall view design of the projects, strategically and technically.

The third and last question focused on how every team should be strategically and technically aligned. A first respondent focused on the communication between the stream leader and the solution architect, stating that in order to assure that all teams are aligned, each stream leader needs to be always in touch with the architect. A second respondent said that the solution architect role must have the overall view of the project, strategically and technically, and should be responsible to align with project's main stakeholders.

Conclusions

Defining a Solution Architect's role is a challenge if we take into account the existence of such different perspectives on the role, whether resulting from concrete experience in projects or simply from theoretical views in the literature. In fact, this role is quite recent as a formal and distinct one in the organizations. Even well-known frameworks for enterprise architecture as TOGAF do not totally address this role making just some references to it in the latter phases of the Architecture Development Method (ADM), Opportunities & Solutions, Migration Planning and Implementation Governance. This traditional approach is being challenged in calls for a more holistic approach to the role to involve the Solution Architect right from the beginning in all phases of enterprise architecture development (Thorn, 2009). A major gap between enterprise and solution architectures frequently is the result of high-level architectures that do not provide concrete guidance for solution architects and developers. Closing the gap is fundamental but role boundaries are still not clear and may overlap (Zimmermann et al., 2004) as we can infer from the interpretation of the Solution Architect's role at the IT and Consultancy Company where this research was undertaken.

This specific problem gains even more momentum when realizing that this specific company deals with very distinct and unique customers. For example, for the Northern Europe customers, certification and standards compliance are mandatory. Even knowing that in Southern Europe it might not always be the case, an organization must be prepared to any reality and be capable of grounding its decisions based on the appropriate set of standards and practices, adequately framed by role definitions. This requirement's relevance grows proportionally to customer's size, new business units and collaborative work with other service providers and higher demands.

At very demanding levels, the worst word to throw at a customer is "Big-Bang". There is simply too much sensitivity in that to use such a method. In this context, the Solution Architect must be quite knowledgeable in all the methodology steps to provide the solution, which modules must be installed and which modules answer the business requirements under an end-to-end perspective.

Project involvement is required from every Solution Architect with a strong influence at early stages of project design and not so strong in systems development but still assuring the quality of the final solution.

Within the first stages of a project, it is clear that in Conference Room Pilot sessions, the Solution Architect shouldn't discuss module-specific requirements but rather focus on the overall coherence and integration of the involved modules. That is not what happens in most of the cases right now at the company. It is the Domain Architect that must be present because the discussions go around a certain business area and since the architect already described how to implement the several modules, the Domain Architect should pick up those blueprints and conduct those Conference Room Pilots sessions to deal directly with the requests from the customer. Until now, the Solution Architect ended up also playing the role of a Domain Architect, when they should be separate roles.

The boundaries of the Enterprise Architect and Solution Architect's roles keep somehow blurred, but the Enterprise Architect is the role who should see the end-to-end of a project while not needing to know the details. Instead, this architect knows exactly who is responsible for a certain domain and redirects the attention to the accountable person. It is a senior role, with knowledge and experience in several business areas and without the need of an integrator or other technology related role to know exactly how a company runs. On the other hand, a Solution Architect is responsible for the end-to-end within a release and the integrity assurance cross-release, and to make it happen, he or she coordinates other Domain Architects.

Acknowledgment

This work is supported by FEDER funding through the Competitiveness Factors Operational Programme – COMPETE and national funding through FCT – Foundation for Science and Technology under the project FCOMP-01-0124-FEDER-022674.

References

Brady, T., Davies, A. and Gann, D. (2005). 'Creating value by delivering integrated solutions.' *International Journal of Project Management*, 23, 360–365.

Bredemeyer, D. and Malan R. (2006). "The Role of the Architect". Architecture Resources for Enterprise Advantage. Bredemeyer Consulting. [Online], [Retrieved May 9, 2011], http://www.bredemeyer.com/pdf_files/role.pdf

Bogue, R. (2005). "Anatomy of a Software Development Role: Solution Architect.", [Online], developer.com, [Retrieved June 10, 2011], http://www.developer.com/mgmt/article.php/3504496.

Glass, R. L. (2002). Sorting out Software Complexity. *Communications of the ACM*. 45(11): pp.19-21.

Helander, A., and Möller, K. (2008). 'How to become a solution provider: System supplier's strategic tools'. *Journal of Business-to-Business Marketing*, 15(3), 247–287.

Herbsleb, J. D. (2007). 'Global Software Engineering: The Future of Socio-technical Coordination'. Proceedings of Future of Software Engineering FOSE'07, ISBN 0-7695-2829-5, 23-25 May 2007, Minneapolis, MN, USA, 188-198.

IEEE (2005). "Guide to the Software Engineering Body of Knowledge (SWEBOK(R)): 2004 Version", IEEE Computer Society Press.

IIBA (2009). "The Guide to the Business Analysis Body of Knowledge", International Institute of Business Analysis.

Kaur, R. and Sengupta, J. (2011), 'Software Process Models and Analysis on Failure of Software Development Projects,' *International Journal of Scientific & Engineering Research*, 2 (2)

Lewis, R. (1998). 'Architect? A Candid Guide to the Profession'. *The MIT Press*.

Matthyssens, P., and Vandenbempt, K. (1998). 'Creating competitive advantage in industrial services'. *Journal of Business & Industrial Marketing*, 13 (4/5), 339–355.

PMI (2008), 'A Guide to the Project Management Body of Knowledge', Project Management Institute.

Slot, R. G. (2010). 'A Method for Valuing Architecture-Based Business Transformation and Measuring the Value of Solutions Architecture'. Faculty of Economics and Business. University of Amsterdam. Doctoral Dissertation.

Storbacka, K. (2011). "A solution business model: Capabilities and management practices for integrated solutions". *Industrial Marketing Management*, 40(5), 699-711.

Storbacka, K., Polsa, P. and Sääksjärvi, M. (2011). "Management practices in solution sales- A multi-level and cross-functional framework". *Journal of Personal Selling & Sales Management*, 31(1), 35–54.

Thorn, S. (2009), "Enterprise Architecture, TOGAF® and Solution Architects", [Online], Architecting the Enterprise, [Retrieved September 4, 2013], http://www.architecting-the-enterprise.com/enterprise_architecture/articles/enterprise_architecture,_togaf_and_solution_architects.php

Wiersem, W. (2009), "Why do we need Solution Architects?" [Online], Capping IT Off, Capgemini's Technology Blog, [retrieved September 5, 2013], http://www.capgemini.com/blog/capping-it-off/2009/03/why-do-we-need-solution-architects

Zimmermann, O., Krogdahl, P. and Gee, C. (2004). Elements of Service-Oriented Analysis and Design. An interdisciplinary modeling approach for SOA projects, [Online], IBM developerWorks, [Retrieved March 21, 2011], http://kacit.ru/upload/iblock/853/8537b4f35143ad7f263b1092faa34a31.pdf