Encyclopedia of E-Business Development and Management in the Global Economy

In Lee
Western Illinois University, USA

Volume III

BUSINESS SCIENCE REFERENCE
Hershey · New York

Director of Editorial Content: Kristin Klinger
Director of Book Publications: Julia Mosemann
Acquisitions Editor: Lindsay Johnston
Development Editor: Julia Mosemann
Publishing Assistant: Sean Woznicki
Typesetter: Deanna Zombro, Callie Klinger
Quality Control: Jamie Snavely
Cover Design: Lisa Tosheff
Printed at: Yurchak Printing Inc.

Published in the United States of America by
Business Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com/reference

Copyright © 2010 by IGI Global. All rights reserved. No part of this publication may be reproduced, stored or distributed in any form or by any means, electronic or mechanical, including photocopying, without written permission from the publisher.

Product or company names used in this set are for identification purposes only. Inclusion of the names of the products or companies does not indicate a claim of ownership by IGI Global of the trademark or registered trademark.

Library of Congress Cataloging-in-Publication Data

Encyclopedia of e-business development and management in the global economy / In Lee, editor.
p. cm.

Includes bibliographical references and index.
Summary: "This research book is a repository for academicians, researchers, and industry practitioners to share and exchange their research ideas, theories, and practical experiences, discuss challenges and opportunities, and present tools and techniques in all aspects of e-business development and management in the digital economy"--Provided by publisher.

ISBN 978-1-61520-611-7 (hardcover) -- ISBN 978-1-61520-612-4 (ebook) 1. Electronic commerce. 2. Electronic commerce--Management. I. Lee, In, 1958-
HF5548.32.E518 2010
658.8'7203--dc22

2009036871

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

All work contributed to this book is new, previously-unpublished material. The views expressed in this book are those of the authors, but not necessarily of the publisher.

Chapter 87
Case Based Web Services

Zhaohao Sun
University of Ballarat, Australia

Gavin Finnie
Bond University, Australia

John Yearwood
University of Ballarat, Australia

INTRODUCTION

Web services are Internet-based application components published using standard interface description languages and universally available via uniform communication protocols (Singh & Huhns, 2005). Web services can be also considered the provision of services over electronic networks such as the Internet and wireless networks (Rust & Kannan, 2003). Web services is a new computing paradigm that has drawn increasing attention in information technology (Deitel, et al, 2004, p.13), information systems, and is playing a pivotal role in service computing and service intelligence (Singh & Huhns, 2005). Web services is a new business paradigm that is playing an important role in e-business, e-commerce and business intelligence (Wang, et al, 2006). The key motive for the rapid development of web services is the ability to discover services that fulfil users' demands, negotiate service contracts and have the services delivered where and when the users request them (Tang, et al, 2007). The current research trend is to add intelligent techniques to web services to facilitate discovery, invocation, composition, and recommendation of web services (Wang, et al, 2006).

Case-based reasoning (CBR) is an artificial intelligence technique that solves problems by reasoning from a case base of previously solved cases, either by finding an exact solution to a previously solved problem or by adapting one or more past solutions (Kolodner, 1994). The most similar set of cases to the current problem will be extracted from

DOI: 10.4018/978-1-61520-611-7.ch087

Copyright © 2010, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.

the case base (Sun & Finnie, 2004). CBR as an intelligent technique has found many successful applications in e-business (Sun & Finnie, 2004) and web services (Lu, et al, 2007), especially in service retrieval, discovery, brokering, composition and recommendation of web services (Yao, 2006). For example, Ladner et al (2008) use a case-based classifier for web services discovery. However, what is the relationship between CBR and web services? How can CBR be applied to the main activities of a web service lifecycle? These problems still remain open. This article will address the above mentioned problems by proposing a unified CBR approach for the main activities of the web service lifecycle. To this end, the remainder of this article is organized as follows: It first looks at web service lifecycle from a requester's demand perspective. Then it proposes CWSR: a case-based web service reasoner. It examines the correspondence relationship between web services and CBR and provides a unified treatment for case-based web service discovery, composition and recommendation. It also looks at some future research directions. The final section ends the article with some concluding remarks and future work.

WEB SERVICE LIFECYCLE: A WEB SERVICE REQUESTER'S PERSPECTIVE

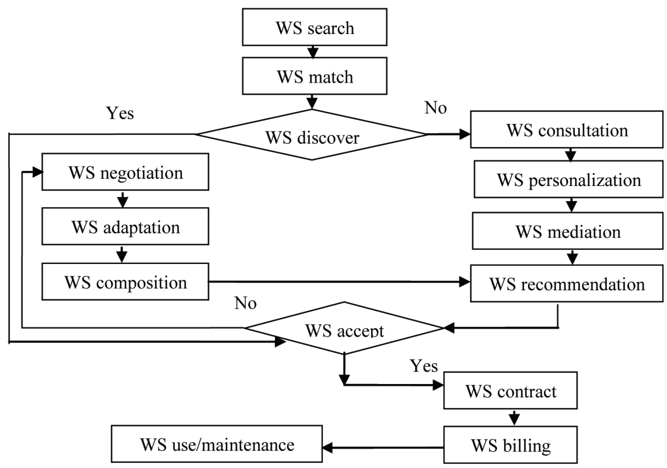
From the perspective of computer science (Pressman, 2001), the software development lifecycle (SDLC) describes the life of a software product from its conception, to its implementation, delivery, use, and maintenance (Pfleeger & Atlee, 2006). A traditional SDLC mainly consists of seven stages: planning, requirements analysis, systems design, coding, testing, delivery and maintenance. Based on this, a web service lifecycle (WSLC) consists of the start of a web service (WS) request (He, et al, 2004) and the end of the WS transaction as well as its evolutionary stages that transform a web service from the start of the request to the end of the transaction.

There have been many attempts to address the WSLC in the web service community. For example, Sheth (2003) proposes a semantic web process lifecycle that consists of web description, discovery, composition and execution or orchestration. Zhang and Jeckle (2003) propose a WSLC that consists of WS modelling, development, publishing, discovery, composition, collaboration, monitoring and analytical control from a perspective of developers. Kwon (2003) proposes a WSLC consisting of four fundamental steps: WS identification, creation, use and maintenance. Narendra and Orriens (2006) consider the WSLC consisting of WS composition, execution, midstream adaptation, and re-execution. Tsalgatidou and Pilioura (2002) propose a WSLC consisting of two different layers: a basic layer and a value-added layer. The former contains WS creation, description, publishing, discovery, invocation and unpublishing. The latter contains the value-added activities of composition, security, brokering, reliability, billing, monitoring, transaction handling and contracting. They acknowledge that some of these activities take place at the WS requester's site, whereas others take place at the WS broker's or provider's site. However, they have not classified the proposed activities based on the WS requester, provider, and broker in detail.

Demand is an important factor for market and economy development (Jackson & McIver, 2004). The demand of WS requesters or customers is the significant force for promoting the research and development of web services. In what follows, we will examine a WSLC from a WS requester's demand perspective.

As a WS requester, he (for brevity, we use he to represent she or he) usually searches, matches web services to meet his demands. For example, if he

Figure 1. A requester's demand-driven web service lifecycle



pays the car registration fee to VicRoad, Australia, he uses Google to search and match "VicRoad" and its web services for car registration. After he discovers a web service that meets his demands, he can pay his registration fee online. Based on this consideration, we can see that WS search, matching and discovery (Ladner, 2008; Tang, 2007) are the common demands of ordinary customers for web services. However, if the service requester cannot discover a satisfactory web service by himself, he has to ask an intermediary or agent for help with providing consultation, mediation and recommendation of WS to her/him. If the agent recommends some web services to the requester after consultation and mediation, the requester accepts one of the recommended WS after evaluation, the WS consultation, mediation (Ladner, 2008) and recommendation are completed. Otherwise, the requester asks the agent to compose the web services to meet his demands. In this case, the agent will negotiate with the requester over the price for composing web services, because he needs WS adaptation. After a successful negotiation, the agent recommends the composite WS to the requester. If the requester accepts the recommended composite WS, then WS composition, adaptation, mediation and recommendation are successful. In practice, the requester also demands personalization, contracting and billing for commercial WS. Therefore, we can illustrate a demand-driven WSLC for WS requesters using Figure 1. This WSLC consists of many activities of web services such as WS search, matching, discovery, adaptation, use/reuse, consultation, personalization, composition, recommendation, negotiation, contracting and billing. All of these have drawn some attention in web services (Singh & Huhns, 2005). In what follows, we only review web service discovery, composition and recommendation in some detail.

Web service (WS) discovery is a process of finding the most appropriate web service for a WS requester (Singh & Huhns, 2005). It identifies a new web service and detects an update to a previously discovered web service (Ladner, 2008). There have been a variety of techniques developed for WS discovery. For example, OWL-S (of W3C) provides classes that describe what the service does, how to ask for the service, what happens when the service is carried out, and how the service can be accessed (Ladner, 2008).

Web service (WS) composition primarily

concerns requests of WS users that cannot be satisfied by any available web services (Narendra & Orriens, 2006). WS composition also refers to the process of creating personalized services from existing services by a process of dynamic discovery, integration and execution of those services in order to satisfy user requirements (Limthanmaphon & Zhang, 2003). WS composition is an important topic for service computing, because composing web services to meet the requirement of the WS requester is one of the most important issues for WS providers and brokers.

Web service (WS) recommendation aims to help WS requesters with selecting web services more suitable to their needs (Lorenzi & Ricci, 2005). WS recommendation can be improved through optimization, analysis, forecasting, reasoning and simulation (Kwon, 2003). Recommender systems have been studied and developed in e-commerce, e-business and multiagent systems (Lorenzi & Ricci, 2005; Sun & Finnie, 2005). Sun and Lau (2007) examine case based web service recommendation. However, how to integrate WS discovery, composition, and recommendation in a unified way is still a big issue for web services. This article will address this issue in later section in more detail.

CWSR: A CASE BASED WEB SERVICES REASONER

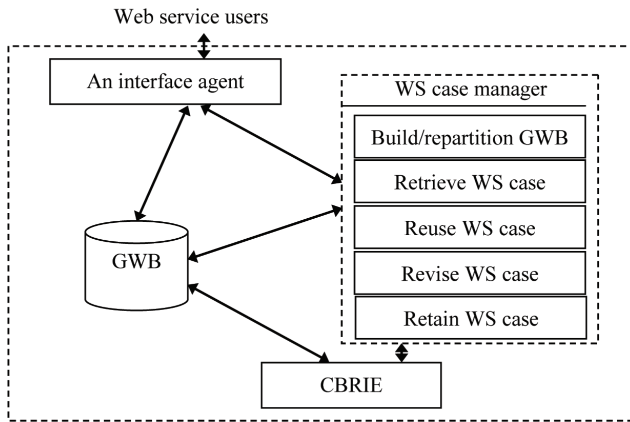
Case-based reasoning (CBR) is a reasoning paradigm based on previous experiences or cases; that is, a CBR system solves new problems by adapting solutions that were used to successfully solve old problems (Ladner, 2008; Sun & Finnie, 2005). Ladner et al (2008) use case-based classification for WS discovery by applying CBR to supervised classification tasks. Kwon (2003) examines how to find the most similar WS case among cases using CBR. Limthanmaphon and Zhang (2003) examine composition of web services using CBR and present a model of web service composition. CBR has been successful in making recommendation of business activities such as in e-commerce to recommend different e-services with high quality of service (QoS) (Sun & Finnie, 2005; Wang & Zhang, 2007). However, how to discover, compose and recommend web services in a unified way remains open for CBR research. To address this issue, we first propose a case based web services reasoner: CWSR. The system architecture of the CWSR mainly consists of an interface agent, a global web service base (GWB), a WS base manager and a CBR inference engine (CBRIE), as shown in Figure 2. The interface agent consists of some kinds of natural language processing systems that allow the user to interact with the web service strategies (Sun & Lau, 2007). The GWB consists of all the WS cases that the system collects periodically and new web service cases discovered when the system is running. The WS base manager is responsible for WS case retrieval, reuse, revise and retention. The CBRIE consists of the mechanism for manipulating the WS cases in GWB to infer web service X based on CBR (where X denotes discovery, composition, recommendation, etc) requested by the user. In order to implement the system architecture of the CWSR, we discuss why CBR can be applied in web services in the next section.

Web Services vs Case Based Reasoning

This section provides a correspondence relationship between the main activities of CBR and those of web services.

A case in CBR can be denoted as $c = (p, q)$, where p is the structured problem description and q is the solution description (Sun & Finnie, 2005). In web services, a service case base stores the collection of service cases (Limthanmaphon & Zhang, 2003). A service case, $w = (d, s)$, consists of the service description d and its service solution s as well as other information including functionally dependency among web services (Kwon, 2003).

Figure 2. A general architecture of the CWSR



The service description corresponds to the requirement of the service user, while the service solution corresponds to the answer to the requirement. In this way, a WS case in web services corresponds to a case in CBR.

When service definitions change or new providers and services are registered within the web services platform such as CWSR, the services need to be adaptive to the change in the environment with minimal user intervention, in order to manage and even take advantage of the frequent changes in the service environment (Dustdar & Schreiner, 2005). In other words, WS adaptation is necessary for web services. In fact, case retrieval, reuse, revise (adaptation) and retention constitute the basic activities of CBR (Sun & Finnie, 2005). WS retrieval (search), reuse, adaptation, and retention in web services can then correspond to the activities of CBR. Therefore, CBR can be used for processing WS retrieval, reuse, adaptation, and retention. This implies that CBR is naturally applicable to web services. This is why CBR has been successfully applied to WS discovery, search and matching (Ladner, 2008). It is significant to apply CBR to the activities of web services such as service discovery, composition and recommendation, which will be examined in the next section.

A UNIFIED TREATMENT OF CASE BASED WEB SERVICES

This section provides a unified treatment for case based web services in a context of the CWSR, based on the previously mentioned requester's demand-driven WSLC.

The WS user's demand is normalized into a structured service description p'. Then the CWSR uses its similarity metric mechanism to retrieve from its GWB, which consists of service cases, each of them is denoted as $c=(p,q)$, where p is the structured service description and q is the service solution description. The inference engine of the CWSR performs similarity-based reasoning that can be formalized as (Finnie & Sun, 2004):

$$\frac{P', P' \approx P, P \rightarrow Q, Q \approx Q'}{\therefore Q'} \tag{1}$$

where P, P', Q, and Q' represent fuzzy compound propositions, $P' \approx P$ means that P and P' are similar. Q and Q' are also similar.

The service case retrieval process from WS search and matching is used to discover the following service cases from the GWB in the CWSR (Finnie & Sun, 2005):

$$C(p')=\{c|c=(p,q),p\approx p'\}=\{c_1,c_2,\ldots,c_n\} \quad (2)$$

This is the result of *case based web service discovery*, where n is a positive integer, c_i, $i = 1, 2, \ldots, n$ are all service cases with their demand description p similar to the current demand description p'. Usually, $C(p')=\{c_1,c_2,\ldots,c_n\}$ satisfies the following property: for any integer i, $1\leq i<n$ and $c_i=(p_i,q_i)$,

$$s(p_i,p')\geq s(p_{i+1},p') \quad (3)$$

where $s(.)$ is a similarity metric, which measures the similarity between one service demand and another.

If n is small, the CWSR will directly recommend the WS solutions of $\{c_1,c_2,\ldots,c_n\}$, $\{q_1,q_2,\ldots,q_m\}$, to the WS requester through the interface agent. If n is very large, the CWSR will recommend the WS solutions of the first m cases of $\{c_1,c_2,\ldots,c_n\}$; that is, $\{q_1,q_2,\ldots,q_m\}$, to the requester, where $1\leq m<n$. This process is *case-based web service recommendation* (Sun & Lau, 2007).

After obtaining the recommended web services from the CWSR, the WS requester will evaluate them and select one of the following:

1. Accept one of the recommended web services, q_k, and contract it, where $1\leq k\leq m$.
2. Adjust his demand descriptions p' and then send them to the CWSR.
3. Reject the recommended e-services and leave the CWSR.

It is obvious that only the first two of these three choices require further discussion. For the first choice, the deal was successfully done and the CWSR routinely updates the service case $c_k=(p_k,q_k)$ in the GWB. At the same time, the CWSR has reused the service case successfully; that is, CWSR completes the process of *case-based web service use* and *reuse*. For the second choice, the demand adjustment is the process of demand adaptation that corresponds to problem adaptation (Sun & Finnie, 2005). After having adjusted the demand, the requester submits it to the CWSR, which will conduct WS retrieval, recommendation and reuse again. This process is *case-based web service adaptation*.

If the WS adaptation is unsuccessful, the CWSR has to conduct *case based web service composition*. Assume that the WS requester's demand is normalized into a structured service description and service solution description $c=(p',q')$, and the CWSR has discovered m web services $\{c_1,c_2,\ldots,c_m\}$ (where m is the least positive number) such that

$$p'\subseteq p_1\cup p_2\cup\ldots\cup p_m \text{ and } q'\subseteq q_1\cup q_2\cup\ldots\cup q_m \quad (4)$$

where $\cup$ is the union operation of the set theory. This is a necessary condition for case based web service composition. Based on (4), the composite web service case $c=(p,q)$ is obtained through *case based web service composition* of the CWSR:

$$p=p_1\oplus p_2\oplus\ldots\oplus p_m \text{ and } q=q_1\otimes q_2\otimes\ldots\otimes q_m \quad (5)$$

where $\oplus$ and $\otimes$ are composition operations for web services. For example, when they are replaced by the ordinary (or fuzzy) union operation of set theory, the composite web service is the same as that discussed in DIANE (Küster, 2007) or similar to the composite web service in (Kwon, 2003). When they are replaced by the "independence" operation taking into account interdependent relationships among the services, the composite service is similar to that discussed in (Limthanmaphon & Zhang, 2003). However, it is still a big issue for case based WS composition to use a more sophisticated composition operation to obtain a composite service case, although service composition can be either performed by composing elementary or composite services (Dustdar & Schreiner, 2005).

After obtaining a composite service case, the CWSR will recommend it to the WS requester for acceptance. This goes to the early mentioned process for acceptance, adaptation or rejection.

So far, we have discussed case-based WS retrieval, discovery, adaptation, reuse, composition and recommendation in a unified way.

FUTURE RESEARCH DIRECTIONS

Applying intelligent techniques to web services is significant for the research and development of business intelligence, web services and service computing. We have argued that CBR as an intelligent technology can be applied to many activities of web services. There are at least four future research directions towards the engineering and management of case based web services. The first future research direction is how to engineer the web service cases and automate the process stages of the requester's demand-driven web service lifecycle based on the existing technologies of data engineering and case engineering. The second future research direction is how to manage huge, heterogeneous web service cases. One intelligent technology cannot be used to automate all activities of web services. Then, the third research direction is how to use hybrid intelligent techniques to provide an integrated solution to web services. The last research direction is how to develop a web based CBR system using the proposed architecture CWSR to improve the following: user satisfaction, finding more web services, better web services, faster web services, and/or more revenue for organizations that would offer it, or less abandoned transactions, etc. To this end, a case c should consist of a problem p, its solution s, and its outcome o: $c = (p, s, o)$. o can be considered as a metric that measures the quality of solution to solve the problem (Bridge, 2005).

CONCLUSION

The article looked at web service lifecycle from a requester's demand perspective, proposed a system architecture of a case based web services reasoner, CWSR, examined the correspondence relationship between the main activities of CBR and those of web services, and explored a unified CBR treatment for the main activities of web services. The proposed approach will facilitate the development of web services, intelligent systems and business intelligence. Service oriented architecture (SOA) is fundamental for service oriented computing (SOC). Web services is an important application field of SOC. A WSLC can be considered as a logical implementation of SOA in web services. The CSWR architecture is a logical realization of the WSLC. These are key ideas behind the above-mentioned discussion in this article. In future work, we will implement a system prototype based on the proposed CWSR for e-solution of sleep deprivation, which can be considered as a part of web health services, and test the effectiveness of the above proposed approach in the context of user satisfaction, finding more web services, better web services, faster web services. We will also integrate web service discovery, composition and recommendation using the CWSR based on soft CBR.

REFERENCES

Alonso, G., Casati, F., Kuno, H., & Machiraju, V. (2004). *Web Services: Concepts, Architectures and Applications*. Berlin: Springer-Verlag.

Bridge, D. (2005). *The Virtue of Reward.* Retrieved May 30, 2009, from www.cs.ucc.ie/~dgb/iccbr05/iccbr05-slides.ppt

Deitel, H. M., Deitel, P. J., DuWadt, B., & Trees, L. K. (2004). *Web Services: A Technical Introduction*. Upper Saddle River, NJ: Prentice Hall.

Dustdar, S., & Schreiner, W. (2005). A survey on web services composition. *International Journal Web and Grid Services*, *1*(1), 1–30. doi:10.1504/IJWGS.2005.007545

Finnie, G., & Sun, Z. (2003). A logical foundation for the CBR cycle. *International Journal of Intelligent Systems*, *18*(4), 367–382. doi:10.1002/int.10093

He, H., Potts, M., & Sedukhin, I. (2004). *Web service management: Service life cycle*. Retrieved May 28, 2009, from http://www.w3.org/TR/wslc/

Hoffman, K. D. (2003). Marketing + MIS = E-Services. *Communications of the ACM*, *46*(6), 53–55. doi:10.1145/777313.777340

ICWS. (2009). Retrieved from http://conferences.computer.org/icws/2009/

Jackson, J., & Mclver, R. (2004). *Microeconomics* (7th Ed.). Brisbane, Australia: McGraw-Hill.

Kolodner, J. (1993). *Case Based Reasoning*. San Mateo: Morgan-Kaufmann Publishers.

Küster, U., König-Ries, B., Stern, M., & Klein, M. (2007). DIANE: An integrated approach to automated service discovery, matchmaking and composition. In *Intl Conf. on World Wide Web (WWW2007),* May 8–12, Alberta, Canada, (pp. 1033-1041).

Kwon, O. B. (2003). Meta web service: building web-based open decision support system based on web services. *Expert Systems with Applications*, *24*, 375–389. doi:10.1016/S0957-4174(02)00187-2

Ladner, R. (2008). Soft computing techniques for web service brokering. *Soft Computing*, *12*, 1089–1098. doi:10.1007/s00500-008-0277-0

Limthanmaphon, B., & Zhang, Y. (2003). Web service composition with case-based reasoning. In *ACM Intl Conf Proc Series; Vol. 143, Proc. 14th Australasian Database Conf, Adelaide*, Australia, (pp. 201-208).

Lorenzi, F., & Ricci, F. (2005). Case-based recommender systems: A unifying view. In Mobasher, B., & Anand, S.S. (Eds.), *ITWP 2003*, (LNAI 3169, pp. 89–113). Berlin: Springer-Verlag.

Lu, J., Ruan, D., & Zhang, G. (2007). E-service Intelligence: An introduction. In J. Lu, D. Ruan, & G. Zhang, (Eds.) *E-Service Intelligence: Methodologies, Technologies and Applications* (pp. 1-33). Berlin: Springer.

Narendra, N. C., & Orriens, B. (2006). Requirements-driven modeling of the web service execution and adaptation lifecycle. In S. Madria, et al, (eds.), *ICDCIT*, (LNCS 4317, pp. 314-324). Berlin: Springer.

Papazoglou, M. P. (2003). Service -Oriented Computing: Concepts, Characteristics and Directions. In *Proceedings of 4th Intl Conf on Web Information Systems Engineering (WISE2003),* (pp. 3-12).

Pfleeger, S. L., & Atlee, J. M. (2006). *Software Engineering: Theory and Practice* (3rd Ed.). Beijing: Pearson Education, Inc.

Pressman, R. S. (2001). *Software Engineering: A Practitioner's Approach,* (5th Ed.). Boston: McGraw-Hill.

Rust, R. T., & Kannan, P. K. (2003). E-service: A new paradigm for business in the electronic environment. *Communications of the ACM*, *46*(6), 37–42. doi:10.1145/777313.777336

Sheth, A. (2003). *Semantic web process lifecycle: Role of Semantics in Annotation, Discovery, Composition and Orchestration*. Invited Talk, WWW 2003 Workshop on E-Services and the Semantic Web, Budapest, Hungary, May 20.

Singh, M. P., & Huhns, M. N. (2005). *Service-Oriented Computing: Semantics, Processes, and Agents*. Chichester, UK: John Wiley & Sons, Ltd.

Sun, Z., & Finnie, G. (2004). *Intelligent Techniques in E-Commerce: A Case-based Reasoning Perspective.* Berlin: Springer-Verlag.

Sun, Z., & Finnie, G. (2005). A unified logical model for CBR-based e-commerce systems. *International Journal of Intelligent Systems*, *20*(1), 29–46. doi:10.1002/int.20052

Sun, Z., Han, J., & Dong, D. (2008). Five Perspectives on Case Based Reasoning. In D.-S. Huang, et al, (Ed.), *ICIC 2008*, (LNAI 5227, pp. 410–419). Berlin: Springer-Verlag.

Sun, Z., & Lau, S. K. (2007). Customer experience management in e-services. In J. Lu, D. Ruan & G. Zhang (Eds.), *E-Service Intelligence: Methodologies, Technologies and Applications,* (pp. 365-388). Berlin: Springer-Verlag.

Tang, X. F., Jiang, C. J., Ding, Z. J., & Wang, C. (2007). A Petri net-based semantic web services automatic composition method. [in Chinese]. *Journal of Software*, *18*(12), 2991–3000.

Tsalgatidou, A., & Pilioura, T. (2002). An Overview of Standards and Related Technology in Web Services. *Distributed and Parallel Databases*, *12*, 135–162. doi:10.1023/A:1016599017660

Wang, H., Zhang, Y. Q., & Sunderraman, R. (2006). Extensible soft semantic web services agent. *Soft Computing*, *10*, 1021–1029. doi:10.1007/s00500-005-0029-3

Wang, M., Cheung, W. K., Liu, J., Xie, X., & Lou, Z. (2006). E-service/process composition through multi-agent constraint management. In *Intl Conf on Business Process Management (BPM 2006)*, Vienna, Austria, (LNCS 4102, pp. 274-289). Berlin: Springer.

Yao, Y., Yang, F., & Su, S. (2006). Flexible decision making in web services negotiation. In J. Euzenat & J. Domingue, (Eds.), *AIMSA 2006*, (LNAI 4183, pp. 108-117).

Zhang, L. J., & Jeckle, M. (2003). The next big thing: Web services composition. In M. Jeckle & L.J. Zhang, (Eds.), *ICWS-Europe*, (LNCS 2853, pp. 1-10).

ADDITIONAL READING

Bell, M. (2008). *Service-Oriented Modeling (SOA): Service Analysis, Design, and Architecture*. Chichester, England: John Wiley and Sons Ltd.

Breu, R. Breu1, M., Hafner, M., & Nowak, A. (2005). Web service engineering – advancing a new software engineering discipline. In Lowe, D. and Gaedke, M. (Eds.): ICWE 2005, LNCS 3579, pp. 8-18, Berlin Heidelberg: Springer-Verlag.

Chen, L., & Tao, F. (2008). An intelligent recommender system for web resource discovery and selection. In Ruan, D., Hardeman F., & van der Meer, K. (Eds). Intelligent Decision and Policy Making Support Systems. Berlin Heidelberg: Springer, pp. 113-140

Cheng, M.-Y., Tsai, H.-C., & Chiu, Y.-H. (2009). Fuzzy case-based reasoning for coping with construction disputes. *Expert Systems with Applications*, *36*(2), 4106–4113. doi:10.1016/j.eswa.2008.03.025

Cheng, R., & Su, S. Yang, F., & Li, Y. (2006). Using case-based reasoning to support web service composition. In Alexandrov, V.N., et al. (Eds.): ICCS 2006, Part IV, LNCS 3994, pp. 87 – 94, Berlin Heidelberg: Springer-Verlag.

Chung, J. Y., Lin, K. J., & Mathieu, R. G. (2003). Web services computing: Advancing software interoperability. *Computer*, *36*(10), 35–37. doi:10.1109/MC.2003.1236469

Costantini, S. (2008). Agents and Web Services. Retrieved April 04, 2009, from http://www.cs.nmsu.edu/~epontell/backbone/aug08/content/Articles/sadri2/paper.pdf

Hammond, K. J. (1989). Case-based Planning: Viewing Planning as a Memory Task. San Diego, CA: Academic Press Professional, Inc.

Han, W., Shi, X., & Chen, R. (2008). Process-context aware matchmaking for web service composition. *Journal of Network and Computer Applications*, *31*(4), 559–576. doi:10.1016/j.jnca.2007.11.008

Jeong, B., Cho, H., & Lee, C. (2009). On the functional quality of service (FQoS) to discover and compose interoperable web services. *Expert Systems with Applications*, *36*(3), 5411–5418. doi:10.1016/j.eswa.2008.06.087

Küster, U., König-Ries, B., Stern, M., & Klein, M. (2007). DIANE: An integrated approach to automated service discovery, matchmaking and composition. In: Intl Conf on World Wide Web (WWW2007), May 8–12, 2007, Alberta, Canada, pp. 1033-1041.

Lau, R. Y. K. (2007). Towards a web services and intelligent agents-based negotiation system for B2B eCommerce. *Electronic Commerce Research and Applications*, *6*(3), 260–273. doi:10.1016/j.elerap.2006.06.007

Lawler, J. P. (2007). Service-Oriented Architecture: SOA Strategy, Methodology, and Technology. Hoboken: Taylor & Francis Ltd.

Lu, J., Ruan, D., & Zhang, G. (Eds.). (2006). E-Service Intelligence. Berlin Heidelberg: Springer Verlag.

Madhusudan, T., & Uttamsingh, N. (2006). A declarative approach to composing web services in dynamic environments. *Journal of Decision Support Systems*, *41*(2), 325–357. doi:10.1016/j.dss.2004.07.003

Park, C.-S., & Park, S. (2008). Efficient execution of composite Web services exchanging intensional data. *Journal of Information Science*, *178*(2), 317–339. doi:10.1016/j.ins.2007.08.021

Psaila, G., & Wagner, R. R. (Eds.). (2008). E-Commerce and Web Technologies: 9th International Conference, EC-Web 2008 Turin, Italy, September 3-4, 2008, Proceedings (LNCS), Berlin Heidelberg: Springer Verlag.

Richter, M. M. (2009). The search for knowledge, contexts, and Case-Based Reasoning. *Engineering Applications of Artificial Intelligence*, *22*(1), 3–9. doi:10.1016/j.engappai.2008.04.021

Rust, R. T., & Kannan, P. K. (Eds.). (2002). E-Service: New Directions in Theory and Practice. Armonk, NY: M.E. Sharpe.

Sahai, A., & Graupner, S. (2005). Web Services in the Enterprise: Concepts, Standards, Solutions, and Management (Network and Systems Management). New York, NY: Springer.

Schank, R. C., & Riesbeck, C. (1989) Inside Case-Based Reasoning. Hillsdale, NJ: Lawrence Erlbaum Associates.

Song, R., Korba, L., & Yee, G. (Eds.). (2007). Trust in E-services: Technologies, Practices and Challenges. Hershey, PA: IgI-Global.

Talukder, A. K., & Yavagal, R. R. (2007). Mobile Computing: Technology, Applications and Service Creation. New York, NY: McGraw-Hill Inc.

Tatemura, J., & Hsiung, W. P. (2006). Web service decomposition: Edge computing architecture for cache-friendly e-commerce applications. *Electronic Commerce Research and Applications*, *5*(1), 57–65. doi:10.1016/j.elerap.2005.08.001

Ting, I.-H., & Wu, H. J. (Eds.). (2009). Web Mining Applications in E-commerce and E-services. Berlin Heidelberg: Springer Verlag.

Web services. Retrieved April 3, 2009, from http://en.wikipedia.org/wiki/Web_service.

Web Services Activity. Retrieved April 3, 2009, from http://www.w3.org/2002/ws/arch/.

Wooldridge, M. (2002). An Introduction to Multiagent Systems. Chichester, England: John Wiley & Sons Ltd.

Zhao, J. L., & Cheng, H. K. (2005). Web services and process management: a union of convenience or a new area of research? *Journal of Decision Support Systems*, *40*(1), 1–8. doi:10.1016/j.dss.2004.04.002

KEY TERMS AND DEFINITIONS

Case-Based Reasoning (CBR): CBR is a reasoning paradigm based on previous experiences or cases; that is: a CBR system solves new problems by adapting solutions that were used to successfully solve old problems. Therefore, CBR is a kind of experience based reasoning. Case-based management, case-based engineering and case-based computing are more general paradigms than CBR.

Intelligent System: An intelligent system is a system that can imitate, automate some intelligent behaviors of human being. Expert systems and knowledge based systems are examples of intelligent systems. Currently, intelligent systems is a discipline that studies intelligent behaviors and their implementations as well as their impacts on human society.

Multiagent Systems: A multiagent system is an intelligent system consisting of many intelligent agents. An intelligent agent can be considered as a counterpart of a human agent in intelligent systems. Google can be considered as an intelligent search agent.

Service Oriented Computing (SOC): SOC is a research field about service science: service intelligence, service technology, service engineering, service management, and service applications. It is the most general form of studying service in computing discipline.

Web Service Architecture: A web service architecture is a high level description for web services, which is free of concrete implementation of a web service system but it is necessary for any implementation of a web service system.

Web Service Discovery: The process of searching, matching a machine-processable description of a Web service. It aims to find appropriate web services to meet the requirement of the customers.

Web Service Lifecycle (WSLC): It consists of the start of a web service request, the end of web service transaction and its evolutionary stages that transform the web service from the start of the request to the end of transaction. Many activities are included in a WSLC such as web service discovery, composition, recommendation and management.

Web Services: General speaking, web services are all the services available on the Web or the Internet from a business perspective. The first web services were information sources (Schneider, 2003). From a technological perspective, web services are Internet-based application components published using standard interface description languages and universally available via uniform communication protocols. Web services is an important application field of service intelligence and service-oriented computing.