Information Technology Journal 3 (2): 184-187, 2004
ISSN 1682-6027

Modeling the Flow in Dynamic Web Services Composition

Muhammad Adeel Talib, Yang Zongkai and Qazi Mudassar Ilyas
Department of Electronics and Information Engineering,
Huazhong University of Science and Technology, Wuhan, Peoples Republic of China

Abstract: With the growing number of Web services, importance of composing existing Web services into more complex services in order to achieve new and more useful solutions is increasing. However, composing Web services is not a trivial task especially when talking about dynamically composing Web services at run-time. Considerable amount of research has been done and currently is underway in this field. A number of frameworks/prototypes using various techniques have been developed by the industry and academia in an attempt to achieve automated composition. This paper is an attempt to describe the approaches currently being adopted by the research community for modeling Web services composition and to critically evaluate them. A novel approach based on process modeling technique is also presented.

Key words: Web services, dynamic composition, composition approaches, workflow

INTRODUCTION

Following the recent explosion in the number of Web services, established enterprises are continuously discovering new opportunities to form alliances with other enterprises, in order to share their costs, skills and resources by offering integrated services (also called composite services). An example of an integrated service is a Travel Planner that uses flight booking, hotel reservation, car rental and itinerary services as components. The component services may be outsourced to business partners and these business partners can in turn outsource part of the activities involved by the delivery of a service to other businesses or organizations.

Unfortunately, the technology to organize, abstract, search, compose, monitor and access Web services has not entirely kept pace with the rapid growth of available opportunities. Indeed, the development of integrated services is still largely ad-hoc, time consuming and requiring a considerable effort of low-level programming[1]. This approach is clearly tedious and demands an automated approach. This has triggered a considerable amount of research and development efforts, both in the academia and in the industry which has resulted in a number of products, standards, frameworks and prototypes addressing the issue of service composition.

Instead of enumerating the techniques used in all these products and frameworks. This study was attempted to provide the conceptual overview of the approaches towards modeling dynamic composition at a higher abstract level i.e. based on the usage of various industry standards and deployment in different industrial/academic communities. This study also described a novel composition approach based on process modeling.

Approaches to model composition: Generally the approaches to compose Web services can be divided into two broad categories viz., standards based syntactic approach and ontology based semantic approach.

Standard based syntactic approach: The syntactic approach is backed by business world that developed a number of XML-based standards to formalize the specification of Web services, their flow composition and execution. The specification of a Web service is expressed in WSDL, which specifies only the syntax of messages that enter or leave a computer program. In which order messages have to be exchanged between services must be described separately in a flow specification. There are many Web services flow specification languages like BPEL4WS and WSCI, but the composition of the flow (i.e., plan) is still manually obtained. Figure 1 shows different layers of the standards based architecture.

Ontology based semantic approach: The semantic web community focuses on reasoning about resources by explicitly declaring their preconditions and effects with

Corresponding Author: Muhammad Adeel Talib, Postal Address: Room 617, Foreign Students Building, Huazhong University of Science and Technology, Wuhan-430074, Hubei, Peoples Republic of China E-mail: adeeltalib@hotmail.com

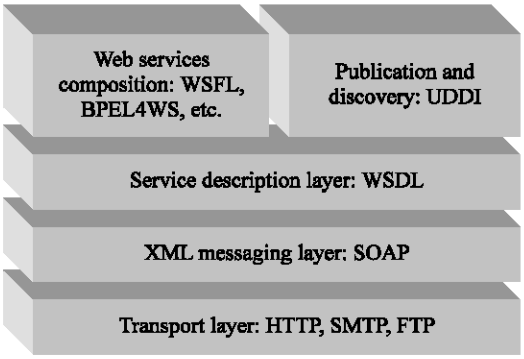

Fig. 1: Standards based architecture

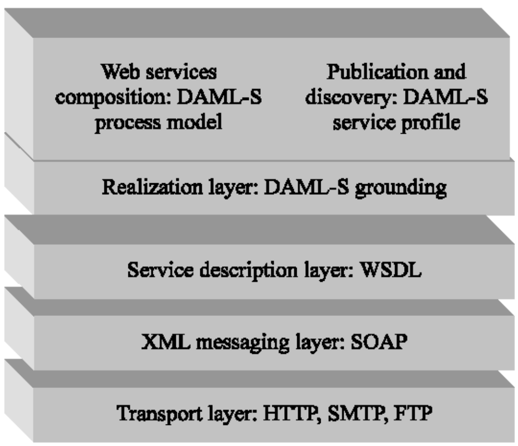

Fig. 2: Semantic based architecture

terms precisely defined in Resource Description Format (RDF) using terms from pre-agreed ontologies[2]. Semantic annotations have been widely discussed in the Semantic web community. OWL-S (formerly DAML-S) is an OWL-based Web service ontology, which supplies Web service providers with a core set of markup language constructs for describing the properties and capabilities of their Web services in unambiguous, computer-interpretable form. Figure 2 presents the technologies involved in this approach. For the composition of Web services, the semantic web community draws on the goal-oriented inferencing from artificial intelligence (AI) planning, which for over three decades, has investigated the problem of how to synthesize complex behaviors given an initial state, an explicit goal representation and a set of possible state transitions.

Critical evaluation of the two approaches: In our opinion none of the two approaches provide true dynamic composition solution. The pros and cons of both of the approaches are also discussed.

Syntactic approach: Any technology supporting a Web services composition language will have to record states for processes that are more complex than a simple request-response. Only by recording the state it is possible to determine what should/can be done, thus enabling long-lived business transactions. WSDL does not provide this functionality. The language also does not provide any information about the control flow. In which order messages have to be exchanged between services must be described separately in a flow specification. This triggered the development of languages like WSFL, XLANG, BPEL4WS, BPML and WSCI etc. However, with the introduction of these languages, a new factor has arisen in which composition using these languages has now become complex and difficult to accomplish[3]. These languages are lacking in an important aspect i.e., semantically representing the activity components of a process[4] and also the process schema has to be manually defined which is a limitation to automation. All these industry based standards describe Web service content in terms of XML syntax, which lacks both a well defined semantics and sufficient expressive power to realize the vision of diverse Web services having wide-scale interoperability[5]. Moreover, it is still confusing which set of standards among the hype of such a large number of standards will provide the best solution[6]. There is a fair amount of traction behind BPEL4WS from major players in the industry and WSCI and BPML have already converged with each other, yet there is no clear winner in this space.

Semantic approach: The need for semantics in service description has been discussed in quite a few of previous publications[7]. Use of ontologies to aggregate the products, services, processes and practices within the industry to realize successful net markets is advocated by Smith[2]. Who states that ontological engineering is the prime requisite for information and service aggregation. To illustrate the need of semantics in service descriptions, let us consider the following scenario. A Web service that provides the location of a user's mobile phone needs to be found. In the search arguments we specify that the inputs for the Web service should be the phone number and the outputs are the longitude and latitude of the phone. Unfortunately, when searching for the said service, an exact match is not found. However, a service, where the input is called MSISDN and the output called location is available. Without knowing anything about the semantics, it would discard this service in the first place. However, if we would have an ontology defining that MSISDN is-kind-of phone number and location is-composed-of longitude and latitude, it could see that in fact the found Web service would serve our needs.

The above example show semantic markup of Web services can provide a way to automate discovery and composition of Web services, but still several challenges remain to be addressed. First of all, the most important question is where the ontologies will come from. Ontology design is a skill that is not widely found in the work force[8]. Current tools, such as Protégé provide only limited help and they have not been widely used outside of prototyping projects and research groups.

Secondly, for the semantic approach to work, semantic markup should exist for the Web services in the form of ontologies. DARPA has proposed DAML-S (and more lately OWL-S), which is a specification to semantically annotate Web services, yet, constructing such ontologies is costly. Currently there is little economical motivation for businesses to semantically annotate their Web services[9].

Thirdly, the semantic approach relies on the use of AI planning techniques to automatically search, compose, orchestrate and execute Web services but these planning techniques cannot be directly applied to service composition problems[10]. The reasons are: I) current planning languages have not yet reached the degree of logical expressivity required for the composition of services; ii) the rich structure of the message objects used in service composition is far more complex then the sparse objects that are currently used by planning community; ii) in contrast to the classical planning where all objects are available in the initial state and the actions change the state of objects, Web services create new objects at runtime and iv) even if service inputs and outputs can be semantically annotated using ontologies, the question is how to give semantics to the input and output specifications of a composite service when different outputs can be produced only when certain complex conditions are satisfied along some composition paths.

Process based service composition: Service composition is an active area of research and development in different fields including component-based frameworks, cross-enterprise workflows, Electronic Data Interchange, XML-based B2B frameworks and agent-based frameworks. Whatever the field be, there is always an execution plan which has to be orchestrated. There is always an underlying process in which the flow of data and control has to be choreographed. Both BPEL4WS and DAML-S provide means for the specification of the underlying process, but the plan/specifications are written manually and no dynamic assembling of complex flows from atomic message exchanges based on a search take place i.e. none has developed a true planning solution.

The process-based composition of Web services is emerging as a promising approach for automating

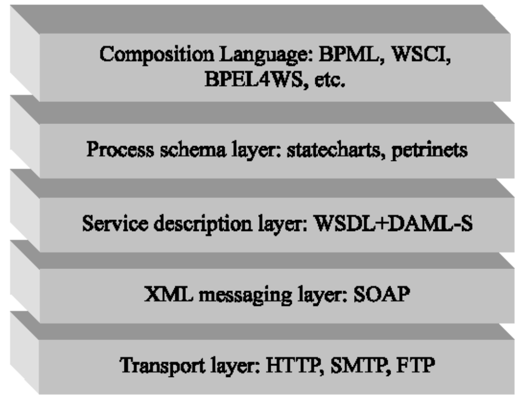

Fig. 3: Process based architecture

business processes within and across organizational boundaries. Several researchers have identified workflows as the computing model that enables a standard method of building Web services applications[11,12]. In this approach, individual Web services are federated into composite Web services. Applications are encapsulated as Web services and the logic of their interactions is expressed as a process model. Business process automation technology such as workflow management systems (WFMSs) can be used to choreograph the component services. This approach provides an attractive alternative to hand-coding the interactions between applications using general-purpose programming languages. The architectural layers of process based composition are depicted in Fig. 3 wherein a process schema/flow is generated with the help of service descriptions which provides the business logic for the composition.

It is our opinion that to achieve Web services' full web-wide and global potential, semantic involvement is necessary. Indeed, there is a need to have a system that enables dynamic composition of Web services based on a process model that also utilize the power of semantics. Some of the prominent research prototypes/frameworks that follow the process based approach[11-14].

Related work: The related work presented are the efforts put forward to compare and critically evaluate Web services composition techniques, technologies, research prototypes and commercial platforms. Currently there are numerous platforms (Microsoft.NET, Sun ONE, IBM webSphere, etc.) and research prototypes (eFlow, SELF-SERV, DYflow, DySCo, Mentor-Lite, etc.) that have been developed by industry and academia. Work on many others is going on (Proteus, FUSION, SHOP2, etc.).

A detailed description of some of the major platforms and frameworks in Web services composition technology is reported by Chakraborty[15]. Wherein the platforms are evaluated and pros and cons are provided individually.

We have provided a critical comparison of the most commonly used approaches from a higher level of abstraction. Srivastava[10] provides a classification of Web services composition based on syntactic and semantic grounds, but more emphasis is given on the issues and challenges involved in service composition from AI planning point of view.

Aalst[3] provides an in depth comparison of the existing service composition standards including XLANG, WSFL and BPEL4WS and some of the major workflow products. The comparison is based on a set of workflow patterns. Each of these workflow patterns correspond to a routing construct often required when designing a workflow.

Medjahed *et al.*[1] compares the technologies involved in B2B interactions based on some dimensions that depend on the usage scenarios, parties involved and business requirements. The dimensions are: coupling among parties, heterogeneity, autonomy, external manageability, adaptability, security and scalability. A comparison of the technologies, research prototypes and commercial platforms is provided in a tabular form in terms of layers which present communication, content and business process.

CONCLUSIONS

Industry looks at composite services mainly from the runtime perspective of functions, data and control flow of processes, while Semantic web focuses on the semantics and on process-centric description of services as actions that are applicable in states. BPEL4WS (syntactic approach) is itself a specification to describe process and DAML-S (semantic approach) also contains the notion of a process model in its Service Model class. Nonetheless these approaches do not provide true dynamicity for composing Web services. There is need to view Web services in the context of specifying, validating and automatically synthesizing complex processes based on formal techniques like process algebras, automata models, or situation calculus.

The discussion above revealed that the two approaches if implemented independently may not accomplish true automated service composition, but if diffused with a blend of business process concepts, the vision of seamless dynamic Web services composition can be achieved.

REFERENCES

1. Medjahed, B., B. Benatallah, A. Bouguettaya, H.H.A. Ngu and A.K. Elmagarmid, 2003. Business-to-business interactions: Issues and enabling technologies. VLDB. J., 12: 59-85.
2. Smith, H., 2002. The role of ontological engineering in B2B net markets. Ontology.org Article. http://www.ontology.org/main/papers/csc-ont-eng.html
3. Wil, M.P. van der Aalst, 2003. Web Service Composition Languages: Old Wine in New Bottles?. Proc. of the 29th EUROMICRO Conference, Turkey, IEEE Comp. Society Press, pp: 298-307.
4. Sivashanmugam, K. *et al.*, 2003. Framework for Semantic Web Process Composition. Technical Report 03-008, LSDIS Lab, Computer Science Dept., University of Georgia Athens. http://lsdis.cs.uga.edu/Projects/METEOR-S/
5. McIlraith, S. and D. Martin, 2003. Bringing Semantics to Web Services. IEEE Intelligent Systems, IEEE Computer Society Press, 18: 90-93.
6. Peltz, C., 2003. Web Services Orchestration: A Review of Emerging Technologies, Tools and Standards. HP Report. http://devresource.hp.com/drc/technical_white_papers/WSOrch/WSOrchestration.pdf
7. Sivashanmugam, K., K. Verma, A. Sheth and J. Miller, 2003. Adding Semantics to Web Services Standards. Proc. of the 1st Int. Conf. on Web Services (ICWS'03), Las Vegas, pp: 395-401.
8. Heflin, J. and M.N. Huhns, 2003. The zen of the web. IEEE Internet Computing, 7: 30-33.
9. Carman, M. and L. Serafini, 2003. Planning for Web Services the Hard Way. Workshop on Service Oriented Computing, Int. Symp. on App. and Internet (SAINT-2003). Florida, USA.
10. Srivastava, B. and J. Koehler, 2003. Web Service Composition: Current Solutions and Open Problems. ICAPS Workshop on Planning for WS., pp: 28-35.
11. Benatallah, B. *et al.*, 2002. Definition and execution of composite Web services: The SELF-SERV project. Bull. IEEE Tech. Comm. Data Eng., 25: 47-52.
12. Casati, F., S. Ilnicki, L. Jin and M. Chen Shan, 2000. An Open, Flexible and Configurable System for E-service Composition. Tech. Report HPL-2000-41, Palo Alto. www.hpl.hp.com/techreports/2000/HPL-2000-41.html
13. Zeng, L., B. Benatallah, H. Lei, A. Ngu, D. Flaxer and H. Chang, 2003. Flexible composition of enterprise Web services. Electronic Markets-Int. J. E-Commerce and Business Media, 13.
14. Ambrozkiowicz, 2003. enTish: An approach to service description and cmposition. Instytut Podstaw Informatyki, PAN, Warszawa. http://www.ipipan.waw.pl/mas/
15. Chakraborty, D. and A. Joshi, 2001. Dynamic Service Composition: State-of-the-art and Rsearch Directions. Technical Report TR-CS-01-19, CSEE, UMBC.