Evaluating Integration Architectures – A Scenario-Based Evaluation of Integration Technologies

Stephan Aier and Marten Schönherr

Technische Universität Berlin, Sekr. FR 6-7, Franklinstr. 28/29, 10587 Berlin
stephan.aier@tu-berlin.de, mschoenherr@sysedv.tu-berlin.de
http://www.sysedv.tu-berlin.de/eai

Abstract. A major aspect of complex Enterprise Architectures is the integration of existing heterogeneous IT-systems in a business process oriented way. The paper starts with the definition of terms as Enterprise Architecture and process orientation. Based on an empirical study the paper shows that there is no significant business process orientation in information system integration projects. Among other reasons this is due to deficits in understanding and managing integration methods and technologies. Therefore the paper addresses the evaluation and comparison of relevant integration architectures as a first step to work on that issue. The paper differentiates individually coded interfaces, centralized hub&spoke and distributed approaches based on standardized interface descriptions (Service Oriented Architecture – SOA). The mentioned empirical study was extended by an action research based prototyping to assure a reliable evaluation and comparison of the three integration architectures. To make them comparable they have been implemented in the same fictitious business scenario which is described briefly. The paper finally compares the integration architectures with a set of 11 criteria which summarize over 400 variables taken into consideration in the evaluation process. The conclusion of the paper is not a solution but a suggestion for further research.

1 Enterprise Architecture – Harmonizing Business Processes and IT Architecture Using Integration Concepts

In the current discussion on integrative enterprise architectures generic approaches are missing – an industry standard is far away. A precondition for the standardization of methods and technologies is the definition of generally accepted requirements and criteria. A common understanding of criteria supports or even enables an evaluation of the appropriateness of both: methods and technologies of integration concepts.

The paper starts with foundations concerning methodological and technological aspects of integrative enterprise architectures followed by a 3 step approach which includes an empirical study on the understanding and use of integration

D. Draheim and G. Weber (Eds.): TEAA 2005, LNCS 3888, pp. 2–14, 2006.

concepts, a prototypical implementation of different integration architectures and finally the evaluation of the chosen integration architectures based on the study and the implementation. The 11 criteria used for the evaluation were taken from an industry funded survey which used more than 400 variables to evaluate integration technology sets including middleware and EAI (Enterprise Application Integration). Considering length restrictions the paper will briefly introduce the chosen integration architectures along the 11 criteria. Two criteria will be described in detail.

An important precondition for a stringent consideration is a coherent understanding of *enterprise architecture*. An architecture can be understood as an abstract, holistic view on structures and patterns [1]. Architectures are usually the result of a planning process and thus represent a master plan for holistic realization of future measures. These general characteristics applied on enterprises lead to the term of enterprise architectures. Enterprise architecture is the combination of organizational, technical, and psychosocial aspects during planning and development of socio-technical business information systems. The paper addresses organizational and technical aspects of enterprise architectures. Therefore we use the terms organizational architecture and IT architecture. Our understanding uses the term integration concept which refers to methods and technologies supporting the integration of information systems in a business process oriented way. In this context the term business process is not following a consistently accepted definition. In our understanding a business process is the sequence of at least two activities which are based on tasks relevant in a business context done by employees responsible for the execution and/or control of the activity or task. Especially in the context of the paper a business process has to be distinguished from technical processes which are describing technical activities or tasks as messaging, exception handling or similar concepts executed by IT infrastructures not human beings (Fig. 1).

Organizational architecture contains all non-technical components. It is similar to the instrumental understanding of an organization which includes all explicit regulations for structures and processes. We differentiate organizational architecture into organizational structure and business processes. IT architecture stands complementary to organizational architecture. It covers all technical

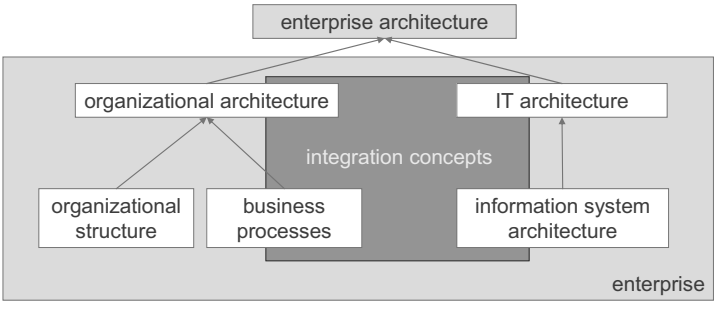

Fig. 1. Enterprise Architecture

components of enterprise architectures - especially information systems with their individual information system architecture.

In scientific literature the terms organizational architecture and IT architecture are used inconsistently. Depending on the authors background organizational architecture addresses technical aspects [2] and IT architecture contains organizational issues [3]. We keep the concepts separated to emphasize their importance. This seems to make sense, since there are complex interdependencies between both of them and they significantly influence operational efficiency of enterprises [4, 5, 6, 7, 8].

Due to changing business requirements business processes usually need to be changed frequently. The IT architecture has to meet the new requirements in an efficient way. Complex, historically grown and therefore heterogeneous IT infrastructures are not designed in a manner which would support changing requirements. The ability of supporting changing requirements is described as sustainability of enterprise architectures [8]. Integration concepts are a possible solution to increase sustainability of enterprise architectures hence reconcile organizational and IT architecture in an agile and volatile surrounding.

The field of sustainable enterprise architecture suffers from a lack of academic, empiric and generic approaches. Therefore we made a study on integration concepts in the context of enterprise architecture which is described in the next chapters.

2 Empirical Study – Objectives and Method

The study analyses how large organizations implement and use system integration concepts focusing on technology and their interdependencies to organizational aspects hence structures and business processes. These research issues have been analyzed in a descriptive and explorative manner. The study has been designed as a non-experimental cross section enquiry over a short period of time primarily using a written standardized questionnaire. Based on the results of the predominant descriptive and quantitative analysis a second questioning has been made. It was based on expert interviews hold individually or in small groups. The purpose of this evaluation was to verify the results and a possibly underlying thesis. Finally a factor analysis has completed the empirical study. Using 14 variables this multivariate analysis generated 4 factors supporting the quantitative analysis. 63 answered questionnaires could be used for data extraction. 5 individual interviews had been done and 25 people participated in a group workshop.

The study categorizes results in the following 4 major topics:

- common understanding of integration concepts
- how to organize the introduction of integration concepts
- interdependencies between integration concepts and business processes
- interdependencies between integration concepts and organizational structures

The next chapter will describe a compendium of the first and the third aspect only due to their relevance for the context of the paper.

3 Integration Concepts – Common Understanding and Interdependencies with Business Processes

Integration concepts are generally understood as an established technology predominantly used in large organizations to implement connectivity between existing IS and to replace individually coded interfaces. It is considered to be a strategic long-term element of IT architecture providing a modular toolset basically including software adapters (connectors), data transformation tools (mapping), monitoring and workflow features and business process management functionality. To operate implemented integration infrastructure even huge organizations employ a few technically well educated staff members only. Compared to the immense effort accepted during the implementation phase it is surprising that just a few people are carrying the valuable knowledge according to a very expensive infrastructural investment hence many organizations do not pay as much attention as necessary to an integrated understanding of the enterprise architecture. Even if the responsibility for integration infrastructures is located in a department responsible for organizational development issues usually there are too few people to manage the challenge. Furthermore the common understanding of integration concepts is technology driven hence in many cases the IT department operates integration infrastructures. IT departments usually provide enough manpower but not the organizational understanding of business processes nor the authority to influence organizational issues. Therefore the IT department is not the right instance to be responsible for architectural issues in the manner of a holistic conception.

The objective to design and run Integration Concepts in a business process oriented fashion has rarely been reached in practice. Implementations done in a process oriented way usually mean technical processes as e.g. message queuing and exception handling not business processes (Fig. 2).

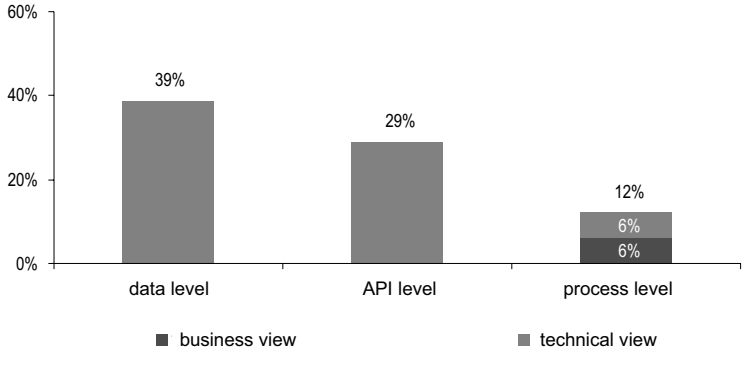

Fig. 2. Process Orientation in Integration Projects

However, in most cases integration projects cause business process changes. These projects shall lead to better, more efficient and faster business processes. The most important process changes are automation and consolidation. These process adaptations result in more formalization of organizational structures. Hence integration approaches are indirect enablers for business processes which only become feasible through such projects.

As a result of the empirical study there are no significant implementations in the companies questioned reconciling business processes and IT architecture in the meaning of a holistic enterprise architecture due to the dominance of a technology driven understanding of integration concepts. The methodological aspect in Integration Concepts mentioned above is not taken into consideration on a sufficient level. This result has motivated further considerations described in the next chapters.

4 Methodology vs. Technology

Integration projects with the objective of harmonizing organizational and IT architecture usually deal with non-technical issues too. Irrespective from the integration topic reflections on methodological and technical aspects have a long tradition in the research on information systems.

After studying the relevant literature we state that holistic architectural concepts consist of interrelated methodological and technical aspects. Methods and Methodologies can be found amongst others in the filed of Enterprise Architecture Frameworks [9, 10] and in the approaches of organizational and technical patterns of integrated architectural design [11, 12, 13].

The paper focuses in a first approach on evaluation criteria of relevant integration architectures. Literature on integration technologies often uses product related categorizations [14, 15]. The present article uses three categories based on the architecture of the integration solution itself. We differentiate between individually coded unidirectional or bidirectional point-to-point interfaces, centralized hub&spoke architectures, and decentralized service oriented architectures (SOA). To some extend it is possible to implement these categories of integration architectures with commercial EAI products. However, the categories are used as abstract concepts for planning and implementing integration projects. Even though real architectures will often mix different architectural types, these abstract concepts are suitable for an analysis. For the field of individual interfaces [18] literature provides detailed comments regarding concepts and implementation, hub&spoke is often described as best practice [14, 15, 8] and SOA [16, 17] being the latest approach is rarely described in literature yet. So far there are no serious sources providing a validation or even comparing description of the integration architectures. Many statements are even emotionally supporting one specific approach to be the best.

Therefore we designed a test scenario at the EAI Competence Center of the Technical University Berlin to implement the three types of architecture for system integration using comparable conditions demanding a business process oriented implementation.

The general facts according to the test scenario and the implementation of the three integration architectures in the test scenario will be described briefly in the next chapter. A detailed description would not fit the length restriction of the paper.

5 Scenario

The fictitious medium-sized company WMYPC (We Make Your PC) sells customized computer systems starting from small multimedia computers to enterprise server systems. The company consists of five departments. The business processes of all departments are modeled using UML activity diagrams. These processes are supported by six individually implemented information systems and one off the shelf software product. Hence the firms IT landscape is heterogeneous. The seven existing information systems use three different concepts of data storage and are distributed over several computers. The aim of the test scenario was to realize an adequately complex business environment with business processes selectively supported by heterogeneous information systems.

After business process definition, modeling and implementation of the scenario's IT infrastructure we have integrated the existing information systems in a business process oriented way with the described architectural types each in an own scenario.

Uni- and bidirectional interfaces have been implemented on the basis of XML documents which are exchanged manually (batch). The hub&spoke implementation has been done on seven commercial EAI products. For process orientation

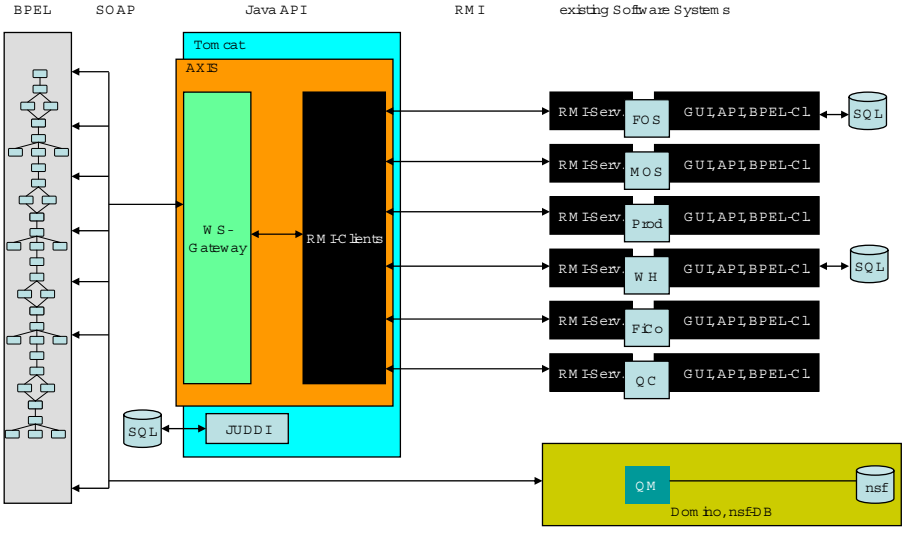

Fig. 3. Integration Architecture SOA

we have used and evaluated the modeling components provided by the respective tool vendors. The SOA has been implemented using web services technology. Orchestration has been done with ORACLE BPEL Manager and BPEL4WS. The seven information systems are encapsulated in 58 services and executed with four BPEL processes. Figure 3 shows the SOA integration architecture. The scenario – business processes as well as IT architecture – are the same for all three architectural implementations.

The following chapter briefly describes 11 criteria used for the evaluation and comparison of the integration architectures implemented. They have been created by merging over 400 detailed variables used in an industrial project surveying integration technologies. The evaluation and comparison uses results from the empirical study and experiences from the prototypical implementation considering the 400 variables. Still the presented results are not generic in the sense of being universally valid in any situation. They have to be taken as a basis for further research.

6 Criteria and Evaluation

To evaluate and compare the integration architectures we used the following criteria considering numerous detailed aspects according to design time and run time characteristics of the implemented architectures.

- *Initial planning efforts:* Implying an existing IT department, the criterion considers efforts needed to support internal staff with the ability/competence/knowledge to design, implement and run the integration technology and/or the costs of mandating external consultancy for supporting this people enabling process. The criterion focuses on the starting phase of an integration project or architectural change.
- *Initial development efforts:* This criterion considers effort put into initial development of the integration technology. By using as many standardized software products as possible the effort is usually comparably low. Developing the integration architecture individually often means to accept a high effort. The criterion is rather meant to evaluate the overall costs of an integration project but to differentiate between the software customizing and coding contingent of a project.
- *Persistent modeling:* There is no common understanding of modeling issues in the field of architectural integration. Most approaches (or commercial products) model technical aspects as message queuing and/or exception/error handling etc. As stated in the paper business process orientation is an important integration paradigm. Persistent modeling means methods, notations and tools for modeling business processes as well as technical workflows in a consistent way to follow a business process driven integration approach.
- *Technical adaptations:* In case of changing integration requirements there is an impact on the integration technology architecture. The criterion considers development efforts that have to be made to enable the integration architecture to meet new requirements.

- *Non-invasive legacy/host integration:* Particularly monolithic coded software systems without standardized interface descriptions or even connectors are very hard to integrate in enterprise architectures. Many organizations running legacy systems are not going to change these systems to improve their adaptability due to reasons of stability, complexity and the risk which comes with a system change. This criterion describes the ability of the considered integration architectures to integrate so called legacy systems in a non-invasive way.
- *Security:* Integrating complex enterprise architectures is even more a security issue than considering security aspects of a single software system. Integration is driven by system communication and multi-user impact. Therefore this criterion evaluates the integration architectures according to their security features. This criterion becomes even more important considering cross-company processes.
- *Maintainability:* The criterion compares features (general administration tools, load balancing, meta data management, monitoring etc.) and efforts to maintain an integration architecture.
- *Customizability:* Customizing integration architectures to meet changing requirements seems to be the complement to 'technical adaptations'. However, there are many situations where changes do not need to affect the technical structure of integration architectures but ask for a specific degree of freedom in customizing existing features according to the requirements. Usually generic elements as customizable objects, adapters or even generic front end forms increase the degree of customizability.
- *Stability:* This criterion considers the technical reliability of integration architectures. Failover or offline state is needed to be minimized in complex enterprise architectures due to direct and indirect impact on secondary areas as data and business process quality and processing time.
- *Transactionality:* Transacting huge amounts of data in a specific time is one of the common tasks in software system communication and integration. Transactionality guarantees a complete processing of a specific number of interactions between information systems as a single coherent step, i. e. either all interactions will be processed or not. This may include a rollback of interactions. This criterion becomes crucial and demanding at the same time with complex business processes distributed among several information systems.
- *Costs of operation:* Operating integration architectures addresses a significant quota of the overall costs. They should be taken into consideration when choosing the right integration architecture.

Table 1 summarizes the results of the evaluation and comparison process. Due to the fact that the results are partly depending on experimental research design the chosen scale seams to be crude but sufficient and adequate for the objective.

The evaluation and especially the comparison have to be interpreted in direct relation between the different architectures. For example a 'very poor' demonstrates a characteristic compared with the other evaluated architectures not an absolute evaluation.

Table 1. Evaluation of integration architectures: −− very poor, − poor, ∘ neutral, + good, ++ very good

criteria	individual interfaces	hub& spoke	SOA
initial planning efforts	+	−−	−
initial development efforts	−	+	−
persistent modeling	−	++	+
technical adaptations	+	∘	+
non-invasive legacy/host integration	−−	+	∘
security	++	+	−
maintainability	−−	++	−
customizability	−−	++	−
stability	++	+	∘
transactionality	++	∘	−
costs of operation	−−	+	∘

Due to lenght restrictions we will explain which arguments led to the respective evaluation of the alternative architectures for two of the criteria only – initial planning efforts and persistent modeling.

Initial planning efforts for designing and planning the implementation of an individually coded point-to-point interface are often quite moderate especially when information systems provide documented interface descriptions. A necessary precondition is existing basic know-how in the integration technologies applied – usually widespread programming languages. This is true for most IT departments of large companies.

In comparison there is often no significant competence for complex EAI integration tools (hub&spoke) which not only aim at the connection of two information systems but on the integration of a whole IT/IS landscape. Such projects are not only technologically demanding but also methodologically because the project has to consider many different requirements and stakeholder interests. As the empiric study points out this usually causes a massive involvement of external consultancy. Figure 4 shows that most integration projects make use of external consultancy in the fields of project management, technical support, IT architecture etc. hence the initial planning efforts are immense compared to the implementation of an individual interface. Prototyping the EAI hub&spoke architectures in the test scenario has been very demanding especially in the first phases when it came to the understanding of proprietary technologies, methods and features offered by different EAI vendors. Once understood the implementation was fast and efficient.

Initial planning efforts implementing an SOA is methodologically as demanding as the hub & spoke architecture especially because there are neither formal guidelines nor many best practices in the field of designing SOA. Technologically it is less demanding due to standardized straightforward technologies used in SOA such as web service descriptions. The use of standards in SOA compared to many proprietary features in EAI (hub&spoke) reduces initial planning

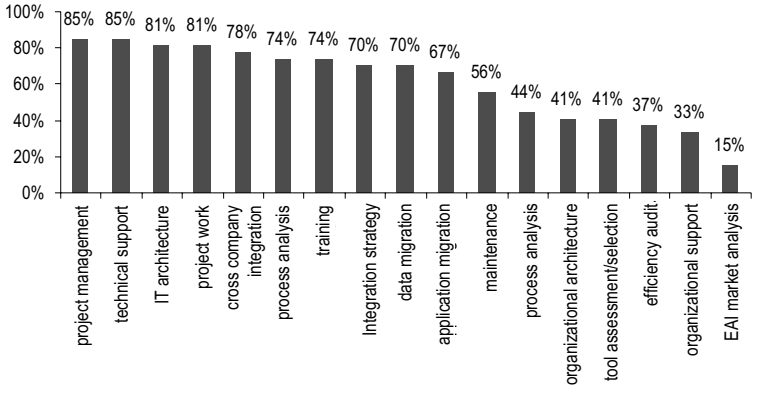

Fig. 4. Frequencies of external consultancy services in EAI projects

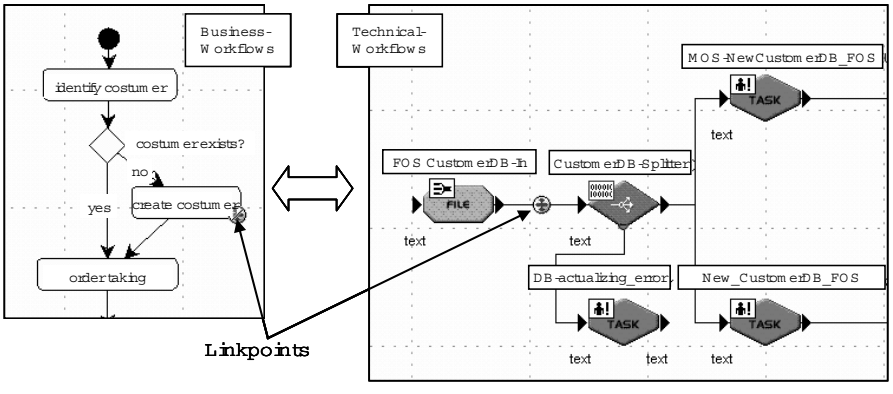

Fig. 5. Connecting business and technical workflow models in EAI

efforts. The effort is much higher compared to an individual interface especially because *Persistent modeling* is rated poor for individually coded interfaces due to the fact that individual interfaces are usually implemented not according to generic business process requirements but individual requirements of connectivity, transactionality and syntactical mappings considering the specific software systems. There is no generic or standardized modeling approach, method, notation or tool used to design and implement individual interfaces considering business processes used in practice. Some might use modeling techniques to support the implementation of individually coded interfaces but the empirical study found out that hub&spoke or SOA replace point to point interfaces for reasons of more generic modeling features offered by these architectures.

EAI (hub&spoke) has been evaluated as 'very good' in the criterion of persistent modeling. All implemented products offer modeling methods, notations and tools with graphical user interfaces. There are definitely differences in quality

and usability between the different vendors but compared to individual interfaces and SOA the features are extremely sophisticated. Usually the products differentiate business and technical workflows. They offer diverse methods to connect these levels of abstraction. See Fig. 5 for an example of link points to connect business and technical workflow models. The modeled workflows often are executable in real time environments supporting reports for controlling and business analysis.

The service oriented integration architecture orchestrates services in a basic or structured way. To do so a script language BPEL (BPEL4WS) and an engine to execute the BPEL files is needed. Usually a simple graphical interface

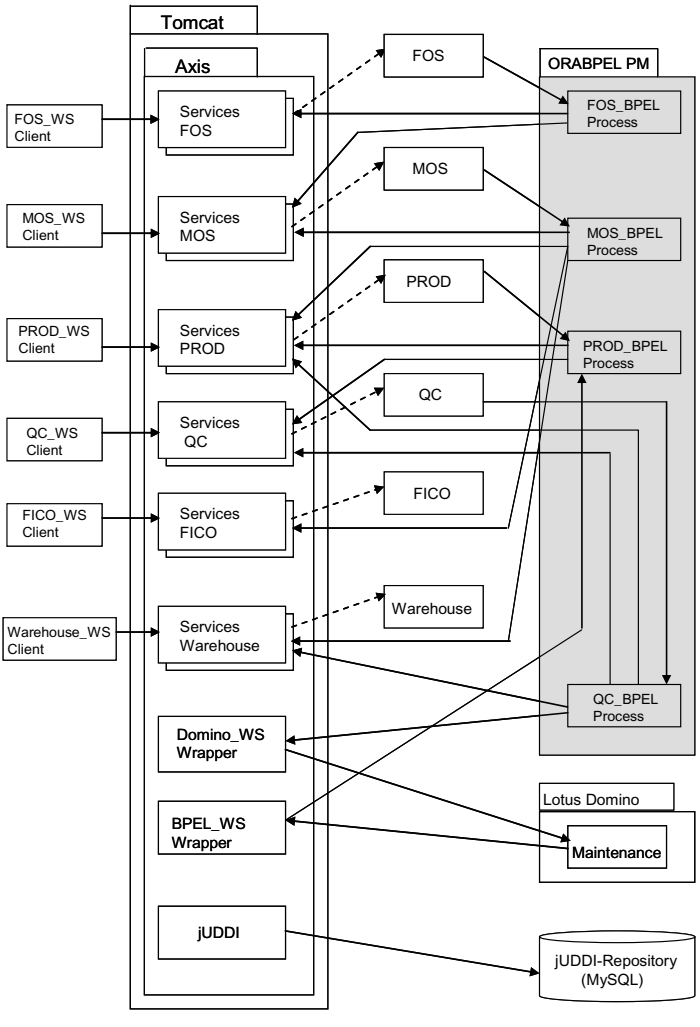

Fig. 6. Encapsulated and orchestrated services in a SOA

is provided to model the service composition. There is a huge variety in orchestrating services especially structures as sequences, parallel flows, switches, loops and event driven activities. Often the graphical interface is not featuring all possible modeling algorithms therefore a development environment to implement script languages is provided. Programming competence is essential for this kind of modeling. A composition of simple services to complex services is possible. The SOA has been evaluated as 'good' in the criterion of persistent modeling due to fact that there is a possibility of orchestrating and composing services along business processes using standardized references and tools. Still there are deficits in usability and full execution of complex business processes by modeled services. There is a high potential but the approach will need some more time to become approved. An important unsolved issue is the granularity used in the service definition phase. Without generic service definition rules there is no structured way to a later orchestration of services according to business process requirements due to the fact that flexible orchestration heavily depends on the technical implementation of a service. The implemented SOA encapsulates 58 services used and orchestrated in four BPEL processes. See Fig. 6 for an overview.

Web services are just one way to implement SOA. Therefore BPEL is not an overall standard for modeling services. The approaches used in modeling SOA are mostly based on UML and BPEL and focused on technological details. A stringent persistent modeling between business and technical processes depends on the specific implementation. The described deficits lead to an evaluation of 'good' especially according to the mentioned future potential.

7 Conclusion

Integration concepts as a central approach to sustainability in enterprise architectures need appropriate methods and technologies. There are few generic approaches guiding the practitioner towards a sustainable enterprise architecture. The paper delivers first results on surveying interdependencies between relevant parts of an enterprise architecture and on evaluating different architectures to integrate the interrelated architectural elements (business processes and IT architecture). After evaluating and comparing integration architectures there are no generic patterns which could be used to suggest a single architectural approach to be the most appropriate integration architecture in general. Specific situational requirements guide to the right technology mix used for an integration concept. As a result of the empirical study described in the paper there is a lack of holistic architectural understanding. Methods need to be developed considering life cycle aspects in persistent modeling. A stringent modeling approach containing a methodology, methods and notations should be developed. Modeling different levels (from business to technical processes) and taking time line issues of the mentioned architectural elements into account are just first ideas for further work on methodical aspects of enterprise architecture.

References

1. Bass, L., Clements, P., Kazman, R.: Software Architecture in Practice. 2nd ed., Pearson Education Inc., Boston (2003)
2. Nadler, D.A., Gerstein, M.S., Shaw, R.B.: Organizational architecture – designs for changing organizations. Jossey-Bass, San Francisco (1992)
3. Zachmann, J.A.: A Framework for Information Systems Architecture. In: IBM Systems Journal **26** (1987) pp. 276–292
4. Markus, M., Robey, D.: Information technology and organizational change: Causal structure in theory and research. Management Science **34** (1988) S. 583–589
5. Leavitt, H., Whisler, T.: Management in the 1980s: New information flows cut new organization flows. Harvard Business Review **36** (1958) S. 41–48
6. Lewin, A.Y., Hunter, S.D.: Information Technology & Organizational Design: A Longitudinal Study of Information Technology Implementations in the U.S. Retailing Industrie, 1980-1996. In: H. Glaser, E.F. Schröder, A. v. Werder, (eds.) Organisation im Wandel der Märkte. Gabler, Wiesbaden (1998) S. 251–286
7. Venkatraman, N.V.: IT-Induced Business Reconfiguration. In: Scott Morton, M.S. (ed.) The Corporation of the 1990s. Information Technology and Organizational Transformation. Oxford University Press, New York (1991) pp. 122–158
8. Aier, S.: Sustainability of Enterprise Architecture and EAI. Soliman, K.S. (ed.) Information Technology and Organizations in the 21st Century: Challenges & Solutions. Proceedings of The 2004 International Business Information Management Conference, International Business Information Management Association (IBIMA), Amman, Jordan, (2004), pp. 182–189.
9. Schekkerman, J.: How to survive in the Jungle of Enterprise Architecture Frameworks. Trafford, Victoria, Canada (2004)
10. Noran, O.S.: A Mapping of Individual Architecture Frameworks onto GERAM. In: Bernus, P., Nemes, L., Schmidt, G. (Ed.) Handbook on Enterprise Architecture. Springer, Berlin (2003) pp. 65–212
11. Fowler, M. (ed.): Patterns of Enterprise Application Architecture. MITP, Boston (2003)
12. Hohpe, G., Woolf, B.: Enterprise Integration Patterns. Addison-Wesley, Boston (2004)
13. Brown, W.J., Malveau, R.C., McCormick, H.W., et al.: Anti Patterns. mitp, Bonn (2004)
14. Linthicum, D.S.: Enterprise Application Integration. Addison-Wesley Longman, Amsterdam (2000)
15. Ruh, W.A., Maginnis, F.X., Brown, W.J.: Enterprise Application Integration. Wiley, John, & Sons, New York (2001)
16. Krafzig, D., Banke, K., Slama, D.: Enterprise SOA: Service-Oriented Architecture. Prentice Hall, Upper Saddle River, NJ (2005)
17. Erl, T.: Service-oriented architecture: A Field Guide to integrating XML and Web Services. Prentice Hall, Upper Saddle River, NJ (2004)
18. Juric, M.B., Basha, S.J., Leander, R., et al.: Professional J2EE EAI. wrox, Birmingham (2001)