

# A value-driven approach to the design of service-oriented information systems—making use of conceptual models

Oliver Thomas · Jan vom Brocke

Received: 4 January 2008 / Revised: 28 July 2008 / Accepted: 2 February 2009 /  
Published online: 27 February 2009  
© Springer-Verlag 2009

**Abstract** We explain the importance of conceptual models for process design with service-oriented architectures (SOA) and illustrate how these models can be used to evaluate the value of SOA. To do so, a methodology is introduced, which comprises both the configuration and the implementation of processes by services and IT components. The approach presented here, makes it possible to evaluate the monetary and objective potentials of SOA, as well as to enhance SOA implementation by providing guidelines. The requirements analysis from an online-mail order company serves as a case study. The results show that up to now, economic aspects were ascribed too little importance in the SOA discussion.

**Keywords** Service-oriented architecture · Web services · Business process management · Value · Methodology · Conceptual modeling

## 1 Introduction

Information systems will change dramatically within the next years. Service-oriented architectures (SOA) make it possible to flexibly create information systems from independent sub-systems (services) (Leymann 2003; Krafzig et al. 2006). The idea is not to keep application systems as complete systems in enterprises, as practiced today, but rather to “pull” the services needed from the Internet and then,

---

O. Thomas (✉)

Institute for Information Systems (IWi), German Research Center for Artificial Intelligence (DFKI),  
Saarland University, Saarbruecken, Germany  
e-mail: oliver.thomas@iwi.dfki.de

J. vom Brocke

HILTI Chair of Information Systems and Business Process Management,  
Institute of Information Systems, Liechtenstein University, Vaduz, Principality of Liechtenstein  
e-mail: jan.vom.brocke@hochschule.li

configure them to make them fit the respective requirements (Fremantle et al. 2002, p. 78; Ferris and Farrell 2003, p. 31; Kreger 2003, p. 33). Already, the spreading of web service standards is leading to the implementation of this idea in operational application systems (Vollmer and Gilpin 2006). More and more, service-oriented product variants are, for example, being developed by the providers of ERP systems (e.g. SAP 2006).

Economic expectations lie in the flexible exchange of information system parts, which should allow a more differentiated coordination between organization and system design (Hagel and Brown 2003). New possibilities for IT/Business Alignment (Henderson and Venkatraman 1993) are seen in process design: on the one hand, an information system configured from services should be capable of adapting itself flexibly to changing operational requirements and on the other, one expects the configurability of services to open new operational fields of action (Luftman 2005). This coordination in organization and system design should result in an SOA that promotes the “agility” of companies and serves as a foundation for the implementation of a “real-time enterprise” (Fingar and Bellini 2004).

All in all, SOA poses the question of the value proposition of information technology anew (Carr 2004). In particular, the value of SOA use remains unclear. It must be said, that euphoric expectations were insufficiently justified by efficiency calculus. The results of the hype circle analysis for application system development carried out by *Gartner* confirm this estimation (Duggan et al. 2006). In fact, the study states that SOA is in a phase of potential over-expectation. Case studies from system providers add fuel to these expectations by showing triple-digit profits (SAP 2006; BEA 2006; Oracle 2007). These studies were carried out in participation with independent analysts, such as the IDC studies at *Goodyear* (Wang 2003b) and *Sasol* (Wang 2003a), which show a return on investment (ROI) of 308% resp. 453% from the introduction of SOA. These studies are highly problematical for decisions concerning the use of SOA, because one cannot simply transfer them to other contexts. It can be assumed that the potential depends on a multitude of influencing factors that vary heavily in different applications. Therefore, we need a methodology that allows us to evaluate the multi-use options of SOA in individual situations and then, to subsequently realize these.

This article will address the value of SOA. We will compare and describe the benefits of an investment in SOA and the costs connected with it. The article is organized as follows: previous work on the design, as well as the evaluation of SOA will be analyzed (Sect. 2). Due to the deficit in research on this topic, we will also consider contributions from neighboring disciplines. To fill the research gap we will design a methodology that can be used for decision support in the introduction of SOA. A framework will be developed for the construction of the methodology (Sect. 3). Then, the methods integrated through the framework will be selected resp. extended and specified (Sect. 4 and 5). The practical benefit of the methodology will be demonstrated with a case study (Sect. 6). We will analyze the case of the company *Matchball* and carry out value calculations on the implementation of an SOA according to the methodology introduced here. We then close with a discussion of the results and an outlook on future research challenges (Sect. 7).

## 2 Related work

The evaluation of the value of investments in information technology is a field controversially discussed within information systems research for some time now (Porter and Millar 1985; Brynjolfsson 1993; Mukhopadhyay et al. 1995; Hitt and Brynjolfsson 1996; Tam et al. 1998; Im et al. 2001). It must however, be considered that the general problems in evaluating investments in IT are augmented by SOA-specific challenges. Thus, the consideration of SOA value is, in addition to its primary intention—i.e. the economic justification of the investment—implicitly given a design function with regard to determining the optimum degree of service-orientation within an enterprise.

The multitude of research work dealing with evaluating the value of investments in information systems can be classified as *ex ante* or *ex post* with regard to the time of the evaluation, whereby the *ex ante* evaluation of investments is especially focused upon by research (for an overview, see, e.g. Walter and Spitta 2004). Up to now, an adaptation of existing concepts with regard to the evaluation of the value of SOA does not exist. Current research primarily concentrates on discussing the generic benefits of an SOA (Woods 2003; Erl 2005; Krafzig et al. 2006; Woods and Mattern 2006), without addressing the special requirements of SOA regarding a value analysis. In addition, a few isolated ROI calculations on the basis of selected practical examples also exist (Barnes 2005a, b).

The problem is complicated further by the fact that most of the previous work on SOA is predominantly technical nature. This goes so far as to reduce SOA to the integration of XML and web services (Hündling and Weske 2003; Erl 2004) or only consider thoughts on technical integration via web services (Ganesh et al. 2004; Mallick et al. 2005). There is, however, work that aims at decision support in selecting web services when the same functionality can be provided by different services. These contributions gather criteria as a distinguishing feature for web services (Kalepu et al. 2003; Zeng et al. 2003; Ran 2003; Yu et al. 2007), which in turn should help decide which service is adequate for a special purpose depending on the enterprise or IT-specific preferences. The “quality of service” is considered here as a central aspect in the orchestration of business processes. Up to now, questions on the economic efficiency of this new technology have hardly been investigated. We lack methods that can put decision makers in a position to economically evaluate and realize the use of SOA from their individual situation.

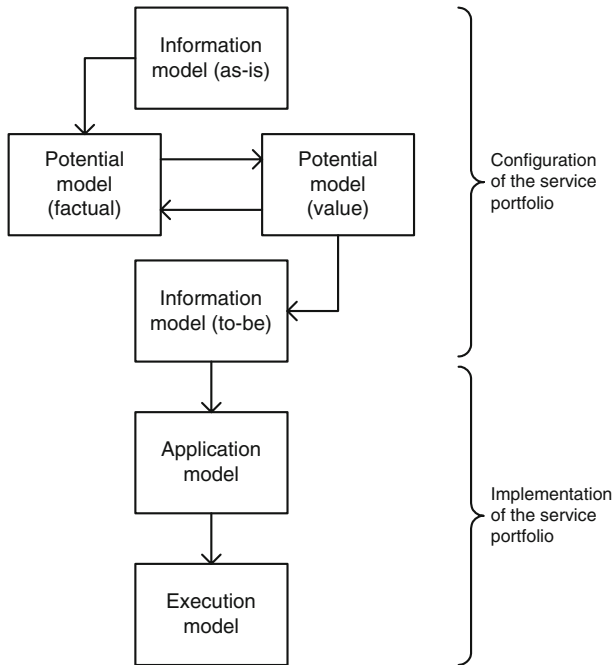
This article will introduce a solution for such a decision support through the usage of conceptual models. By constructing conceptual models, the attempt is made to create abstracting artifacts that make the complexity of information systems manageable (Hirschheim et al. 1995; Weber 1997; Mylopoulos 1998; Wand and Weber 2002). In information systems research it is widely accepted that conceptual models have a lasting influence on the quality of the software developed (e.g. Wolff and Frank 2005). Moreover, from a practical point of view, a current study from *Gartner* points out that the documentation of business processes with their processing times and respective responsibilities can lead to an increase in productivity of more than 12% (Melenovsky 2005). Consequently, conceptual models can also be used meaningfully in the development of service-oriented systems.

However, the idea of model-driven SOA design is not completely unheard of. Focusing on the transactional properties of web services, the use of UML class diagrams as a transactional layer above UML statechart diagrams describing the service's structure is proposed by Schmit and Dustdar (2005). Baresi et al. (2003) use UML models to decide about the aptitude of certain middleware platforms for business information platforms. Their approach shows some convergence between conceptual modeling and SOA but concentrates solely on the validation of SOA and not on the specific needs for designing an SOA. Emig et al. (2006) examine how the concept of programming in the large can be applied practically to SOA development based on process models. They use the business process modeling notation (BPMN) as a process programming language, which can be mapped to the business process execution language (BPEL). Because process languages are led by IT-relevant aspects, they are hardly relevant for business analysts designing an SOA driven by business needs. On the level of technical integration, the synergy effects between process modeling and SOA has been recognized by approaches related to the concept of model-driven architecture (MDA) (Frankel 2003). Pfadenhauer et al. (2005a, b) have proposed a model-driven service architecture (MDSA) but fail to integrate pure business requirements formulated by non-technical managers or domain experts. Contrasting to this prevailing technical understanding, Wenhui et al. (2006) confirm that the true potential of process-driven SOA lies in innovative business models, which are facilitated by web service infrastructure.

To sum up, among these existing approaches there is a missing link between the moderate innovations an SOA brings to IT integration and unprecedented potentials to be leveraged for business integration. None of the related contributions examined acknowledges the fact that business process modeling has established throughout companies by means of semi-formal diagrams that are easy to understand and handle by non-technicians. Our approach is to leverage this treasure of existing conceptual models created during longsome BPM projects. In addition, to complement the existing works, we should however aim at the evaluation of SOA value using conceptual models. Therefore, the models should not just span an alternative set that shows potentially usable services for the support of business tasks, but rather provide decision support to those responsible when selecting these services. Because this decision support must be deduced from considerations on value, the models must be enriched with evaluation criteria for the implementation of SOA.

### 3 A methodology for the model-based design of service-oriented systems

A fundamental idea in service-oriented architecture is the separation of process logic and process control in system design. This aims at structuring processes with respect to situational requirements and using selective services for the execution of activities. Thus, in principle, a predefined spectrum of system functionalities is not the starting point for the design of SOA, but rather an enterprises-individual situation. An adaptation between the business requirements and the possibilities on the technical side, as suggested in the framework of the model-based design of service-oriented systems (cp. Fig. 1), must be carried out.



**Fig. 1** Framework of the model-based design of service-oriented systems

Understanding the adaptation possibilities between the organization and system design is not simply a case of selecting technical components. In fact, process restructurings must be considered with regard to questions of added value, system integration and enterprise cooperation. Before the services and infrastructure elements used for the institutionalization of a process—here referred to as a *service portfolio*—can be implemented, the respective configurations must be carried out. The design possibilities given by the SOA must be listed (potential model factual) and evaluated according to their economic consequences (potential model value) for the configuration of the service portfolio. This way, the design alternatives profitable in a specific decision situation (as-is) can be found and then serve as guidelines for the implementation of the service portfolio (to-be) (cp. Fig. 1).

Despite the decision support available with this approach, the coupling of the service portfolio configuration and the service portfolio implementation remains difficult. The consequent transfer of business requirements into a service-oriented system is not guaranteed here due to the existing gap. The problem is the lack of continuity in modeling from the configuration to the implementation level. The basic idea of the integration of business and technical process description in our approach is to insert the process logic shared in both levels into a third level (cp. Fig. 1). Based on the given processes, information models, application models and execution models could then be differentiated between dependent on their proximity to IT.

This separation between information and application models is based on an understanding of information systems consisting of three main components: people,

tasks and technology (Bostrom and Heinen 1977; Orlikowski 1992). In our terminology, information models represent information about people, tasks and technology. They contain the potentially business-relevant framework for each of the tasks, functions or activities defined in the process organization. These can be, for example, organizational units, resources, respectively, costs or legal restraints. The components “tasks” and “technology” of an information system form the sub-system “application system”. In this sense, an application model represents a component of an information system. Application models represent the information needed for the IT-support of the processes. These can be for example, the services involved, required input and output data or the interface information needed for the integration of external partners. Lastly, execution models represent the technical component of an information system and contain all of the information required for the process execution.

The methodology for the model-based design of service-oriented systems therefore, follows the paradigm of a *method and model-based design theory for companies*, as postulated in the field of business engineering (Schelp and Schwinn 2005; Thomas et al. 2006; Österle 2007). With the methods available by business engineering (here: conceptual modeling) companies and strategic fields of business should be designed according to engineering principles by using the potentials of IT (here: SOA). In addition, the framework is oriented on *problem-solving strategies* from software engineering. The fundamental system development strategy “top-down” with its origin in human problem-solving (Newell and Simon 1972) can, in particular, contribute to solving problems within the framework of SOA projects. In analogy to human problem-solving behavior it is *first* assumed that one must proceed systematically for the model-based design of service-oriented systems, because at the beginning neither a complete set of alternatives, nor a final design goal can be defined. And *second*, a problem is broken down successively through desaggregation into sub-problems and relationships in the top-down approach. This breakdown is repeated until sub-problems emerge that are seen as solvable by the participants. These are then integrated to form an overall solution via the derived relationships.

Very different challenges exist for the development of methods in the areas of configuration and implementation of service portfolios. In the configuration phase, potentials must first be gathered and evaluated value-based. Because the representation of this information has not been dealt with in existing works, questions pertaining to the conceptual and representational extension of the modeling languages available for use come to the foreground. In the implementation phase however, the value-based best alternatives should be transferred step-by-step in IT components. Thus, the methodical support pertains more to action guidelines that describe how models should be created using modeling languages (cp. Fig. 1).

## 4 Configuring the service portfolio

### 4.1 Preliminary considerations

In this section, we will study information models that can be used by decision makers for the configuration of service portfolios. In accordance with the framework

introduced, the focus will lie on methods for constructing *potential models*. With these methods, the SOA potentials can first, be recorded *factually* and then compared with each other *value-based*. Because the evaluation takes place directly on the basis of the factual representation, service portfolio (re)configurations can be carried out, which are understandable with regard to their economic consequences.

Three levels of analysis can be derived from the technical works on SOA (Krafzig et al. 2006) and these can be transferred to the design of processes (Österle 1995, p. 16; Becker and Kahn 2003, pp. 4 ff):

- *Activity level*: sub-step in a process, which describes one change of state. Activities can be connected to each other logically and improved as a process.
- *Service level*: an alternative to the execution of an activity, which can exhibit different forms of institutionalization. In particular, a comparison between internal and external, as well as automated and non-automated services must be made.
- *Infrastructure level*: all of the requirements necessary for the integration of the services into the process. Non-technical requirements must also be considered here, which for example, pertain to organizational aspects of the service integration.

Thus, different aspects of service-oriented designed processes are recorded on these levels. They form the parameters for the configuration of a service portfolio. To study their adjustment methodically, they must be examined from a factual, as well as a value-based point of view:

- *Factual representation*: The factual representation is important for recording the design alternatives given by SOA. Thus, it must be shown which selection possibilities are available. These possibilities consist of specific combinations of decisions on activities, services and infrastructures that each correspond with a specific configuration of the service portfolio.
- *Value-based representation*: The individual, factually documented design alternatives are evaluated with regard to their economic consequences. In order also, to record long-term consequences, the payments connected to a service portfolio configuration must be calculated and compressed to key figures.

The integration of the dimensions described should make it possible to understand the effect of modifications in individual design areas on other areas. Examples are changes in the factual requirements on the service choice or the specific conditions of individual service providers. However, process reengineering can also be taken into account here. The mechanisms of the method integration will be discussed in detail in the following.

## 4.2 Factual representation of the potentials of SOA

Process modeling languages can be used for the factual representation of design options. For reuse reasons it makes sense to use languages already existing for the as-is modeling in specific application contexts. Examples where meta models exist are the EPC (Scheer 1999; Korherr and List 2007), Petri nets (Murata 1989, pp. 542

ff; Billington et al. 2003, p. 5) and UML activity diagrams (Rumbaugh et al. 1998, pp. 81 ff; Fowler 2003).

In contrast to conventional information models, potential models are characterized by *optionality*. Alternatives in the terminology of decision theory for the process design are represented, which again, must be tested with regard to their value. We can differentiate between two principles for the construction of models:

- *Explicit representation*: Explicit models are created for individual design alternatives. The optionality is expressed in the relationships between the models that, in turn, can be expressed in a model system with a corresponding operator.
- *Implicit representation*: The number of design alternatives is represented within individual models. Constructs for the explication of individual design options and rules for controlling any dependencies between options are required.

The value of an explicit or implicit representation is influenced by the number of alternatives, as well as by their diversity. If, for example, an out-tasking is being considered where alternative services from several service providers must be selected for the institutionalization of a local activity (Knolmayer 1997, p. 100; Keen and McDonald 2000), then the advantages of an implicit representation take effect. If however, restructurings are connected with the decision, in the case for example, of applications for process integration (Linthicum 2000, p. 18; Sutherland and van den Heuvel 2002), then an explicit representation may be superior for reasons of model clarity. The number of alternatives for explicit representation can be limited in order to enhance the value of model construction.

In the following, the new requirements on modeling languages for the implicit representation of design alternatives will be discussed in detail. The basic principle is illustrated in Fig. 2 on an EPC.

The representation of institutionalization alternatives in process models is pivotal for recording the potentials of SOA. Constructs, which make the description of processes on the activity, infrastructure and service level possible, must be allowed for here. As can be expected, process modeling languages provide constructs for the description of processes on the activity level, for example events, functions and operators of the EPC. In addition, constructs for executing activities are also provided (Becker et al. 2003, p. 21). According to our approach, these must be classified in the service level. In doing so however, the resources needed for executing activities are strictly assigned.

To record the design potentials of SOA, it seems expedient to extend modeling languages with constructs for the representation of *services* and *infrastructures*. Services fulfill a specific task and use a specific amount of resources to do so. The individual requirements necessary for the integration are recorded as infrastructure elements. They are determined in part, by the specific type of service and in part, by the application context and can thus be explained in the correlation between activities and services. The SOA's optionality is registered through the fact that the services available for each activity are represented as *alternatives*. Potential models can therefore, be characterized as build-time models (Remme 1995). To make their



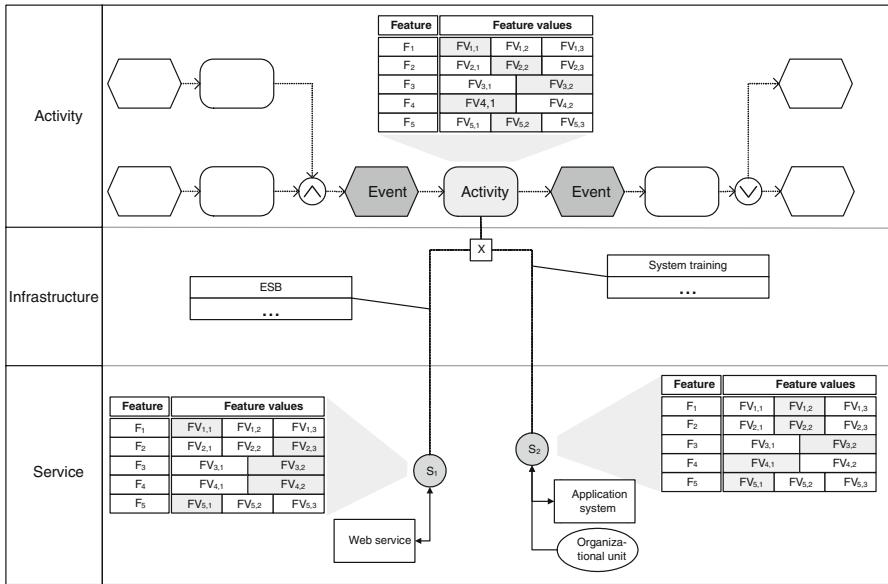


Fig. 2 Principle for the representation of design options for SOA in process models

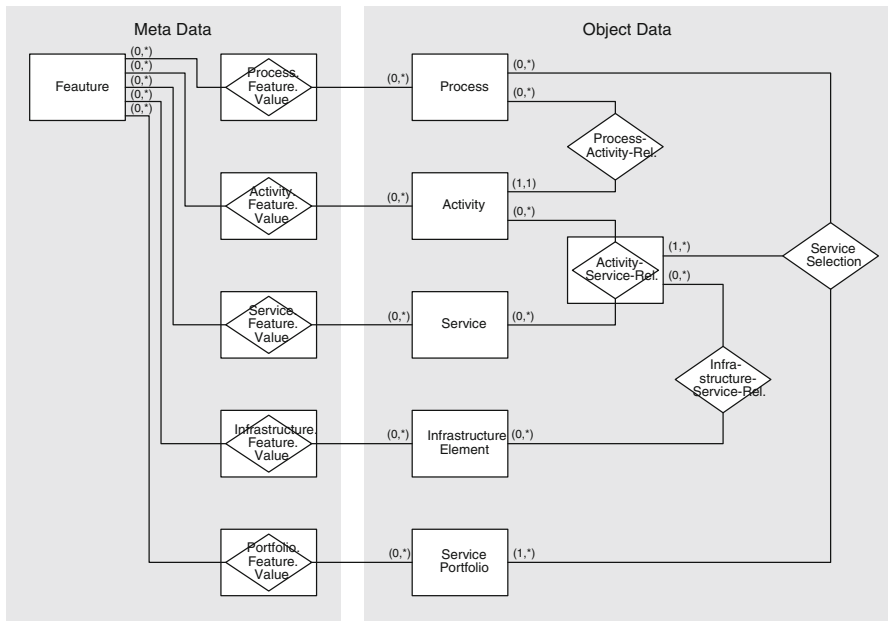


Fig. 3 Meta model for the representation of design options for SOA in process models

execution possible, a choice of individual services for all functions must be made. The relationships are illustrated in the meta model (cp. Fig. 3).

The excerpt of the model shown here specifies which constructs are relevant for the representation of potential models when using an SOA. It shows that alternative services are used for the execution of an activity (Activity-Service-Rel.). In doing so, individual services can make specific demands with regard to the integration infrastructure (Infrastructure-Service-Rel.). The configuration of a service portfolio is thus characterized by the fact that a specific service is selected for each activity in a process (Service Selection). An aggregation of all of the activities assigned to a process must be carried out here, in the course of which the infrastructural requirements must be consolidated.

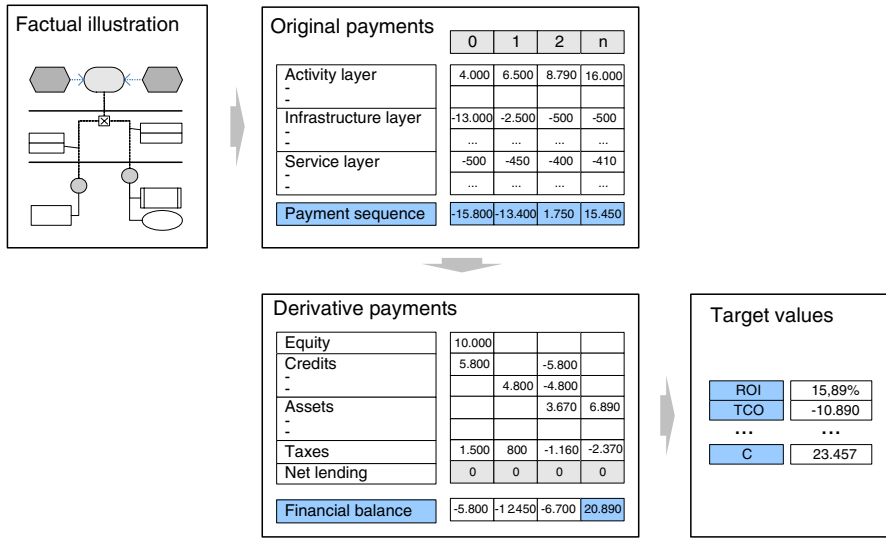
Feature-driven description methods can be used for decision support with regard to the selection of services. Feature-driven techniques allow the construction of classification systems (Buchanan 1979) already used in information systems research (e.g. Glezer and Yadav 2001). Feature catalogs can be used here to specify the requirements of the execution of activities (to-be quality), as well as the performance from individual services (as-is quality). Thus, principles from web service selection (Padovitz et al. 2003; Cardoso et al. 2004; Day and Deters 2004) can be transferred to the modeling by way of a comparison of features. Added to this, are features for the characterization of individual infrastructure elements, which also serve the identification of specific offers for the requirements defined from the service view.

Further types of features focus on the management of the service portfolio configuration. Using features on the process level, relationships between feature characteristics can be modeled across several activities. This makes it possible to describe situations where typical specialized requirements exist for the execution of the activities. Thus, for example, it could be necessary to select different service levels due to different seasons, which are then adjusted over all of the activities by marking the seasonal characteristic. Similarly, knowledge about the context factors of the configuration can be saved based on features regarding the service portfolio, for example data on the term of planning.

### 4.3 Value-based representation of the potentials of SOA use

Methods from process controlling can be used to evaluate the processes (Leymann and Altenhuber 1994; zur Muehlen and Rosemann 2000; Kang and Kang 2005). In particular, works that allow model-based evaluation, as available for activity-based costing (Cooper and Kaplan 1991), are interesting for the construction of potential models. Because however, arrangements for the design of processes must be made in potential models, the consideration of the long-term economic consequences of the individual design decision becomes necessary. Thus, the introduction of SOA is typically characterized by the decision situation of an *investment*, whose anticipated benefit is seen in a potentially higher flexibility in process design.

Works where an investment appraisal was conceived on the basis of process models (Grob and vom Brocke 2006) can be used to evaluate value. These works allow one to determine the *economic value* connected with the implementation of a



**Fig. 4** Principle for the evaluation of process alternatives based on controlling methods

process alternative in a concrete decision situation based on factual descriptions. This principle is illustrated in Fig. 4.

First, the original payments derived from the depositions documented in the process model must be recorded. On the service and infrastructure levels payments arise, which must be calculated differently depending on the specific decision situation. Generally, one differentiates between a market and a resource-oriented calculation:

- *Market-oriented calculation:* Services and infrastructure elements are calculated according to the conditions customary on the market. Different models can be used (Vandermerwe and Rada 2007). In the case of flat pricing models, performance-neutral (or fixed-step developing) payments are available. Payments must be characterized as performance-driven for transaction-based price models. The performance can be measured in different reference units such as, for example, the process frequency and duration.
- *Resource-oriented calculation:* The valuation is calculated on the basis of the resources needed for the value performance. Different models can be used here for pricing. Thus, for example, the cost rate for a demand-oriented calculation is dependent of the utilization of individual resources, while constant cost rates are used for a supply-oriented calculation.

In addition to the pay-outs, the pay-ins connected with a process alternative must also be calculated. These were recorded in our approach on the activity level. The representation of such payments can take place in general—and thus performance-neutral—or performance-distinguished. Performance-neutral valuations are recorded by period-individual payments, which must be calculated in relation to the total process. A differentiated evaluation can be carried out with prices that can be

calculated depending on individual institutionalizations of activities. They function as agio respectively disagio and are calculated in a transaction-based manner:

- *Process-wide calculation:* Payments that arise when a process is carried out once can be registered by the usage of prices based on the whole process. This facilitates the planning of pay-ins that can be generated by the use of process performance measures.
- *Activity-specific calculation:* differences in quality (for example: service level) can be evaluated on the monetary level with prices for special service alternatives. To do so, agio resp. disagio must be calculated in dependency of the feature characteristics formulated on the activity level.

The basic constructs for recording original payments are illustrated in the meta model in Fig. 5. Payments on the activity, infrastructure and service levels must be

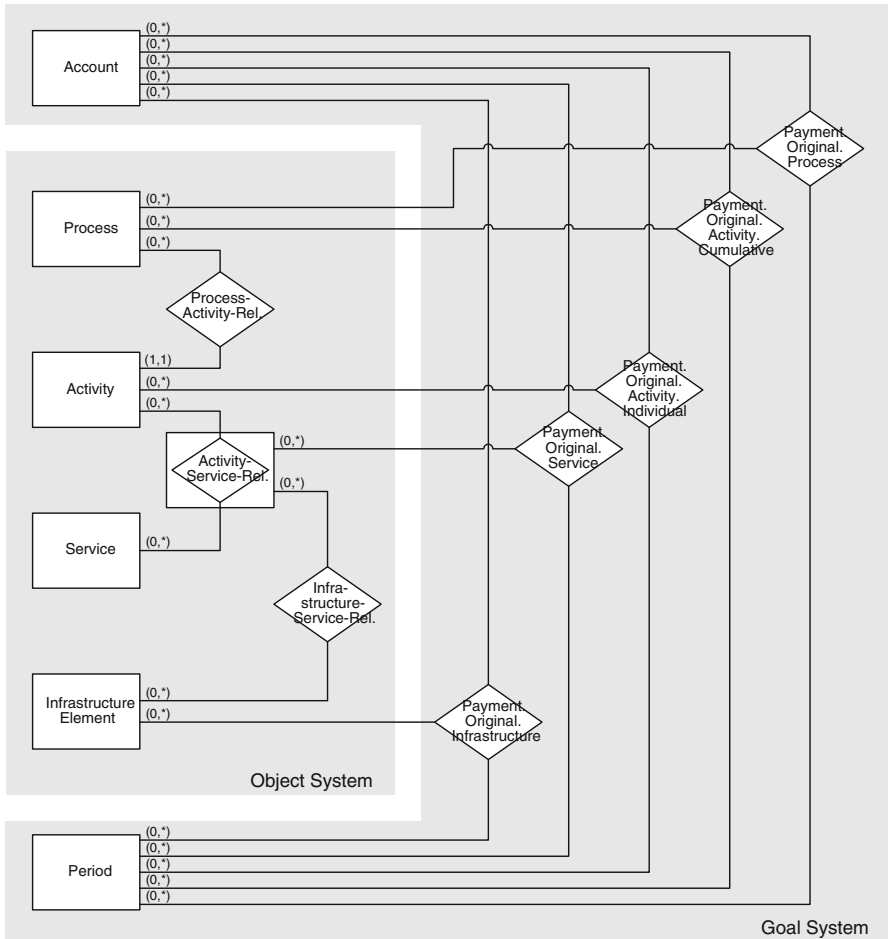


Fig. 5 Meta model for the evaluation of design options for SOA in process models

summed up for the individual configurations of the service portfolio. To do so, locally recorded values must be aggregated in accordance with the specific process structure. Methods from process analysis and simulation, which allow the consideration of probabilities in process chains can be used here (Giaglis and Paul 1996). The result is a sequence of payments that represents all of the monetary consequences of the design alternatives in a process.

Further economic consequences result from the original payments. These can be referred to as derivative payments. In order to analyze these payments, the financial situation of the use case enterprise must be considered. Established methods from investment controlling can be used here through the determined sequence of payments per process alternative. Decisive for the selection of methods is how differentiated the finance situation can be mapped. While for example, classic, dynamic methods from investment appraisal allow for a consistent, adequate target rate for debit and credit interest, finance plan-oriented methods allow the explicit modeling of various finance alternatives and tax conditions (Grob 1993, pp. 121 f).

The transparency of the original and derivative payments connected with a process alternative allow the calculation of a wide spectrum of key figures on the value of a process alternative (vom Brocke 2007). Among these figures are for example, the total cost of ownership (TCO), which traditionally serves the life cycle-oriented recording of application system costs (Ellram and Siferd 1993). The TCO of a process alternative is particularly significant when no pay-ins must be calculated in the use case. Alternatively, it is also possible to show value, such as for example the ROI. Based on these key figures, different configurations of the service portfolio can be compared with each other. Optimization, as well as experimental procedures can be used.

All in all, the method described above supports decision makers by *first*, making the options for process design given by the SOA transparent and *second*, using these options for the purpose of corporate goals. The result of the configuration is a to-be model, which provides for a clear assignment of services to activities in a specific process structure. This model is the guideline for the implementation of the service portfolio.

## 5 Implementing the service portfolio

### 5.1 Preliminary considerations

The methods in the previous section support the configuration of the service portfolio during the model-based design of an SOA. The result is a model that represents the information system to be implemented on a conceptual level. This representation must be adjusted to the corresponding application system for the purpose of implementing the service portfolio—in accordance with the research design selected in this paper in order to then, be transformed into an executable process model. Thus, the application system design constitutes the link between information system design and the process execution.

In the transitions from information to application model and from application to execution model, a reuse of model elements occurs. In order for this reuse to be successful, the corresponding models must be constructed so that they first, abstract from the individual content of the systems represented and then serve as a guideline for the model construction. An example for this is the concentration of organizational relationships (for example: responsibilities) in the information model, as well as the conscious abandonment of these simultaneously combined with the enhancement of technical details in the transition to the application model (for example: detailed specification of the system behavior, consideration of transactions and system errors). Altogether, the implementation of service portfolios is characterized by creativity and exhibits a high level of complexity. To make this complexity controllable, previous works from the field of reference modeling (Thomas 2005; Fettke and Loos 2007; Becker and Delfmann 2007) on procedure models and construction techniques can be consulted, due to the established correlation of model reusability. Furthermore, modeling languages and tools must be selected for the implementation of the service portfolios.

## 5.2 Modeling languages

For the implementation of the service portfolio, the process logic represented by an information model can be taken as a starting point for the derivation of an application model. While the basic structure defined for the process flow is enriched for example, by the annotation of organizationally relevant constructs in the information model, an extension of the application model occurs through technical details for process execution. These pertain in particular, to the detailed specification of the behavior of a system carrying out the process in error-situations, as well as the consideration of transactions. Thus, the modeling language BPMN can be used for the representation of application models. It was significantly developed by IBM and accepted as a specification in February of 2006 by the Object Management Group (OMG 2006a). It was designed to be understandable by all business users and to be transferred into technical process description languages for process execution. The language constructs made available by BPMN thus allow the construction of models, which contain the process logic of the information model level, as well as consider the technical details of the execution model level via attributes. The representation of responsibilities or resources and thus, the representation of the organizational components of an information system are not in the interest of BPMN. Alternatives to the representation of the application system during the implementation of the service portfolio are: UML activity diagrams (OMG 2005), UML EDOC Business Processes (OMG 2006b), IDEF (Knowledge Based Systems Inc. 1993), ebXML BPSS (OASIS 2001), activity-decision flow (ADF) diagram (Mentzas et al. 2001), line of visibility enterprise modeling (LOVEM) (IBM 2007).

Overall, BPMN helps to specify the behavior of an information system more precisely as EPC. This can be seen for example, on the use of a transaction concept allowing for time-based events (OMG 2006a). The creation of this sort of models requires an IT-close way of thinking that puts the focus on the possibilities and limitations of process support through application systems and thus, differs from organization design.

The design of an execution model should build upon the graphic representation of the control flow and the textual attribution of the process elements in the application model. While the configuration of the service portfolio and the transition between information and application model were affected primarily by organizational aspects, the encoding of the requirements of the graphic application system model into an execution model lies in the sphere of IT development and represents the most technical task. In doing so, information gaps rooted in the semi-formal definition of the modeling language used on the application level and hindering the process execution (Recker and Mendling 2006) must be filled in (for example: specifying partner links or expressing data manipulation). The BPMN specification has mapping rules for this purpose, which allow the conversion process, i.e. the transfer of existing BPMN models in code-based, executable BPEL models.

BPEL is an XML-based language, whose specification was initiated by IBM, BEA and Microsoft and is currently being developed by the OASIS standardization initiative in the Version 2.0 (Alves et al. 2006). Due to the high acceptance of the language in the software industry, BPEL can be referred to as the de-facto-standard for business process implementation based on web services. A BPEL process is generally understood as a set of service calls that are subject to a logical and chronological order. This is also referred to as the “orchestration” or “composition” of services (Alonso et al. 2004). Thus, a BPEL process model can be seen as the IT counterpart to a business process model. BPEL builds upon the Web Service Description Language (WSDL) by combining the web services described in WSDL to form a process (Alves et al. 2006).

BPEL processes are executed on so-called BPEL engines, which in turn can be integrated into application servers. For the graphic illustration and modification of BPEL processes some BPEL engines make separate modeling tools available, which use a proprietary notation (e.g. Klückmann 2007) despite the contextual proximity of the used visualizations to BPMN.

Alternatives to BPEL are XL, a XML programming language for the specification and composition of Web Services (Florescu, Grünhagen & Kossmann 2003), the XML Process Definition Language (XPDL), an XML-based standard of the Workflow Management Coalition (WfMC) for the description of workflows (Workflow Management Coalition 2007), Wf-XML, a standard for the interoperability of process instances (Swenson et al. 2004), and the Business Process Modeling Language (BPML), whose further development has however, been given up by the Business Process Management Initiative (BPMI). The XPDL version 2.0, available since May of 2005, especially supports BPMN.

### 5.3 Modeling tools

Tool support can considerably increase the efficiency of the service portfolio implementation procedure. This refers to the “manual” process that must be carried out by the model user in the transition from an information to an application model, as well as to the (semi-) automatic adaptation process in the transition from an application to an execution model. Thus, in addition to selecting modeling languages, one must also decide upon a preferred modeling tool. The “Magic

Quadrant for Business Process Analysis” from Gartner (Blechar and Sinur 2006) can be drawn upon to clarify the wide range of professional modeling tools available on the market. It is however—despite its annual updating—hardly suitable for decision support in selecting one (or more) of these tools, unless the development potential of the tool provider is—as is common in business practice—a decisive selection criterion. Market surveys and studies (e.g. Berlecon 2005) extend the model developer’s horizon on modeling tools, for example by comparing the tool’s range of functionalities or identifying application possibilities in selected use cases. However, these “snapshots” often exhibit weaknesses with regard to content. Therefore, the conceptual framework for the evaluation of modeling tools for business process management from Nüttgens (2002), which provides explicit modeling-specific selection criteria, is recommended as an auxiliary means for the selection of tools. The conceptual framework is divided up into five main categories “Product and Pricing Model”, “Producer and Customer Base”, “Technology and Interfaces”, “Methodology and Modeling”, as well as “Application and Integration”. These central attributes are operationalized in a multi-level manner via sub-features and comprise a total of 350 individual features.

Due to the lack of integration of the various modeling and development tools, the implementation of the service portfolio based on the information, application and execution model, trouble-free and free of breakage, was not possible up to now. It is to be expected however, due to today’s convergence of different product categories, that this situation will strongly improve in the future. Thus for example, *Forrester* predicts that tools that concentrate primarily on human interaction (“human-centric BPM suites”) will overlap more and more with tools aimed at automated business processes (“integration-centric BPM suites”) (Vollmer and Moore 2006) so that companies can cover the expert analysis and documentation of processes, as well as their technical integration with integrated development environments. *Gartner* differentiates between the tools in “Business Process Analysis” and “Integrated Service Environments” and thus, differentiates between tools in expert-analytic and integration-oriented categories and predicts their merge to “Business Process Platforms” (Graham et al. 2004).

All it all, the problem of creating continuity in concepts from the business level to IT-supported process execution by way of tool providers has already been recognized and will be considered in the next generation of products. This will allow the comprehensive IT-support of the methodology suggested here and simplify the implementation of a service portfolio.

## 6 The usage of the methodology for the model-based design of service-oriented systems

### 6.1 An introduction into the case study

The methodology developed here for the model-based design of service-oriented systems has been tested in a case study. Our partner was the small business *Matchball Sportartikel Vertriebs-GmbH* (short: *Matchball*), Germany, which is



specialized in selling ball game items on the Internet. To support their business processes, the company uses an ERP system, a shop system based on a database with shopping basket functionality and in their call center, a CRM system. *Matchball* provides its customers with the possibility of paying per credit card or direct debit. The results from market studies that showed that 78.6% of the customers in internet shops prefer payment by invoice triggered the business process reengineering examined in the case study. Credit transfer is the second most used payment method for shopping on the internet with 60 % before credit cards with 59.5% (Postbank 2004; van Baal et al. 2005). However, a higher payment risk is to be expected with the introduction of payment by invoice. To reduce this risk, payment by invoice at *Matchball* was to be connected with a customer creditworthiness check: only customers who were creditworthy (resp. whose rating exceeded a minimum value) were to be allowed to pay by invoice. Different design options for the creditworthiness check were examined and in association with this, an SOA was introduced. Three alternatives were worked out:

1. *Crefosprint online*: This is an IT solution from *Creditreform*, which allows the user to call up information from an economic database with approx. 3.6 million data sets via an online-interface and can be integrated into the *Matchball* ERP system. The customer's creditworthiness data is transferred in an encoded, structured form and linked with the customer master data in the ERP system (Creditreform 2007). The creditworthiness check's corporate task is completely out-tasked to *Creditreform* with this solution.
2. *Creditworthiness index*: Within this individual solution, existing internal evaluation rules and customer data should be enriched with external information. A probability index from the SCHUFA (2007) provides this external information on possible payment losses of customers. The contractual partners receive a customer-specific score value with a rating level. The *creditworthiness index* is provided over a proprietary interface technology from the SCHUFA. The SCHUFA Computer Direct Information (SCDI) facilitates communication between the customer's application and the SCHUFA for order processing via a dial-in.
3. *MediaFinanz web service*: With this web service specialized in internet commerce from *MediaFinanz GmbH & Co. KG* creditworthiness data about people living in Germany can also be obtained. The information is queried directly from the contractual partner's system via an SOAP interface. It is however, possible to integrate the web service into *Matchball's* ERP system with little effort (MediaFinanz GmbH & Co.KG 2007).

The subject of the case study was the analysis of the value of each alternative. The results we received using our methodology will be discussed in detail in the following.

## 6.2 The configuration of the service portfolio

As the starting point for our analysis, the design alternatives at *Matchball* will be described in potential models (cp. Fig. 6).

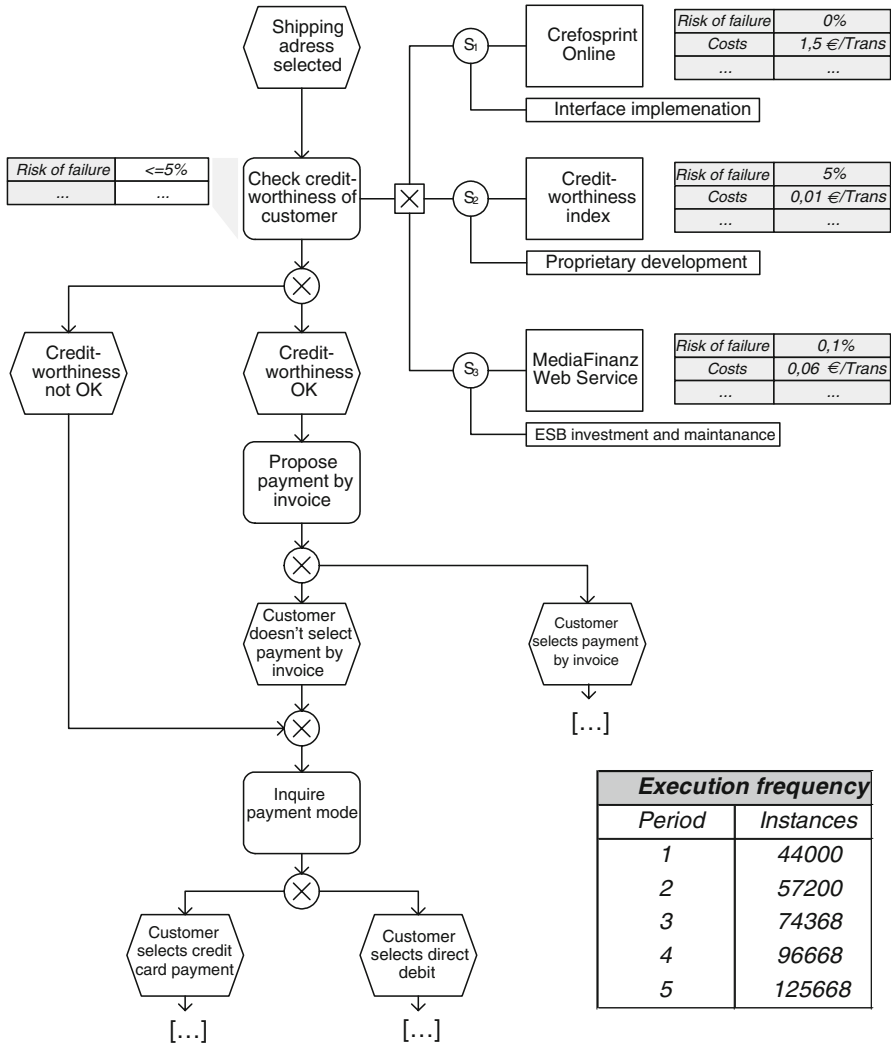


Fig. 6 Potential model for an order transaction at Matchball

The model shows that alternative services are available for the implementation of the activity “Check creditworthiness of customer”. The introduction of each individual service however, has different infrastructural requirements. It becomes clear that the decision is relevant *locally* in the process model, because it does not show structural or institutional effects on other parts of the process. However, different economic consequences do go hand-in-hand with the alternatives and these can often only be discovered in a value-based examination.

To register the payments, an analysis based on a partial model (vom Brocke 2007) was carried out in the case study. It lend itself due to the locally, relatively limited effect of the design decision. Thus, we examined which *additional*

payments—in comparison to continuing with the status quo—were connected with the selection of each alternative. In accordance with the methods introduced, the payments were calculated differently according to the activity, infrastructure and service level. The results are summarized in Table 1 and, in the case of the *MediaFinanz Web Service*, broken down in detail.

The calculation allowed us to map and discuss the various economic consequences in detail. The resulting planned values form the basis for the following thoughts:

- *Activity level: Matchball* linked the introduction of a method of payment by invoice with the expectation of minimizing the rate of abort events in the order process. Log file-analyses have shown that the rate of abort events was at an average of 95% up to now. Due to path analyses, it is assumed that this rate could be improved by six percent to 89% with the introduction of the new payment method. Market basket analyses show that an average turnover of 40 €

**Table 1** Calculation of the payments for the usage of an SOA at *Matchball*

Original payments	0	1	2	3	4	5
<b>MediaFinanz Web Service</b>	<b>-25.000 €</b>	<b>949.984 €</b>	<b>1.236.379 €</b>	<b>1.609.493 €</b>	<b>2.095.190 €</b>	<b>2.726.383 €</b>
<b>Activity layer</b>	<b>0 €</b>	<b>959.040 €</b>	<b>1.246.752 €</b>	<b>1.620.778 €</b>	<b>2.107.010 €</b>	<b>2.739.099 €</b>
<b>Payments-in</b>	<b>0 €</b>	<b>959.040 €</b>	<b>1.246.752 €</b>	<b>1.620.778 €</b>	<b>2.107.010 €</b>	<b>2.739.099 €</b>
(1) Interruption frequency without CC		95%	95%	95%	95%	-95%
(2) Interruption frequency with CC		89%	89%	89%	89%	89%
(3) Risk of failure of the web service		0,100%	0,100%	0,100%	0,100%	0,100%
(4) Process frequency		400.000	520.000	676.000	878.000	1.142.440
Additional ordering frequency		23.976	31.169	40.519	52.675	68.477
Average volume per order		40	40	40	40	40
Volume of sales effect		959.040 €	1.246.752 €	1.620.778 €	2.107.010 €	2.739.099 €
<b>Infrastructure layer</b>	<b>-25.000 €</b>	<b>-8.000 €</b>	<b>-9.000 €</b>	<b>-9.500 €</b>	<b>-9.500 €</b>	<b>-9.700 €</b>
<b>Pattern (payments for ESB)</b>	<b>-25.000 €</b>	<b>-8.000 €</b>	<b>-9.000 €</b>	<b>-9.500 €</b>	<b>-9.500 €</b>	<b>-9.700 €</b>
Payment for aquisition	-25.000 €					
- Maintenance		5.000 €	5.000 €	5.000 €	5.000 €	5.000 €
- Adaptions		3.000 €	4.000 €	4.500 €	4.500 €	4.700 €
Amounts of depreciation (not direct affecting payment)		5.000 €	5.000 €	5.000 €	5.000 €	5.000 €
<b>Service layer</b>	<b>0 €</b>	<b>-1.056 €</b>	<b>-1.373 €</b>	<b>-1.785 €</b>	<b>-2.320 €</b>	<b>-3.016 €</b>
<b>Service payments</b>	<b>0 €</b>	<b>-1.056 €</b>	<b>-1.373 €</b>	<b>-1.785 €</b>	<b>-2.320 €</b>	<b>-3.016 €</b>
<b>Service cost rate (price per transaction)</b>		<b>0,06 €</b>	<b>0,06 €</b>	<b>0,06 €</b>	<b>0,06 €</b>	<b>0,06 €</b>
Ordering frequency, absolute		44.000	57.200	74.360	96.668	125.668
* Frequ. of activity checking of liquidity, rel.		40%	40%	40%	40%	40%
= Frequency of activity checking of liquidity		17.600	22.880	29.744	38.667	50.267
* Service cost rate		0,06 €	0,06 €	0,06 €	0,06 €	0,06 €
= <b>Service payments</b>		<b>-1.056 €</b>	<b>-1.373 €</b>	<b>-1.785 €</b>	<b>-2.320 €</b>	<b>-3.016 €</b>
<b>Crefosprint Online</b>	<b>-1.000 €</b>	<b>932.100 €</b>	<b>1.212.980 €</b>	<b>1.577.084 €</b>	<b>2.050.519 €</b>	<b>2.665.939 €</b>
<b>Activity layer</b>	<b>0 €</b>	<b>960.000 €</b>	<b>1.248.000 €</b>	<b>1.622.400 €</b>	<b>2.109.120 €</b>	<b>2.741.840 €</b>
<b>Infrastructure layer</b>	<b>-1.000 €</b>	<b>-1.500 €</b>	<b>-700 €</b>	<b>-700 €</b>	<b>-600 €</b>	<b>-500 €</b>
<b>Service layer</b>	<b>0 €</b>	<b>-26.400 €</b>	<b>-34.320 €</b>	<b>-44.616 €</b>	<b>-58.001 €</b>	<b>-75.401 €</b>
<b>Creditworthiness index</b>	<b>-32.500 €</b>	<b>902.495 €</b>	<b>1.175.393 €</b>	<b>1.530.971 €</b>	<b>1.993.653 €</b>	<b>2.594.133 €</b>
<b>Activity layer</b>	<b>0 €</b>	<b>912.000 €</b>	<b>1.185.600 €</b>	<b>1.541.280 €</b>	<b>2.003.664 €</b>	<b>2.604.748 €</b>
<b>Infrastructure layer</b>	<b>-32.500 €</b>	<b>-9.500 €</b>	<b>-10.200 €</b>	<b>-10.300 €</b>	<b>-10.000 €</b>	<b>-10.600 €</b>
<b>Service layer</b>	<b>0 €</b>	<b>-5,09 €</b>	<b>-6,62 €</b>	<b>-8,61 €</b>	<b>-11,19 €</b>	<b>-14,54 €</b>

per transaction should be fixed. A market forecast furnishes the estimated process frequencies with which additional turnover can be determined. This amounts to 960,000 € in the first period. In the case of the out-tasking of creditworthiness checks to *Crefosprint Online*, this potential would be guaranteed by the service contract, payment risks however must be taken into account for the other alternatives. While the risk in the case of *MediaFinanz Web Service* only amounts to 0.1% in accordance with the service level agreements, one must assume a risk of 5% in the case of the development of the *creditworthiness index*. Correspondingly, the additional payment would be lower.

- *Infrastructure level:* The use of the *MediaFinanz Web Service* requires the introduction of an enterprise service bus (ESB). In the use case it was planned to follow an incremental implementation strategy (Hagel 2002), so that payments to the amount of 25,000 € and the periodically new payments for maintenance and adaptation could be assigned completely to the alternative. Such payments would be lower in the case of out-tasking the creditworthiness checks to *Crefosprint Online*. Here a monetary equivalent for the personal expense was calculated that would result in the implementation of an online-interface between the economic database and the ERP system through the in-house IT consultants at *Matchball*. Similarly, payments for the alternative to the *creditworthiness index* were calculated. These were higher due to the extent of the task.
- *Service level:* A transaction-oriented price model with a cost rate of 6 Cent per transaction was fixed for the *MediaFinanz Web Service*. The resulting payments per period take the absolute frequency of the order transaction from the sales planning into account, so that in the first period an activity frequency of 17,600 and payments amounting to 1,056 € result. In the case of the out-tasking to *Crefosprint Online*, a cost rate of 1.5 € per transaction can be fixed at the same quantity structure, so that correspondingly higher payments result. The payments fixed for the *creditworthiness index* are the basis for a calculation where the price models from the *SCHUFA*, as well as the internal cost rates for resources were considered.

The payments recorded were successively aggregated for the purposes of decision support. A succession of payments for each individual alternative was determined by using the methods introduced. Based on this, the derivative payments were determined in further analyses, in order to determine the *economic value* generated in the concrete decision situation at *Matchball* per alternative. The results are documented in the finance plan in Table 2 for the *MediaFinanz Web Service* alternative.

In the financial plan created for the calculation, the original payments are first taken over as a succession of payments. In order to determine the capital flow connected with this, concrete positions for different types of investments and credit from *Matchball* were listed. It was assumed that their own liquid assets amounted to 10,000 €. In addition, a loan with amortization for 2 years at 6% and a disagio of 5%, as well as an advance on a current account at 8% interest on debit were an obvious choice. An interest rate at an average of 6% could be assumed for

**Table 2** Calculation of the payments for an SOA usage at *Matchball*

Derivative payments	0	1	2	3	4	5
<b>Payment sequence</b>	<b>- 25.000 €</b>	<b>949.984 €</b>	<b>1.236.379 €</b>	<b>1.609.493 €</b>	<b>2.095.190 €</b>	<b>2.726.383 €</b>
Equity capital						
+ Opening balance	10.000 €	0 €	0 €	0 €	0 €	0 €
- Withdrawal	0 €	0 €	0 €	0 €	0 €	0 €
+ Deposit	0 €	0 €	0 €	0 €	0 €	0 €
Credit with bullet repayment (debit interest rate 6%)						
+ Borrowing	10.000 €	0 €	10.000 €	0 €	0 €	0 €
- Paying back	0 €	0 €	0 €	0 €	0 €	0 €
- Debit interests (incl. 5% disagio)	500 €	600 €	600 €	0 €	0 €	0 €
Open account credit (debit interest rate 8%)						
+ Borrowing	5.500 €	0 €	0 €	0 €	0 €	0 €
- Paying back	0 €	5.500 €	0 €	0 €	0 €	0 €
- Debit interests	0 €	440 €	0 €	0 €	0 €	0 €
Standard investment (credit interest rate 6%)						
+ Investment	0 €	424.412 €	591.847 €	837.734 €	1.160.751 €	1.601.247 €
- Dissolution	0 €	0 €	0 €	0 €	0 €	0 €
+ Interests	0 €	0 €	25.465 €	60.976 €	111.240 €	180.885 €
Tax payments		55,0%	52,5%	50,0%	47,5%	45,5%
- Payments-out		519.032 €	659.397 €	832.734 €	1.045.679 €	1.306.020 €
+ Payments-in		0 €	0 €	0 €	0 €	0 €
<b>Financial balance</b>	<b>0 €</b>	<b>0 €</b>	<b>0 €</b>	<b>0 €</b>	<b>0 €</b>	<b>0 €</b>
Stock figures						
Loan portfolio	15.500 €	10.000 €	0 €	0 €	0 €	0 €
Funds	0 €	424.412 €	1.016.259 €	1.853.994 €	3.014.744 €	4.615.991 €
<b>Stock balance</b>	<b>- 15.500 €</b>	<b>414.412 €</b>	<b>1.016.259 €</b>	<b>1.853.994 €</b>	<b>3.014.744 €</b>	<b>4.615.991 €</b>

investments. The specific fiscal effects of the individual alternatives were also considered, whereby a sinking average tax rate was assumed. The relevancy of the fiscal analysis can be seen in the calculations, for example in the ability to capitalization of the ESB solution, which means that a decision for the *MediaFinanz Web Service* would result in *fiscal* advantages caused by the depreciation and amortization of fixed assets, which then lead to a lower tax base.

The accumulated values of all design alternatives were calculated through the extrapolation of loans and assets. In the planning at *Matchball* documented here, an accumulated value for the investment in process reorganization using *MediaFinanz Web Service* amounted to 4,615,991 €, an accumulated value of 4,535,077 € resulted with *Crefosprint Online* and an accumulated value of 4,443,736 € resulted with the *creditworthiness index*. Compared with the accumulated value of the opportunity—a 6 % interest rate of the owner’s equity, which results in a value of 13,382 €—it was shown that the introduction of payment by invoice would be advantageous for *Matchball*. Due to the rate of the accumulated values, several variations of the planned values were carried out, in order to investigate the sensitivity of the results. Thus for example, it was shown that (on the example of the *MediaFinanz Web Service*) a critical reduction of the frequency of abort events of 4.23% could be achieved: if this percent mark is not exceeded, then the investment in the reorganization proves to be disadvantageous.

The order of the alternatives shows that at *Matchball* the highest additional accumulated value could be generated through a decision for the web service alternative. The added value amounts to 80,914 € in contrast to *Crefosprint Online* and 172,255 € in contrast to the *creditworthiness index*. Therefore, the introduction

of payment by invoice using the *MediaFinanz Web Service* was set for the implementation.

### 6.3 The implementation of the service portfolio

During the implementation of the service portfolio, the to-be information model for order transactions at *Matchball* must be transferred into a corresponding application model as a first step. The modeling language BPMN will be used here to represent the application model. The starting point for the adaptation process is the process logic represented in the EPC potential model (cp. Fig. 6). It will be complemented with regard to the behavior of the service-oriented IT system supporting the order process. Figure 7 shows the BPMN model resulting from the potential model. As in the EPC model, we have gone without the representation of returns by the customer, which serve informative purposes or corrective measures, in favor of a more clearly arranged representation. For the same reasons, the process is represented after the selection of a delivery address by the customer.

After the customer has selected his or her preferred address, the customer's creditworthiness is checked. This check takes place via data exchange with *MediaFinanz*, the service provider for the creditworthiness check, which is represented in a separate lane as an external organizational unit. Because errors in the creditworthiness check or the communication with *MediaFinanz* cannot be ruled out, error correction is defined with the help of an intermediate error event. The error event is shown graphically on the lower edge of the respective activity. The sequence flow coming from this leads directly to the activity "Inquire payment method", which guarantees the operation of the shop system even if the creditworthiness check fails.

The result of the creditworthiness check affects the choice of payment methods offered to the customer. The corresponding decision is represented as a data-based XOR gate in the form of a rhombus in the BPMN application model. A positive result assumes the existence of explicit information about the customer's sufficient creditworthiness. In all other cases, i.e. should the check come up negative or creditworthiness information not be available for the customer, the result of the check is negative (default flow).

If the customer is creditworthy, then he is asked if he wishes to pay by invoice. If the customer confirms, then the activity "Prepare invoice" is triggered immediately. This may comprise several sub-steps, such as the calculation the gross/net amounts of the invoice and the display of the value-added tax. Because these steps are not shown in the graphic model, the relevant activity is marked with a "+" (collapsed sub-process). Should the customer decide against payment by invoice, he is shown a selection of other methods of payment, just as if the creditworthiness check had turned out negative.

A selection between direct debit and credit card is only possible in this step of the process. The shop system however, recommends direct debit as a standard. In both cases, the payment transaction is prepared—similar to payment by invoice, i.e. all of the information required for processing the later payment is collected. The use of compensation activities ensures that, in the case of an aborted or failed order

transaction, these payment preparations are recalled. Compensation activities can be recognized by an arrow pointing backwards (compensation marker) in the lower part of the activity rectangle. They are connected with the respective activity they compensate via a dotted line (association). These in turn, are marked by a compensation event, whose graphic appearance is similar to a “rewind”-symbol.

When the procedures connected with the selection of the payment method are concluded, the order is completely defined and only requires the legally binding confirmation by the customer. If this occurs, then the transactional sub-process “Order transaction” is successfully concluded and the corresponding transaction, represented by a frame with a double line (transaction sub-process), can be left successfully. What follows is the order processing activity not discussed here, which concludes the process as a whole. If the customer does not confirm the order, then the order transaction is cancelled and the sequence flow leads to an abort event. This in turn, leads to the process being stopped and triggers the abort event of the transaction. This is represented at the lower edge of the transaction frame and can also be triggered by external events such as a defined period being exceeded since the last interaction of the customer with the online shop system.

The abort event of the transaction now causes the transaction to roll back by activating all of the compensation events within the transaction and thus, triggering the execution of the compensation activities connected with them. Lastly, the abort event of the transaction triggers the activity “Delete order”, which removes the temporary shopping basket from the system.

Overall, the example once again illustrates the added value of the BPMN application model in contrast to the EPC information model. After the decision for support through a web service on the organizational level was made, the representation concentrates on the precise illustration of the behavior of the information system in the application level.

The second step in the implementation of the service portfolio is to transfer the BPMN application model of the order transaction at *Matchball* into a corresponding execution model. For this adaptation, further information in the form of attributes must be added to the graphic model, as for example, partner links or variables for XML data structures. The BPMN specification stipulates mapping rules defining how such attributed BPMN models should be transformed into code-based, executable BPEL models. BPMN is often used as a graphic notation for the visualization of processes in web services development environments (a current list of providers can be found at [http://www.bpmn.org/BPMN\\_Supporters.htm](http://www.bpmn.org/BPMN_Supporters.htm)). This functionality has however, not yet been realized in many of the BPMN tools currently on the market, despite a standardized transformation of the BPMN models into executable BPEL process descriptions by the OMG. Figure 1. Framework of the model-based design of service-oriented systems Fig. 8 shows the BPEL code resulting from the BPMN model in Fig. 7. Omissions are marked by “...”.

The entire process is first surrounded by a scope-element (lines 4 and 58), which defines the transaction context. An exception is defined in the lines 5–9 within the faultHandlers-element. The transactional order process is located within a sequence-element (lines 10 and 57). As can be seen in the basic structure of the BPEL model, the four data-based gates of the BPMN model are transformed to four

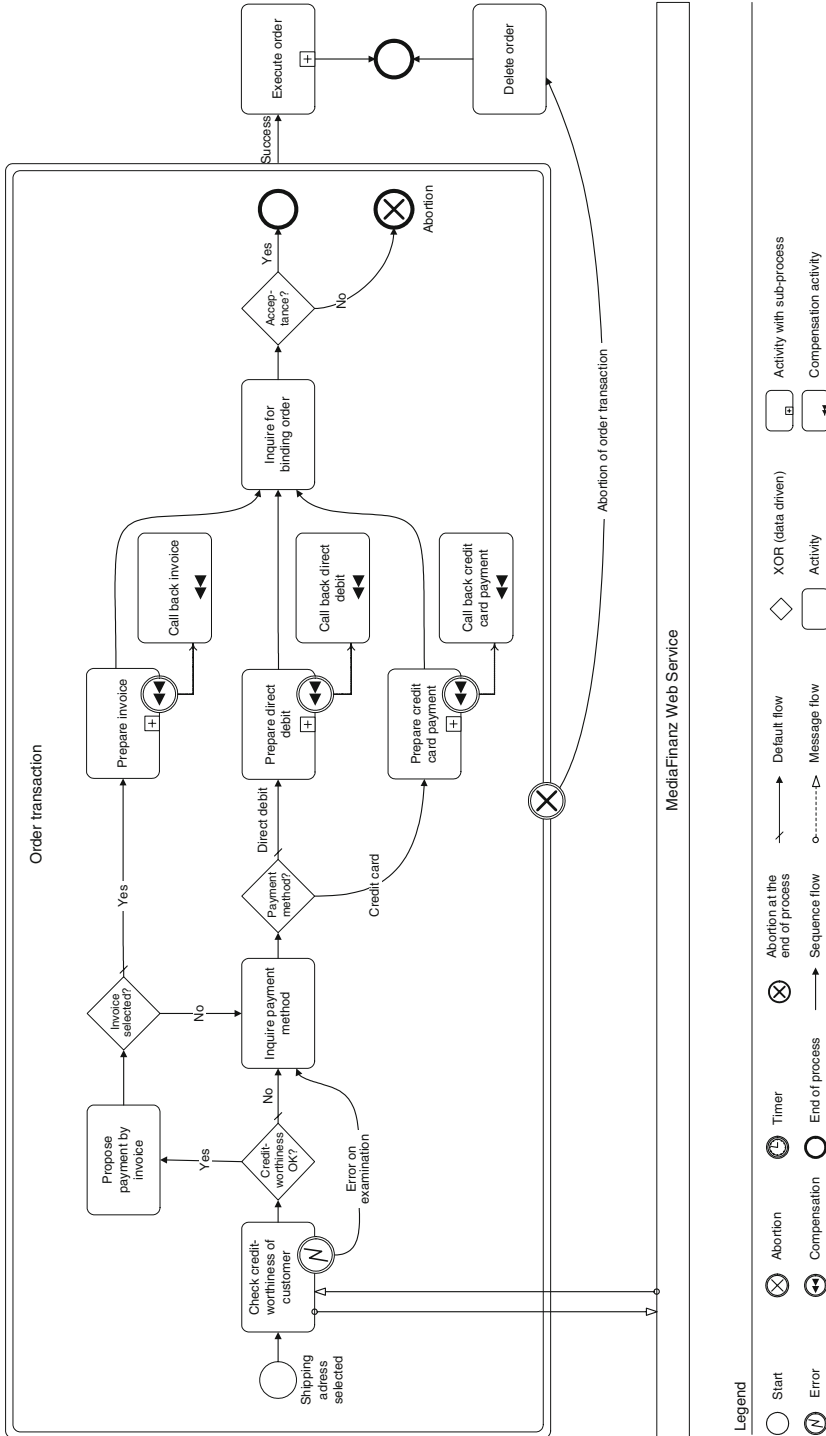


Fig. 7 Application model for an order transaction at Matchball



```

1 <process ... >
2 <partnerLinks> ... </partnerLinks>
3 <variables> ... </variables>
4 <scope>
5   <faultHandlers>
6     <catch faultName="cancel">
7       <invoke operation="DeleteSalesOrder" ... />
8     </catch>
9   </faultHandlers>
10  <sequence name="orderprocess">
11    <invoke operation="checkCreditWorthiness" ... />
12    <switch>
13      <case condition="CustomerRating = A">
14        <invoke operation="confirmInvoicePayment" ... />
15        <switch>
16          <case condition="InvoiceConfirmation = false">
17            <invoke operation="selectPaymentMethod" ... />
18          </case>
19          <otherwise>
20            <invoke operation="prepareInvoicePayment" ... >
21              <compensationHandler>
22                <invoke operation="cancelInvoicePayment" ... />
23              </compensationHandler>
24            </invoke>
25          </otherwise>
26        </switch>
27      </case>
28      <otherwise>
29        <invoke operation="selectPaymentMethod" ... />
30      </otherwise>
31    </switch>
32    <switch>
33      <case condition="PaymentMethod = Creditcard" >
34        <invoke operation="prepareCreditCardPayment" ... >
35          <compensationHandler>
36            <invoke operation="cancelCreditCardPayment" ... />
37          </compensationHandler>
38        </invoke>
39      </case>
40      <otherwise>
41        <invoke operation="prepareDebitPayment" ... >
42          <compensationHandler>
43            <invoke operation="cancelDebitPayment" ... />
44          </compensationHandler>
45        </invoke>
46      </otherwise>
47    </switch>
48    <invoke operation="confirmSalesOrder" ... />
49    <switch>
50      <case condition="SalesOrderConfirmation = true" >
51        <invoke operation="processSalesOrder" ... />
52      </case>
53      <otherwise>
54        <throw name="cancel" faultName="cancel"/>
55      </otherwise>
56    </switch>
57  </sequence>
58 </scope>
59 </process>

```

**Fig. 8** Execution model for an order transaction at *Matchball* (excerpt)

switch-constructs. Each outgoing sequence flow corresponds with an underlying case-element, Default flows are mapped via otherwise-elements. A possible reset of the transaction is triggered by the throw-element in line 54 and leads to the activation of the compensationHandler with its respective invoke-activities in the

lines 21, 35 and 42 via the execution of the operation DeleteSalesOrder stated in the faultHandlers-element.

## 7 Summary and outlook

The leading providers of IT solutions see service-orientation as the paradigm of the future. With this management and system architecture concept, it appears possible for the first time to connect business process modeling with the physical execution in a software platform. Nevertheless, the economic analysis of the investments connected with the introduction of an SOA should not be neglected. In order to make this evaluation of a long-term, strategic realignment of enterprise-wide IT architecture connected with SOA possible, this article emphasized the usage of a service-oriented architecture for the design of business processes. The introduced methodology makes recommendations for process design that are economically founded. With a cost/benefit analysis, one can determine whether the result (the benefit) of the introduction of SOA justifies its effort (the costs). With the methodology introduced in this paper, such an assessment can be carried out in consideration of the individual economic parameters of a company.

The intended connection of two fields—on the one hand, conceptual modeling and on the other, process controlling—for the evaluation of the value of an SOA is new for two reasons: *first* up to now, no noteworthy research results on the value of SOA were available and *second*, the lack of continuity of business requirements towards IT implementations often criticized can be reacted to with the results in this article.

The scientific preoccupation with SOA in this analysis must however, be distinguished from empirical research. Nevertheless, the findings won here can be consulted as a starting point for further empirical research work. Thus, analyses are possible for investigating whether the more deductively won findings conform to the reality observed in business practice. It is also possible to analyze to what extent effects could be achieved on the effectiveness and efficiency of the procedure in SOA implementation projects by using our methodology. And lastly, a future challenge can be seen in the investigation of the effects of this methodology in business reality.

**Acknowledgments** This work has been funded by the German Federal Ministry of Education and Research in the research project PIPE (Process-oriented Integration of Product Development and Service Documentation), reference: 01FD0623. The authors would like to thank the anonymous reviewers for their comments which helped to improve an earlier version of this paper.

## References

- Knowledge Based Systems Inc. (ed) (1993) Integration Definition for Function Modeling (IDEF0). <http://www.idef.com/pdf/idef0.pdf>
- Alves A et al (eds) (2006) Web services business process execution language Version 2.0: Committee Draft, 17th May 2006. Billerica, MA: OASIS

- Alonso G, Casati F, Kuno H, Machiraju V (2004) Web services: concepts, architectures and applications. Springer, Berlin
- Baresi L, Heckel R, Thöne S, Varró D (2003) Modeling and validation of service-oriented architectures: application vs. style. In: Proceedings of the 9th European software engineering conference held jointly with 11th ACM SIGSOFT international symposium on foundations of software engineering. Helsinki, Finland: ACM Press, pp 68–77
- Barnes M (2005a) Mitsui-Soko drives service-oriented architecture for business agility. Gartner, Stamford, CT
- Barnes M (2005b) Superpartners uses service-oriented architecture to increase market share. Gartner, Stamford, CT
- BEA (2006) SOA case studies. BEA Systems Inc. <http://www.bea.com/framework.jsp?CNT=customers.htm&FP=/content/solutions/soa/library/>
- Becker J, Delfmann P (eds) (2007) Reference modeling: efficient information systems design through reuse of information models. Springer, Berlin
- Becker J, Kahn D (2003) The process in focus. In: Becker J, Kugeler M, Rosemann M (eds) Process management: a guide for the design of business processes. Springer, Berlin, pp 1–11
- Becker J, Kahn D, Wernsmann C (2003) Project management. In: Becker J, Kugeler M, Rosemann M (eds) Process management: a guide for the design of business processes. Springer, Berlin, pp 13–40
- Berlecon (2005) BPM in practice—success factors, project experiences and vendors. Berlecon Research GmbH, Berlin
- Billington J, Christensen S, van Hee K, Kindler E, Kummer O, Petrucci L, Post R, Stehno C, Weber M (2003) The petri net markup language—concepts, technology and tools. In: van der Aalst W, Best E (eds.) applications and theory of petri nets: 24th international conference, Icatpn 2003, Eindhoven, the Netherlands, June 23–27 2003, Proceedings. Berlin: Springer, pp 483–505
- Blechar MJ, Sinur J (2006) Magic quadrant for business process analysis tools, 2006. Gartner Res, Stamford, CT
- Bostrom R, Heinen S (1977) MIS problems and failures: a socio-technical perspective, Part 1: the causes. *MIS Quarterly* 1(3):17–32
- Brynjolfsson E (1993) The productivity paradox of information technology. *Commun ACM* 36(12):66–77
- Buchanan B (ed) (1979) Theory of library classification. Clive Bingley, London
- Cardoso J, Sheth A, Miller J, Arnold J, Kochur K (2004) Quality of service for workflows and web service processes. *J Web Semantics* 1(3):281–308
- Carr N (ed) (2004) Does it matter? Information technology and the corrosion of competitive advantage. Harvard Business School Press, Boston
- Workflow Management Coalition (ed) (2007) XPDL support & resources. <http://www.wfmc.org/standards/xpdl.htm>
- Cooper R, Kaplan RS (1991) Profit priorities from activity-based costing. *Harvard Business Rev* 69(3):130–135
- Creditreform (ed) (2007) Crefosprint online. [http://www3.creditreform.de/downloads/flyer/Produktblatt\\_Crefosprint\\_Online.pdf](http://www3.creditreform.de/downloads/flyer/Produktblatt_Crefosprint_Online.pdf)
- Day J, Deters R (2004) Selecting the best web service. In: Lutfiyya, H, Singer J, Stewart DA (eds) Proceedings of the 2004 conference of the Centre for Advanced Studies on Collaborative Research. IBM Press, Markham, pp 293–307
- Duggan J et al (2006) Hype cycle for application development, 2006. Gartner, Inc. [http://www.gartner.com/DisplayDocument?doc\\_cd=127755](http://www.gartner.com/DisplayDocument?doc_cd=127755)
- Ellram LM, Siferd SP (1993) Purchasing: the cornerstone of the total cost of ownership concept. *J Business Logistics* 14(1):163–184
- Emig C, Weisser J, Abeck S (2006) Development of SOA-Based Software Systems – an Evolutionary Programming Approach. In: Proceedings of the Advanced Int'l Conference on Telecommunications and Int'l Conference on Internet and Web Applications and Services. IEEE Computer Society, pp. 182–188
- Erl T (2004) Service-oriented architecture: a field guide to integrating XML and Web services. 5th print. Upper Saddle River, NJ: Prentice Hall PTR
- Erl T (2005) Service-oriented architecture: concepts, technology, and design. Prentice Hall PTR, Upper Saddle River, NJ
- Ferris C, Farrell J (2003) What are Web services? *Commun ACM* 46(6):31
- Fettke P, Loos P (eds) (2007) Reference modeling for business systems analysis. Idea Group Publishing, London

- Fingar P, Bellini J (2004) *The real-time enterprise: competing on time using the revolutionary business SEX machine*. Meghan-Kiffer Press, Tampa
- Florescu D, Grünhagen A, Kossmann D (2003) XL: an XML programming language for web service specification and composition. *Comput Network* 42(5):641–660
- Fowler M (ed) (2003) *UML distilled: a brief guide to the standard object modeling language*, 3rd edn. Addison-Wesley Professional, Boston
- Frankel DS (2003) *Model driven architecture: applying MDA to enterprise computing*. Wiley, Indianapolis
- Fremantle P, Weerawarana S, Khalaf R (2002) Enterprise services. *Commun ACM* 45(10):77–82
- Ganesh J, Padmanabhuni S, Moitra D (2004) Web services and multi-channel integration: a proposed framework. In: *Proceedings of the IEEE international conference on web services (ICWS'04)*, June 6–9, 2004, San Diego, California, USA. IEEE Computer Society, pp 70–79
- Giagliis GM, Paul RJ, (1996) It's time to engineer re-engineering. Investigating the potential of simulation modelling for business process redesign. In: Scholz-Reiter B, Stickel E (eds): *Business process modelling*. Berlin: Springer, pp 313–332
- Glezer C, Yadav SB (2001) A conceptual model of an intelligent catalog search system. *J Org Comput Electronic Commerce* 11(1):31–46
- MediaFinanz GmbH & Co.KG (ed) (2007) SOAP Web-Service: Beschreibung der MediaFinanz-SOAP-Schnittstelle. [http://www.mediafinanz.de/files/pdfs/MediaFinanz\\_SOAP.pdf](http://www.mediafinanz.de/files/pdfs/MediaFinanz_SOAP.pdf) (in German)
- Graham C, Latimer N, Biscotti F, Correia J, Eschinger C, Pang C, Topolinski T (2004) *Software Market Research Methodology and Definitions, 2003–2004: Gartner Dataquest Guide*. Stamford : Gartner
- Grob HL (1993) *Capital budgeting with financial plans: an introduction*. Gabler, Wiesbaden
- Grob HL, vom Brocke J (2006) Profitability of business processes. In: Becker J, Kugeler M, Rosemann M (eds) *Process management: a guide for the design of business processes*, 2nd edn. Berlin, Springer
- Hagel JI (2002) Edging into web services. *McKinsey Quart* 4:29–37
- Hagel JI, Brown S (2003) *The agile dance of architectures—reframing IT enabled business opportunities*. [http://www.johnhagel.com/paper\\_agiledance.pdf](http://www.johnhagel.com/paper_agiledance.pdf)
- Henderson JC, Venkatraman N (1993) Strategic alignment: leveraging information technology for transforming organisations. *IBM Syst J* 32(1):272–284
- Hirschheim R, Klein HK, Lyytinen K (1995) *Information systems development and data modelling: conceptual and philosophical foundations*. Cambridge University Press, Cambridge
- Hitt LM, Brynjolfsson E (1996) Productivity, business profitability, and consumer surplus: three different measures of information technology value. *MIS Quart* 20(2):121–142
- Hündling J, Weske M (2003) Web services: foundation and composition. *Electronic Markets* 13(2):108–119
- IBM (ed) (2007) *Line of visibility enterprise modeling (LOVEM)*. <http://www-935.ibm.com/services/us/index.wss/offering/bcs/a1006593>
- Im KS, Dow KE, Grover V (2001) Research report: a reexamination of IT investment and the market value of the firm—an event study methodology. *Inform Syst Res* 12(1):103–117
- Kalepu S, Krishnaswamy S, Loke SW (2003) Verity: a QoS metric for selecting web services and providers. In: *Fourth international conference on web information systems engineering workshops (WISE 2003 Workshops)*, pp 131–139
- Kang Y, Kang K (2005) Process controlling and monitoring scheme for distributed systems with fault-tolerance by using web services. In: Fan, W, Wu Z, Yang J (eds) *Advances in web-age information management*, 6th international conference, WAIM 2005, October 11–13 2005. Hangzhou, China, Proceedings. Springer, pp 876–881
- Keen P, McDonald M (2000) *The eProcess edge: creating customer value & business in the internet Era*. Osborne/McGraw-Hill, Berkeley
- Klückmann J (2007) 10 steps to business-driven SOA. Saarbruecken: IDS Scheer AG (ARIS Expert Paper, Inventory Number SOA 0607-E-EP)
- Knolmayer G (1997) A hierarchical planning procedure supporting the selection of service providers in outtasking decisions. In: Krallmann H (ed) *Wirtschaftsinformatik '97: internationale Geschäftstätigkeit auf der Basis flexibler Organisationsstrukturen und leistungsfähiger Informationssysteme*. Physica, Heidelberg, pp 99–119
- Korherr B, List B (2007) Extending the EPC with performance measures. In: *Proceedings of the 2007 ACM symposium on Applied computing*, Seoul, Korea, March 11–15, 2007. ACM Press, New York, pp 1265–1266
- Krafzig D, Banke K, Slama D (2006) *Enterprise SOA : service-oriented architecture best practices*, 5th print. Prentice Hall PTR, Upper Saddle River

- Kreger H (2003) Fulfilling the web services promise. *Commun ACM* 46(6):29–34
- Leymann F (2003) Web services: distributed applications without limits. In: Weikum G, Schöning H, Rahm E (eds) *BTW 2003, Datenbanksysteme für Business, Technologie und Web, Tagungsband der 10. BTW-Konferenz, 26–28 Februar 2003, Leipzig*. Leipzig: Springer, pp 2–23
- Leymann F, Altenhuber W (1994) Managing business processes as an information resource. *IBM Syst J* 33(2):326–348
- Linthicum DS (2000) *Enterprise application integration*, 3rd edn. Addison-Wesley, Boston
- Luftman J (2004) Key Issues for IT Executives 2004. *MIS Quarterly Executive* 4(2):269–285
- Mallick S, Sharma A, Kumar B, Subrahmany S (2005) Web services in the retail industry. In: *SADHANA—Academy proceedings in engineering sciences* 30, April/June, pp 159–177
- Melenovsky MJ (2005) Business process management's success hinges on business-led initiatives. Gartner Research, Stamford, CT
- Mentzas G, Halaris C, Kavadias S (2001) Modeling business processes with workflow systems: an evaluation of alternative approaches. *Int J Inform Manage* 21(2):123–135
- Mukhopadhyay T, Kekre S, Kalathur S (1995) Business value of information technology: a study of electronic data interchange. *MIS Quart* 19(2):137–156
- Murata T (1989) Petri nets: properties, analysis and applications. *Proc IEEE* 77(4):541–580
- Mylopoulos J (1998) Information modeling in the time of the revolution. *Inform Syst* 23(3/4):127–155
- Newell A, Simon HA (1972) *Human problem solving*. Prentice-Hall, Englewood Cliffs
- Nüttgens M (2002) An evaluation framework for business process modelling tools. <http://www.cimosa.de/IctSupport/ToolSt-mn02.html>
- OASIS (ed) (2001) ebXML Business process specification schema: version 1.01. UN/CEFACT, Oasis. <http://www.ebxml.org/specs/ebBPSS.pdf>
- OMG (ed) (2005) Unified Modeling Language: Superstructure, version 2.0, formal/05–07-04. Needham : OMG
- OMG (ed) (2006a) Business process modeling notation specification: final adopted specification dtc/06-02-01. OMG, Needham
- OMG (ed) (2006b) UML Profile for enterprise distributed Object Computing (EDOC). <http://www.omg.org/technology/documents/formal/edoc.htm>
- Oracle (ed) (2007) Bringing SOA value patterns to life: an oracle white paper. Redwood Shores, Oracle Corp., CA, USA
- Orlikowski WJ (1992) The duality of technology: rethinking the concept of technology in organizations. *Org Sci* 3(3):398–427
- Österle H (1995) *Business in the information age: heading for new processes*. Springer, Berlin
- Österle H (2007) *Business engineering: transition to the networked enterprise*. *Electronic Markets* 6(2):14–16
- Padovitz A, Krishnaswamy S, Loke SW (2003) Towards efficient selection of web services. In: *Workshop on web services and agent-based engineering*, Melbourne, Australia, July 14/15, 2003
- Pfadenhauer K, Dustdar S, Kittl B (2005a) Challenges and solutions for model driven web service composition. In: *Proceedings of the 14th IEEE international workshops on enabling technologies: infrastructure for collaborative enterprise*. IEEE Computer Society, pp 126–134
- Pfadenhauer K, Dustdar S, Kittl B (2005b) Comparison of Two Distinctive Model Driven Web Service Orchestration Proposals. In: *Proceedings of the seventh IEEE international conference on E-Commerce technology workshops—Volume 00*. IEEE Computer Society, pp 29–36
- Porter ME, Millar VE (1985) How information gives you competitive advantage. *Harvard Business Rev* 63(4):149–160
- Postbank (ed) (2004) *eCommerce 2004: Strukturen und Potenziale des eCommerce in Deutschland aus Kunden- und Händlersicht*. Deutsche Postbank AG, Bonn (in German)
- Ran S (2003) A model for web services discovery with QoS. *ACM SIGecom Exchanges* 4(1):1–10
- Recker J, Mendling J (2006) On the translation between BPMN and BPEL: conceptual mismatch between process modeling languages. In: Latour T, Petit M (eds) *Proceedings of the 18th international conference on advanced information systems engineering: proceedings of workshops and doctoral consortiums*, pp 521–532
- Remme M (1995) Systematic development of information systems using standardised process particles. In: Doukidis GJ et al (eds) *Proceedings of the 3rd european conference on information systems—ECIS '95: Athens, Greece, vol II. June 1–3 1995*, pp 963–969
- Rumbaugh JE, Jacobson I, Booch G (1998) *The unified modeling language reference manual*. Addison-Wesley Longman, Amsterdam

- SAP (ed) (2006) Enterprise SOA Case Studies. <https://www.sdn.sap.com/irj/sdn?rid=/webcontent/uuid/f4f1fe1f-0b01-0010-dba1-e1c3bdc6cd64>
- Scheer A-W (1999) ARIS–business process modeling, 2nd edn. Springer, Berlin
- Schelp J, Schwinn A (2005) Extending the business engineering framework for application integration purposes. In: Proceedings of the 2005 ACM symposium on applied computing, Santa Fe, New Mexico, March 13–17, 2005. ACM Press, New York, pp 1333–1337
- Schmit BA, Dustdar S (2005) Model-driven development of web service transactions. *Enterprise Model Inform Syst Architect* 1(1):46–56
- SCHUFA (ed) (2007) SCHUFA Scoring Services: SCHUFA-Score für Handelsunternehmen—Mehr Sicherheit. [http://www.schufa.de/downloads/PIB\\_SSC\\_Handel\\_0505.pdf](http://www.schufa.de/downloads/PIB_SSC_Handel_0505.pdf) (in German)
- Sutherland J, van den Heuvel W-J (2002) Enterprise application integration and complex adaptive systems. *Commun ACM* 45(10):59–64
- Swenson KD, Pradhan S, Gilger MD (2004) Wf-XML 2.0: XML based protocol for run-time integration of process engines. WfMC
- Tam KY, Moinzadeh K, Berke E (1998) The impact of information technology investments on firm performance and evaluation: evidence from newly industrialized economies. *Inform Syst Res* 9(1):85–98
- Thomas O (2005) Understanding the term reference model in information systems research: history, literature analysis and explanation. In: Kindler E, Nüttgens M (eds) *Business process reference models : proceedings of the workshop on business process reference models (BPRM 2005)*; Nancy, France, September 5, 2005. Nancy, pp 16–29
- Thomas O, Horiuchi M, Tanaka M (2006) Towards a reference model management system for business engineering. In: *Applied computing 2006: Proceedings of the 2006 ACM symposium on applied computing*; April 23–27, 2006. Dijon, France, vol 2. ACM, New York, pp 1524–1531
- van Baal S, Krüger M, Hinrichs JW (2005) *Internet-Zahlungssysteme aus Sicht der Händler: Ergebnisse der Umfrage IZH3*. Köln: Inst. für Handelsforschung, Univ. Köln (in German)
- Vandermerwe S, Rada JF (2007) Servitization of business: adding value by adding services. *Euro Manage J* 6(4):314–324
- Vollmer K, Gilpin M (2006) *The forrester wave: enterprise service bus, Q2 2006*. Forrester Research, Inc, Cambridge
- Vollmer K, Moore C (2006) Demand for business process management suites will accelerate through 2009. Forrester, Cambridge
- vom Brocke J (2007) Service portfolio measurement: evaluating financial performance of service-oriented business processes. *Int J Web Services Res* 4(2):1–32
- Walter S, Spitta T (2004) Approaches to the Ex-Ante evaluation of investments into information systems. *Wirtschaftsinformatik* 46(3):171–180
- Wand Y, Weber R (2002) Research commentary: information systems and conceptual modeling—a research agenda. *Inform Syst Res* 13(4):363–376
- Wang A-L (2003a) Case study and return of investment: Use of SAP NetWeaver at Sasol. IDC, Framingham
- Wang A-L (2003b) SAP netweaver enables a profitable new business model at goodyear. IDC, Framingham
- Weber R (1997) *Ontological foundations of information systems*. Coopers & Lybrand and the Accounting Association of Australia and New Zealand, Melbourne
- Wenhui S, Feng L, Jinyu Z, Gang D (2006) Develop a telecommunication service system using service-oriented architecture. In: *Proceedings of the IEEE international conference on e-Business engineering*. IEEE Computer Society, pp 674–677
- Wolff F, Frank U (2005) A multi-perspective framework for evaluating conceptual models in organisational change. In: *Proceedings of the 13th European conference on information systems, information systems in a rapidly changing economy, ECIS 2005, Regensburg, Germany, May 26–28, 2005*, pp 1283–1294
- Woods D (2003) *Enterprise services architecture*, 1st edn. O'Reilly, Beijing
- Woods D, Mattern T (2006) *Enterprise SOA: designing IT for business innovation*, 1st edn. O'Reilly, Beijing
- Yu T, Zhang Y, Lin K-J (2007) Efficient algorithms for Web services selection with end-to-end QoS constraints. *ACM Transac Web I* 1:1

- Zeng L, Benatallah B, Dumas M, Kalagnanam J, Sheng QZ (2003) Quality driven web services composition. In: Proceedings of 12th international conference on World Wide Web (WWW'03), Budapest, Hungary, 2003. ACM Press, New York, pp 411–421
- zur Muehlen M, Rosemann M (2000) Workflow-based process monitoring and controlling—technical and organizational issues. In: Sprague RH (ed) Proceedings of the 33rd Hawaii International Conference on Systems Sciences, January 04–07, 2000, Wailea, HI. IEEE Computer Society, Washington