

COMPOSING E-WORKFLOWS USING SYNTACTIC, OPERATIONAL AND SEMANTIC INFORMATION

Jorge Cardoso
University of Georgia
LSDIS Lab, Computer Science Department
University of Georgia, USA
jcardoso@uga.edu

ABSTRACT

With the development and maturity of infrastructures that support e-services, we expect organizations to incorporate Web services as part of their business processes. While in some cases Web services may be utilized in an isolated form, it is natural to expect that Web services will be integrated as part of workflows. The composition of workflows to model e-service applications differs from the design of traditional workflows due to the number of Web services available during the composition process, their heterogeneity, and their autonomy. The composition process raises two main problems that need to be solved: how to efficiently discover Web services and how to facilitate their integration with the workflow host. To support, facilitate, and assist the composition of process, we present a solution based on ontologies. We have developed an algorithm and prototype that workflow systems and discovery mechanisms can use to find Web services with desired interfaces, operational metrics, and to assist designers in resolving heterogeneity issues among Web services. Our approach provides an important solution to enhance Web service discovery and integration.

KEYWORDS

Workflow management systems (WfMSs), quality of service (QoS), workflow composition, web services, business process management.

1. INTRODUCTION

The Internet has created new organizational and working models which are based on electronic transactions. For example, e-services have been heralded as the next wave of Internet-based business applications that will dramatically change the use of the Internet (Fabio Casati et al., 2001). With the development and maturation of infrastructures and solutions that support e-services, we expect organizations to incorporate Web services into their business processes. Workflow management systems are capable of integrating business objects for setting up e-services in an amazingly short time and with impressively little cost (Shegalov et al., 2001).

The modeling and enactment of e-services using workflows raises two challenges for workflow systems. First, Web services must be located that might contain the desired functionality and operational metrics to carry out the realization of a given task. What makes the e-service vision attractive is the ability to automatically discover the e-services that fulfill users' needs (Fabio Casati et al., 2001). It is necessary to efficiently discover Web services from the potentially thousands of services available on the Internet. The discovery of a Web service cannot only be based on syntactic information; it also has to account for operational metrics and semantics. Second, once the desired Web services have been found, mechanisms are needed to facilitate the resolution of structural and semantic differences with other components present in an e-workflow host.

The main motivation for our work is the need to enhance workflow systems with better mechanisms for Web service discovery and integration. Our approach relies on the use of ontologies that describe Web service interfaces and relies on a quality of service (QoS) model to describe Web services operational

metrics. Ontology-based approaches have proved to be an important solution for information integration in order to achieve interoperability (Uchold and Gruninger, 1996).

This paper is structured as follows. Section 2 presents the problems associated with the compositions of e-workflows. Section 3 focuses on the extension of traditional workflow tasks specifications to semantically describe their interfaces, on the specification of Web services, and on the association of a QoS model to specify operational metrics for both tasks and Web services. In section 4, we describe how the discovery and integration of Web service is achieved. Finally, section 5 presents our conclusions.

2. E-WORKFLOW COMPOSITION CHALLENGES

E-services can be orchestrated with hard-coded applications or by using workflows. In this paper, we call a workflow which manages e-services and possibly traditional workflow tasks an e-workflow.

During the composition of an e-workflow the designer manually selects tasks from a repository or registry to model an abstraction of a business process. A workflow comprises a number of logic steps (known as tasks or activities), dependencies among tasks, routing rules, and participants. In a workflow, a task can represent a human activity or a software system. We use the term task can refer to a traditional workflow task or to a Web service.

The process of finding a traditional task from a workflow repository is humanly manageable since the number of tasks to choose from is modest (a repository typically contains only tens to a few hundreds of tasks.) Therefore, no sophisticated search or discovery mechanisms are required and basic query interfaces are sufficient (Song, 2001). In the other hand, when a designer is searching for a Web service to insert in an e-workflow, the potential number of services available can be extremely large. Potentially tens or hundreds of thousands of on-line Web services may be available. We cannot expect a designer to discover a Web service manually. This process is time consuming, cumbersome, and tedious. Thus, efficient discovery mechanisms must be available.

Once a Web service is found, it is not realistic to expect that its interfaces will perfectly match and interoperate with the hosting e-workflow without additional work. Web services are heterogeneous by nature; the designer will need to manually establish connections among the Web service interfaces and the tasks present in an e-workflow, accounting for structural and semantic differences. Therefore, it is important to develop mechanisms to facilitate the resolution of structural and semantic differences. An e-workflow designer should be assisted by mechanisms that suggest the establishment of connections between interfaces that maximizes the degree of integration.

The composition of e-workflows cannot be undertaken while ignoring the importance of operational metrics. Trading agreements between suppliers and customers modeled with e-workflow include the specification of QoS items such as products or services to be delivered, deadlines, quality of products, and cost of service. The correct management of such specifications directly impacts the success of organizations participating in e-commerce and also directly impacts the success and evolution of e-services itself.

3. TASKS AND WEB SERVICES SPECIFICATIONS

We rely on the use of ontologies to semantically describe task and Web service interfaces. Semantics have been a strong candidate for increasing the success of information discovery and integration on the Internet (Fensel and Musen, 2001). Ontologies are introduced as an “explicit specification of a conceptualization” (Gruber, 1993). The use of ontologies for the explication of knowledge is a possible approach to overcome the problem of integrating heterogeneous workflow tasks and Web services.

In our implementation, tasks and Web services interfaces are semantically described by concepts that are defined in ontologies constructed with DAML+OIL (Horrocks et al., 2001). This will facilitate the resolution of structural and semantic heterogeneity. Since there is a strong analogy between interface attributes and data classes and the concepts classes defined in an ontology, the establishment of mappings between the two is facilitated. Our approach is not dependent on DAML+OIL; other ontology representation languages could be employed.

For the prototype that we have developed we have selected DAML-S (Ankolekar et al., 2001) to specify Web services; more precisely, we use the *Service Profile* ontology. As with WSMF (Fensel and Bussler, 2002), our approach to e-workflow composition is not dependent on the method chosen to specify Web services. Therefore, other specification languages can be employed. The service profile ontology describes the functionality of a Web service. It tells “what the service does” (Ankolekar et al., 2001) and is employed to advertise Web services availability and capability. We have decided to use DAML-S because in the same way we did with workflow tasks, we need to establish associations among the interface parameters of a Web service with ontological concepts. Since the DAML-S specification semantically describes Web services, there is an explicit association of Web services interface with concepts.

The operational metrics of workflow, tasks, and Web services are described using a QoS model. QoS addresses the operational issues of workflows components, rather than components function. QoS should be seen as an integral aspect of workflows, and therefore it should be integrated with tasks and Web services specifications. We have developed a theoretical model for the automatic computation of workflow QoS based on tasks QoS metrics (Cardoso et al., 2002a, Cardoso et al., 2002b). Based on our model, we have developed an ontology for the specification of QoS metrics for tasks and Web services. This information will allow for the discovery of Web services based on operational metrics.

4. WEB SERVICE DISCOVERY AND INTEGRATION

In our work we have devised an algorithm and implemented a system to discover Web services and facilitate the resolution of structural and semantic differences during the integration process with an e-workflow. The system determines and evaluates the best mapping between the interfaces of a service object (a Web service specification) and the e-workflow host that yields the highest degree of integration.

The Web service discovery and integration process is carried out by a key operation: the match function. The matching step is dedicated to finding correspondences between a service template (ST, *i.e.*, a query) and a service object (SO). During the discovery phase, the match function is employed to successively match a service template against a set of service objects, which are advertised in a registry.

The system evaluates the degree of similarity between a ST and a SO and provides the means for the integration of services through the analysis and suggestion of connections between the SO interfaces that maximize the degree of integration with the ST. The service objects are ranked based on their degree of similarity and integration with the service template. The user may then select the Web service with the highest degree of similarity and manually solve the schematic differences not already solved by the system.

The function uses syntactic, operational, and semantic information as a way to increase the precision of the match.

Syntactic Similarity: The syntactic similarity of a ST and a SO is based on attributes such as *service names* and *service descriptions*. At this stage, only syntactic information is taken into account, since both fields are simply expressed using a set of words, without attaching any tag of concepts to each one.

Operational Similarity: The autonomy of Web services does not allow for users to identify their operational metrics at design time. Operational metrics characterize Web services according to their QoS, which includes their timeliness, quality of products delivered, cost of service, and reliability. When composing a workflow it is indispensable to analyze and compute its overall QoS (Cardoso et al., 2002a, Miller et al., 2002). Syntactic and semantic information allows for the selection of Web services based on their functionality, but without accounting for operational metrics. The operational similarity of a ST and a SO is calculated based on the metrics specified in their QoS model. The purpose is to determine how close two Web services are, based on their operational capabilities.

Semantic Similarity: To achieve interoperability, it is necessary to address the problem of semantic integration (Sheth and Larson, 1990, Kashyap and Sheth, 1996) – the identification of semantically similar objects that belong to different systems and the resolution of their schematic differences. Purely syntactical methods that treat terms in isolation from their contexts are insufficient since they deal with syntactic but not with semantic correspondences. Since users may express the same concept in different ways, we rely on semantic information to evaluate the similarity of concepts that define the ST and SO interfaces. This

evaluation will be used to calculate their degree of integration. When tasks and Web services are put together, their interfaces need to interoperate; therefore, structural and semantic heterogeneity needs to be resolved. Structural heterogeneity exists because Web services use different data structures and class hierarchies to define the parameters of their interfaces.

5. CONCLUSION

In this paper we have presented a set of challenges that the emergence of Web services and e-services has brought to organizations. Designing workflows managing Web services and e-services entails research in two areas. It is necessary to develop mechanisms to efficiently discover Web services during an e-workflow composition process and to facilitate their subsequent integration with the e-workflow host.

Our approach for Web service discovery is based on ontologies and accounts for three dimensions: syntax, operational metrics, and semantics. The approach allows for Web service discovery not only based on functional requirements, but also on operational metrics and semantics. The need to discover workflow components based on operational metrics has a greater importance when Web services are involved, as compared to workflow tasks. The autonomy of Web services does not allow for users to identify their operational metrics at design time. The development of mechanisms for the discovery of Web services based on operational metrics allows organizations to translate their vision into their business processes more efficiently, since e-workflows can be designed according to QoS requirements, goals, and objectives.

To facilitate the discovery and posteriori integration of Web service into workflows we propose an approach based on the use of ontologies to describe workflow tasks and Web service interfaces. Ontology-based approaches are an important solution to information integration in order to achieve interoperability. Our system automatically computes the similarity of two services, efficiently and without human intervention, and suggests potential mappings between tasks and Web services interfaces which maximize their degree of integration.

REFERENCES

- Ankolekar, A., et al., 2001, DAML-S: Semantic Markup for Web Services. *Proceedings of the International Semantic Web Working Symposium (SWWS)*. Stanford University, California, pp 39-54
- Cardoso, J., et al., 2002a. Modeling Quality of Service for Workflows and Web Service Processes. *the Very Large Data Bases Journal*, Vol. submitted in May 2002, No., pp.
- Cardoso, J., et al., 2002b, Workflow Quality of Service. *International Conference on Enterprise Integration and Modeling Technology and International Enterprise Modeling Conference (ICEIMT/IEMC'02)*. Valencia, Spain, pp
- Fabio Casati, et al., 2001. E-Services - Guest editorial. *The VLDB Journal*, Vol. 10, No. 1, pp. 1.
- Fensel, D. and Bussler, C., 2002. The Web Service Modeling Framework. Vrije Universiteit Amsterdam (VU) and Oracle Corporation. <http://www.cs.vu.nl/~dieter/ftp/paper/wsmf.pdf>.
- Fensel, D. and Musen, M., 2001. The Semantic Web: A Brain for Humankind. *IEEE Intelligent Systems*, Vol. 16, No. 2, pp. 24-25.
- Gruber, T., 1993. A translation approach to portable ontology specifications. *Knowledge Acquisition*, Vol. 5, No. 2, pp. 199-220.
- Horrocks, I., et al., 2001. DAML+OIL. DAML. <http://www.daml.org/2001/03/daml+oil-index.html>.
- Kashyap, V. and Sheth, A., 1996. Schematic and Semantic Similarities between Database Objects: A Context-based Approach. *Very Large Data Bases (VLDB) Journal*, Vol. 5, No. 4, pp. 276-304.
- Miller, J. A., et al., 2002, Using Simulation to Facilitate Effective Workflow Adaptation. *Proceedings of the 35th Annual Simulation Symposium (ANSS'02)*. San Diego, California, pp 177-181
- Shegalov, G., et al., 2001. XML-enabled workflow management for e-services across heterogeneous platforms. *The VLDB Journal*, Vol. 10, No. 1, pp. 91-103.
- Sheth, A. and Larson, J., 1990. Federated database systems for managing distributed, heterogeneous, and autonomous databases. *ACM Computing Surveys*, Vol. 22, No. 3, pp. 183-236.
- Song, M., 2001. RepoX: A Repository for Workflow Designs and Specifications. Department of Computer Science, University of Georgia, Athens.

Uschold, M. and Gruninger, M., 1996. Ontologies: Principles, methods and applications. *Knowledge Engineering Review*, Vol. 11, No. 2, pp. 93-155.