SITIO - Semantic Business Processes based on Software-as-a-Service and Cloud Computing

FRANCISCO GARCIA-SANCHEZ  
ENEKO FERNANDEZ-BREIS  
RAFAEL VALENCIA-GARCIA  
University of Murcia  
Facultad de Informatica  
30100 Espinardo (Murcia)  
SPAIN  
frgarcia@um.es  
enekofb@um.es  
valencia@um.es

ENRIQUE JIMENEZ  
JUAN M. GOMEZ  
JAVIER TORRES-NIÑO  
Universidad Carlos III de Madrid  
Computer Science Department  
28911 Leganes (Madrid)  
SPAIN  
enrique.jimenez@uc3m.es  
juanmiguel.gomez@uc3m.es  
javitonino@gmail.com

DANIEL MARTINEZ-MAQUEDA  
Indra Software Labs  
Alcanto 11, 28911 Madrid  
SPAIN  
dmmaqueda@indra.es

Abstract: Recently, the economy has taken a downturn, which has forced many companies to reduce their costs in IT. According to experts, one interesting approach could be to outsource parts of their business through third parties. This will allow companies to free resources invested in outsourced business processes and to focus their investments on those processes that are unique to the organization. The future Internet will be based on services and this new trend will have significant impact on domains such as e-Science, education and e-Commerce. Recently, SaaS (Software-as-a-Service) has emerged as a model of software deployment suitable for implementing this new service-oriented strategy. However, there are few academic works in this area and many gaps are still open. In this paper, we present a Business Process based on Semantics platform where services are executed from a SaaS perspective. The SITIO platform allows external developers to create add-on applications that integrate into the main SITIO application and are hosted on Cloud Computing infrastructure.


1 Introduction

The Web is evolving from a mere repository of information to a new platform for business transactions and information interchange. Large organizations are increasingly exposing their Business Processes through Web Services Technology for the large-scale development of software, as well as for the sharing of their services within and outside the organization. New paradigms for Software and Services Engineering such as the Software-as-a-Service (SaaS) and the cloud computing model promise to create new levels of efficiency through large-scale sharing of functionality and computing resources. SaaS is a software distribution model in which additionally to providing software, the supplier offers additional services such as maintenance and support. The advantage of SaaS is that the software is distributed and hosted on the Internet, eliminating the requirement for the users to install the software on their own computing infrastructure, as well as any related data.

Tightly related to the SaaS model is cloud computing: an IT infrastructure provisioning model in which applications and services from many organizations are hosted in a single large-scale facility. Cloud computing and SaaS open the doors for large economies-of-scale, but are faced with a number of challenges. Foremost among these challenges are: (i) the lack of proven models for determining under which conditions it is cost-effective for IT user organizations to migrate towards these models, taking into account legal, business and technical factors; and (ii) the lack of proven methods (e.g. architectural guidance) for facilitating such migration. Specific challenges include how to represent SaaS and cloud computing capabilities and requirements, and how to enable brokers to match between such capabilities and requirements.

Accordingly, one of the key aims of our work was to develop a framework, including models and methods, to facilitate cost-effective access to hosted business services and cloud computing services, from the perspective of a broker. The SITIO platform can also deal with the problem of the increasing number of Business Processes available as Services on the Web, which gives rise to a new challenging problem: the integration of heterogeneous applications.
Semantic Technologies deal with adding machine-understandable and machine-processable metadata to Web resources through its key-enabling technology: ontologies. Ontologies are a formal, explicit and shared specification of a conceptualization. The breakthrough of adding semantics to Business Processes for SaaS leads to an architecture to integrate various Information Systems from the perspective of emerging scenarios which benefit from the use of such technologies. The aim of SITIO is gathering these emerging concepts (SaaS, Semantic Technologies, Business Process Modeling, Cloud Computing) to foster dramatic evolution of a new platform oriented towards interoperability and cost reduction which can impact significantly in industry. Thus, SITIO can be defined as a platform for reliable, privacy-aware, secure and cost-efficient Semantics-based Software-as-a-Service Creation, Integration and Management.

The rest of the paper is organized as follows. Section 2 contains an overview of the state-of-the-art of the technologies involved in this project. The components that take part in the platform and its overall architecture are described in Section 3. Finally, conclusions and future work are put forward in Section 4.

2 Context
2.1 Technological Background
Four main technologies are involved in this project, namely, Software-as-a-Service, ontologies, Business Process Modeling and Cloud Computing. Combining the state of the art of these technologies, as it is the goal of our project, one can construct a platform oriented towards interoperability and cost reduction which can impact significantly in industry. A brief review of these technologies is provided next.

Software as a Service (SaaS) [2] is a software distribution model in which additionally to providing software, the supplier offers additional services such as maintenance, help and support. The advantage generated by the use of SaaS is that the software is distributed and hosted on the Internet, eliminating the requirement for the user to install the software on his personal computer, as well as any related data. The distribution model is based on the premise that the supplier company offers the maintenance and support service. This entails the assumption that all of the information, processing and business logic is stored in the same location, a fact from which ICT companies greatly benefit, given that the lack of such a concept (which is in many cases, the reality) frequently leads to the dispersion and lack of integration and communication between distinct business processes.

The Semantic Web [17], on the other hand, aims to extend the current Web standards and technology so that the semantics of the Web content is machine-processable. Therefore, it can be said that the Semantic Web is characterized by the association of machine-accessible formal semantics with the traditional Web content. The knowledge representation technology used in the Semantic Web is the ontology. A number of different ontology definitions can be found currently in literature. In this work we have adopted the following one: “an ontology is a formal and explicit specification of a shared conceptualisation” [18]. In this context, formal refers to the need of machine-understandable ontologies. Besides, the ontology language selected in this work was OWL (Web Ontology Language) [12], which is the the ontology language recommended by the World Wide Web Consortium (W3C).

The idea of a service-based architecture comes from the eighties. However, this practice did not come into widespread use until relatively recently. Nowadays, the most common use of Service-Oriented Architectures (SOA) [11] is at a functional level, where it is required to integrate the internal applications of the company. The main problem was connecting and communicating the different units and departments in an easy and scalable way. By means of a SOA, this communication can be solved more easily because its functions are published as services, and they are consumed independently by other departments which utilize the same or other systems. The second use of SOA was derived from the results of the previous advantage, migrating the idea to a superior level, to the business processes instead of functional units. Therefore, Business Process Management (BPM) [6] was developed along with execution languages such as BPEL (Business Process Execution Language), notation languages (BPMN) and monitoring tools (BPA). As a result, a business process can be defined in a specific language and each activity within this process can be connected to the functional unit it depends on.

Finally, cloud computing [4] represents a paradigm shift in the delivery architecture of information services. There is little consensus on how to define cloud computing, but we have adopted the following definition [7]: “A large-scale distributed computing paradigm that is driven by economies of scale, in which a pool of abstracted, virtualized, dynamically-scalable, managed computing power, storage, platforms, and services are delivered on demand to external customers over the Internet”. The National Institute of Standards and Technology (NIST) has established a group focused on promoting the effective and secure use of cloud technology within government and industry. They identify five essential characteristics (on-demand self-service, ubiquitous net-
work access, location independent resource pooling, rapid elasticity and pay per use), three service models (Software-as-a-Service -SaaS-, Platform-as-a-Service -PaaS-, and Infrastructure-as-a-Service -IaaS-) and four development models (private, community, public and hybrid clouds). Virtualization constitutes the cornerstone technology for cloud computing and can be referred to as the abstraction of physical IT resources from the people and applications using them.

2.2 Related Work: Semantic Technology for Cloud Computing

In [13], the authors emphasize the importance that cloud computing techniques have for addressing the problems of scale in processing semantic data. They point out that a number of applications, particularly search engines, will have to cope with the ever increasing amount of semantic-based structured data that is being made available on the Web. The properties of the Web makes it suitable for a distributed approach to the management of data, and the authors believe that cloud computing is the most promising solution.

In [19], the authors undertook a task to construct a unified ontology of cloud computing, where the cloud is sectioned into five main layers - applications, software environments, software infrastructure, software kernel, and hardware. This ontology is used to classify different cloud systems and to capture the interrelations between the different cloud components to compose one cloud service from one or more other cloud services.

In [5], the authors propose an infrastructure composition model that aims at increasing the adaptability of the capabilities exposed through it by dynamically managing their non-functional requirements of on-demand services provisioned from the cloud. This is achieved by a mediator exposing a virtualized interface of the resource enhanced with a Secured Profile (SP). The SP itself will be formed using semantic technologies, which will describe the different components along with their constraints and allow for their dynamic selection and composition. The current implementation allows an enterprise to expose different capabilities as web services in a secure, dynamic, and virtualized manner.

In [10], the authors combine logic programming and communication abstractions to first specify independently from implementation Technologies cloud applications and then synthesize them automatically. The specifications are given for data, operation, and connectivity layer. Target cloud is characterized with a set of agent types, a communication policy and a data access policy. In this abstraction, a cloud consists of a finite set of nodes connected by ideal (no information loss), bidirectional, and point-to-point communication channels.

In [15], the authors advocate the rise of service parks, which would serve sets of Web services with their own sets of rules for combining and modifying them. These service parks are very constrained and thus technically feasible. Instead of providing access to Web services with heterogeneous semantics from a large variety of providers they offer Web services with homogeneous semantics by a selection of providers. These service parks offer customization possibilities, however limited, as well as guaranteed service-level agreements. Service parks typically include common runtime systems that greatly facilitate implementation of the services. Thus it would be natural to assume that these parks could operate on clouds from cloud computing providers.

In [9], the authors combine logic programming and communication abstractions to first specify independently from implementation Technologies cloud applications and then synthesize them automatically. The specifications are given for data, operation, and connectivity layer. Target cloud is characterized with a set of agent types, a communication policy and a data access policy. In this abstraction, a cloud consists of a finite set of nodes connected by ideal (no information loss), bidirectional, and point-to-point communication channels.

3 SITIO Architecture

The proposed architecture of the SITIO platform is depicted in Fig. 1. Three main elements can be distinguished: the user interface, the process and business services, and the metadata services. Next, these components are described in detail.

3.1 User Interface

The aim of the user interface is to enable users with different profiles and necessities to access to the SITIO platform and to use and manage their accounts and applications. The users in the platform can play two roles, namely, tenant or consumer. Tenants are software developers and IT providers that make use of the platform in order to deploy their applications. By means of the utilities provided by our platform, tenants can manage the applications they have placed in the system, setup various customized versions of these applications, upload new applications, and check, with different levels of details, the operating costs of their applications for using the underlying infrastructure. The objective is to assist in managing these applications in the most seamless way possible,
Figure 1: SITIO Architecture

and to abstract all maintenance duties from the developers themselves. Consumers, on the other hand, can be also referred to as the final users of the platform. They are allowed to search in the applications repository through a human-like natural language search interface [3], and get subscribed to those they are interested in. The final users have also access to an application management view in which they can review their subscriptions (i.e., add new applications and remove the unwanted ones) and monitor the applications they are subscribed to.

A set of JavaScript (JS) libraries constitutes the technology to build these user interfaces. In the last few years, several JS libraries have emerged in the dynamic HTML framework. Much of these libraries provide free access to a number of functions for creating Ajax controls and generating different visual effects. Furthermore, in most cases, these libraries are standard-compliant. For the purposes of our work, the user interfaces will be developed using dynamic HTML with JS libraries and AJAX, when possible, to provide an agile and responsive communication capability with the central server. By using these technologies, we guarantee the provision of a usable, efficient and attractive user interface with the additional advantages of being the cross-browser compatible and OS independent.

3.2 Process and Business Services

These two layers of the SITIO architecture provide the ability to run arbitrary web services in the cloud computing platform. To achieve this, they add distribution, load balancing and data persistence services to a given application transparently. This is made possible by imposing some restrictions on the services that run on top of the platform, as being distributed as a Java EE 5 standard application.

Distribution is handled by uploading the same application to multiple application servers in the same cluster. These servers are distributed among different data centers, so a problem in a datacenter does not implies a service disruption. This is combined with the load-balancing service that distributes requests across all the instances running application servers available at the given time. This means that if an instance fails, all the traffic gets redirected to other instances, while keeping the service status by means of an distributed session database.

Furthermore, all data saved to the database provided by SITIO is automatically replicated in multiple database nodes for reliability and availability. The data from these nodes is also backup to an external backup device for added data security. The load balancing service also works in conjunction with the database servers to balance the load among all database servers, splitting read and write queries for increased read performance, which is the most common type of query in most applications.

3.3 Metadata Services: Semantic Annotation of Web Services

In order to facilitate users to locate the desired application quickly, a mere syntactic description of the applications in the cloud registered in the platform is not enough. By making use of machine-processable content, it is possible to enable the automatic processing of users’ queries. Thus, ontologies are employed in SITIO to provide the semantic description of the
available services. In order to properly annotate a Web service, the annotator must be aware of the real purpose of each method of the service. Therefore, in most cases it will be the Web services developers themselves who are in charge of semantically annotating the services. Regrettably, Web services developers are hardly ever aware of the methods and principles concerning the Semantic Web and ontology-based knowledge representation mechanisms. It is thus necessary to elaborate a methodology to assist users in the annotation process by identifying relevant information in the description and implementation of the services and then suggesting annotations.

Many research groups have worked on the development of tools and methodologies with the goal of automating the semantic annotation process. Some of these tools have been subject to a detailed investigation to establish their pros and cons. The platforms included in our study are “METEOR-S: Web Service annotation framework” (MWSAF) [14], “Automated Semantic Service Annotation with Machine Learning” (ASSAM) [8], “Automatic Annotation of Web Services Based on Workflow Definitions” (AAWSW) [1], “A proposal for semi-automatically annotating web services using SAWSDL” (ASWSDL) [16].

From the results of our analysis, we elaborated a semi-automatically three-steps annotating methodology. The annotator must follow three sequential phases:

1. Collect information from the web service;
2. Find mappings between domain ontologies and the web service;
3. Web service annotation and expert user validation for the suggested annotations.

In the first step, to make easier the web service candidate elements identification process, two complementary approaches are followed: (i) manually annotating the services by using special tags and commentaries, and (ii) using natural language techniques. The assumption in the first approach is that while implementing a Web service, the developer adds keywords or named entities that precisely define the functionality provided by the methods that form part of the service. The natural language processing tools of the second approach aim at identifying “named entities” from the information available concerning a particular Web service (implementation, WSDL description, etc.). Based on the Web services-related information gathered in the previous step, the annotator looks for the elements in domain ontologies that are relevant for describing these services. The aim is, therefore, to carry out a classification process in which the Web services-related identified information elements are mapped to ontology components (concepts, relations, attributes, instances, etc.). Taking into account this mapping, the semantic description of the services is produced and the user is asked to validate the proposed annotations and to set the mappings for the elements that the tool has not been able to process automatically.

4 Conclusion

The impact of new paradigms of Integration and Management of Information Systems has been dramatic in most scenarios of society’s day-to-day life. The revolution undergone and provoked by Information Technologies (IT) over the last years was unforeseeable, particularly the transformation of the Web from a mere repository of information to a vehicle for business transactions. Emerging approaches such as Semantic Technologies, Software-as-a-Service (SaaS), Business Process Management and Service-oriented Architectures are changing the current business landscape, as well as domains such as e-Science, education and e-Health, especially regarding their adaptability and interoperability, and derived from both of these characteristics, the incorporation of cognitive and reasoning capabilities.

However, the potential synergies of these emerging trends in IT have not yet been explored. In this work, we aim to gather these emerging concepts (SaaS, Semantic Technologies, Business Process Modeling and Cloud Computing) to foster dramatic evolution of a new platform oriented towards interoperability and cost reduction which can impact significantly in industry. SITIO can be defined as a Business Process based on Semantics platform where services are executed from a SaaS perspective. It allows external developers to create add-on applications that integrate into the main SITIO application and are hosted on Cloud Computing infrastructure.

Several issues remain open for future work. Semantics is currently only focused on the description of Web Services functionality and capabilities. This approach is satisfactory in most cases to assist users in finding the applications they are looking for and to enable the automatic composition of services to generate added-value applications, since Web services represent the most prominent technology for service provision on the Web. However, in order to make the SITIO platform as broadly applicable as possible, other service-based technologies (e.g. RMI, CORBA, REST, etc.) must be contemplated. On the other hand, security has received little attention in literature, even though it is a major concern in this kind of environ-
Acknowledgments: This work has been partially supported by European Union through the EUREKA Initiative (Σ!4989), and the Spanish Ministry for Industry, Tourism and Commerce through projects SITIO (TSI-0204000-2009-148), GO2 (TSI-020400-2009-127), SONAR II (TSI-020100-2009-263) and INNOVA (TSI-020100-2009-612).

References:


