

A survey on adaptation decision-making of business processes and the affected web service compositions

Qinghua Lu^{1,2*}, Shanshan Li¹, Weishan Zhang¹

¹ College of Computer and Communication Engineering, China University of Petroleum, Qingdao, China

² Software Systems Research Group, NICTA, Sydney, Australia

Received 1 November 2014, www.cmnt.lv

Abstract

When changes occur in the business processes and their implemented Web service compositions, the Web service compositions are required to be adapted at run-time to accommodate the changes. To guide development of new decision-making algorithms and corresponding frameworks for management of Web service compositions, there is a need to examine and classify problems involved in adaptation. In this paper, we studied the existing work in the area of adaptation of Web service compositions and performed a classification of problems in adaptation decision-making for business processes and their affected Web service compositions. We classify the adaptation decision-making problems using the 5WH question and propose six dimensions of classifications of problems in adaptation decision-making: causes of the adaptation (corresponding to “When”), goals of the adaptation (corresponding to “Why”), scope of the adaptation (corresponding to “Where”), adaptation decision-maker (corresponding to “Who”), planning of the adaptation (corresponding to “How”), and changes made by adaptation (corresponding to “What”). The classification identifies aspects that are not studied sufficiently in the past works and helps us explore the requirements for adaptation decision-making software. The study observes that autonomic business-driven decision-making is a promising research area, with many unsolved challenges and there is a gap on business-driven decision-making for concurrent adaptation of multiple Web service composition instances.

Keywords: Business-driven IT management; Runtime adaptation; Decision-making; Web service management; Business process management; Autonomic computing

1 Introduction

In the modern world, the globalized economy increasingly requires and supports various business relationships between diverse enterprises. Technical changes (e.g., server is down) and business changes (e.g., new customers added) happen frequently and increasingly in long-running business processes. Enterprise information technology (IT) systems are required to seamlessly interconnect with diverse IT systems of business partners irrespective of their implementations and to handle various technical and business changes. Service-oriented computing was developed to implement business processes and address these challenges. Service-oriented architecture (SOA) implemented with Web service technologies [1] is the most popular approach to build distributed information technology (IT) systems for business processes, within and across organisations. Service-oriented systems [2] (such as Web services and their compositions) are composed in a loosely-coupled manner, possibly at runtime, and exposed as implementation independent services. However, diverse technical and business changes that affect business processes and their Web service compositions cannot fully be handled by just implementing and composing service-

oriented systems. To fully handle changes, business processes and the affected Web service compositions have to be adapted. Adaptation of business processes and the affected Web service compositions is a set of controlling activities undertaken to ensure correct and secure operations, accommodate changes, maximise quality of service (QoS) and business benefits. There are usually more than one ways to adapt business processes and the affected Web service compositions.

In order to determine how to perform adaptation, decision-making algorithms are required to be developed. To be able to combine and compare various decision-making algorithms, there is a need to identify and classify various problems of decision-making in adaptation. Also, the classification of problems is required to direct new decision-making algorithm development and corresponding adaptation decision-making framework. Therefore, in this paper, we develop a novel classification of problems in decision-making for adaptation of business processes and the affected Web service compositions.

This rest of the paper is organized in the following way. Section II discusses the state of the art. Section III concludes the paper.

* Corresponding author's e-mail: dr.qinghua.lu@gmail.com

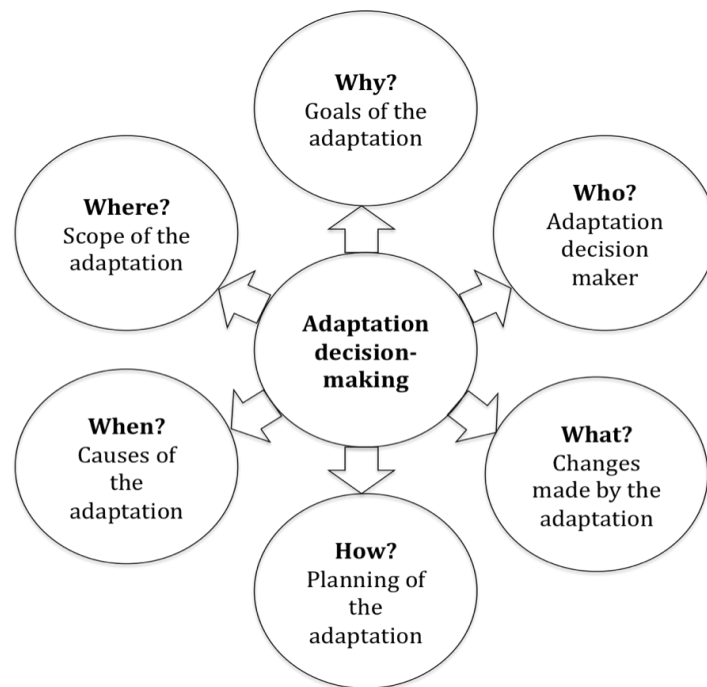


FIGURE1. Six dimensions of classifications of problems in adaptation decision-making.

2 Classifications of Problems in Adaptation Decision-Making

To successfully perform adaptation, it is necessary to precisely specify what is to be achieved by adaptation and how, where, when, and by whom this should be done. In journalism and some other areas, there are six questions (known as 5WH questions) used to guide the description of a situation: “What”, “Where”, “When”, “Why”, “Who”, and “How”. Therefore, we classify the adaptation decision-making problems using these six questions.

We examined the meaning of the 5WH questions in the context of adaptation of business processes and the affected Web service compositions. Then we proposed six dimensions of classifications of problems in adaptation decision-making: causes of the adaptation (corresponding to “When”), goals of the adaptation (corresponding to “Why”), scope of the adaptation (corresponding to “Where”), adaptation decision-maker (corresponding to “Who”), planning of the adaptation (corresponding to “How”), and changes made by adaptation (corresponding to “What”). Fig. 1 illustrates the above six dimensions. In this section we present the classification of problems and discuss which types of problems have been examined by representative past research projects.

2.1 GOALS OF THE ADAPTATION

Different goals set for adaptation of Web service compositions have different influences on decision-making. Here is an example showing how the goals influence adaptation decision-making. Web service X in a Web service composition costs \$10 per month and promises that the response time is less than 10 ms. Assume that Web service X suddenly becomes unavailable for

some reason. There are four alternative Web services which can replace Web service X. The response time and cost of each alternative Web service are shown in Table 1. Assume if the response time guarantee is not met, the penalty for each incident is \$0.10 per incident. Also assume that there are no more than 100 invocations of the Web service per month.

TABLE 1 Four Web service alternatives for Web service X.

Options	Response time	Cost
Web service A	10 ms	\$12
Web service B	10.5 ms	\$10
Web service C	7 ms	\$13
Web service D	12 ms	\$4

As shown in Fig. 2, the goals of adaptation are classified into:

- **Stick to the contract** [3]. In this case, the goal of adaptation is to restore the previous situation, which is as the defined technical metrics in the contract, as much as possible. Thus, for the given example, Web service A is selected to achieve the goal, although the cost is higher than \$10.
- **Minimize the impact of the change** which is caused the adaptation [4]. In this case, the prices and penalties should be considered in the decision-making. Thus, this goal is broader than the above goal “stick to the contract”. Thus, for the given example, Web service B would be the best option as it has minimal impact and it costs less than Web service A and C.

- **Achieve the maximal technical quality of service – QoS** [5-8, 47-49]. In this case, the goal of adaptation is to optimise technical QoS metrics, without taking into account costs. Thus, for the given example, Web service C is selected as the response time of Web service C is the fastest.
- **Achieve maximal business value** [9]. In this case, the goal of adaptation is to achieve the optimal business value, which means the decision is made from a business perspective. This type of goal is

opposed to the above goal which is from the technical perspective. Achieving business value considers not only penalties and costs to be paid but also considers intangible business metrics (such as customer satisfaction and market share) as they have impact on long-term business value. For the given example, Web service D is cheaper than the other options while its response time is higher than the others. Thus, Web service D is selected as it is the best option from the business perspective.

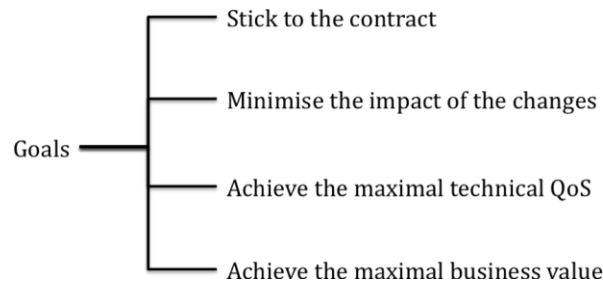


FIGURE 2. Goals of adaptation.

Most of the past works on adaptation decision-making mainly focus on optimization of technical QoS (such as availability and response time) [10-12]. However, business users view technical QoS metrics only as ways to optimise business value (i.e., business worth). A business metric (such as profit and market share) is any measure of business worth of a situation or an activity to a particular business party [13]. To business owners of IT systems, business metrics are more important than technical QoS metrics. For example, when response time of a service changes from 10ms to 30ms, a business owner is usually not very interested in the response time change, but wants to know the business impact of this change, e.g. how much it will cost the owner's business. The business-driven IT management (BDIM) research area [10-12] studies models, algorithms, and architectures for mapping business metrics and technical QoS metrics and making IT management decisions best from the business viewpoint based on such mappings. For example, it tries to quantify impact on business profits of increased/decreased availability. Autonomic business-driven IT management, which is the intersection area of autonomic computing and business-driven IT management, adds processing of business metrics to decision-making components of autonomic systems. As the complexity of business processes and Web service compositions is growing, autonomic business-driven IT management has become a hot research topic for service-oriented systems [14-15].

Most past works on business-driven management of business processes have focused on maximising short-term profit of enterprises. However, business managers often differ in how they prioritise different business metrics in different time frames (e.g., short-term vs. long-term). Maximising short-term profit may not always be the best approach to achieve long-term business goals [13]. It is appropriate to incorporate business motivation in autonomic business-driven IT management solutions to direct decision-making in managing IT systems in response

to runtime changes (e.g., system performance changes and business goal changes). Business motivation represents the factors (such as business vision, goals, and strategies) that motivate the establishing of business, which is a major differentiator of enterprises in a competitive market. Also, IT systems usually run as support of business actions to fulfil operational and strategic objectives of an organization. The execution and management of IT systems should be guided to meet the organisation's business vision. The Business Motivation Model (BMM) [16] is the Object Management Group's (OMG) standard for specification of high-level business motivation and intent as input into design, development and execution of IT systems. Therefore, it is beneficial to make adaptation decisions taking into account long-term business concerns described in the Business Motivation Model.

2.2 SCOPES OF THE ADAPTATION

As shown in Fig.3, we classify the scopes of the adaptation into two types:

- **Schema level adaptation.** In this case, the business process description (e.g., BPEL) is modified. All instances of the affected Web service composition are affected. Schema level adaptation is for all instances, which are further classified into two different sub-types: one same adaptation for all users [17-19], or different adaptations for different classes of user [3].
- **Instance level adaptation** [20]. In this case, instances are classified into different types according to their characteristics, such as current positions in the running process and classes of user. Thus, different classes of instances are adapted in different ways. For example, if there are ten instances with same characteristics and the resources are not enough to provide 99.999% availability to all instances, then some instances (this number depends on the available resources) are adapted to achieve 99.999% availability, while the others are adapted to achieve lower availability (e.g., 98%).

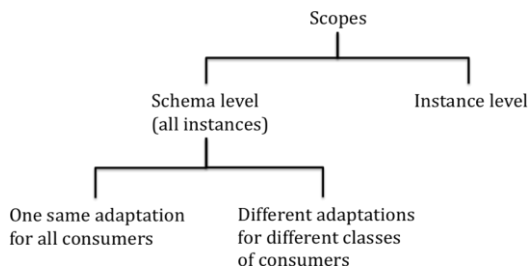


FIGURE 3 Scopes of adaptation.

Most of past research projects make adaptation decisions in terms of one same adaptation for all users, while only a few research projects make different adaptation decisions for different classes of user. It is observed that instance level adaptation is studied rarely. Here, the concept “instance” is a runtime invocation of a business process or a Web service composition.

A particular layer of complexity in adaptation decision-making is that often the change affects not only one instance, but multiple instances and possibly different Web service compositions at the same time.

Since the affected concurrent instances can have different characteristics, it is often necessary to adapt them in different ways. When a change occurs, instances in different classes of services or at different runtime positions in the business process model might require different adaptation. Adapting all affected instances in the same way can lead to sub-optimal solutions, e.g., due to limited underlying resources. Therefore, it is beneficial to examine how to concurrently adapt all affected instances individually to find the globally optimal adaptation.

2.3 CAUSES OF THE ADAPTATION

As shown in Fig. 4, we classify the causes of the adaptation of Web service compositions into:

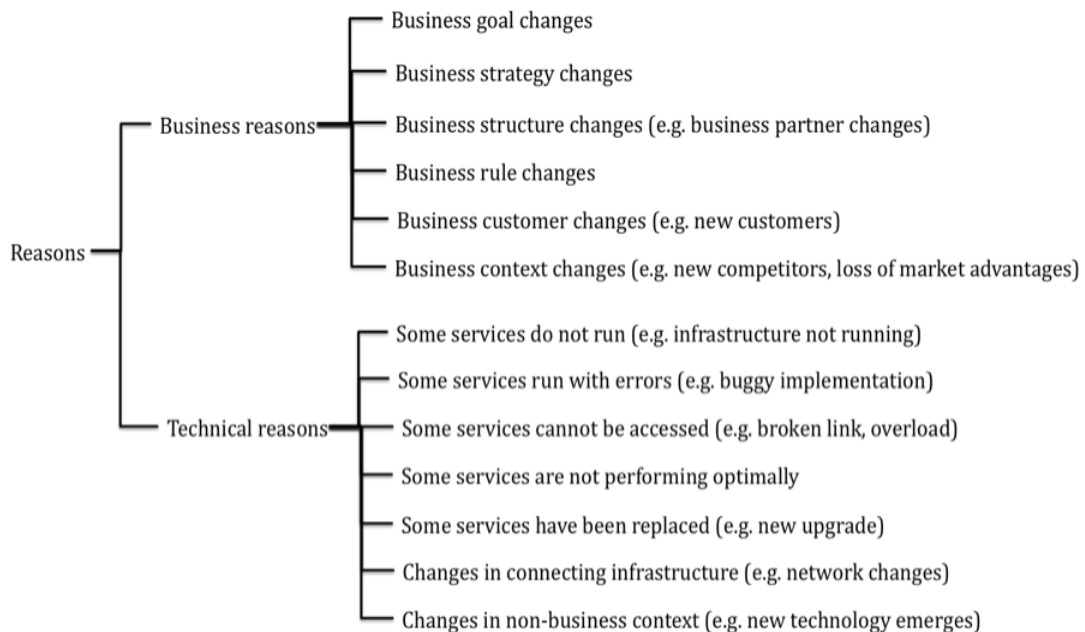


FIGURE 4. Causes of adaptation.

- **Technical reasons** [21-28]. The technical reasons are further classified into: some services do not run (e.g., implementations of services do not work), some services run with errors (e.g., because of a wrong input), some service cannot be accessed (e.g., due to overloaded services or broken network), some services are not performing optimally (e.g., response time increases), some services have been replaced (e.g., new upgrade with backward compatibility), changes in connecting infrastructure (e.g., orchestrator software changes), changes in non-business context (e.g., new emerging technology, technology changes).
- **Business reasons** [18, 24]. The business reasons are further classified into: business goal changes (e.g., triple the current annual profit goal”), business strategy changes (e.g., business focus changes from gold consumers to silver consumers), business structure changes (e.g., business partner changes), business rule changes (e.g., increase to 20 servers for gold consumers), business customer changes (e.g., customer preference changes), business context changes (e.g., new emerging competitors).

The majority of past works primarily examine adaptations caused by technical reasons, while only a few research projects [18, 24] concern the impact of business reasons without detailed analysis. It is necessary to understand various reasons causing adaptation as they may need different monitoring and adaptation. For example, if business related events are not monitored, the corresponding required adaptation will not be triggered by the management system.

This rest of the paper is organized in the following way. Section II discusses the state of the art. Section III concludes the paper.

2.4 PLANNING OF THE ADAPTATION

As shown in Fig. 5, according to whether a single action or a set of actions is to be determined, we classify adaptation decision-making into:

- **Single action** [17, 19, 29, 30]: Only one optimal adaptation action is selected from several alternatives.
- **Single plan** [31-32]: a single plan with a group of loosely couple sequential actions is determined.
- **Multiple parallel plans**: this type of planning is relevant to instance level adaptation which is discussed in Scope of the Adaptation. In this case, Different plans are needed for different classes of instances.

The majority of past research projects mainly focus on determining single actions for adaptations. As the complexity of long-term plan determination, only a few past projects focus on plan set up, particularly on multiple parallel plans.

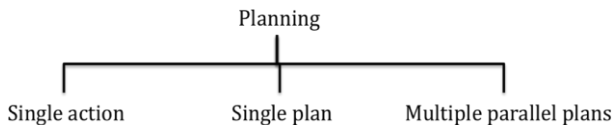


FIGURE 5 Planning of adaptation

2.5 CHANGES CAUSED BY THE ADAPTATION

As shown in Fig. 6, adaptation could lead to changes in various aspects of system execution and configuration:

- **Changing the business process structure** [17, 19, 22, 33, 34], which includes creating a new business process, modifying the current business process, and terminating the services for some time.
- **Changing contracts with customers** [35-36] include making new contracts, switching customers between existing contracts, changing existing extra-functional contracts, changing existing functional contracts.
- **Changing execution of the business process** [18, 23, 25], by taking some actions such as skip, rollback, termination and retry of the business process instance.

Research projects [18, 24] concern the impact of business reasons without detailed analysis. It is necessary the past research projects support the above types of changes well, although there is no one supporting all types of changes. The classification of changes caused by adaptations can guide the outputs of adaptation decision-making algorithms.

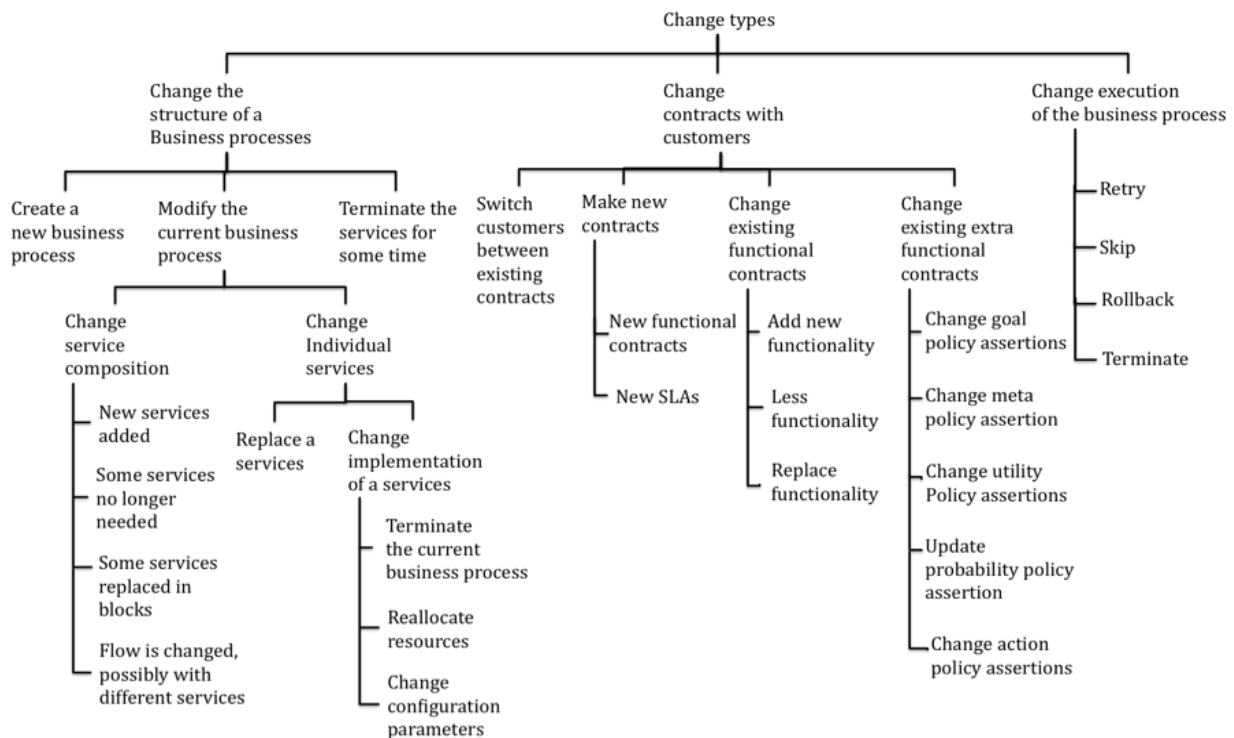


FIGURE 6 Changes caused by the adaptation.

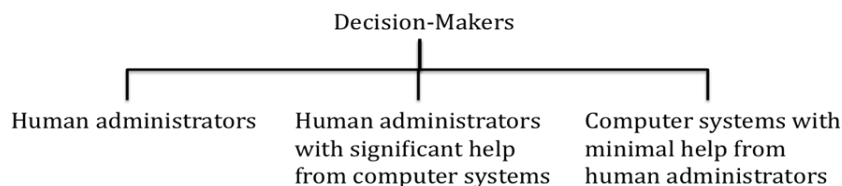


FIGURE 7 Adaptation decision-makers.

2.6 ADAPTATION DECISION-MAKERS

As shown in Fig. 7, we classify decision-makers into: computer systems with minimal help from human administrators [22, 25], human administrators with support from computer systems [37], human administrators. Correspondingly, adaptation decisions can be determined automatically, semi-automatically, or manually.

In industry practice, adaptation decision-making of business processes and their affected Web service compositions is still manually done by human administrators. The majority of past management products for adaptation mainly act as decision-making support tools for human administrators. These products show summaries of monitored information and often automate execution of some simple control actions, but human administrators are ultimately responsible for making complex decisions about control actions. However, human administrators might not always be available, might take too long to analyse complex situations and respond, or might be too expensive. Further, it is difficult for human administrators to handle a large number of Web services and understand multifaceted interdependencies between many diverse components and metrics at various levels of abstraction. Due to the high complexity of management tasks and cost of experienced human system administrators, IT systems should be managing themselves with minimal human intervention. Self-management has been a research goal for several decades, but was made prominent by the vision of autonomic computing. Autonomic computing [38] is an approach towards reducing complexity in IT system management, where IT systems self-manage themselves directed by configurable high-level management policies with minimal human intervention. A prerequisite for performing system adaptation is existence of a machine process able and precise format for specification of adaptations [39-40]. Policies are a frequent approach to IT system management, which is not limited to autonomic computing. A policy formally specifies a collection of high-level and implementation-independent operation, goals, and rules.

Autonomic systems should be carefully designed because various policies and the interdependencies between them may increase the complexity of systems, which is in contrast to the aim of autonomic computing. Further, most of previous works in autonomic computing mainly assume that autonomic systems are fully specified via perfect policies written by human administrators and the autonomic system execution is predictable. However, sometimes human administrators are not sure about exact values for policies or the values they know might have become obsolete due to runtime unpredictable runtime changes or human errors. Therefore, it is necessary to introduce specification of uncertainty in policies and algorithms for decision-making under uncertainty.

Table 2 shows different features of several represen-

tative research projects in the area of self-adaptation for service-oriented systems. There are some other research projects, but with similar strengths and weaknesses as the discussed representative projects, so the overall conclusions of the analysis also apply to them. Most of the autonomic systems only focus on some of the self-management capabilities. For example, Unity [44] focuses on self-configuration, self-healing and self-optimization while Dynamo [42] and Adaptive Server Framework [43] mainly work on the self-healing capabilities of autonomic systems. [18, 24] identify and classify adaptation problems at high-level. Many research projects have developed self-healing approaches using policies [18, 41]. Most of existing research projects focus on self-healing framework development [21, 25, 28, and 41]. Only a few research projects [45-46] work on recovery without taking into account business metrics and they do not consider long-term business metrics.

TABLE 2 Representative research projects on self-adaptation of business processes and the affected Web service compositions.

Paper	Self-healing	Self-optimisation	Policy-based	Uncertainty handling
[22]	Yes	No	No	No
[24]	No	No	No	No
[18]	Yes	No	Yes	No
[21]	Yes	No	No	No
[25]	Yes	No	No	No
[28]	Yes	No	No	No
[41]	Yes	No	Yes	No
[42]	Yes	No	Yes	No
[43]	Yes	No	Yes	No
[44]	Yes	Yes	Yes	No
[45]	Yes	No	No	No
[46]	Yes	No	No	No

3 Conclusion

In this paper, we performed a classification of problems in adaptation decision-making for business processes and their affected Web service compositions. The classification identifies aspects that are not studied sufficiently in the past works and helps us explore the requirements for adaptation decision-making software.

Acknowledgement

This project is supported by National Natural Science Foundation of China (Grant No. 61402533), "the Fundamental Research Funds for the Central Universities" (No. 14CX02140A) and "the Scientific Research Foundation of China University of Petroleum" (No. Y1307021).

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

References

- [1] Thomas Erl, 2004 Service-Oriented Architecture: A Field Guide to Integrating XML and Web Services, Prentice Hall.
- [2] M.P.Papazoglou, D.Georgakopoulos 2003 Service-Oriented Computing *Communication of the ACM* 46, 25-28.
- [3] [3] Claudio Bartolini, Abdel Boulmakou, Athena Christodoulou, Andrew
- [4] Farrell, Mathias Sallé, David Trastour 2004 Management by
- [5] Contract: IT Management Driven by Business Objectives
- [6] *Proceedings of 11th HP Openview University Association Workshop*,
- [7] T. Dumitra et al., Dynamic Change Management for Minimal Impact on Dependability and Performance in Autonomic Service-Oriented Architectures *CyLab*, 48.
- [8] Chaffle et al., 2006 Adaptation in Web Service Composition and Execution in *Proceedings of IEEE International Conference on Web Services*, 549-557.
- [9] H. Tong, S. Zhang 2006 A fuzzy multi-attribute decision making algorithms for web services selection based on qos in *Proceedings of IEEE Asia-Pacific Conference on Services Computing*, 51-57.

- [10] R. Aschoff, A. Zisman 2011 QoS-Driven Proactive Adaptation of Service Composition *Service-Oriented Computing*, 421-435.
- [11] T. Yu et al 2005 Efficient algorithms for Web services selection with end-to-end QoS constraints *Information Systems and e-Business Management* 3(2),103-126.
- [12] S. Aiber et al. 2004 Autonomic self-optimization according to business objectives in *Proceedings of 1st International Conference on Autonomic Computing*, 206-213.
- [13] J. Sauve et al. 2006 An Introductory Overview and Survey of Business-Driven IT Management in *Proceedings of 1st IEEE/IFIP International Workshop on Business-Driven IT Management*, 1-10.
- [14] A. Moura et al. 2007 Research challenges of business-driven IT management in *Proceedings of the 2nd IEEE/IFIP International Workshop on Business-Driven IT Management, Munich, Germany*, 19-28.
- [15] C. Bartolini, C. Stefanelli 2011 Business-driven IT management in *Proceedings of IFIP/IEEE International Symposium on Integrated Network Management*, 964-969.
- [16] V. Tosic 2010 Autonomic business-driven dynamic adaptation of service-oriented systems and the WS-Policy4MASC support for such adaptation *International Journal of Systems and Service-Oriented Engineering*, 1, 9-95.
- [17] V.C. Emeakaroha 2010 Low level metrics to high level SLAs – LoM2HiS framework: Bridging the gap between monitored metrics and SLA parameters in Cloud environments in *Proceedings of International Conference on High Performance Computing and Simulation*, 48-54.
- [18] M. Litoiu et al. 2010 A business-driven cloud optimization architecture in *Proceedings of the 2010 ACM Symposium on Applied Computing*, 380-385.
- [19] OMG-BMM 2004 Business Motivation Model, Version 1.0 *Object Management Group*. <http://www.omg.org/spec/BMM/1.0/PDF>.
- [20] T. Yu K.J. Lin 2005 Adaptive algorithms for finding replacement services in autonomic distributed business processes in *Proceedings of 2005 Autonomous Decentralized Systems*, 427-434.
- [21] R. Hamadi et al. 2008 Self-adapting recovery nets for policy-driven exception handling in business process *Distributed and Parallel Databases*, 23, 1-44.
- [22] T. Yu et al. 2007 Efficient algorithms for Web services selection with end-to-end QoS constraints *ACM Transaction on the Web* 1, 6.
- [23] P. Leitner et al. 2013 Cost-Based Optimization of Service Compositions in *IEEE Transactions on Services Computing*, 6(2), 239-251.
- [24] G.K. Mostefaoui et al. 2007 On modeling and developing self-healing web services using aspects in *Proceedings of 2nd International Conference on Communications Systems Software and Middleware*, 1-8.
- [25] G. Chafle et al. 2006 Adaptation in Web Service Composition and Execution *IEEE*, 549-557.
- [26] F. Cabrera et al. 2005 Web services transaction (WS Transaction) <http://www.ibm.com/developerworks/library/specification/ws-tx/>.
- [27] N. Desai et al. 2006 An overview of business process adaptations via protocols in *Proceedings of the 5th International joint conference ON autonomous agents and Multiagent Systems*, 1326-1328.
- [28] O. Nabuco 2008 Model-Based QoS-Enabled Self-Healing Web Services in *Proceedings of 19th International Conference on Database and Expert Systems Application*, 711-715.
- [29] E. Zahoor 2010 Disc: A declarative framework for self-healing web services composition, *IEEE*, 25-33.
- [30] G. Friedrich et al. 2010 Exception Handling for Repair in Service-Based Processes *IEEE Transactions on Software Engineering* 36, 198-215.
- [31] H. Naccache, G.C. Gannod 2007 A self-healing framework for web services in *Proceedings of 2007 International Conference on Web Services*, 345-398.
- [32] C. Herzsens et al. 2008 Context-driven autonomic adaptation of sla *2008 International Conference on Service-Oriented Computing*, 362-377.
- [33] H. Tong, S. Zhang 2006 A fuzzy multi-attribute decision making algorithms for web services selection based on qos in *Proceedings of 2006 IEEE Asia-Pacific Conference on Service Computing*, 51-57.
- [34] J. Sauve et al. 2006 Business-Driven Decision Support for Change Management: Planning and Scheduling of Changes, in *Proceedings of 17th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management*, 173-184.
- [35] G. Tesauro et al., 2005 Utility function driven resource allocation in autonomous systems in *Proceedings of the 2nd IEEE International Conference on Autonomic Computing*, 342-343.
- [36] P. Doshi et al. 2004 Dynamic workflow composition using Markov decision processes in *Proceedings of Second International Conference on Web Services*, 576-582.
- [37] T. Yu, K. Lin 2004 Service selection algorithms for Web services with end-to-end QoS constraints in *Proceedings of 2004 IEEE International Conference on e-Commerce Technology*, 129-136.
- [38] A. Erradi et al. 2007 MASC-NET-based middleware for adaptive composite Web services in *Proceedings of 2007 IEEE International Conference on Web Services*, 727-734.
- [39] V. Tosic et al. 2003 Management Application of the Web Service Offerings Language (WSOL) *Information Systems* 30, 564-586.
- [40] N. Samaan, A. Karmouch 2009 Towards Autonomic Network Management: an Analysis of Current and Future Research directions *IEEE Communications Surveys & Tutorials* 11, 22-36.
- [41] J.O. Kephart, D.M. Chess 2003 The Vision of Autonomic Computing *Computer* 36(1), 41-50.
- [42] A. Keller, H. Ludwig 2003 The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services *Journal of Network and Systems Management* 11(1), 57-81.
- [43] V. Tosic et al. 2005 WSOL –Web Service Offerings Language *the International Workshop on Web Services, E-Business, and Semantic Web*, Spring, 57-67.
- [44] L. Zeng, et al. 2005 Policy-driven exception-management for composite web services in *Proceedings of 7th IEEE International Conference on E-Commerce Technology*, 355-363.
- [45] L. Baresi, et al. 2007 Policies and Aspects for the Supervision of BPEL processes in *Proceedings of the 19th International conference on Advanced Information Systems Engineering*, 340-354.
- [46] Y. Liu, I. Gorton 2007 Implementing adaptive performance management in server applications in *Proceedings of 2007 International Workshop on Software Engineering for Adaptive and Self-Managing Systems*, 12.
- [47] D.M. Chess, et al. 2004 Unity: Experiences with a prototype autonomic computing system in *Proceedings of 2004 International Conference on Autonomic Computing*, 140-147.
- [48] G. Friedrich, et al. 2010 Exception Handling for Repair in Service-Based Processes *IEEE Transaction on Software Engineering* 36(2), 198-215.
- [49] E. Zahoor, et al. 2010 Disc: A declarative framework for self-healing web services composition in *Proceedings of 2010 IEEE International Conference on Web Services*, 25-33.

Author	
	<p>< Qinghua Lu >, <1983.6>, < Qingdao, Shandong Province, P.R. China></p> <p>Current position, grades: Qinghua Lu is a lecturer at Department of Software Engineering, China University of Petroleum, and Qingdao, China.</p> <p>University studies: She received her PhD from University of New South Wales (UNSW) in 2013.</p> <p>Scientific interest: Her research interest field includes software architecture, dependability of cloud computing, and service engineering.</p> <p>Publications: She has more than 20 papers published in various journals and conferences proceedings.</p> <p>Experience: She has teaching experience of 2 years, has completed five scientific research projects.</p>
	<p>< Shanshan Li >, <1990.10>, < Qingdao, Shandong Province, P.R. China></p> <p>Current position, grades: Shanshan Li is a postgraduate student at College of Computer and Communication Engineering, China University of Petroleum.</p> <p>Scientific interest: software architecture, API and cloud computing.</p>
	<p>< Weishan Zhang >, <1970.4>, < Qingdao, Shandong Province, P.R. China></p> <p>Current position, grades: Weishan Zhang is a full professor, deputy head for research of Department of Software Engineering, College of Computer and Communication Engineering, China University of Petroleum. He is the founding director of the Autonomous Service Systems Lab.</p> <p>Scientific interest: His current research interests are big data platforms, pervasive cloud computing, and service oriented computing.</p> <p>Publications: Weishan has published over 50 papers and his current h-index according to google scholar is 10.</p>