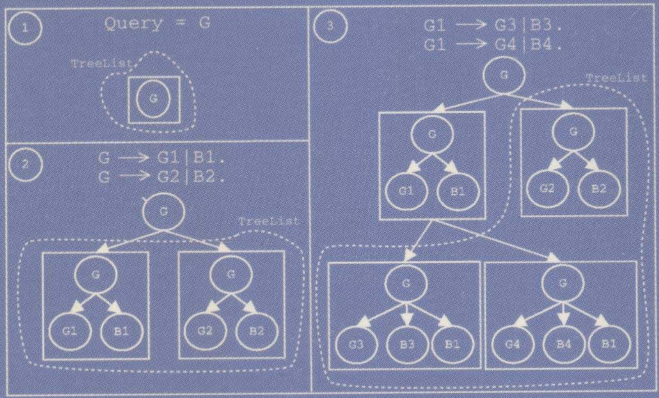
Hot Topics

LNCS 3387

Jorge Cardoso
Amit Sheth (Eds.)

Semantic Web Services and Web Process Composition

First International Workshop, SWSWPC 2004
San Diego, CA, USA, July 2004
Revised Selected Papers



Springer

Jorge Cardoso   Amit Sheth (Eds.)

Semantic Web Services and Web Process Composition

First International Workshop, SWSWPC 2004
San Diego, CA, USA, July 6, 2004
Revised Selected Papers

Springer

Volume Editors

Jorge Cardoso
Universidade da Madeira, Departamento de Matemática e Engenharias
Funchal, 9000-390 Portugal
E-mail: jcardoso@uma.pt

Amit Sheth
University of Georgia, LSDIS Lab, Computer Science Department
415 Boyd GSRC, DW Brooks Dr., UGA, Athens, GA 30602-7404, USA
E-mail: amit@cs.uga.edu

Library of Congress Control Number: 2004117658

CR Subject Classification (1998): H.3, H.4, H.2, I.2, H.5, C.2

ISSN 0302-9743
ISBN 3-540-24328-3 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11376736 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3387

Lecture Notes in Computer Science

For information about Vols. 1–3266

please contact your bookseller or Springer

Vol. 3387: J. Cardoso, A. Sheth (Eds.), Semantic Web Services and Web Process Composition. VIII, 147 pages. 2005.

Vol. 3386: S. Vaudenay (Ed.), Public Key Cryptography - PKC 2005. IX, 436 pages. 2004.

Vol. 3385: R. Cousot (Ed.), Verification, Model Checking, and Abstract Interpretation. XII, 483 pages. 2004.

Vol. 3382: J. Odell, P. Giorgini, J.P. Müller (Eds.), Agent-Oriented Software Engineering V. X, 239 pages. 2004.

Vol. 3381: M. Bieliková, B. Charon-Bost, O. Sýkora, P. Vojtáš (Eds.), SOFSEM 2005: Theory and Practice of Computer Science. XV, 428 pages. 2004.

Vol. 3376: A. Menezes (Ed.), Topics in Cryptology – CT-RSA 2005. X, 385 pages. 2004.

Vol. 3363: T. Eiter, L. Libkin (Eds.), Database Theory - ICDT 2005. XI, 413 pages. 2004.

Vol. 3362: G. Barthe, L. Burdy, M. Huisman, J.-L. Lanet, T. Muntean (Eds.), Construction and Analysis of Safe, Secure, and Interoperable Smart Devices. IX, 257 pages. 2004.

Vol. 3360: S. Spaccapietra, E. Bertino, S. Jajodia, R. King, D. McLeod, M.E. Orlowska, L. Strous (Eds.), Journal on Data Semantics II. XI, 233 pages. 2004.

Vol. 3358: J. Cao, L.T. Yang, M. Guo, F. Lau (Eds.), Parallel and Distributed Processing and Applications. XXIV, 1058 pages. 2004.

Vol. 3357: H. Handschuh, M.A. Hasan (Eds.), Selected Areas in Cryptography. XI, 355 pages. 2004.

Vol. 3356: G. Das, V.P. Gulati (Eds.), Intelligent Information Technology. XII, 428 pages. 2004.

Vol. 3353: J. Hromkovič, M. Nagl, B. Westfechtel (Eds.), Graph-Theoretic Concepts in Computer Science. XI, 404 pages. 2004.

Vol. 3350: M. Hermenegildo, D. Cabeza (Eds.), Practical Aspects of Declarative Languages. VIII, 269 pages. 2004.

Vol. 3348: A. Canteaut, K. Viswanathan (Eds.), Progress in Cryptology - INDOCRYPT 2004. XIV, 431 pages. 2004.

Vol. 3347: R.K. Ghosh, H. Mohanty (Eds.), Distributed Computing and Internet Technology. XX, 472 pages. 2004.

Vol. 3344: J. Malenfant, B.M. Østvold (Eds.), Object-Oriented Technology. ECOOP 2004 Workshop Reader. VIII, 215 pages. 2004.

Vol. 3342: E. Şahin, W.M. Spears (Eds.), Swarm Robotics. X, 175 pages. 2004.

Vol. 3341: R. Fleischer, G. Trippen (Eds.), Algorithms and Computation. XVII, 935 pages. 2004.

Vol. 3340: C.S. Calude, E. Calude, M.J. Dinneen (Eds.), Developments in Language Theory. XI, 431 pages. 2004.

Vol. 3339: G.I. Webb, X. Yu (Eds.), AI 2004: Advances in Artificial Intelligence. XXII, 1272 pages. 2004. (Subseries LNAI).

Vol. 3338: S.Z. Li, J. Lai, T. Tan, G. Feng, Y. Wang (Eds.), Advances in Biometric Person Authentication. XVIII, 699 pages. 2004.

Vol. 3337: J.M. Barreiro, F. Martin-Sanchez, V. Maojo, F. Sanz (Eds.), Biological and Medical Data Analysis. XI, 508 pages. 2004.

Vol. 3336: D. Karagiannis, U. Reimer (Eds.), Practical Aspects of Knowledge Management. X, 523 pages. 2004. (Subseries LNAI).

Vol. 3334: Z. Chen, H. Chen, Q. Miao, Y. Fu, E. Fox, E.-p. Lim (Eds.), Digital Libraries: International Collaboration and Cross-Fertilization. XX, 690 pages. 2004.

Vol. 3333: K. Aizawa, Y. Nakamura, S. Satoh (Eds.), Advances in Multimedia Information Processing - PCM 2004, Part III. XXXV, 785 pages. 2004.

Vol. 3332: K. Aizawa, Y. Nakamura, S. Satoh (Eds.), Advances in Multimedia Information Processing - PCM 2004, Part II. XXXVI, 1051 pages. 2004.

Vol. 3331: K. Aizawa, Y. Nakamura, S. Satoh (Eds.), Advances in Multimedia Information Processing - PCM 2004, Part I. XXXVI, 667 pages. 2004.

Vol. 3329: P.J. Lee (Ed.), Advances in Cryptology - ASIACRYPT 2004. XVI, 546 pages. 2004.

Vol. 3328: K. Lodaya, M. Mahajan (Eds.), FSTTCS 2004: Foundations of Software Technology and Theoretical Computer Science. XVI, 532 pages. 2004.

Vol. 3327: Y. Shi, W. Xu, Z. Chen (Eds.), Data Mining and Knowledge Management. XIII, 263 pages. 2004. (Subseries LNAI).

Vol. 3326: A. Sen, N. Das, S.K. Das, B.P. Sinha (Eds.), Distributed Computing - IWDC 2004. XIX, 546 pages. 2004.

Vol. 3323: G. Antoniou, H. Boley (Eds.), Rules and Rule Markup Languages for the Semantic Web. X, 215 pages. 2004.

Vol. 3322: R. Klette, J. Žunić (Eds.), Combinatorial Image Analysis. XII, 760 pages. 2004.

Vol. 3321: M.J. Maher (Ed.), Advances in Computer Science - ASIAN 2004. XII, 510 pages. 2004.

Vol. 3320: K.-M. Liew, H. Shen, S. See, W. Cai (Eds.), Parallel and Distributed Computing: Applications and Technologies. XXIV, 891 pages. 2004.

Vol. 3317: M. Domaratzki, A. Okhotin, K. Salomaa, S. Yu (Eds.), Implementation and Application of Automata. XII, 336 pages. 2004.

Vol. 3316: N.R. Pal, N.K. Kasabov, R.K. Mudi, S. Pal, S.K. Parui (Eds.), Neural Information Processing. XXX, 1368 pages. 2004.

Vol. 3315: C. Lemaître, C.A. Reyes, J.A. González (Eds.), Advances in Artificial Intelligence – IBERAMIA 2004. XX, 987 pages. 2004. (Subseries LNAI).

Vol. 3314: J. Zhang, J.-H. He, Y. Fu (Eds.), Computational and Information Science. XXIV, 1259 pages. 2004.

Vol. 3312: A.J. Hu, A.K. Martin (Eds.), Formal Methods in Computer-Aided Design. XI, 445 pages. 2004.

Vol. 3311: V. Roca, F. Rousseau (Eds.), Interactive Multimedia and Next Generation Networks. XIII, 287 pages. 2004.

Vol. 3309: C.-H. Chi, K.-Y. Lam (Eds.), Content Computing. XII, 510 pages. 2004.

Vol. 3308: J. Davies, W. Schulte, M. Barnett (Eds.), Formal Methods and Software Engineering. XIII, 500 pages. 2004.

Vol. 3307: C. Bussler, S.-k. Hong, W. Jun, R. Kaschek, D.. Kinshuk, S. Krishnaswamy, S.W. Loke, D. Oberle, D. Richards, A. Sharma, Y. Sure, B. Thalheim (Eds.), Web Information Systems – WISE 2004 Workshops. XV, 277 pages. 2004.

Vol. 3306: X. Zhou, S. Su, M.P. Papazoglou, M.E. Orlowska, K.G. Jeffery (Eds.), Web Information Systems – WISE 2004. XVII, 745 pages. 2004.

Vol. 3305: P.M.A. Sloot, B. Chopard, A.G. Hoekstra (Eds.), Cellular Automata. XV, 883 pages. 2004.

Vol. 3303: J.A. López, E. Benfenati, W. Dubitzky (Eds.), Knowledge Exploration in Life Science Informatics. X, 249 pages. 2004. (Subseries LNAI).

Vol. 3302: W.-N. Chin (Ed.), Programming Languages and Systems. XIII, 453 pages. 2004.

Vol. 3300: L. Bertossi, A. Hunter, T. Schaub (Eds.), Inconsistency Tolerance. VII, 295 pages. 2004.

Vol. 3299: F. Wang (Ed.), Automated Technology for Verification and Analysis. XII, 506 pages. 2004.

Vol. 3298: S.A. McIlraith, D. Plexousakis, F. van Harmelen (Eds.), The Semantic Web – ISWC 2004. XXI, 841 pages. 2004.

Vol. 3296: L. Bougé, V.K. Prasanna (Eds.), High Performance Computing - HiPC 2004. XXV, 530 pages. 2004.

Vol. 3295: P. Markopoulos, B. Eggen, E. Aarts, J.L. Crowley (Eds.), Ambient Intelligence. XIII, 388 pages. 2004.

Vol. 3294: C.N. Dean, R.T. Boute (Eds.), Teaching Formal Methods. X, 249 pages. 2004.

Vol. 3293: C.-H. Chi, M. van Steen, C. Wills (Eds.), Web Content Caching and Distribution. IX, 283 pages. 2004.

Vol. 3292: R. Meersman, Z. Tari, A. Corsaro (Eds.), On the Move to Meaningful Internet Systems 2004: OTM 2004 Workshops. XXIII, 885 pages. 2004.

Vol. 3291: R. Meersman, Z. Tari (Eds.), On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE, Part II. XXV, 824 pages. 2004.

Vol. 3290: R. Meersman, Z. Tari (Eds.), On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE, Part I. XXV, 823 pages. 2004.

Vol. 3289: S. Wang, K. Tanaka, S. Zhou, T.W. Ling, J. Guan, D. Yang, F. Grandi, E. Mangina, I.-Y. Song, H.C. Mayr (Eds.), Conceptual Modeling for Advanced Application Domains. XXII, 692 pages. 2004.

Vol. 3288: P. Atzeni, W. Chu, H. Lu, S. Zhou, T.W. Ling (Eds.), Conceptual Modeling – ER 2004. XXI, 869 pages. 2004.

Vol. 3287: A. Sanfeliu, J.F. Martínez Trinidad, J.A. Carrasco Ochoa (Eds.), Progress in Pattern Recognition, Image Analysis and Applications. XVII, 703 pages. 2004.

Vol. 3286: G. Karsai, E. Visser (Eds.), Generative Programming and Component Engineering. XIII, 491 pages. 2004.

Vol. 3285: S. Manandhar, J. Austin, U.B. Desai, Y. Oyanagi, A. Talukder (Eds.), Applied Computing. XII, 334 pages. 2004.

Vol. 3284: A. Karmouch, L. Korba, E.R.M. Madeira (Eds.), Mobility Aware Technologies and Applications. XII, 382 pages. 2004.

Vol. 3283: F.A. Aagesen, C. Anutariya, V. Wuwongse (Eds.), Intelligence in Communication Systems. XIII, 327 pages. 2004.

Vol. 3282: V. Guruswami, List Decoding of Error-Correcting Codes. XIX, 350 pages. 2004.

Vol. 3281: T. Dingsøyr (Ed.), Software Process Improvement. X, 207 pages. 2004.

Vol. 3280: C. Aykanat, T. Dayar, İ. Körpeoğlu (Eds.), Computer and Information Sciences - ISCIS 2004. XVIII, 1009 pages. 2004.

Vol. 3279: G.M. Voelker, S. Shenker (Eds.), Peer-to-Peer Systems III. XI, 300 pages. 2004.

Vol. 3278: A. Sahai, F. Wu (Eds.), Utility Computing. XI, 272 pages. 2004.

Vol. 3275: P. Perner (Ed.), Advances in Data Mining. VIII, 173 pages. 2004. (Subseries LNAI).

Vol. 3274: R. Guerraoui (Ed.), Distributed Computing. XIII, 465 pages. 2004.

Vol. 3273: T. Baar, A. Strohmeier, A. Moreira, S.J. Mellor (Eds.), <<UML>> 2004 - The Unified Modelling Language. XIII, 454 pages. 2004.

Vol. 3272: L. Baresi, S. Dustdar, H. Gall, M. Matera (Eds.), Ubiquitous Mobile Information and Collaboration Systems. VIII, 197 pages. 2004.

Vol. 3271: J. Vicente, D. Hutchison (Eds.), Management of Multimedia Networks and Services. XIII, 335 pages. 2004.

Vol. 3270: M. Jeckle, R. Kowalczyk, P. Braun (Eds.), Grid Services Engineering and Management. X, 165 pages. 2004.

Vol. 3269: J. Lopez, S. Qing, E. Okamoto (Eds.), Information and Communications Security. XI, 564 pages. 2004.

Vol. 3268: W. Lindner, M. Mesiti, C. Türker, Y. Tzitzikas, A. Vakali (Eds.), Current Trends in Database Technology - EDBT 2004 Workshops. XVIII, 608 pages. 2004.

Vol. 3267: C. Priami, P. Quaglia (Eds.), Global Computing. VIII, 377 pages. 2004.

Preface

This book constitutes the refereed proceedings of the 1st International Workshop on Semantic Web Services and Web Process Composition, SWSWPC 2004, held at the Westin Horton Plaza Hotel, San Diego, California, USA, July 6, 2004, in conjunction with the IEEE International Conference on Web Services (ICWS 2004).

The workshop intended to bring researchers, scientists from both industry and academics, and representatives from different communities together to study, understand, and explore the phases that compose the lifecycle of Semantic Web processes. The workshop presented what can be achieved by the symbiotic synthesis of two of the hottest R&D and technology application areas, Web services and the Semantic Web, as recognized at the 12th International World Wide Web conference (WWW 2003) and in the industry press.

The emphasis of the workshop was mainly on Web services, Web processes and semantics which are important movements emerging in the World Wide Web. Web services and Web processes promise to ease several current infrastructure challenges, such as data, application, and process integration. Web services are truly platform-independent and allow the development of distributed, loosely coupled applications, a key characteristic for the success of dynamic Web processes.

The 9 revised full papers presented were carefully reviewed and selected from 20 submissions, after a double-blind review process. In addition, we were honored by the presence of two distinguished invited speakers, namely Prof. Munindar Singh (North Carolina State University, USA) and Prof. Boualem Benatallah (University of New South Wales, Australia). The workshop also included a panel entitled "Do Academic Research and Industry Differ in the Role and Approach to the Use of Semantics for Web Processes?" where John Miller (University of Georgia), Jeff Pollock (Network Inference Ltd., USA), Jianwen Su (UC, Santa Barbara, USA), Ryusuke Masuoka (Fujitsu, USA), and Shishir Garg (France Telecom, France) participated.

We would like to express our sincere gratitude to all the authors, who provided the rich material discussed at the workshop, and the members of the Program Committee who reviewed and assessed the scientific merit of each submitted paper, thus ensuring high quality standards.

July 2004

Jorge Cardoso
Amit Sheth

Organization

SWSWPC 2004 was organized by the Department of Mathematics and Engineering, University of Madeira, Portugal and by the Department of Computer Science, University of Georgia, USA, in cooperation with the 2004 IEEE International Conference on Web Services (ICWS 2004).

Program Committee

Conference Chair	Jorge Cardoso (University of Madeira, Portugal)
Program Chairs	Jorge Cardoso (University of Madeira, Portugal) Amit Sheth (University of Georgia, USA)
Organizing Chairs	Jorge Cardoso (University of Madeira, Portugal) Amit Sheth (University of Georgia, USA) Leonid A. Kalinichenko (Russian Academy of Sciences, Russia) Francisco Curbera (IBM, USA)

Referees

R. Akkiraju	L. Kalinichenko	D. Plexousakis
C. Bussler	M. Little	A. Sheth
J. Cardoso	R. Masuoka	S. Staab
F. Curbera	J. Mischkinsky	R. Studer
S. Damodaran	M. Nez	S. Thatte
A. Dogac	L. Obrst	K. Verma
S. Garg	M. Paolucci	

Sponsoring Institutions

Network Inference, Inc., USA (http://www.networkinference.com/)

Table of Contents

Introduction

Panel

Talk

Full Papers

Introduction to Semantic Web Services and Web Process Composition

Jorge Cardoso[1] and Amit Sheth[2]

[1] Departement of Mathematics and Engineering,
University of Madeira, Funchal, Portugal
jcardoso@uma.pt
[2] Large Scale Distributed Information Systems (LSDIS) Lab,
Department of Computer Science,
University of Georgia, GA, USA
amit@cs.uga.edu

Abstract. Systems and infrastructures are currently being developed to support Web services. The main idea is to encapsulate an organization's functionality within an appropriate interface and advertise it as Web services. While in some cases Web services may be utilized in an isolated form, it is normal to expect Web services to be integrated as part of Web processes. There is a growing consensus that Web services alone will not be sufficient to develop valuable Web processes due the degree of heterogeneity, autonomy, and distribution of the Web. Several researchers agree that it is essential for Web services to be machine understandable in order to support all the phases of the lifecycle of Web processes. This paper deals with two of the hottest R&D and technology areas currently associated with the Web — Web services and the Semantic Web. It presents how applying semantics to each of the steps in the Semantic Web Process lifecycle can help address critical issues in reuse, integration and scalability.

1 Introduction

E-commerce and e-services have been growing at a very fast pace. The Web coupled with e-commerce and e-services is enabling a new networked economy [1]. The scope of activities that processes span has moved from intra-enterprise workflows, predefined inter-enterprise and business-to-business processes, to dynamically defined Web processes among cooperating organizations.

There is a remarkable range for growth in trade through electronic interactions, simply because it can eliminate geographical distances in bringing buyers and sellers together. With the Internet dissemination and the e-commerce growth there is a shift from the traditional off-line distribution process based on organization's catalogs to on-line services. A shift that is marked by isolated initiatives guided by the business-to-customer and business-to-business promise of increased profit margins and reduced commission values. This leads us to the present situation where we can find diverse and numerous groups of on-line systems, most of them focused in one or in a few types of products. Therefore, organizations are increasingly faced with the challenge

J. Cardoso and A. Sheth (Eds.): SWSWPC 2004, LNCS 3387, pp. 1 – 13, 2005.

of managing e-business systems and e-commerce applications managing Web services, Web processes, and semantics. Web services promise universal interoperability and integration. The key to achieving this relies on the efficiency of discovering appropriate Web services and composing them to build complex processes. We will start this section by explaining what semantics are and their role and relationships with ontologies. We then explain the purpose of each of the Web process lifecycle phases.

2 Semantic Web Process Lifecycle

Semantic Web services will allow the semi-automatic and automatic annotation, advertisement, discovery, selection, composition, and execution of inter-organization business logic, making the Internet become a global common platform where organizations and individuals communicate among each other to carry out various commercial activities and to provide value-added services.

In order to fully harness the power of Web services, their functionality must be combined to create Web processes. Web processes allow representing complex interactions among organizations, representing the evolution of workflow technology. Semantics can play an important role in all stages of Web process lifecycle. The main stages of the Web process lifecycle are illustrated in Figure 1.

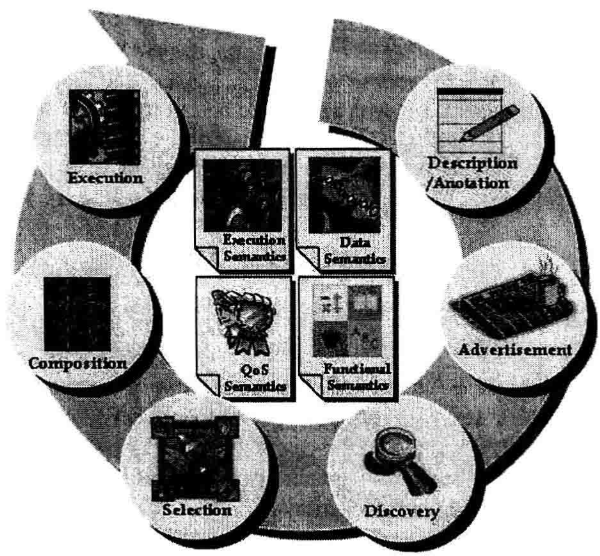


Fig. 1. Web process lifecycle and semantics.

The lifecycle of semantic Web processes includes the description/annotation, the advertisement, the discovery, the selection, the composition of Web services that makeup Web processes, and the execution of Web processes. All these stages are significant for the Web process lifecycle and their success.

2.1 Semantics and Ontologies

There is a growing consensus that Web services alone will not be sufficient to develop valuable and sophisticated Web processes due the degree of heterogeneity, autonomy, and distribution of the Web. Several researchers agree that it is essential for Web services to be machine understandable in order to allow the full deployment of efficient solutions supporting all the phases of the lifecycle of Web processes.

The idea and vision of the "Semantic Web" [2] catches on and researchers as well as companies have already realized the benefits of this great vision. Ontologies [3] are considered the basic building block of the Semantic Web as they allow machine supported data interpretation reducing human involvement in data and process integration.

An ontology "is a formal, explicit specification of a shared conceptualization. *Conceptualization* refers to an abstract model of phenomena in the world by having identified the relevant concepts of those phenomena. *Explicit* means that the type of concepts used, and the constraints on their use are explicitly defined. *Formal* refers to the fact that the ontology should be machine readable. *Shared* reflects that ontology should capture consensual knowledge accepted by the communities" [4].

When the knowledge about a domain is represented in a declarative language, the set of objects that can be represented is called the universe of discourse. We can describe the ontology of a program by defining a set of representational terms. Definitions associate the names of entities in the universe of discourse (e.g. classes, relations, functions or other objects) with human-readable text describing what the names mean and formal axioms that constrain the interpretation and well-formed use of these terms.

A set of Web services that share the same ontology will be able to communicate about a domain of discourse. We say that a Web service commits to an ontology if its observable actions are consistent with the definitions in the ontology.

Example: Benefits of Ontologies for the Travel Industry. The Web has permanently changed the manner travel packages can be created. Consumers can now acquire packages from a diversity of Web sites including online agencies and airlines. With the spread of Web travel, a new technology has surfaced for the leisure travel industry: dynamic packaging. For the development of dynamic packaging solutions it is necessary to look in detailed at the technology components needed to enhance the online vacation planning experience. By transitioning from a third-party service in most markets, dynamic packaging engines can better tailor its package offerings, pricing and merchandising to consumer demand.

Currently, the travel industry has concentrated their efforts on developing open specifications messages, based on eXtensible Markup Language (XML), to ensure that messages can flow between industry segments as easily as within. For example, the OpenTravel Alliance (OTA) [5] is an organization pioneering the development and use of specifications that support e-business among all segments of the travel industry. The cumulative effort of various teams, individuals, associations, companies, and international organizations, including air, car, cruise, rail, hotel, travel agencies, tour operators and technology providers, has produced a fairly complete set of XML-based specifications for the travel industry (more than 140 XML specification files exist).

The current development of open specifications messages based on XML, such as the OTA schema, to ensure the interoperability between trading partners and working groups is not sufficiently expressive to guaranty an automatic exchange and processing of information. The development of a suitable ontology for the tourism industry is indispensable and will serve as a common language for travel-related terminology and a mechanism for promoting the seamless exchange of information across all travel industry segments.

The development of such an ontology can be used to bring together autonomous and heterogeneous Web services, Web processes, applications, data, and components residing in distributed environments. Semantics allow rich descriptions of Web services and Web processes that can be used by computers for automatic processing in various tourism related applications. The deployment of ontologies help articulate a well-defined set of common data elements or vocabulary that can support communication across multiple channels, expedite the flow of information, and meet travel industry and customer needs.

For the travel industry, the simplest form to construct an ontology is to retrieve rich semantic interrelationships from the data and terminology present in the XML-based OTA specifications already implemented [5] and available to organizations. This procedure is illustrated in Figure 2.

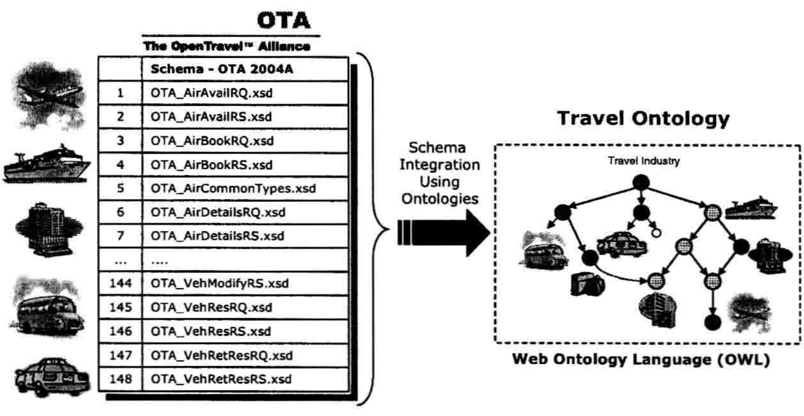


Fig. 2. Ontology for the travel industry

One possible language to construct such an ontology is using the Web Ontology Language (OWL) [6] designed by the World Wide Web Consortium (W3C). The OWL is designed for use by applications that need to process the content of information instead of just presenting information to humans. OWL facilitates greater machine interpretability of Web content by providing additional vocabulary along with a formal semantics. It can be used to explicitly represent the meaning of terms in vocabularies and the relationships between those terms.

OWL is appropriate to develop an ontology for the travel industry since it is intended to be used when the information used by Web services needs to be processed by applications, as opposed to situations where the content only needs to be presented to humans.

The development of such an ontology lead to the spearhead and foster the cross-industry consensus needed to establish and maintain the most effective and widely used specifications designed to electronically exchange business data and information among all sectors of the travel industry.

This effort represents what can be achieved by the symbiotic synthesis of two of the hottest R&D and technology application areas: Web services and the semantic Web, as recognized at the Thirteenth International World Wide Conference (2004) and in the industry press. The intelligent combination of Web services and the semantic Web can start off a technological revolution with the development of semantic Web processes [7]. These technological advances can ultimately lead to a new breed of Web-based applications for the travel industry.

2.2 Semantics for Web Services

In Web services domain, semantics can be classified into the following types [8] illustrated in Figure 1:

- Functional Semantics
- Data Semantics
- QoS Semantics and
- Execution Semantics

These different types of semantics can be used to represent the capabilities, requirements, effects and execution of a Web service. In this section we describe the nature of Web services and the need for different kind of semantics for them.

Functional Semantics. The power of Web services can be realized only when appropriate services are discovered based on the functional requirements. It has been assumed in several semantic Web service discovery algorithms [9] that the functionality of the services is characterized by their inputs and outputs. Hence these algorithms look for semantic matching between inputs and outputs of the services and the inputs and outputs of the requirements. This kind of semantic matching may not always retrieve an appropriate set of services that satisfy functional requirements. Though semantic matching of inputs and outputs are required, they are not sufficient for discovering relevant services. For example, two services can have the same input/output signature even if they perform entirely different functions. A simple mathematical service that performs addition of two numbers taking the numbers as input and produce the sum as output will have the same semantic signature as that of another service that performs subtraction of two numbers that are provided as input and gives out their difference value as output. Hence matching the semantics of the service signature may result in high recall and low precision. As a step towards representing the functionality of the service for better discovery and selection, the Web services can be annotated with functional semantics. It can be done by having an ontology called Functional Ontology in which each concept/class represents a well-defined functionality. The intended functionality of each service can be represented as annotations using this ontology.

Data Semantics. All the Web services take a set of inputs and produce a set of outputs. These are represented in the signature of the operations in a specification file. However the signature of an operation provides only the syntactic and structural details of the input/output data. These details (like data types, schema of a XML complex type) are used for service invocation. To effectively perform discovery of services, semantics of the input/output data has to be taken into account. Hence, if the data involved in Web service operation is annotated using an ontology, then the added data semantics can be used in matching the semantics of the input/output data of the Web service with the semantics of the input/output data of the requirements. Semantic discovery algorithm proposed in [9] uses the semantics of the operational data.

QoS Semantics: After discovering Web services whose semantics match the semantics of the requirements, the next step is to select the most suitable service. Each service can have different quality aspect and hence service selection involves locating the service that provides the best quality criteria match. Service selection is also an important activity in web service composition [10]. This demands management of QoS metrics for Web services. Web services in different domains can have different quality aspects. For organizations, being able to characterize Web processes based on QoS has several advantages: a) it allows organizations to translate their vision into their business processes more efficiently, since Web processes can be designed according to QoS metrics, b) it allows for the selection and execution of Web processes based on their QoS, to better fulfill customer expectations, c) it makes possible the monitoring of Web processes based on QoS, and d) it allows for the evaluation of alternative strategies when Web process adaptation becomes necessary.

Execution Semantics. Execution semantics of a Web service encompasses the ideas of message sequence, conversation pattern of Web service execution, flow of actions, preconditions and effects of Web service invocation, etc. Some of these details may not be meant for sharing and some may be, depending on the organization and the application that is exposed as a Web service. In any case, the execution semantics of these services are not the same for all services and hence before executing or invoking a service, the execution semantics or requirements of the service should be verified.

Some of the issues and solutions with regard to execution semantics are inherited from traditional workflow technologies. However, the globalization of Web services and processes result in additional issues. In e-commerce, using execution semantics can help in dynamically finding partners that will match not only the functional requirements, but also the operational requirements like long running interactions and complex conversations. Also, a proper model for execution semantics will help in coordinating activities in transactions that involve multiple parties.

3 Phases of the Web Process Lifecycle

As stated previously, the lifecycle of semantic Web processes includes the description/annotation, the advertisement, the discovery, and the selection of Web services, the composition of Web services that makeup Web processes, and the execution of Web processes. In this section, we discuss the characteristics of each of these stages.

3.1 Semantic Web Service Annotation

Today, Web service specifications are based on standards that only define syntactic characteristics. Unfortunately, it is insufficient, since the interoperation of Web services/processes cannot be successfully achieved. One of the most recognized solutions to solve interoperability problems is to enable applications to understand methods and data by adding meaning to them.

Many tools are available to create Web services. Primarily programs written in Java or any object oriented language can be made into Web services. In technical terms any program that can communicate with other remote entities using SOAP [11] can be called a Web service. Since the development of Web services is the first stage in the creation of Web services, it is very important to use semantics at this stage. During Web service development data, functional and QoS semantics of the service needs to be specified.

All the Web services (operations in WSDL file [12]) take a set of inputs and produce a set of outputs. These are represented in the signature of the operations in a WSDL file. However the signature of an operation provides only the syntactic and structural details of the input/output data.

To effectively perform operations such as the discovery of services, semantics of the input/output data has to be taken into account. Hence, if the data involved in Web service operation is annotated using an ontology, then the added data semantics can be used in matching the semantics of the input/output data of the Web service with the semantics of the input/output data of the requirements.

The Meteor-S Web Service Annotation Framework (MWSAF) [13] provides a framework and a tool to achieve automatic and semi-automatic annotation of web services using ontologies.

Figure 3 illustrates one solution to annotate WSDL interfaces with semantic metadata based on relevant ontologies [14]. A Web service invocation stipulate an input interface that specifies the number of input parameters that must be supplied for a proper Web service realization and an output interface that specifies the number of outputs parameters to hold and transfer the results of the Web service realization to other services.

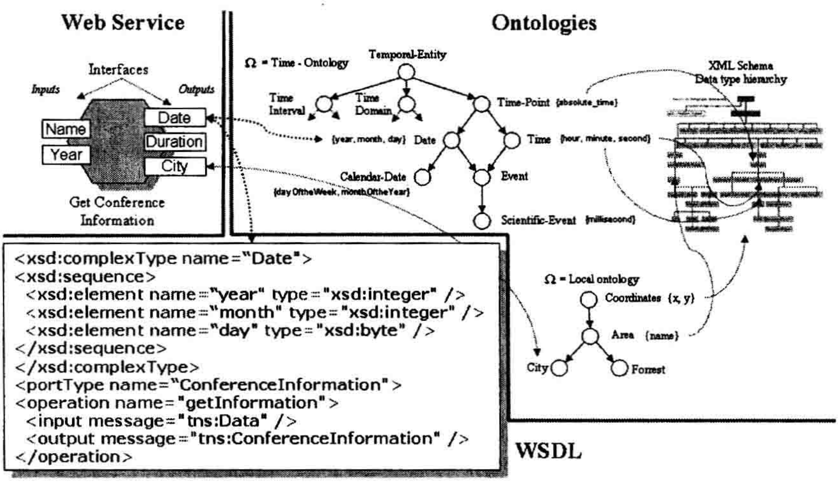


Fig. 3. Semantic annotation of a Web service specified with WSDL

3.2 Semantic Web Service Advertisement

After the service is developed and annotated, it has to be advertised to enable discovery. The UDDI registry is supposed to open doors for the success of service oriented