Special Issue:
Web Services Discovery and Composition

Editorial Preface
Liang-Jie Zhang, IBM T.J. Watson Research Center, USA

Services discovery and composition are among the key issues that decide the success of Web services technology. This issue of the International Journal of Web Services Research (IJWSR) highlights the theme of a special issue: “Web Services Discovery and Composition.” Guest Editors M. Brian Blake, William K. Cheung, Andreas Wombacher, and Kwok Ching Tsui organized four articles in this special issue focused primarily on various approaches on services discovery and composition.

The fifth paper, “A Dynamic Two-Phase Commit Protocol for Adaptive Composite Services,” Weihai Yu and Calton Pu propose enhancement to traditional two-phase commit protocol to serve Web services-based applications.

Guest Editorial Preface
M. Brian Blake, Georgetown University, USA
William K. Cheung, Hong Kong Baptist University, Hong Kong
Andreas Wombacher, Ecole Polytechnique Federale de Lausanne, Switzerland
Kwok Ching Tsui, Hong Kong Baptist University & Hong Kong and Shanghai Banking Corp. Ltd., Hong Kong

INTRODUCTION
Web services are at the core of service-oriented architectures (SOA) and the service-oriented computing paradigm (Papazoglou, 2003). Web services can be defined as networked capabilities with openly accessible interfaces such that they can be discovered and executed by other machines in real time. Web service composition involves putting together a chain of multiple, related services.

The process of composing services depends greatly on the ability to determine how the output message names of one service can be mapped to the input message names of other services. Moreover, any composition request contains, at a minimum, predicate conditions that must match to the input message names of the potential initial services of the chain, and the desired results must relate to the outputs of the final services in the chain.
Although there are many solutions to the problem of composition (Koehler & Shristiva, 2003), this notion of sequentially grouping services is similar to the well-established notions of planning and backwards/forward chaining in artificial intelligence literature (Rao & Su, 2004).

The notion of service composition is practical when you consider the example of a travel agent who is scheduling a trip for a client. If the travel agent is able to use a web service to purchase airline tickets then it is likely that there is enough information to submit queries to other applicable services such as hotels, rental cars, and entertainment events. A commonly used example scenario is the travel planning scenario where web services represent flight search, flight reservation, rental car search and reservation. An intelligent software agent can act as a proxy for the human travel agent in providing information to initiate a travel-specific composition routine. As shown in Figure 1., some set of provided information can trigger the composition of a chain of services until the required resultant information is obtained. In each step in the chain, a subset of the provided information in addition to the new output information from any of the preceding underlying services is available for the discovery and execution of the next service to be integrated in the chain.

This special issue explores several approaches to accomplishing this notion of web services discovery and composition. This issue is similar to recent venues that explore implementation issues in the form of a competition (Blake, 2005; WS-Challenge, 2006). In the next section, we discuss the general aspects of service discovery and composition that are independent of implementation details. In the subsequent section, the various contributions of the special issue with respect to the discovery/composition implementation approaches are summarized.

**SERVICE DISCOVERY AND COMPOSITION CHALLENGES**

Although the notion of service discovery and composition seems straightforward, there are several implementation challenges when constructing service management systems. These challenges can be stratified systematically into six areas (summarized in Figure 2.) that represent the tasks within the overall discovery and composition process.

- **Capturing and Loading Composition Constraints.** A human user of an external software module must be able to manage the information that depicts what is provided and what is desired in a composition routine. An accurate description and an effective retrieval method are necessary when managing these requirements and provisions such that the most effective service is discovered from a potentially large repository.

- **Service Discovery and Filtering.** Software modules must be able to exploit composition constraints to generate a set of potential services and then accurately prune the set such that errant services are removed.

- **Identifying the Initial Services in the Chain.** When identifying the initial services for the composition procedure, the composition module must match potential services to the composition provisions. Also, the software module can make some predictions about the search space and perhaps collect candidate services. Additionally, if initial services are not found then the composition routine can be terminated.

- **Identifying Subsequent Services.** As the composition module proceeds through the chain, out messages must match with input messages among services. A syntactical approach may be sufficient within an enterprise with previously-established schemas, but inter-organizational composition will require semantic matching.

- **Choosing the Solution Chain.** A composition routine may identify a number of potential solution chains. In addition, solutions chains can be continually discovered considering the size of the service repository. The composition module has to evaluate efficiency versus quality when determining when to stop building chains and in determining which chain to return from multiple options.

- **Presenting the Solution.** The solution of the composition must be translated into a representation that the business process mediation system can schedule and execute.

**CONTRIBUTIONS WITHIN THE ISSUE**

Articles within this special issue represent a diverse set of approaches for service discovery and
Three articles discuss approaches to discovering and composing services from the service repositories available for the Web Services Challenge (http://ws-challenge.org). The fourth article presents a general approach for querying and composing services from a service registry.

Seog-Chan Oh, Dongwon Lee, and Sundar R.T. Kumara present an approach to service discovery and composition called WSPR. WSPR uses an artificial intelligence planning approach where forward chaining and regression searching techniques are combined. WSPR is implemented using Python. Also using an artificial intelligence planning approach, Jurgen Dorn, Peter Hrastnik, and Albert Rainer use answer set programming (ASP) as the core of their software module. ASP
has traditionally been used for the combinatorial search problems in plan generation. Using the Java programming language and the WSDL4J libraries, composition constraints (message inputs and outputs) are translated into the Datalog with Disjunction (DVL) representation and the planning capabilities of ASP are exploited.

Aoying Zhou, Shen Huang, and Xiaoling Wang take a bottom-up approach. They introduce an approach called Binary Tree Web Service Composition (BITS), where message names are grouped together in a Java-based binary tree representation. BITS contains efficient search strategies to traverse the binary tree for both completeness and optimality. The binary tree serves as an index that facilitates multiple compositions within the same repository. The final article, by Walter Binder, Ion Constantinescu, and Boi Faltings, introduces a declarative query language called DQL that can be used for discovery and composition within a service registry. The language is robust enough to declaratively define actions with embedded efficiency.

The approaches and performance results of the contributions within this issue provide an effective foundation for existing and future systems to evaluate their designs and responsiveness.

ACKNOWLEDGMENT

Michael Fitzgerald Nowlan helped in managing the review process for the articles in this issue. Work towards this special issue was partially funded by the National Science Foundation under award number 0548514.

REFERENCES


Liang-Jie Zhang (zhanglj@us.ibm.com), PhD, is a chief architect of Industrial Standards of IBM Software, the founding chair of the Services Computing Professional Interest Community (PIC), and a research staff member at the IBM T.J. Watson Research Center. He is part of the business informatics research team with a focus on SOA and Web services for industry solutions and business performance management services. He has filed more than 30 patent applications in the areas of e-commerce, Web services, rich media, data management, and information appliances, and he has published more than 80 technical papers in journals, book chapters and conference proceedings. Dr. Zhang is the chair of the IEEE Computer Society’s Technical Committee on Services Computing and the editor-in-chief of the International Journal of Web Services Research (JWSR). In 2005, he served as the general co-chair of the IEEE International Conference on Web Services (ICWS 2005) and the general co-chair of the IEEE Conference on Services Computing (SCC 2005). Liang-Jie received a BS in electrical engineering at Xidian University in 1990, an MS in electrical engineering at Xi’an Jiaotong University in 1992, and a PhD in computer engineering at Tsinghua University in 1996.
M. Brian Blake is an associate professor of software engineering at the Department of Computer Science, Georgetown University, Washington, D.C. He conducts applied research in the area of service-oriented computing, adaptive workflow management, and enterprise integration. He has published over 70 technical papers in the area of enterprise systems integration. He was the program co-chair of the 2006 IEEE International Conference on Services Computing and co-founder of the Web Services Challenge (http://ws-challenge.org).

William K. Cheung is an associate professor in the Department of Computer Science at Hong Kong Baptist University. His research interests include model-based pattern recognition, machine learning, and artificial intelligence with applications to data mining, information extraction, recommender systems, and Web and grid service management. He received his PhD in computer science from the Hong Kong University of Science and Technology.

Andreas Wombacher is a post-doc at the Distributed Information Systems Laboratory at the Ecole Polytechnique Federale de Lausanne (EPFL), Switzerland. His research interest in cross-organizational workflows, service oriented architectures and P2P systems, where he focuses in particular on the decentralization aspects of data and processes. He has published in various journals and conferences and serves in several organization, program and steering committees.

Kwok Ching Tsui is an IT manager at HSBC focusing on data centre infrastructure management. He is also an adjunct assistant professor at the Computer Science Department of the Hong Kong Baptist University. His research interest includes software agent, nature-inspired computation, and e-transformation technology and their practical applications.