

A Governance-Based Approach for the Development of Process-Aware Service-Oriented Applications

Carlos Fernando García Hernández¹, Armando Maldonado Talamantes²

¹ Instituto Tecnológico Autónomo de México (ITAM), Computer Engineering Department,
Mexico D.F., Mexico
cgarciahe@comunidad.itam.mx

² Instituto Tecnológico Autónomo de México (ITAM), Digital Systems Department,
Mexico D.F., Mexico
armando@itam.mx

Abstract. As Service-Oriented Architecture (SOA) starts moving beyond a hype phase, new technical and cultural issues arise. Traditional development methodologies, aimed at defining Service-Oriented Analysis and Design processes, fail at leveraging an overall governance initiative as a primary foundation for the construction of an SOA. In this paper, we present a methodology that considers the creation of a Center of Excellence (CoE) for the reinforcement of governance rules and policies as the core structure in the SOA lifecycle. This methodology is the result of an effort performed in collaboration with some institutions in the financial services industry. It is based on a macro-level development model that leverages the concepts of service orchestration and Service Component Architecture (SCA) in a Business Process Management (BPM) scenario.

Keywords: Service-Oriented Architecture, SOA Governance, Center of Excellence, Service Orchestration, Business Process Management

1 Introduction

Enterprises are striving to take advantage of the windows of opportunity that Service-Oriented Architecture (SOA) implies both for business and for IT. Increased agility and flexibility, reduced time to market and development costs are just some of the many benefits promised by service orientation in its hype phase.

Despite the increasing support by vendors, industry analysts and researchers, we have found, however, that when applied into a real business execution context, the promises of SOA might not be realized in such a straightforward way. Far beyond the technical issues, the reuse mechanisms promoted by SOA require a cultural shift when it comes to developing services that can be shared among several lines of business. There exist legacy development practices that are difficult to change to become service-oriented. The SOA paradigm requires the enforcement of new rules,

policies and control mechanisms. It is also needed the right leadership to deal with the intolerance to change and the difficulty to share that are inherent to human nature. We need much more than the right technological resources to deal with an SOA initiative.

Professional services providers have their own methodologies to tackle this need, but they are not openly available and frequently depend on proprietary products and methods. On the other hand, academic efforts tend to focus on the low-level technical details for the implementation of an SOA. They seem little concerned about the importance of the human side as a decisive factor in the success of the SOA initiative. Some methodologies for service design and development have been proposed ([1], [2]), but they lack an overall vision of governance in an enterprise SOA, or the governance planning is left as a final phase in the SOA lifecycle. We have realized that governance must instead be an enabler of the SOA initiative from its inception and throughout its deployment. Understanding this is critical to the success of SOA.

In the following sections, we explore the need to put in place a governance initiative, not as a phase within an SOA development methodology, but as a pillar at the very heart of such methodology. The SOA can thus be constructed and evolved over time around a set of control mechanisms that enable sharing and change.

The document is organized as follows. In section 2, we give a brief overview of process-driven SOA to put in context the problem of automating business processes. Next, in section 3 we reference the previous work that we have done regarding SOA as a foundation for Business Process Management (BPM). Section 4 describes SOA Governance and the need for a Center of Excellence (CoE). In section 5, we propose a methodology for the development of SOA applications that is based on what we call a “Macro-level Development Model”.

2 Process-Driven SOA

Business dynamics requires agility and flexibility to respond to challenges and opportunities. Key to achieving such attributes is to become process-driven by centering the organization on cross-functional, end-to-end business processes that span several functional units of the enterprise. Additionally, the right techniques and tools to support those processes must be leveraged. The SOA paradigm [3] provides useful features (loose coupling, standards-based interoperability, portability, scalability, etc.) that have made it become a popular approach to dealing with the demanding technological requirements of process orientation. Nevertheless, the real business value of SOA can only be achieved by acquiring an overall discipline of Business Process Management (BPM) [4]. BPM enforces the correct definition, implementation, monitoring and continuous improvement of business processes.

SOA is based on the concept of services, where each service groups together a set of common functionality. The services make up building blocks used to compose/recompose high-level business processes in a flexible, agile and extensible fashion. These services are registered in a registry/repository, enabling their clients to discover them through a well-defined mechanism. The communication with services takes place over an Enterprise Service Bus (ESB) [5], which acts as the main infrastructure element to enable the implementation of an SOA. A service exposes its behavior

through one or more interfaces, where each interface uses one or more open and standard communication protocol, e.g., messaging, publish/subscribe (JMS), web access (HTTP, SOAP), APIs. Application components provide the implementation of one or more service interfaces. Components within a service –and services themselves- can be replaced by other components/services with similar functionalities without affecting the business logic of their consumers. This allows the process-aware system the flexibility to plug-in different implementations for services as long as the service interfaces that are exposed to the clients are not modified.

The nature of services as independent collaborative entities requires a set of governance rules through their development cycle. The lack of governance will increase development redundancy and complexity, and will decrease reuse capabilities and business value.

3 Related Work

This paper is based on a previous work [6] that we published to clarify the relationship between BPM and SOA and to share our findings after our first experiences with their integration.

3.1 Integrating Workflows and System Flows with a Multilayered Approach

In [6] we summarized the role of process awareness in the construction of information systems through the evolution of software development practices. The importance of process-driven initiatives was highlighted, and a multilayered model for the development of BPM-SOA integrated solutions was proposed. We emphasized the differentiation between workflows and system flows depending on the role played by users within the activities performed in a business process.

Workflows are those where there is a direct interaction with a human user, and they can be specified using the Business Process Modeling Notation (BPMN) [7]. On the other hand, system flows automate activities performed by autonomous systems. System flows are implemented by means of an orchestration of services, which can be defined in BPMN but needs to be later translated into a Business Process Execution Language (BPEL) [8]. The multilayered approach is described in terms of human users, a business process layer, a service layer and an enterprise applications layer.

3.2 A Reference Framework for the Integration of BPM-SOA

In order to achieve the desired BPM-SOA integration, in [6] we also proposed a Reference Framework (RF) consisting of five functional entities: i) Business Model; ii) BPM Model; iii) BPM-SOA Infrastructure; iv) Portfolio of Services; v) Available Services (coupling entity). We have realized that this framework is a good high-level guidance in practice. Given its abstract nature, it works as a blueprint for understanding the functionality required by a BPM-SOA integration project. We deliberately constructed it at this abstraction level to provide flexibility in its

implementation. However, we have also been experiencing the need for a more detailed methodology that along with the RF can lead the integration process according to industry standards.

In this scenario, the proposed RF works in the context of the OASIS SOA Reference Model [9] as an architectural blueprint that indicates the “What” of a BPM-SOA solution. Our RF is part of the architecture work towards a particular SOA implementation (see Figure 1).

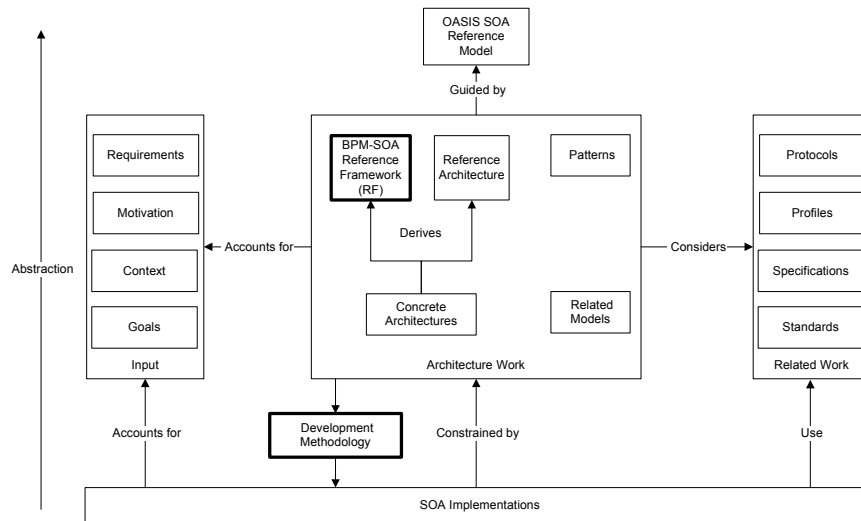


Fig. 1. BPM-SOA Reference Framework (RF) and a Development Methodology in the context of the OASIS SOA Reference Model

As shown in Figure 1, a development methodology (the “How” of the SOA initiative) should be located between the “Architecture Work” and “SOA Implementations” blocks.

4 SOA Governance and the Center of Excellence

SOA governance is a subset of IT governance aimed at defining the constraints, rules and control mechanisms that govern how services interact with their consumers and providers in an organization’s SOA. It comprises the enforceable policies and decision rights for building, deploying and managing such services.

SOA governance should consist mainly of: i) policies and guidelines regulating the roles and responsibilities for service ownership, definition and enhancements; ii) procedures and standards for appropriate service design and use.

Governance plays a critical role in the adoption and ongoing operation of any enterprise SOA. According to [10], SOA Governance can be considered in terms of three aspects of a service’s lifecycle: i) *Design-time governance* stands for policies

and procedures to ensure that the right services and its supporting artifacts are being developed, deployed and used; ii) *Run-time governance* refers to policies that affect the binding of consumers and providers, including service performance, monitoring and control, Service Level Agreements (SLAs) management, etc.; iii) *Change-time governance* handles service enhancements by providing versioning and change management policies and procedures.

Run-time governance policies are implemented by a repository or other infrastructure component. Design-time and change-time governance are implemented through well-defined governing processes.

A formal structure will need to be established to manage governance. The SOA Center of Excellence (CoE) is one of the mechanisms that can be set up to provide a governing body as well as serving as a focal point of expertise for SOA. This structure leverages the skills, methodologies, best practices and available knowledge so that benefits can be realized as a result of increased reuse. The SOA CoE must: i) promote the reusability and adoption of best practices and standards across the enterprise; ii) provide a consistent approach for the identification of reusable business services and for application development; iii) build a governance framework and formulate the SOA governance model; iv) provide education, assistance, mentoring, communication and feedback across projects.

In general, the CoE organizational structure will include experienced project architects and product specialists led by a senior architect. Product specialists mentor development teams on the product best practices required by the SOA implementation. The project architect, on the other hand, leads project teams, helps customers to identify and define reusable services, and provides guidance on how to leverage SOA to reduce costs. Finally, a senior architect leads the activities of the CoE and reports the ongoing status of the SOA initiatives to the executive management team. This leading architect acts as the primary technology contact with customers and works with them to deliver solutions to their needs. Although these are the main roles involved in the core structure of the CoE, it is important to note that this kind of effort is a shared responsibility. For the CoE to evolve and become an organizational platform it is required the active involvement of diverse participants. Software engineering and enterprise architecture representatives, developers and business users must also participate in the CoE working sessions. They are encouraged to communicate their concerns and give feedback to improve the governance models that will be influencing their particular activities. In order to enforce participation, executive leadership is essential to the success of the SOA initiative.

5 A Process-Aware Service-Oriented Methodology for the Development of SOA Applications

This section describes the proposed methodology for developing SOA solutions in a BPM scenario with a governance enforcer CoE at its core. It is based on a Macro-level Development Model that considers a number of concepts that we have identified to be relevant in the construction of an SOA application.

5.1 Towards a Macro-level Development Model

We use the term “Macro-level Development Model” to express the high-level concepts needed for business process automation in the context of service-oriented development. We emphasize the mechanisms available to the developer and their relationship with other components in a general governed setting.

This model is under refinement and will not be treated comprehensively in this work. However, it plays an important role as an abstract conceptual view of the desired development and execution environment.

The main focus of the proposed model lies on the ability to use distributed functionalities by composing services. These composed services can later construct business processes with an evolutionary approach.

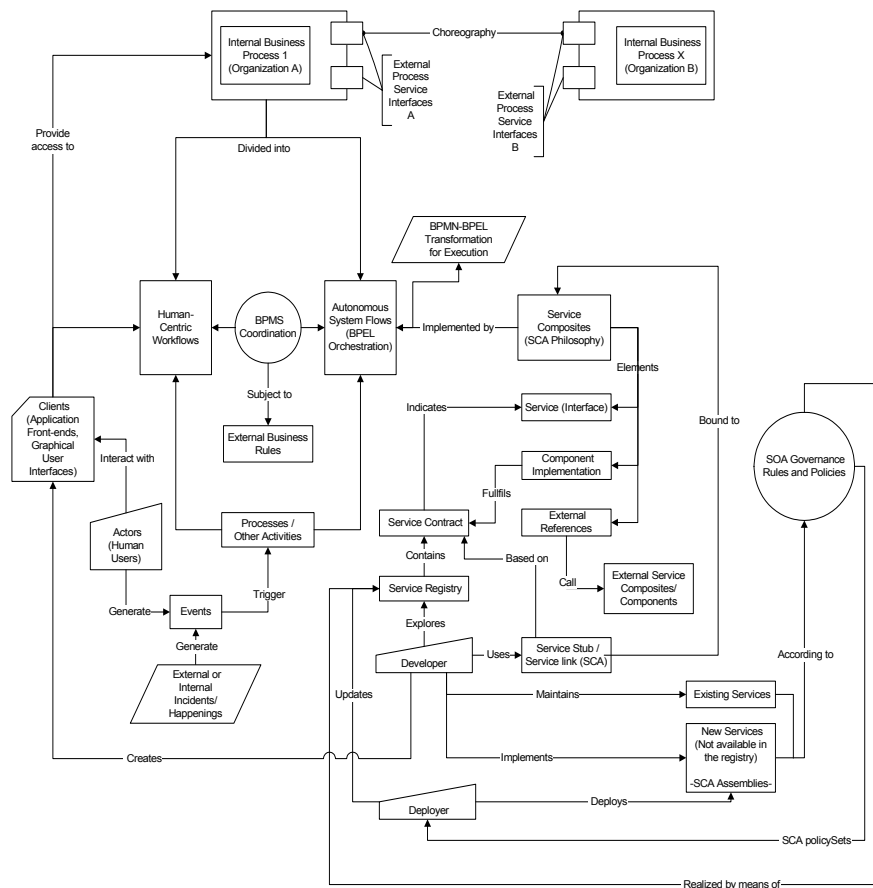


Fig. 2. Logical view of a Macro-level Development Model for SOA Applications

Figure 2 shows the logical view of the proposed development model. Its structure implies the definition and automation of business processes through high-level service compositions –both inside and outside the enterprise–, and it relies on the following concepts: i) Model-Driven Development, for modeling workflows and system flows, allowing the translation of business process models described in BPMN into executable BPEL models; ii) Service Component Architecture (SCA) [11] to implement business services according to standard composition and assembly rules; iii) Event Processing, as the point of user interaction and external happenings that trigger the execution of business processes; iv) Business Rules-based Development, as a mechanism to manage external rules that are independent of the process logic where they are used. This allows the simplification of process models and a centralized governance of those rules.

As depicted in Figure 2, a central point in the development model resides in the coordination capabilities enabled by the Business Process Management System (BPMS). It acts as a fundamental mechanism to manage the execution of business processes through the coordination of workflows and system flows instantiated by user events. The composites in SCA are implemented by BPEL code that defines the orchestration of discrete services into higher-value business services. The role of service development is highlighted regarding its interaction with the registry/repository as an enabler for SOA Run-time governance policies.

We consider Macro-level Development as an extension to what has been traditionally defined as Programming in the Large [12]. The broader scope of our model goes far beyond the mere composition of services into coarser-grained processes. Similarly, a subsequent Micro-level Development Model could be established to support the actual implementation of elemental services that will be later used in Macro-level Development. However, its scope would be equivalent to that of Programming in the Small [12].

5.2 The Methodology Scenario and Phases

In a typical SOA development methodology, we can identify several common activities. Generally, we can divide the development lifecycle into Analysis, Design, Implementation, Execution and Monitoring phases. The output of each phase realizes the requirements of the functional entities that we proposed in our Reference Framework (RF) [6]. Figure 3 depicts the proposed methodology having a CoE for the enforcement of governance policies as a core element of the overall initiative. The SOA CoE acts on three domains: design-time, run-time and change-time governance, depending on the corresponding development phase.

According to Figure 3, it is required an initial planning phase conducted by senior business and IT management. In this phase, the business context and drivers, strategic goals, and the tactical roadmap to accomplish such goals must be defined. The project team is established, and initial information is gathered.

SOA Governance must be planned to grow incrementally, and the organization must be aware of governance issues from the beginning. The envisioned roadmap has to consider these both for short and long-term objectives.

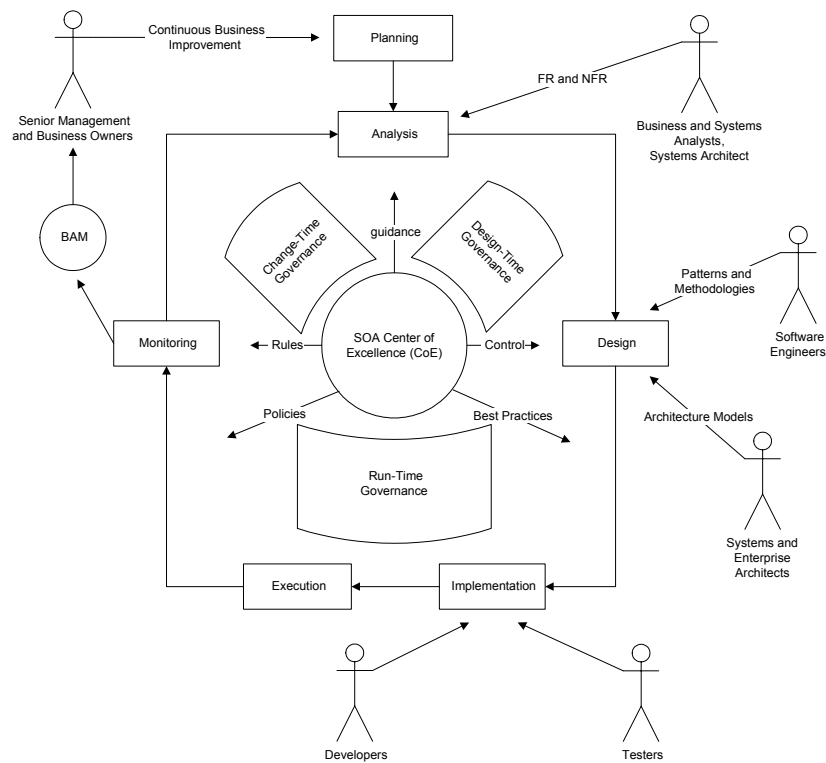


Fig. 3. BPM-SOA Development Methodology Centered on Governance Processes

The tasks in the analysis phase are carried out by business and systems analysts and architects. They assess the gathered information to decide which business processes should be supported based on the business model attributes. Then, they develop a reviewed as-is business process model of the selected processes. Finally, Functional and Non-Functional Requirements (FR and NFR) are defined. Technical issues should be considered in order to identify possible restrictions due to limitations of the current systems. The analysis phase must include the definition of a preliminary catalogue of legacy systems and candidate services. Service identification can be approached from a top-down (strategic) and/or bottom-up (tactical) perspective depending on the goals pursued by the project. The SOA CoE will suggest guidelines on this matter and provide feedback based on past experiences and ongoing projects. Since there is not one single solution for governance, particular choices must be made in order to meet the specific requirements of the enterprise.

In the design phase, the as-is model is optimized to reflect the goals of the business. This optimization can be achieved through simulation techniques embedded in BPMS systems, producing a to-be business process model, which will be supported by the new application. The new optimized processes are then mapped to candidate services. The required logical flows of information and user interfaces are also

defined. System use cases are specified and detailed. Governance in this phase is enforced by means of procedures aimed at the correct definition of services (design-time governance).

The implementation phase deals with the new requirements of the to-be business process model. The candidate services are built (or reused), tested and orchestrated into high-level composites that can be consumed by the BPMS during process execution. It is also needed to select and put in place the right infrastructure (both for BPM and SOA technology, including the ESB). The Front-end interfaces that the user will be utilizing to interact with and trigger business processes are built and tested.

Execution phase implies the installation and deployment of the final SOA application. Governance in the implementation and execution phases is leveraged by the service registry/repository at run-time.

Finally, the monitoring phase considers the use of a Business Activity Monitoring (BAM) method so that business owners can have real-time information about the performance of business processes. Depending on such performance, another cycle of planning/analysis can be initiated to enhance the business processes, instantiating the development methodology once again in an iterative and evolutionary fashion. The new business objectives will be based on previous results and expected future improvements.

The SOE CoE must have a communication mechanism accessible to all the participants in order to facilitate knowledge sharing through all the phases of the methodology. Its members (product specialists, lead and support architects) play an active role in the definition and management of policies, rules and practices used by project teams. Workshops will be held at specific points to present results, recommendations, receive feedback and state future, short and long-term objectives.

The described development process follows a sequential path and a top-down approach (from strategic business models and processes to tactical integration solutions), but this is not always the recommended scenario. The approach to be taken depends on the particular characteristics and needs of the organization. In general, the best overall results we have had in applying our methodology come from a combination of top-down and bottom-up approaches. The activities to perform within each phase will depend on the approach being considered. Equally important is to emphasize that the proposed methodology must be viewed not as a replacement, but as an extension of other methodologies existing in the software engineering field, e.g., RUP (Rational Unified Process), XP (Extreme Programming), Scrum, etc. The best practices contained in these methodologies can be leveraged to implement our development methodology. We have found that, initially, it is better to start with extensive methodologies like RUP to put in place the right management foundation for the SOA. Agile approaches like XP and Scrum can be engaged later on to deal with individual SOA projects.

6 Conclusions

Governance is essential in the SOA implementation process. The macro-level development model and methodology presented here provide an empirical technique

to lead a BPM-SOA initiative on a governance foundation. The SOA CoE is an important platform to enable the execution of governance practices within the development process in an enterprise SOA. In order for the CoE to grow, expand, and improve, it must have the acceptance, support and active involvement of all stakeholders. Business and IT leadership is key to succeed with SOA governance.

In addition, we have realized that the best BPM-SOA solutions arise from the combination of different development approaches depending on the SOA and process orientation maturity of the enterprise. This is discovered and analyzed in the initial phase of the project.

The presented methodology shows a brief summary of the work that we have done in the BPM and SOA fields in collaboration with leading institutions in the financial services industry, particularly in the consumer banking domain. Our academic research is thus being constantly applied to practical industry problems, which leads to a continuous improvement of our methods.

Currently, our methodology is being tested in different projects, which is continuously keeping us informed of its capabilities and limitations. Definitely, this is just a first step. There is still a lot of work to do in the governance field when approaching a BPM-SOA integration project. A further, deeper assessment must be conducted so that our methodology can be enriched with best practices and detailed guidelines and recommendations.

References

1. Papazoglou, M. P., van den Heuvel, W.: Business Process Development Lifecycle Methodology. In: Communications of the ACM, vol. 50, no. 10, pp. 79--85. ACM, New York (2007)
2. Erradi, A., Anand, S., Kulkarni, N.: SOAF: An Architectural Framework for Service Definition and Realization. In: Proceedings of the IEEE International Conference on Services Computing, pp. 151--158. IEEE Computer Society, Washington (2006)
3. Erl, T.: Service-Oriented Architecture: Concepts, Technology and Design. Prentice Hall, Upper Saddle River (2005)
4. Van der Aalst, W.M.P., ter Hofstede, A.H.M., Weske, M.: Business Process Management: A Survey. In: Van der Aalst, W.M.P., ter Hofstede, A.H.M., Weske, M. (eds.) BPM 2003. LNCS, vol. 2678, pp. 1--12. Springer, Heidelberg (2003)
5. Chappell, D.: Enterprise Service Bus. O'Reilly, Sebastopol (2004)
6. García, C., Maldonado, A.: Integrating Workflows and System Flows for the Development of SOA Applications (in Spanish). In: Proceedings of the XXXIII Latin American Conference on Informatics (CLEI 2007). San José, Costa Rica (2007)
7. OMG Business Process Modeling Notation (BPMN) v1.1, <http://www.bpmn.org>
8. OASIS Web Services Business Process Execution Language (WS-BPEL) v2.0, <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>
9. OASIS SOA Reference Model v1.0, <http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.html>
10. SOA Governance: Adoption and Best Practices, <http://www.soainstitute.org/articles/article/article/soa-governance-adoption-and-best-practices.html>
11. OSOA Service Component Architecture (SCA), <http://www.osoa.org>
12. Leymann, F.: Web Services: Distributed Applications without Limits. In: Weikum, G., Schöning, H., Rahm, E. (eds.) BTW 2003. LNI, vol. 26, pp. 2--23. Springer, Leipzig (2003)