



Business Operating Environment for Service Clouds

Sven Graupner, Sujoy Basu, Sharad Singhal

HP Laboratories
HPL-2011-13

Keyword(s):

services, cloud, service customer perspective, service integration, service management, business relationships with service providers

Abstract:

Currently, most research in services focuses on the needs of the service provider. Examples include research to efficiently create, operate, and deliver services. Much less attention is paid to what the service customer needs to do in order to procure, integrate, or consume a service within the context of a service supply chain. In this paper, we introduce a Services Business Operating Environment (SBOE) to support services from the perspective of the service consumer. An SBOE provides capabilities for defining, contracting, integrating and operating service supply chains as viewed from the perspective of an enterprise customer. The SBOE acts like an operating system within which the customer can connect, combine and transform services purchased from others into higher level abstractions that are relevant for business. We anticipate such an environment to be critical to the "future of IT," where business and IT services are procured over the cloud from specialized service providers rather than created in-house as is done traditionally. We describe research at HP Labs that identifies the requirements and challenges posed by an SBOE, and present an initial prototype around IT supplier management that we have used to explore some of the concepts.

External Posting Date: February 6, 2011 [Fulltext]

Approved for External Publication

Internal Posting Date: February 6, 2011 [Fulltext]

To be published and presented at SRII Conference, San Jose, Mar 29 - Apr 02, 2011, Conference proceedings of SRII conference.

© Copyright SRII Conference 2011.

Business Operating Environment for Service Clouds

Sven Graupner, Sujoy Basu, Sharad Singhal

Hewlett-Packard Laboratories

1501 Page Mill Road

Palo Alto, CA 94304, USA

{ sven.graupner, sujoy.basu, sharad.singhal }@hp.com

Abstract — Currently, most research in services focuses on the needs of the service provider. Examples include research to efficiently create, operate, and deliver services. Much less attention is paid to what the service customer needs to do in order to procure, integrate, or consume a service within the context of a service supply chain.

In this paper, we introduce a *Services Business Operating Environment (SBOE)* to support services from the perspective of the service consumer. An SBOE provides capabilities for defining, contracting, integrating and operating service supply chains as viewed from the perspective of an enterprise customer. The SBOE acts like an operating system within which the customer can connect, combine and transform services purchased from others into higher level abstractions that are relevant for business. We anticipate such an environment to be critical to the “future of IT,” where business and IT services are procured over the cloud from specialized service providers rather than created in-house as is done traditionally.

We describe research at HP Labs that identifies the requirements and challenges posed by an SBOE, and present an initial prototype around IT supplier management that we have used to explore some of the concepts.

Keywords—services, cloud, service customer perspective, service integration, service management, business relationships with service providers.

I. INTRODUCTION

Services have become a substantial part of modern economies encompassing manufacturing, distribution and the consumption of goods. “Everything as a service” [1] describes an ongoing transformation in modern economies where traditional product-centered business is increasingly surrounded by services. In some instances, services even dominate products or are themselves treated like products, creating new value propositions for customers. The Internet has significantly accelerated this trend by enabling more and more business functions to be procured as services from external providers as web services. We use the term *service cloud* to describe this emerging electronic business-business market.

Over the last century, standardization and mass-production at scale has enabled efficient supply chains for products where parts can be widely reused and assembled into larger parts in modern manufacturing. A similar “industrialization” of electronic services has not yet emerged beyond a few basic services such as communication (e.g. email, phone) or information access

(e.g. web). For these standardized services, massive delivery infrastructures have been built to deliver services efficiently at scale. These industrialized service delivery infrastructures are described as “service parks” or “service farms” in [2]. Even for these services, the focus has been on delivering services at massive scale—creation of higher order services based on assembling other services in a global supply chain remains a challenge. Thus, while a deep understanding has emerged for the design, assembly, distribution, consumption and recycling of material goods through complex, globally interconnected supply chains, the same is not true for cloud-based service supply chains, which remain fragmented.

This challenge arises because most innovation in services has focused on the service provider’s perspective. The development of efficient supply chains also requires a service consumer’s perspective, where services are procured and integrated into value added services by intermediaries who “assemble” higher-ordered services from “parts”, either for their own use or for delivery into other service supply chains.

Unfortunately, the “assembly” of independently delivered services into higher-order services remains a hard problem. We believe that solving this problem requires a shift of perspective from the service provider to the service customer. Fundamental problems need to be addressed to enable a service customer to “consume” services by assembling them into other higher-order services either for itself, or for sale to others as part of a larger supply chain.

In this paper, we introduce the metaphor of a *Services Business Operating Environment (SBOE)* to support the customer’s perspective for services. As an operating environment, the SBOE allows the customer to procure, connect and combine externally provided services and transforming them into higher abstractions that are relevant for the customer’s own business.

This paper is structured as follows: It first provides a more detailed problem statement in section II. Section III introduces the metaphor of a Services Business Operating Environment. Section IV briefly describes one aspect of an SBOE: the domain of supplier management for externally procured IT services. It introduces the abstractions based on which the SBOE for this domain operates and presents its architecture and implementation. Then in section IV-D, a concrete instance and example of an SBOE is presented for externally procured IT services. Its customers are internal IT departments in enterprises who have the choice

of either producing desired IT services in-house or procuring them from outside. In this particular instance, the service supply chain is modeled based on the ITIL.v3 methodology [3]. Section V discusses the state of the art and related work. Section VI summarizes the paper.

II. PROBLEM STATEMENT

Services ultimately need to serve business needs of customers. Enterprises have the choice of either building services themselves, or procuring them from third parties in a variety of outsourcing models. However, IT-centric services have remained surprisingly resilient to externalization. Beyond organizational resistance, challenges arise because existing IT practices do not easily account for new factors (e.g. risks) that need to be considered when procuring services. Other problems arise because of the need to establish and manage business relationships with external service providers, which has traditionally not been done within IT, and IT managers often lack business authority to contract external services.

In this paper we follow a methodology driven from business needs by identifying essential “parts”, or sub-services, that are needed for a customer’s business. In the first phase, these business needs and potential services are identified followed by a selection of potential service providers offering the identified services. The second phase then includes the contracting and “assembly” (or integration) of services as part of a service supply chain. This phase also includes ongoing maintenance of business and technical relationships with service providers. From the service customer’s perspective, our methodology has two stages:

1. *Business definition and service discovery* relates to a class of problems in defining and designing business needs and activities, identifying the necessary “parts” for those activities and their relationships and matching them with services that can either be produced in-house (e.g. by internal IT departments) or that can be discovered, selected and procured from external service providers.

2. *Service Integration and Operation*. Once business decisions have been made to procure services from external service providers, issues of integration and operation need to be addressed. Examples here relate to data exchange with external entities and the associated considerations such as compliance with regulations and business policies. In addition, technical problems such as application and data integration must be addressed.

In both stages, as with traditional suppliers contracted by the enterprise, risks need to be assessed, strategies for failure or disruption must be established and relationships with suppliers or service providers must be managed. Today, little technical support exists for handling these complex issues.

III. SERVICES BUSINESS OPERATING ENVIRONMENT

An SBOE provides capabilities for the definition, the selection, the contracting of service providers, the integration and the operation of service supply chains within the context of a service customer. In an SBOE,

services contracted from external service providers can be connected, combined and transformed into higher abstractions that are then suitable for the customer’s business, either for its own internal use or for delivery into further service supply chains.

In this section, we describe research at HP Labs towards creating such a Services Business Operating Environment. The goal of this research is to develop a deeper understanding of issues that emerge when supporting the full lifecycle of service relationships from the definition, selection, integration, operation to the termination of services in a service supply chain. We discuss a number of requirements that can be derived for a SBOE. We assume that service relationships can be established and managed over the Internet. The SBOE itself can be an internal or an external service that mediates these business relationships with service providers on behalf of a service customer.

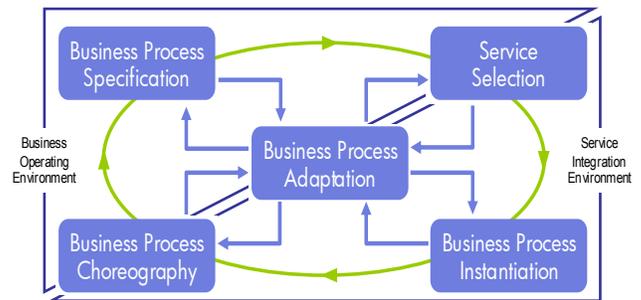


Figure 1. Service Lifecycle from a service consumer's perspective

Figure 1 shows an overall lifecycle of an external service from the perspective of the service consumer. The lifecycle consists of a number of inter-related stages that can be broadly categorized into a business operating environment, which primarily deals with challenges associated with business-level concerns, and a services integration environment, where technical issues associated with consuming the services are addressed. The lifecycle is broken into the following stages:

A. Business Process Specification

The business needs to specify the business level goals, identify business activities that meet those goals, and map those activities to a set of services. Business requirements are typically gathered by consultants who interview various stakeholders and capture the information semi-formally in form of text documents, spreadsheets, or presentation slides. These requirements are usually intended for human communication. They tend to become scattered across teams, and are rarely retained consistently across the entire lifecycle. Recently, research has shown the application of lightweight formal reasoning based on parsing of natural-language requirements documents [4]. However, given the unstructured nature of human-oriented documentation, these methods do not generalize well.

The challenge for capturing business process specifications is to guide a business user through the specification or requirements based on which services later can be contracted. Requirement must be comprehensive and include functional and non-functional requirements. It is important to provide both languages and dialogs which can capture the appropriate information, while not posing an undue cognitive burden on the end-user who is specifying what he or she needs in business terms.

The business requirements then need to be transformed into a set of service-level requirements that must be met by the service candidates that can potentially be contracted. Thus, for example, business continuity requirements may translate to particular service availability or redundancy requirements in each service. Anticipated transaction volumes may have impact on required service scale or performance requirements. Establishing semantic agreement between the business requirements and the underlying services offered by the service providers, and understanding how to transform the business-level requirements into a set of technical requirements for each service are hard problems.

Furthermore, the requirements need to be maintained across the process lifecycle stages to ensure that the overall service supply chain conforms to them. Thus, for example, data privacy requirements may need to be considered at runtime during service choreography to ensure that data is not inadvertently shared with service providers who cannot guarantee privacy of business information being entrusted to them. Retention requirements may force the business (or its service providers) to maintain the data even after the business relationship has been terminated. Since run-time stages are usually automated, this implies that the specifications have to be captured in machine-interpretable form that are generic enough to be re-used, but precise enough to be interpreted correctly at a later stage. This also makes it important to be able to analyze the requirements such that decisions based on them can be explained to business users.

It is necessary to ensure that the business user has the ability to change the requirements at any later stage. Since business conditions may necessitate change in requirements, change management needs to be implemented. It should be possible for users to track the history of changes they have made to their requirements, and the impact of these changes on the service or fabric specifications.

B. Service Selection

The challenge in service selection is to enable the business to select appropriate services from the service cloud that meet all requirements.

Current IT services have grown and evolved organically to meet specific business needs. For a small set of services, it is possible to use manual service discovery of the best service candidates according to business goals, budget, and requirements. As the number of services and

service providers increases, it may become impossible to discover services manually.

The service customer needs to understand the capabilities offered by the services. Current standards for service discovery do not provide the level of abstraction necessary [13]. Challenges remain in describing the services and data, and the available controls over the services and data in terms that the users can understand at the level of abstraction they desire.

The service customer needs to develop trust in order to use services over the cloud. The customer needs to know that the service can be relied upon to meet its declared specifications. In most business environments, “soft” criteria such as the reputation of the service provider, recommendations from others, or past experience with the same provider play a major role in decisions about which service providers are selected by the customer, along with more quantifiable criteria such as price or service levels.

Requirements that exist between service providers and consumers are often captured as Service Level Agreements (SLA) [14], [15], which are written legal contracts (e.g., in business process outsourcing). The Information Technology Service Management (ITSM) defines a framework for service level management [16]. A technical reference for SLA is the ITIL SLA toolkit [17].

In practice, customers use expert recommendations, product review sites, comparative shopping, search engines and social networks to make choices about which services or products to purchase, or which vendor to establish relationships with. Sites such as Amazon and Netflix make product recommendations based on the customer’s and the community’s habits and interests and past history. Typically, prediction is performed using pattern recognition and (more broadly) machine learning techniques such as collaborative filtering, statistical clustering and reputation systems [18]-[21]. However, the state of the art still does not often enable the business user to discover and select best services from the Internet.

C. Business Process Instantiation

The challenge for business process instantiation is to establish the data and control flow from the service customer’s context with the services that are participating in a service supply chain.

Currently, a plethora of standards are used by service implementations. However, these standards largely focus on enabling communication between services at a “wire level” (BPEL is an example) and leave open the harder issues surrounding sharing of business information between services. Hence, each service expects data to be presented in a form dictated by the service writer, and presents data in a similar manner. Moreover, services tend to assume that they hold authoritative information, and are often not structured in ways that allow capabilities to be seamlessly spread across services.

In addition, current web services composition mechanisms have not focused on non-functional characteristics such as reliability, recoverability, or safety of the overall composite service or of the business

information being passed between them. Such issues are relevant to a service customer who runs business-critical operations across a service supply chain. While service level agreements may provide violation penalties, it is much more important to the business to maintain continuity of operations.

Finally, additional issues relate to security, risk management, privacy, and regulatory compliance need to be addressed for using services. Because service customers delegate responsibility for handling business data to the service providers, it becomes important that the customers have knowledge of the extent to which services can meet such requirements, and are able to specify the degree to which such requirements must be met by the underlying services.

As part of business process instantiation, the various services of a services supply chain need to be composed to ensure that data and control flow are coordinated among participating services. Many of the techniques assume static environments, and are not suitable when only short-term relationships are involved. Within the enterprise, there is growing awareness of the availability of on-demand platforms within the enterprise architecture [22].

D. Business Process Choreography

Once communication and delivery channels across services have been established, the business process needs to be orchestrated and observed to ensure that business goals are being met. Large enterprise applications such as SAP provide comprehensive libraries of business processes instantiated in the software [28], but these systems are typically hard to manage and customize for service cloud environments.

In traditional IT environments, the customer maintains control over the IT in-house. In case the customer moves IT functions to the cloud, it becomes even more important that the services provide transparency to the customer about the state of the business process being choreographed on its behalf. Ensuring that the proper information is provided to the customer as feedback during choreography in federated environments is a non-trivial task.

Information may need to be replicated across services to ensure that it is not lost during operation. Audit and logging capabilities are needed to ensure that non-repudiation capabilities are provided when data is passed between multiple providers. Data needs to be logged for billing and charging purposes. Run-time availability, performance, reliability, etc. are also needed. All of these pose additional challenges over existing IT environments.

E. Business Process Adaptation

The challenge here is to have the ability to adapt the process to changes according to needs, faults or failures in the environment. On one hand, service providers are able to optimize the services to business needs by observing large volumes of data and transactions at large scales. On the other hand, the business customer needs to understand how the business objectives could (or should) be changed

to better leverage available capabilities as the underlying component services evolve.

To change the specification of a service, techniques for identifying versions of service interfaces have been proposed [38], [39]. A complete solution requires both the implementation and specification problems be solved, and to this point no single, integrated approach, which can be applied across the board in service development practices, tools and run-times, has emerged or been accepted in standards.

New mechanisms for handling faults and failures in the process are needed. Unlike traditional environments, most large-scale services assume that failure is the norm in the environment. Similar assumptions are necessary at the business process level. Additional issues arise when the business process needs to be executed across administrative domain boundaries. Most current systems provide access to data using some form of federated identity management [31]. However, identity-based access management leads to administrative problems because multiple systems need to maintain identity information.

IV. SBOE FOR IT SERVICE SUPPLY MANAGEMENT

It is obvious that the scope of an SBOE can be very large. In this section, we present the description of a prototype that we have used to understand how to address some of the challenges in developing an SBOE within the context of a concrete example. Our focus is specifically on the business process specification and service selection stages of the service lifecycle. In this prototype we address the issue of supplier management, which is a critical part of sourcing services. In our prototype, we use best practice guidelines established as part of the IT Information Library (ITIL) [3].

An SBOE within this context must address business definition and service discovery as well as service integration and operation. The ITIL methodology defines steps and considerations that need to be taken into account when IT decides to procure services from external providers rather than producing those services in-house. ITIL is a collection of guidelines that summarize a broad set of experiences of IT strategy, design, transformation and operation. It has been well-established in the IT industry. However, ITIL practices typically need to be interpreted within the specific context in order to be implemented.

In this section we briefly introduce ITIL's Supplier Management for externally procured IT services in section A. Section B describes the abstractions and the approach, and section C presents the architecture and technical building blocks of the SBOE system. Section D then describes the implementation of the prototype.

A. Supplier Management in ITIL

The section about Supplier Management in ITIL (4.7 "Supplier Management", ITIL Vol. 2 Service Design [3]) provides guidance on how external IT services should be provisioned. This section takes the view of the contracting

organization of IT services (service customer). Examples of such services are basic information services such as networking, email or archiving services, but can also be business services such as payroll or accounting services. Since it is assumed that these services are delivered by external service providers, which are separate legal entities (external companies), business aspects must be considered beyond the technical aspects. Business aspects include establishing and maintaining legally valid service provider/customer relationships, monitoring contracts and delivered services, maintaining a contract database, etc.

The guidance in ITIL’s Supplier Management section is informally described in form of text over about 20 pages. The following is an excerpt from this text describing the scope of the Supplier Management Process [3]:

*“The **Supplier Management process** should include:*

- *Implementation and enforcement of the **supplier policy***
- *Maintenance of a **Supplier and Contract Database***
- *Supplier and contract categorization and risk assessment*
- *Supplier and contract evaluation and selection*
- *Development, negotiation and agreement of contracts*
- *Contract review, renewal and termination*
- *Management of suppliers and supplier performance*
- *Agreement and implementation of service and supplier improvement plans*
- *Maintenance of standard contracts, terms and conditions*
- *Management of contractual dispute resolution*
- *Management of sub-contracted suppliers.”*

Figure 2. Quote from ITIL's Supplier Management [3].

Our goal is to create an interactive work environment in which this high-level guidance can be represented and interpreted such that interactions among people (or organizations) can be guided within the SBOE.

B. Abstractions for Supplier Management

Like an operating system, a SBOE must be based on proper abstractions based on which it can function. Unlike an operating system, these abstractions do not refer to machine components and their interactions, but to people and their interactions in context of supplier management.

Concept	Sub-concepts
Thing – an identifiable entity.	
Context – Identifiable set of connections between things.	Relationship, Organization, Project.
Activity – Identifiable motion of something over time.	Task – A thing to do, Process – planned & sync., multi-step activity, Step – atomic unit of a process, Event – notification that something happened, Conversation – interaction between actors for purpose of information gathering.
State – Identifiable condition of something at a point in time.	Goal – end state of an activity, Lifecycle State – stages over: plan -> existence -> remembrance.

Item – An identifiable thing to work on or work with.	Document, Schedule.
Actor – Someone or something that carries activity forward.	Person – a human being in a role, System – system performing activity, Service – something or someone doing something for someone else.
Role – function assumed by an actor in a context.	

Table 1: Core abstractions for supplier management process.

Table 1 defines the core concepts. The two top-level concept categories are “thing” and “activity”, which can often be associated with nouns and verbs (or phrases with nouns and verbs) in informal text. Other top-level concepts are “context”, “actor”, “role” and “relationship”. Relationships among concepts lead to graphs, which are often used as a way to formalize and represent domain knowledge. The concepts described in the table can be used to create a variety of interpretation patterns that can be used to drive behavior within the SBOE. For example, an “activity” could be sent to an “actor” associated with a given “role” to be performed. Note that the pattern does not necessarily specify how the activity would be performed—based on such a pattern, the SBOE would simply direct the activity to the appropriate actor. Similarly, “items” could be modified as part of an activity, and such changes would trigger other activities to be performed.

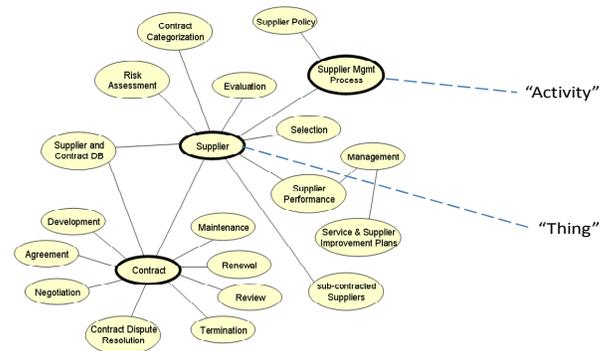


Figure 3. Top level concept graph for supplier management.

In the supplier management case, the main concepts of “Contract” and “Supplier” are categorized as “things”, while “Supplier Management Process” is an “activity”. Figure 3 shows the nodes and relationships as a concept graph describing the guidelines in Figure 2 based on the categorization in Table 1.

The SBOE can interpret this graph using the basic concepts and direct the tasks described, e.g. by asking the proper role to provide a supplier evaluation as a necessary step to qualify a service provider candidate by either using the existing interpretation patterns, or by augmenting or refining existing interpretation patterns by domain-specific patterns.

C. Architecture of Supplier Management SBOE

Figure 4 shows the building blocks of the proposed system. It consists of three main components: an information repository, a logic layer, and a portal as web access layer.

The information layer stores the core information used by the system. Most fundamental are the core and domain representations of knowledge graphs, which define the concepts that are understood by the logic and the interpretation patterns that refer to them. Executable templates link concepts from knowledge graphs and interpretation patterns together. They are used to capture domain information. Templates fall into two categories, general domain templates and context templates, which are specific for projects.

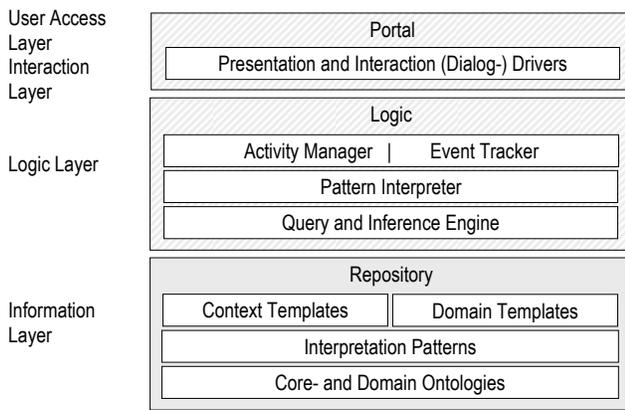


Figure 4. SBOE architecture for Supplier Management.

The logic layer accesses the information of the information layer for querying and inference. This layer provides a generic access layer, based on which a pattern interpreter can load and interpret patterns. Pattern interpretation is triggered from the Event Tracker and the Activity Manager. Activities are triggered either as result of user activity via the user interface portal (e.g. a manager assigns a task to a person), or as result of an event (e.g. the due date for a deliverable has arrived). Events are associated with interpretation patterns which describe the reaction to the event. The Activity Manager is an interface between our system and existing process execution engines. It enables transferring the events captured by Event Tracker regarding the progress of a best practice activity to the corresponding process instance in the process engine, and vice versa, to return the changes in the process execution (e.g., creation of a new task in the workspace of another person) to the user access layer in our system. The activity manager is also responsible for the update of the process definition and process instance when an activity template is refined in our system.

The user access layer is a web-based portal which mediates the user interactions with the system. It performs two major functions. One is the presentation of concepts (information), the other is to initiate actions and present information regarding the progress of activities to the user.

The overall methodology for mapping process descriptions from best practices frameworks into actionable steps consists of four steps.

(1.) The first step is to identify the concepts that are relevant in the best practices framework and relate those concepts in a graph. This work is done by an expert in the best practices framework domain.

(2.) The second step is to assign those domain concepts to pre-defined categories that are understood in our system. For example, the Supplier and Contract Database mentioned in the domain topic list falls into the category of a “thing”, while Supplier and Contract Evaluation and Selection refers to a “process” that needs to be established and performed. This step is also performed by the domain expert.

(3.) The third step is to refine the domain concepts. Concept categories have semantics associated, which means they have properties such as name, definition, etc. The other aspect of a template is the actions represented in form of interpretation patterns. For instance, in our example, Supplier and Contract Evaluation and Selection had been categorized as an activity. One of interpretation patterns on activity is the process definition refinement pattern. Applying this pattern on Supplier and Contract Evaluation and Selection leads our system to perform a dialog with the domain expert asking for more detail about this particular process such as individual steps, the required inputs and outputs, the triggering events, relationships with other concepts (steps and conversations), etc. The domain expert answers this dialog to define the activity template as part of domain templates.

Note that the template can be further refined after being created (creation pattern) to be used in context of a new domain example to create context templates. The result is a definition of a refined Supplier and Contract Evaluation and Selection process. Answers provided by the domain expert in this context lead to more detailed concepts and supplemental information which becomes available to the system.

(4.) Step four refers to the use of the refined process definition, which includes the creation of instances of this process and the application of further interpretation patterns that are defined for the process category, such as the assignment pattern or the execution pattern.

Invoking the assignment pattern means that an instance of the refined Supplier and Contract Evaluation and Selection process will be taken through a dialog in which individual roles and responsibilities are assigned to the tasks of this process. In contrast to the definition pattern, the assignment pattern will not be invoked by the ITIL domain expert, rather by a role that has been put in charge of a particular Supplier and Contract Evaluation and Selection activity.

Invoking the execution pattern means that actual actions are triggered for the assigned roles. Individuals behind those roles receive actionable items about expected deliverables and timelines as well as the input material they need to perform the task assigned to them.

D. SBOE Implementation

We have implemented an initial prototype to demonstrate the proposed approach. We use RDF to represent the information for knowledge graphs, interpretation patterns and templates. We use the Jena toolkit [19] which includes a variety of model stores for the repository as well as libraries for query and inference. The pattern interpreter was implemented in Java. It reads the patterns encoded in RDF (N3 notations) and creates HTML pages for the user interface or invokes operations from the activity manager.

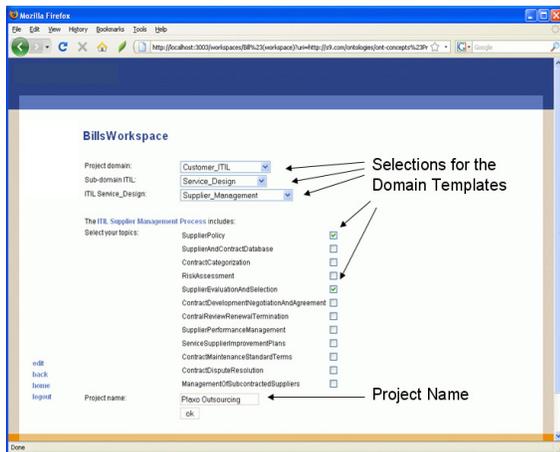


Figure 5. Making choices during the “Creation Dialog” for ITIL Supplier Management Project.

Figures 5 and 6 show screens from the ITIL Supplier Management example presented earlier. The scenario is that a person (Bill) is tasked with a new supplier management project. He creates a new workspace for his new project, which triggers the interpretation of a number of creation patterns starting with a general pattern asking for a dialog about the domain. Choosing ITIL triggers a more specific pattern asking for a domain within ITIL. If Supplier Management is chosen, the creation pattern for Supplier Management is activated. Figure 5 shows the end state of the creation patterns with the choices made.

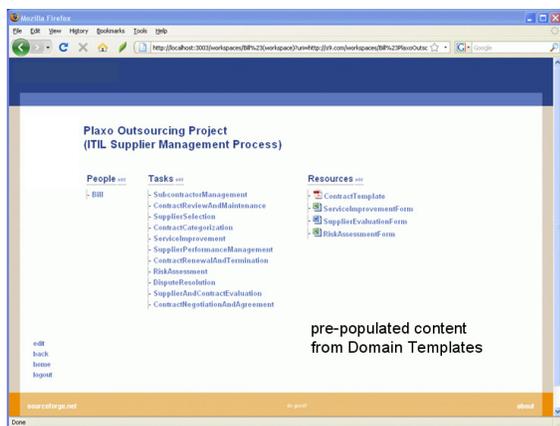


Figure 6. New workspace is populated with concepts from ITIL Supplier Management.

Figure 6 shows the workspace of the new project which is structured based on the Supplier Management process listing its parts in the task column. Bill (the creator) is at this point the only person involved in this project. The resources column shows work material that has been included by the system, also as result of the creation pattern. Each object on the page has further interpretation patterns associated, which are activated when clicked. People can be added or removed, as well as tasks and resources.

The activity manager and event tracker components of the logic layer are implemented in Java. As mentioned in Section III.B, the activity manager is implemented on top of JBoss jBPM [21]. When a process template is created (using the creation pattern), the corresponding process definition is deployed in jPBM process engine. When the activity template is refined (e.g., a task is added), the corresponding process definition is updated by the activity manager and a new version of the process is deployed to the process engine. If the refinement happens while the process is running (there is an associated process instance), then the process instance is migrated to use the new process definition in the jPBM process engine. Given that in best practice processes there is no strict ordering between many of tasks of an activity (unless explicitly specified via dependencies in templates), all tasks are created as child elements of a task node in jPDL. It allows us to achieve the goal of no strict order, while other constraints such as actor, start-date and due-date are enforced for each task.

E. Status and Future work

In our system, we have to build concept graphs (domain ontologies) from the best practices frameworks. The goal of our future work is to assist in the generation of concept graphs from best practice frameworks by populating the RDF database with concepts and relationships extracted the documents for these frameworks. The domain expert should be able to search the database, and include some of the concepts and relationships found as a result into the concept graphs.

This involves knowledge acquisition from text based on natural language processing and machine learning techniques. The taxonomic relationships leading to the hierarchy of classes in the ontology have received a lot of attention. These are the transitive ‘is a’ relationships. The non-taxonomic relationships express the properties of classes and instances and are more challenging. The common approach in the past has been finding anonymous relationships from the text that are frequent or important and then labeling them in a subsequent phase. Generating the relationship label can be done with a chunk parser that maps sentences to structures containing noun phrase (NP) and verb (V) with optional preposition (P). Common patterns can be NP-V-NP and NP-V-P-NP. The verb with optional preposition then becomes the relationship label. When domain knowledge is available, it can guide the generation of an RDF graph. As an example, in [39], the Unified Medical Language System (UMLS) is used to

determine the relationships that should be extracted from abstracts of biomedical publications available in PubMed.

We also plan to observe and learn from people's interactions in order to enrich domain templates. This can be done by processing all context templates after projects finish.

V. RELATED WORK

We categorize the related work into four areas: modeling and specification tools, best practice frameworks and business processes, knowledge and document management systems, and collaboration systems.

Modeling and specification tools: Tools assisting the requirement gathering and analysis [5] stage may include the use of UML [6] for capturing use cases and structural requirements. Systematic requirements analysis is also known as requirements engineering [7]. A large body of literature exists on business process modeling [8]. Standards for business process modeling have evolved over the years, such as XPD [9], BPMN [10], and EPC [11]. Commercial business process modeling tools such as Aris Enterprise Architect from Software AG [12] can capture requirements; however, they are targeted towards specialists in the field, and are rarely usable directly by business people.

Best practice frameworks: processes from best practice frameworks such as ITIL [40] and eTOM [41] has been often described as textual descriptions. There have been efforts to provide automated support for various aspects of best practice frameworks. For example, [42] proposes a Semantic Wiki for ITIL. The work provides query capabilities on RDF generated manually from the ITILv3 glossary (treated as a knowledge base). However, the process aspect is not considered. We propose capturing processes in best practices as templates and making them actionable. We also introduce interpretation patterns to auto-generate dialog pages for each concept in the RDF model in order to refine the model, as well as drive interactions among people.

Business processes: Business process modeling and management tools such ARIS [12] or SAP [43] allow definition of well-defined, rigidly-structured business processes. However, many processes in the enterprise, especially in the context of best practice processes, involve human interactions that are semi-structured or ad-hoc. While our solution takes advantage of existing business process management in driving the interactions, it allows best practice processes to be defined and refined in a much more "lightweight" approach and provides support of a collaborative environment rather than executing hard-coded processes in software.

BPEL4People [44] (as extension to WS-BPEL) addresses the need to capture human interaction in business processes. Until then, one could use WS-BPEL to express the orchestration of web services. It is complementary to our framework, and our activity manager can build on top of BPEL4People engine to enact processes that include human involvements. However, BPEL4People engine has to handle complex human

interaction patterns, such as manual nomination of a task by a supervisor to an employee, escalation and independent decision-making by 2 humans (4-eye principle).

Recent research [23] aims to derive executable business process specifications from higher-ordered business goals. There is also work in the academic community related to automatically composing services [24]-[26], and emerging work applying similar techniques to supply chains [27]. However, such tools are not generally available yet for service instantiation and integration.

Further, the inter-enterprise technologies such as ebXML have specified schema that supports the definition of business transactions and the choreography of business transactions into business collaborations [29]. Another example for business process choreography is the definition by Rosettanet of Partner Interface Processes (PIPs) to help define business processes between trading partners. These define business logic, message flow, and message content to enable commerce [30].

Definition of ad-hoc and flexible processes has also gained attentions recently. For instance, Caramba [45] enables definition of ad-hoc process definitions in the context of virtual team. In this work, the process definition has to be explicitly defined by the team members using graphical tools. However, in the context of best practices: (i) process are not well-structured to be defined formally, (ii) formal process definition is at a lower level of abstractions for knowledge workers that are interested to work with familiar environments such as Wiki and MS Office, and (iii) the proposed template-based approach provides a living knowledge-base for best practice processes. It enables making the domain and context knowledge of processes available to knowledge workers, an advantage that does not exist in works enabling the definition of ad-hoc processes.

A substantial body of work exists for the business process management [32] domain, where the lifecycle for business processes over stages of design, modeling, execution, monitoring and optimization is studied as a continuous process. The aim is continuous adaptation of business processes based on a changing business environment.

Service-oriented business continuity [33] describes ensuring reliable delivery of business functions in SOA environments; the common solution today to service or process adaptation is simply governance. Governance policies are typically applied at the service level, and provide best practices or required policies when changing any aspect of a service within an SOA [34], [35]. Automated or declarative approaches for allowing services to adapt without introducing down-time can be grouped by whether the focus is on the implementation of the service or the specification of the service. The approaches for implementation deal with how a service's "business logic" can be changed while the service is running, and focus on middleware techniques for changing the binding between a

service interface and its implementation at run-time [36], [37].

Knowledge and document management systems as well as collaboration environments: Many existing requirement capture and management tools [46] and business process analysis tools such as ProVison [47] simplify the tasks of gathering, documenting, tracking and managing requirements and process definitions in an enterprise. Typically these tools help document requirements and processes, and in some instances simulate the impact of changes. They are geared towards implementing and executing projects and processes in IT systems, not among people.

Our solution differs from document management systems [48] such as Documentum [49] in that it is targeted at not just managing documents or document workflows, but business interactions between participants. A major hurdle to wide-spread adoption of knowledge management tools [50] today is the poor linkage between them and the surrounding human processes: they are only repositories of information. Our work addresses this hurdle by providing the link so that information in the knowledge base is now actionable. That is they can be interpreted and executed. The other key differentiator of our work from knowledge management systems or business process management tools is that our system does not have to be used “after the fact” (after processes are thoroughly defined). Often knowledge bases or process design tools are used to design processes, but people rarely go back to update information in them. This reduces the ability of enterprises to reuse information and processes as a result of ad-hoc changes needed by people.

The proposed Wiki-based platform differs from Semantic Wikis (e.g. Semantic Media Wiki [51], IkeWiki [52], OntoWiki [53] or KaukoluWiki [54]) because we incorporate domain-independent semantics into the Wiki to organize people’s activities by defining concepts in the RDF model, which relate to functionality implemented in the Wiki back end. We also differ in the ability to upload RDF models templates with domain knowledge into the Wiki.

The social computing concepts, approaches and tooling for knowledge sharing and collaboration incorporated in the enterprise systems, collectively known as Enterprise 2.0 [55], are complementary to our work. Our platform benefits from leveraging abstractions and methods from this area for knowledge capture and sharing between business users.

VI. SUMMARY

In this paper, we described some research at HP Labs and presented an initial prototype of a part of a Services Business Operating Environment (SBOE), a system for establishing and managing a service supply chain for the particular domain of contracting IT services from external service providers using industry best practices guidelines from ITIL.

This SBOE system addresses a need from the observation that most present service innovation focuses

on the service provider side and less attention is given to the service customer side where services need to be procured and integrated into service supply chains.

With the SBOE, we aim at supporting the service customer perspective. The SBOE provides capabilities for the definition, contracting, integration and the operation of a service supply chain following the abstraction of a service lifecycle. An example of an SBOE is presented for externally procured IT services. As customers of this particular instance were assumed IT departments in enterprises who have choices to produce IT services in-house or procure them from outside service providers following a methodology from ITIL (Supplier Management). An initial prototype was developed for this instance of an SBOE with particular focus on the first two stages of the service lifecycle (business definition and specification and service provider selection). A more detailed description of this SBOE prototype is available in [56].

REFERENCES

- [1] Robison, S.: Everything as a Service, Hewlett-Packard, 2010, <http://www.hp.com/hpinfo/initiatives/eaas/index.html>.
- [2] Petrie, C.: Service Parks, IEEE Services, 2008.
- [3] ITIL, version 3, Vol. 2 Service Design, Section 4.7 Supplier Management.
- [4] V. Gervasi and B. Nuseibeh. Lightweight validation of natural language requirements. Software: Practices and Experience, 2002; 32:113-133.
- [5] Requirements analysis, http://en.wikipedia.org/wiki/Requirements_analysis.
- [6] UML Specification v. 1.1 (OMG document ad/97-08-11), <http://www.omg.org/cgi-bin/doc?ad/97-08-11>.
- [7] Wiegers, Karl E. (2003). Software Requirements 2: Practical techniques for gathering and managing requirements throughout the product development cycle, 2nd ed., Redmond: Microsoft Press. ISBN 0-7356-1879-8.
- [8] Rolland, C.; Pernici, C. Thanos (June 1998). A Comprehensive View of Process Engineering. Proceedings of the 10th International Conference CAiSE'98, B. Lecture Notes in Computer Science 1413. Pisa, Italy: Springer.
- [9] XML Process Definition Language (XPDL), <http://www.wfmc.org/standards/XPDL.htm>.
- [10] Business Process Modeling Notation (BPMN), <http://www.bpmn.org>.
- [11] Event-driven process chain (EPC), August-Wilhelm Scheer, ARIS - Business Process Modeling, ISBN 3540658351, Springer; 3rd ed. Edition, March 30, 2000.
- [12] Aris Enterprise Platform, <http://www.software-ag.com>.
- [13] Umapathy, Karthikeyan;Purao, Sandeep A theoretical investigation of the emerging standards for web services, Journal: Information Systems Frontiers; Volume 9, pp 119-134 (2007).
- [14] Robert Benyon and Rob Johnston , Service Agreements: A Management Guide (ITSM Library), Van Haren Publishing, 2006.
- [15] Rick Sturm and Wayne Morris, Foundations of Service Level Management, Sams publishing, 2000.
- [16] Andrew Hiles, The Complete Guide to IT Service Level Agreements: Aligning IT Service to Business Needs (3rd Edition), Rothstein Associates Inc., 2002.
- [17] ITIL, The SLA toolkit, <http://www.service-level-agreement.net>.

- [18] B. Martin, Collaborative filtering: A machine learning perspective, Thesis, Graduate Department of Computer Science, University of Toronto, 2004.
- [19] R. Salakhutdinov, A. Mnih, and G. Hinton, Restricted Boltzman machines for collaborative filtering, *Int. Conf. in Machine Learning*, 2007.
- [20] M. Figueiredo, D. Cheng, and V. Murino, Clustering under prior knowledge. *NIPS*, 2006.
- [21] A. Josang, R. Ismail, and C. Boyd, A survey of trust and reputation systems for online service provision, *Decision Support Systems*, vol. 43, pp. 618-644, 2007.
- [22] David S Linthicum, Leveraging an On-Demand Platform for Enterprise Architecture, Preparing for change., online publication, http://www.salesforce.com/assets/pdf/datasheets/Leveraging_an_On_Demand_Architecture.pdf.
- [23] Matthias Kaiser: Toward the Realization of Policy-Oriented Enterprise Management, special issue on Service Orientation in *IEEE Computer*, Nov 2007.
- [24] Danny Gagne, Marwan Sabbouh, Dr Scott Bennett and Susan Powers, Using Data Semantics to enable Automatic Composition of web services. *IEEE International Conference on services Computing* 2006.
- [25] E Sirin, J Hendler, B Parsia, Semi-automatic composition of web services using semantic descriptions - all 8 versions, *Web Services: Modeling, Architecture and Infrastructure*, 2002 - mindswap.org <http://www.mindswap.org/papers/composition.pdf>.
- [26] Danny Gagne, Marwan Sabbouh, Dr Scott Bennett and Susan Powers, Using Data Semantics to enable Automatic Composition of web services. *IEEE International Conference on services Computing* 2006.
- [27] Liu, Yongjun; Nie, Guihua, Automatic Business Coordination in Supply Chains Using Semantic Web Service Composition *International Conference on Wireless Communications, Networking and Mobile Computing*, 2007. 21-25 Sept. 2007 Page(s):4830 – 4833
- [28] SAP R/3 Business Blueprint: Understanding the Business Process Reference Model by Thomas A. Curran, Gerhard Keller, Andrew Ladd, Prentice Hall, 1997.
- [29] UN/CEFACT and OASIS, ebXML Business Process Specification Schema 6 Version 1.01, 2001. <http://www.ebxml.org/specs/ebBPSS.pdf>.
- [30] RosettaNet, Overview Cluster, Segments, and PIPS, Version 02.01.00 http://www.rosettanet.org/cms/export/sites/default/RosettaNet/Downloads/RStandards/ClustersSegmentsPIPsOverview_UpdateVer2_April2007.pdf.
- [31] Liberty Alliance, <http://www.projectliberty.org>.
- [32] Jörg Becker, Martin Kugeler, Michael Rosemann (Eds.) *Process Management* ISBN 3-540-43499-2.
- [33] Michelsen, J, Services-Oriented Business Continuity, whitepaper from iTKO company, http://www.ebizq.net/executive_corner/topics/ent_tech/features/7862.html.
- [34] Brown, K.; Michael, E. Best practices for web services versioning.
- [35] Lublinsky, B. Versioning in SOA. *Microsoft Architect Journal*. <http://msdn2.microsoft.com/en-us/arcjournal/bb491124.aspx>.
- [36] Kaminski, Mueller, Litoiu, Dynamic Reconfiguration of Evolving Web Services, In *SEAMS 2006 – ICSE Workshop*.
- [37] Florian Irmert, Marcus Meyerhoefer, Markus Weiten, Towards Runtime Adaptation in a SOA Environment.
- [38] Fang, R. et al. A version-aware approach for web service directory. *ICWS 2007*. Page(s):406 - 413. <http://doi.ieeecomputersociety.org/10.1109/ICWS.2007.26>.
- [39] Erik Wilde, Semantically Extensible Schemas for Web Service Evolution.
- [40] Information Technology Infrastructure Library (ITIL), Version 3, 5 Volumes, <http://www.itil-officialsite.com>.
- [41] TeleManagement Forum, Enhanced Telecom Operations Map (eTOM) – The Business Process Framework, <http://www.tmforum.org/BestPracticesStandards/BusinessProcessFramework/1647/Home.html>, visited May 2009.
- [42] S. Battle, D. Booth, K. Jahn, R. Lawson, C. Peltz, RDF, Ontologies and ITIL – Reducing the pain of IT information management, <http://battle-s-1:8080/IkeWiki/index.jsp>.
- [43] SAP, SAP NetWeaver Capabilities - Business Process Management, <https://www.sdn.sap.com/irj/sdn/nw-bpm>, Visited May 2009.
- [44] WS-BPEL Extensions for People—BPEL4People: A Joint White Paper by IBM and SAP, July 2005 Accessed on May 14, 2009 from: <http://www.sdn.sap.com/irj/servlet/prt/portal/prtroot/docs/library/uid/cfab6fdd-0501-0010-bc82-f5c2414080ed>.
- [45] S. Dustdar, Caramba — A Process-Aware Collaboration System Supporting Ad hoc and Collaborative Processes in Virtual Teams, *Distrib. Parallel Databases* 15, 1 (Jan. 2004), pp. 45-66.
- [46] Volere. List of Requirement Management Tools, <http://www.volere.co.uk/tools.htm>.
- [47] Metastorm, ProVision, <http://www.metastorm.com>
- [48] Hci Journal, List of Document Management Software, http://www.hci.com.au/hcisite3/journal/List_of_document_management_software.htm
- [49] EMC. EMC Documentum, <http://www.emc.com/products/category/subcategory/collaboration-and-document-management.htm>
- [50] M. Philippe, Knowledge Representation and Translation in RDF+OWL, N3, KIF, UML and the WebKB-2 languages, <http://www.webkb.org/doc/model/comparisons.html>,
- [51] University of Karlsruhe, Semantic Media Wiki, <http://semantic-mediawiki.org>
- [52] Ike Wiki, <http://ikewiki.salzburgresearch.at>.
- [53] OntoWiki, <http://ontowiki.net>.
- [54] KaukoluWiki, <http://kaukoluwiki.opendfki.de>.
- [55] T. Davenport, Enterprise 2.0: The New, New Knowledge Management? *Harvard Business Online*, Feb. 19, 2008.
- [56] Graupner, S., Motahari, H. R., Singhal, S., Basu, S.: Making Process from Best practices Frameworks Actionable, *Second International Workshop on Dynamic and Declarative Business Processes (DDBP 2009)*, Auckland, New Zealand, September 2009.