Service Engineering for the Internet of Services

Jorge Cardoso, Konrad Voigt, and Matthias Winkler

SAP Research CEC, Chemnitzer Strasse 48, 01187 Dresden, Germany
{jorge.cardoso,konrad.voigt,matthias.winkler}@sap.com

Abstract. The Internet and the Web have extended traditional business networks by allowing a Web of different digital resources to work together to create value for organizations. The most industrialized countries have entered a post-industrial era where their prosperity is largely created through a service economy. There is a clear transition from a manufacturing based economy to a service based economy. From the technological perspective, the development of Web-based infrastructures to support and deliver services in this new economy raises a number of challenges. From a business perspective, there is the need to understand how value is created through services. In this paper, we describe how we propose to address these two perspectives and realize the vision of the Internet of Services (IoS), where Web-based IT-supported service ecosystems form the base of service business value networks. This paper addresses the main challenging issues that need to be explored to provide an integrated technical and business infrastructure for the Internet of Service.

Keywords: Internet of Services, service engineering, service, e-service, web service, business models.

1 Introduction

Throughout the years organizations have always tried to introduce new business models to gain a competitive advantage over competitors or to explore hidden markets. For example, IKEA introduced the concept that people could transport the merchandise and assemble the furniture by themselves. eBay gained an early competitive advantage by being the first-to-market with a new business model based on auctions. Dell was able to bypass distributors, resellers, and retailers and use the Internet to reduce costs. In all these examples, the new or adapted business models are often derived from the human perception that something could be done in a different way. The idea comes very often from intuition and it is driven by a business need.

Recently, the vision of the Internet of Services (IoS) [13] emerged and can be seen as a new business model that can radically change the way we discover and invoke services. The IoS describes an infrastructure that uses the Internet as a medium for offering and selling services. As a result, services become tradable goods. Service marketplaces, where service consumers and providers are brought together to trade services and so engage in business interaction, are an enabling technology for the IoS vision. Thus, the IoS provides the business and technical base for advanced business models where service providers and consumers form business networks for service

J. Filipe and J. Cordeiro (Eds.): ICEIS 2008, LNBIP 19, pp. 15–27, 2009.

provision and consumption. Within these business networks, organizations work together to deliver a service to consumers. For example, a service-based value network may include the research, development, design, production, marketing, sales and distribution of a particular service. All these phases work interchangeably to add to the overall worth of a service. Value is created from the relationship between the company, its customers, intermediaries, aggregators and suppliers.

In the Internet of Services, the underlying IT perspective provides a global description of standards, tools, applications, and architectures available to support the business perspective. Currently, the service-oriented architecture paradigm has gained mainstream acceptance as a strategy for consolidating and repurposing applications to be combined with new applications in more dynamic environments through configurable services. Services that are composed into advanced business processes can interoperate with other services in order to support business processes spanning across organizational boundaries. In this paper we describe the challenges to support the concept of the Internet of Services. We enumerate the areas that need to be explored to provide fundamental insights on research to enable a radically new way to trade services on the Internet. A special focus will be laid on service marketplaces as an enabling technology for the IoS and to service engineering, which we see as a fundamental approach for creating services.

The remaining of this paper is structured in four main sections. In Section 2 we describe the role of marketplaces and clarify the concept of service for the IoS. Section 3 identifies a set of requirements that needs to be addressed to support the concept of IoS to provide, create and drive a new "service industry" for producing, changing, adapting, (re)selling, and operating services in a Web-based business service economy. In Section 4, we discuss the importance of Service Engineering (SE) for the IoS. SE is a new discipline that will enable the development and implementation of technological solutions based on the Internet of Services. Finally, Section 5 presents our conclusions.

2 Marketplaces for the Internet of Services

Electronic marketplaces for products have gained much attention over the last years enabling business interaction between providers and consumers of physical goods. Examples of such marketplaces include eBay and Amazon. In the IoS vision, services are seen as tradable goods that can be offered on service marketplaces by their providers to make them available for potential consumers. [2] describe service marketplaces as one example of web service ecosystems which represent "… a logical collection of web services whose exposure and access are subject to constraints, which are characteristic of business service delivery." On a service marketplace multiple providers may offer their services. Providers may be large providers as well as small companies offering specialized services. As such, an ecosystem of competing as well as collaborating services may be created.

2.1 What Are Services?

The terms Service, e-Service and Web Service have been widely used to refer, sometimes, to the same concept and other times to different concepts. These terms are generally used to identify an autonomous software component that is uniquely identified by a URI and that can be accessed using standard Internet protocols such as

XML, SOAP, or HTTP. [1] have identified that the terms Service, e-Service and Web Service actually address related concepts from different domains such as computer science, information science and business science. We believe that a deeper understanding of those concepts needs to be made in order to conceptually separate and address the various stakeholders involved when architecting an enterprise wide solution based on services. Therefore, we introduce a set of definitions for the IoS.

Business Service. In business and economics, a service is the non-material equivalent of a good. In these domains, a service is considered to be an activity which is intangible by nature. Services are offered by a provider to its consumers. We adopt the definition from [1] who defined business services as business activities provided by a service provider to a service consumer to create a value for the consumer. This definition is consistent as it is also understood in the business research community. In traditional economies, business services are typically discovered and invoked manually, but their realization maybe performed by automated or manual means (Figure 1). Therefore, business services may be performed by humans. Examples include *cutting hair*, *painting a house*, *typing a letter*, or *filling a form*. If a service is executed by means of automated mechanisms then *processing an insurance claim* is also considered a service. Services are considerably different from products primarily due to their intangible nature. Most products can be described physically based on observable properties, such as size, color, and weight. On the other hand, services lack of concrete characteristics. Thus, services must be defined indirectly in terms of the effects they have on consumers. Products have usually a well defined set of possible variants for customization. For example, if a consumer requires a faster laptop, a more powerful CPU can be designed, built and attached to the motherboard. If an important consumer (e.g. Yellow Cab Co.) desires yellow cars, a manufacturer only needs to notify the production chain to select a new color. The same cannot be easily achieved for services. This makes the description of services one of the most important undertakings for the IoS.

e-Service. With the advances made by the Internet, companies started to use electronic information technologies for supplying services that were to some extend processed with the mean of automated applications. At this stage, the concept of e-service [12], electronic- or e-commerce was introduced to describe transactions conducted

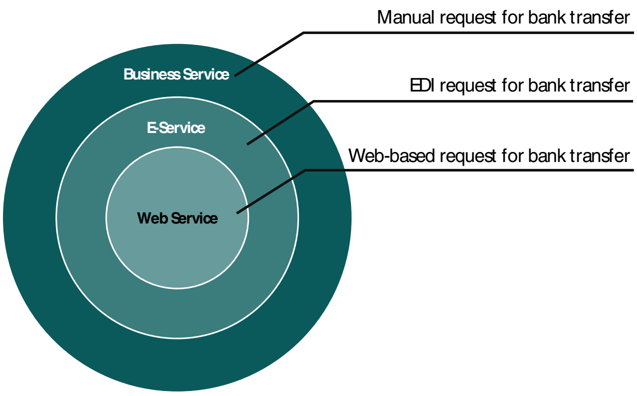

Fig. 1. Examples of invocations of Business Services, e-Services and Web Services

over the Internet. The main technology that made e-commerce a reality was computer networks. Initial developments included on-line transactions of buying and selling where business was done via Electronic Data Interchange (EDI). Examples of such transactions include an *EDI request for bank transfer* (Figure 1) or a *money transfer via a private network*. While many definitions for e-services can be found in the literature, we will use the definition given by [6] since it adequately matches our service concept: "a e-service is a collection of network-resident software services accessible via standardized protocols, whose functionality can be automatically discovered and integrated into applications or composed to form more complex services." Therefore, we consider that e-services are a subset of business services (Figure 1). E-services are services for which the Internet (or any other equivalent network such as mobile and interactive TV platforms) is used as a channel to interact with consumers. Virtually any service can be transformed into an e-service if it can be invoked via a data network. It should be pointed out that this definition implies that the ability to withdraw money from an ATM machine is supplied trough an e-service. E-services are independent of the specification language used to define its functionality, non-functional properties or interface. The term Internet Services will be used be used to refer to the discovery and invocation of e-services using the Internet as a channel. In this paper, when no ambiguity arise, we will use the term service to refer to an Internet service.

Web Service. Web services are e-services that are made available for consumers using Web-based protocols or Web-based programs. Separating the logical and technical layers specifications of a service leaves open the possibility for alternative concrete technologies for e-services. Nowadays, we can identify three types of Web services: RPC Web Services, SOA Web Services, and RESTful Web services. *RPC Web Services* bring distributed programming functions and methods from the RPC world. Some researchers view RPC Web services as a reincarnation of CORBA into Web services. *SOA Web Services* implement an architecture according to SOA, where the basic unit of communication is a message, rather than an operation. This is often referred to as "message-oriented" services. Unlike RPC Web services, loose coupling is achieved more easily since the focus is on the "contract" that WSDL provides, rather than the underlying implementation details. *RESTful Web Services* are based on HTTP and use a set of well-known operations, such as GET, PUT, and DELETE. The main focus is on interacting with stateful resources, rather than messages or operations (as it is with WSDL and SOAP). [4] describes REST objectives in the following way: "The name 'Representational State Transfer' (REST) is intended to evoke an image of how a well-designed Web application behaves: a network of web pages (a virtual state-machine), where the user progresses through an application by selecting links (state transitions), resulting in the next page (representing the next state of the application) being transferred to the user and rendered for their use." We also consider that any e-service that can be invoked using Web standards, such as HTTP, is also a Web service.

2.2 Discovery, Invocation and/or Execution of Services

The lifecycle of services includes two main phases that are of importance to the IoS (Figure 1): *discovery/invocation* and *execution*. Discovery and invocation refer to the medium and technology used to find and request for the execution of a particular service (human-based, via EDI, Web-based, etc.). The execution describes how the realization of a service is carried out. A service maybe carried out only with human involvement, with a conjunction of humans and automated devices, or resorting purely on automated machines (Figure 2). Therefore, services in the vision of the IoS may lay anywhere in the spectrum of services executed by humans, on the one side, or purely automated services on the other side. Nonetheless, in the IoS, the discovery and invocation of all services is IT-based. Service marketplaces provide the access point to the services made available by their providers. One example of a service where invocation is IT-based but execution is performed by humans would be a house painting service where a consumer selects the painter (service provider) and the color of the house using a service marketplace. The painting of the house is, of course, done by humans.

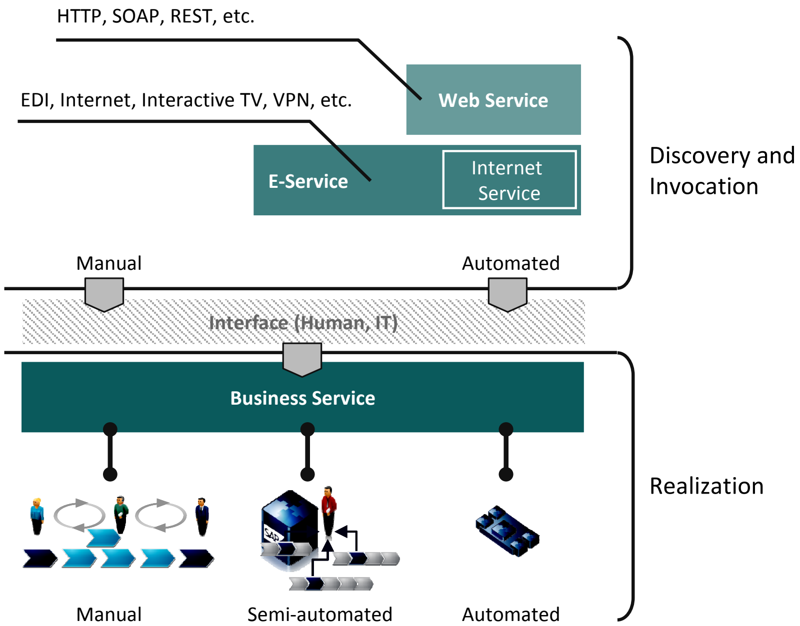

Fig. 2. Relationships between Business Service, e-Service and Web Service

2.3 Atomic vs. Composite Services

The vision of marketplaces for services was previously described by [11]. They describe two steps in the evolution of marketplaces for offering services. As a first step basic services are offered on an electronic marketplace. The focus is on making services available via a new medium – the Internet – as opposed to the real world. Advanced mechanisms for searching and finding suitable services based on their properties were created and support for automatic negotiation of provisioning terms between the service provider and the service consumer was established. Consumers were able to create a one-to-one relationship with a provider. As a second step more complex relationships between consumers and providers are to be formed. Basic services can, in many cases, only provide basic functionality. To provide more complex functionality, composite services may be created from basic services or other composite services and offered on the marketplace. This might mean that there is a need for interaction of the different services with each other.

In the scope of our work we are striving for the second step. There will not only be service providers and service consumers involved in business interactions. A further role often called service aggregator will be performed by entities that specialize on combining services into more complex service compositions or bundles which are then offered on the marketplace to be found and used by service consumers. While the aggregated services are sold to the service consumers by the aggregator, the individual parts they consist of (atomic services or other service compositions) will be provided by their original providers who will receive a payment from the aggregator. The added value created by the service aggregator stems from the selection and composition of available services into more complex services or bundles which constitute an advantage for the consumer.

Aspects of service supply and demand such as service discovery, monitoring, charging and payment as well as publishing, composition, and re-provisioning (leasing, licensing) that are important for the different stakeholders (service providers, aggregators and clients) need to be regulated by the service marketplace [2]. In order to do so, service marketplaces need to provide a broad range of functionality which we will describe in more detail in the next section of this paper.

3 Requirements for the IoS and Marketplaces

Marketplaces, as enabler of the IoS vision, need to offer complex functionality for bringing together a large number of consumers, aggregators, and providers and enable the means for business interactions among them. Service providers offer their services on the marketplace. Consumers select services from different providers based on functionality, best pricing, offered quality of service, or rating. After the service has been selected, it is delivered by the provider. Finally, the client will have to pay a due amount for the service consumption. In order to pave the path for the IoS there is a need for identifying and understanding the underlying challenges that need to be address to provide solutions realizing the vision. The IoS bridges business and IT infrastructures perspectives. As a result, IoS requirements need to have a strong emphasis on the business and IT sides. Therefore, the following topics need to be analyzed, studied and framed within the IoS:

Legal and Community Aspects, and Business Models. The implications of business value networks need to be studied from a legal perspective. The combination and integration of world-wide regulations is fundamental. A special emphasis has to be given to the generation of new business models for all stakeholders (i.e., service providers, aggregators, and consumers) and corresponding incentive mechanisms. Community aspects encourage cooperation and boost innovations through the extensive exchange of knowledge.

Service Search based on Advanced Service Description. Marketplaces need to offer search mechanisms that allow for heterogeneous search criteria. At the base for such search mechanisms, a framework for describing different aspects of services is needed. A suitable service description framework covers not only the functional description of a service but also aspects such as pricing, quality of service, user rating, and legal aspects among others. Consumers will be able to search for service functionality based on functional classifications such as UN/SPSC or eCl@ss [5] or natural language based descriptions. The search results may then be further refined taking into consideration a large variety of non-functional properties, thus leading to improved search results [8].

Negotiation of Service Level Agreements (SLA). Prior to interacting with a service, the consumer needs to create an agreement with the service provider stating the terms under which the service need to be provided. Rights as well as obligations of both parties regarding the service consumption will be described. The aspects specified in an SLA (e.g. QoS and pricing) will be derived from the service description as it provides the base for negotiation.

Service Monitoring. In order to enable trust among the participants of a business interaction, there is a need for monitoring the interaction. The goal is to make sure that service providers deliver the service under the terms promised to the consumer. The monitoring of functionality may be provided by the marketplace itself or by a trusted third party (e.g. a monitoring service). The base for the monitoring is the SLA negotiated between the provider and the consumer.

Billing and Payment. Consumers need to pay for the services they have consumed. The marketplace needs to provide the means to assure that clients pay the correct amount of money. An invoice will be created based on the usage information gathered during the interaction with services. During the payment process, a consumer will need to transfer the amount due according to a predefined payment method. Billing and payment functionality may be provided either by the marketplace or by a third party service.

Service Governance. Governance addresses the strategic alignment between business services and business requirements thereby reducing risks and assuring compliance with rules and regulations. The ability to freely compose and orchestrate business functions which are available as services on a diversity of marketplaces bears overwhelming opportunities. Trust and trustworthiness of service offerings must be facilitated by the platform, balancing individual requirements, policies, and must be capable of adapting to the given business context.

Service Delivery Platform. An infrastructure for service delivery has to be provided for technically enabling businesses to engage in business transactions. This infrastructure has to be scalable with respect to complexity, i.e., its consumers must be able to counter the intricacies of distributed systems. Platform services need to be provided directly by the platform. These fundamental services include brokering, mediation, billing, security and trust services.

Service Engineering. Service engineering (SE) is a new approach to the analysis, design and implementation of service-based ecosystems in which organizations and IT provide value for others in the form of services. SE not only provides methodologies to handle the increased complexity of numerous business actors and their value exchanges, but also provides tools for constructing and deploying services that merge the IT and business perspectives.

These requirements are a basis for the TEXO project [14] which main goal is to develop new business models for the Web. It targets the development of an (open) platform for the development, distribution and provision of (business) services. While all these topics are important to support the vision of the IoS, in this paper we will concentrate our study on the emerging research discipline termed Service Engineering. The ISE methodology, which we describe in the next section, provides the means for suppliers of services to create new services.

4 Service Engineering

One recent development that is believed to allow organizations to support the notion of IoS is the adoption of Service-Oriented Architectures (SOA). The OASIS SOA Reference Model defines SOA as "*a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains*" [9]. With SOA, designers of services are facing the challenge of gaining a deep understanding of the business for which they are developing solutions with the right scope and granularity. Designing services is not only a technical undertaking; it is the job of analyzing the business environment and business processes, and identifying business functions that could be implemented as a service. It should be noticed that it is frequently impossible to implement an innovative business model without, eventually, rely on the underlying IT infrastructure. This constitutes a major problem since there is a considerable gap between these two complementary worlds. Therefore, one challenge lies on bridging the gap between business and IT. This challenge requires a set of design principles, patterns, and techniques that has not been precisely identified yet. Therefore, the IoS cannot be realized without giving a strong emphasis on the business side of services.

4.1 Definition

The set of activities involved in the development of service-based solutions in a systematic and disciplined way that span, and take into account, business and technical perspectives can be referred to as service engineering.

*"**Service Engineering** is an approach that provides a discipline for using models and techniques to guide the understanding, structure, design, implementation, deployment, documentation, operation, maintenance and modification of electronic services (e-services)."*

Service Engineering is a structured approach for describing a part of an organization from a service perspective that expresses the way the organization works. The approach should systematically translate an initial description from a natural language that expresses the way stakeholders think and communicate about the organization through a sequence of representations using various models to a representation that is accepted and understood. Developing and implementing services is a major chore for organizations. Dealing with hundreds of services may be seen, from a management point-of-view, as difficult as managing hundreds of human resources inside an organization, requiring a dedicated department, specialized staff, and adequate methodologies.

4.2 The ISE Methodology

Compared to other approaches, the methodology which we are developing (the ISE methodology), not only focuses on a technical perspective, but also focuses on a deep and prominent business perspective when developing services for the IoS. Since the notions of abstraction (perspective) and dimensions (entities) were important for our approach, we have followed an solution based on the Zachman framework [16] to support service engineering.

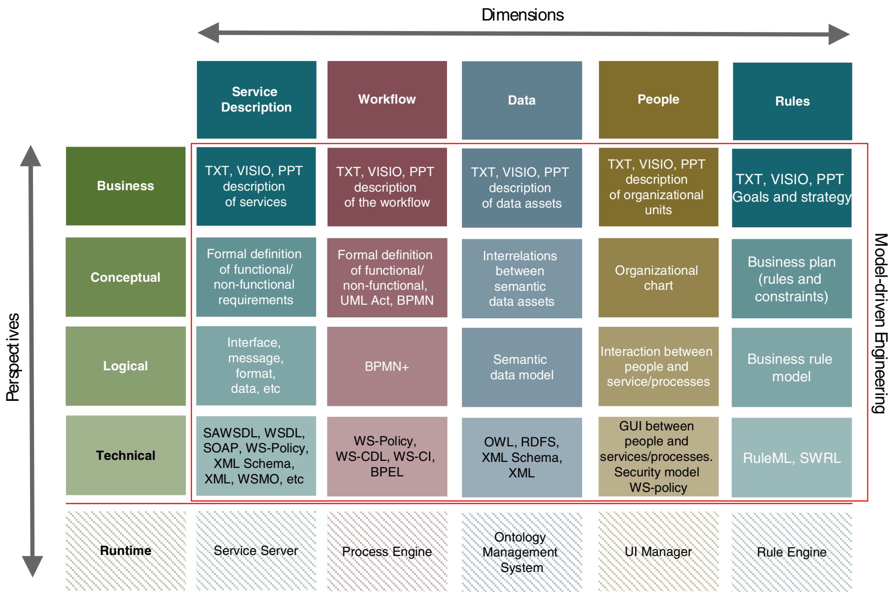

Fig. 3. Perspectives and dimensions of the ISE methodology

Each of the perspectives (layers or rows) of the ISE methodology (Figure 3) can be regarded as a phase in the development process of services. Thus, the models and methods which are assigned to each of the layers support the development process from different view points (e.g., business, conceptual, logical, technical, and runtime).

For all the cells of the matrix we have developed major artifacts which should be considered in the business service development process. These artifacts include important elements such as balanced scorecards, UML, mind maps, BPMN, BPEL, OWL, OCL, etc. Artifacts are assigned to the intersection of an abstraction layer and a dimension (Figure 3).

At the business layer, the development of a service is triggered, typically but not always, by the planning process, where strategies, objectives and performance measures (KPI) that can help an organization to achieve its goals. Fundamental services can often be derived from the strategic planning activity of an organization. Other elements that are typically part of the strategy, or direction, of an organization include resources, capital and people. Models and techniques that can be used to identify fundamental services at the business layer include the SWOT and PEST analysis. Once a list of services is identified – that is deemed necessary for the organization to stay competitive – the service engineering process will proceed with the analysis of the conceptual layer, the logical layer and the technical layer. Once a full technical specification of the service is created, the service is sent to the runtime platform for execution. For services, the business layer, the conceptual layer, the logical layer, the technical layer, and the runtime layer give a different perspective for stakeholders (i.e. CEO, CTO, CIO, architects, IT analysts, programmers, etc.) to services.

4.3 Service Model Integration

In order to implement the ISE methodology different stakeholders have to develop different models defining a service. Since the union of all models defines a service they need to be integrated and synchronized. This integration task is facing major challenges because of the various people involved within the development process and the rising complexity of the models. To cope with these challenges we propose to integrate the models automatically supporting each role in ISE.

The ISE models contain artifacts representing modeled information. Following the separation of concerns paradigm raised by [10], the ISE methodology divides a service into five dimensions, namely: service description, workflow, data, people and rules.

Furthermore, each of these models is divided into four layers (levels) of abstraction. Throughout one dimension artifacts are modeled with respect to different views and refined until they conform to a technical specification. This leads to multiple representations of information on different layers of abstraction in the corresponding dimensions. Each of these models has to be specified and maintained. Changes in one model have to be propagated into the affected models holding the overlapping information. This is a time-consuming and challenging task since each of the models has to be aware of changes and needs to be adjusted. Due to several people involved in the development this leads also to an increased communication effort. For a structured approach we separate the dependencies between models into two classes: vertical and horizontal.

Vertical Dependencies cover the synchronization of dependencies between models on different layers of abstraction in one dimension. It represents the bridging of layers of abstraction by transforming between multiple representations of artifacts.

Horizontal Dependencies define the synchronization of models on the same layer of abstraction. This describes dependencies between models of different dimensions which refer to artifacts of other dimensions. This also includes multiple representations of an artifact on the same layer of abstraction.

These dependencies form the integration of the models and have to be implemented manually or by automatic support. Being more precise, a dependency is defined by a mapping. Formally a mapping assigns to a set of artifacts a set of artifacts; where one sets corresponds to the other. That means the different representations of information are assigned to each other. To illustrate the dependencies, Figure 4 shows an example which depicts the dependencies between two layers of abstraction as well as between models on the same layer but of different dimensions. The workflow dimension shown is specified regarding the conceptual and logical layers. The conceptual layer is represented by an UML activity diagram. The Business Process Modeling Notation (BPMN) is used to represent the logical layer. The artifacts of the logical layer of the data dimension are modeled using an OWL-UML profile. The arrows depict artifacts that need to be synchronized and are mapped onto each other.

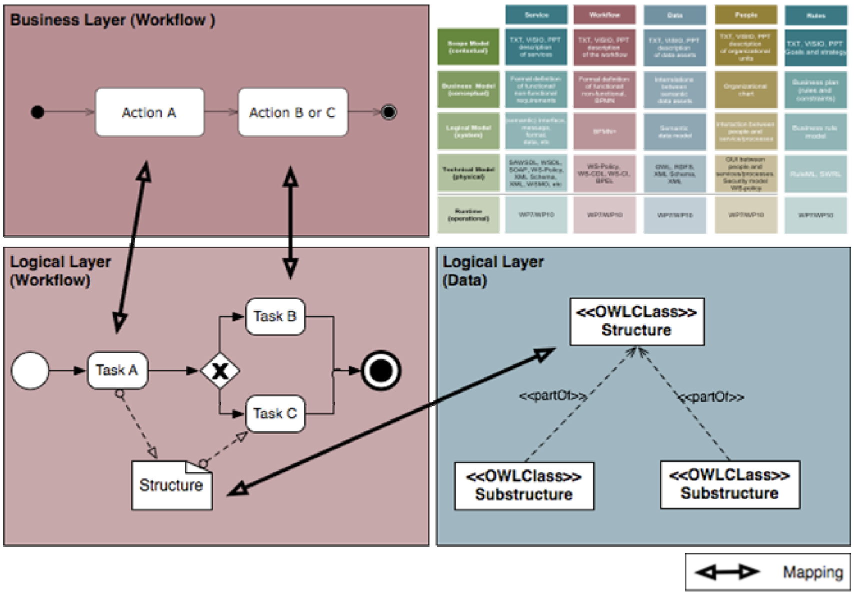

Fig. 4. Example of models (UML activity diagram, BPMN and OWL-UML profile) synchronization

Actions modeled in the activity diagram are again represented in BPMN as tasks. Therefore, *Action A* needs to be in synchronized with *Task A*. That means that UML actions need to be mapped to BPMN tasks. The *XOR* between *Task B* and *Task C* of the BPMN model is mapped from *Action B or C* of the UML model. Furthermore, the *Information I* artifact used in the workflow is defined in the OWL-model (i.e., it depends on it). When one model changes (e.g. renaming or deletion), the depending models have to be updated. These updates can be done manually or by providing an automatic support. One solution to enable an automatic approach is by using model transformations for implementing mappings.

The first step to enable the implementation of model transformations is to define one common formal representation of models. This can be done using ontology formalism or more mature concepts like the Meta Object Facility (MOF). Based on this formalism, a domain specific language for model transformation can be used to define rules and apply them to the models. During the last years many model transformation languages have been proposed, both by academia and industry. For an overview, we refer to [3] classification of today's approaches. The two most prominent proposals in the context of Model Driven Architecture (MDA) are Query, View and Transformation (QVT) and the ATLAS Transformation Language (ATL).

We have chosen to rely on MDA to support model transformations because of matured concepts, well established infrastructure for model management and transformation, and available OMG standards. The MDA guide (2003) defines a model transformation as "the process of converting one model to another model of the same system". Thus a model transformation is an implementation of a mapping (model dependency specification). We follow [7] refining this definition to an automatic generation of a target model from a source model, following a transformation definition. A transformation definition is a set of rules describing formally how a source model can be transformed into a target model. Using a rule-based language like QVT to define model transformations executed by an engine allows for incremental and traceable transformations.

For an automatic model integration we argue for model transformations as the implementation of mappings. Using and applying these concepts enables an automatic model synchronization. This supports both the implementation of vertical and horizontal dependencies, thus reducing the complexity, effort and errors in modeling a service using ISE.

5 Conclusions

The Internet of Services (IoS) will provide the opportunity to create and drive a new "service industry" for producing, changing, adapting, (re)selling, and operating services. By providing a holistic approach, the IoS will be able to contribute to the larger topic of a Web-based business service economy. Business value networks based on the IoS can only be successfully achieved if important topics, such as legal issues, community aspects, new business models, service innovation, service governance and service engineering are exploited.

Service marketplaces act as enablers for business interactions between various stakeholders in the IoS where business services are offered, composed, sold, and invoked by the means of IT. In order to support all stakeholders in their business, marketplaces need to provide advanced functionality such as service search based on functional and non-functional service properties, negotiation and monitoring of SLA and the means for billing and payment. A major constituent of a service marketplace include a common service description framework forming the base for the service lifecycle on the marketplace.

Based on the requirements from marketplaces and based on the concept of IoS, we have introduced a new service engineering methodology for developing and describing services. By covering the technical and business perspectives, ISE provides a structured approach for service engineering. The structuring is achieved by following the separation of concerns and model-driven design. Therefore, we divide a service into several models and identify the need for model integration. Finally, we adopt a model-driven approach by using model transformations to integrate individual models on different layers of abstraction. This leverages service engineering as a discipline and enables the realization of the IoS.

Acknowledgements. The TEXO project was funded by means of the German Federal Ministry of Economy and Technology under the promotional reference 01MQ07012. The authors take the responsibility for the contents.

References

1. Baida, Z., Gordijn, J., Omelayenko, B.: A Shared Service Terminology for Online Service Provisioning. In: The 6th International Conference on Electronic Commerce (ICEC 2004) (2004)

2. Barros, A., Dumas, M., Bruza, P.: The Move to Web Service Ecosystems. BPTrends Newsletter 3(3) (2005)
3. Czarnecki, K., Helsen, S.: Feature-based Survey of Model Transformation Approaches. IBM Systems Journal 45(3) (June 2006)
4. Fielding, R.T.: Architectural Styles and the Design of Network-based Software Architectures, Ph.D. Thesis, University of California, Irvine, California (2000)
5. Hepp, M., Leukel, J., Schmitz, V.: A quantitative analysis of product categorization standards: content, coverage, and maintenance of ecl@ss, UNSPSC, eOTD, and the rosettanet technical dictionary. Knowl. Inf. Syst. 13(1), 77–114 (2007)
6. Hull, R., Benedikt, M., Christophides, V., Su, J.: E-services: a look behind the curtain. In: Proceedings of the twenty-second ACM SIGMOD-SIGACTSIGART symposium on Principles of database systems, pp. 1–14. ACM Press, New York (2003)
7. Kleppe, A., Warmer, J.: MDA Explained. The Model Driven Architecture: Practice and Promise. Addison-Wesley, Reading (2003)
8. O'Sullivan, J.: Towards a Precise Understanding of Service Properties. PhD thesis, Queensland University of Technology (2006)
9. OASIS. OASIS SOA Reference Model (2006) (retrieve on 8 April 2008), `http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=soa-rm`
10. Parnas, D.L.: On the criteria to be used in decomposing systems into modules. Communications of the ACM (12), 1053–1058 (1972)
11. Piccinelli, G., Mokrushin, L.: Dynamic Service Aggregation in Electronic Marketplaces. TechReport HPL-2001-31, Hewlett-Packard Company (2001)
12. Rust, R.T., Kannan, P.: E-service: a new paradigm for business in the electronic environment. Communications of the ACM 46(6), 36–42 (2003)
13. Schroth, C., Janner, T.: Web 2.0 and SOA: Converging Concepts Enabling the Internet of Services. IT Professional 3, 36–41 (2007)
14. Texo, TEXO – Business Webs in the Internet of Services (retrieve on 8 April 2008), `http://theseus-programm.de/scenarios/en/texo`
15. Theseus (retrieve on 8 April 2008), `http://theseus-programm.de/`
16. Zachman, J.A.: A Framework for Information Systems Architecture. IBM Systems Journal 26(3) (1987)