Grant Agreement N° 215483

Title:	*Consolidated and updated state of the art report on Service-Based Applications*
Authors:	CITY, CNR, FBK, LERO, Polimi, SEERC, SZTAKI, Tilburg, TUW, UniHH, UOC, UPM, USTUTT, VUA,
Editor:	*Michele Mancioppi (USTUTT)*
Reviewers:	*Barbara Pernici (Polimi)*
	Manuel Carro (UPM)
	Dragan Ivanovic (UPM)
	Andreas Metzger (UniDue)
Identifier:	*CD-IA-1.1.7*
Type:	*Deliverable*
Version:	*1*
Date:	*30 November 2011*
Status:	*Final*
Class:	*External*

Management Summary

This deliverable discusses the advancements of the State of the Art of the topics studied in S-Cube since the beginning of the NoE on the basis of both S-Cube and extra-S-Cube research efforts.

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 under grant agreement n° 215483 (S-Cube).

Members of the S-Cube consortium:

University of Duisburg-Essen (Coordinator)	Germany
Tilburg University	Netherlands
City University London	U.K.
Consiglio Nazionale delle Ricerche	Italy
Center for Scientific and Technological Research	Italy
The French National Institute for Research in Computer Science and Control	France
Lero - The Irish Software Engineering Research Centre	Ireland
Politecnico di Milano	Italy
MTA SZTAKI – Computer and Automation Research Institute	Hungary
Vienna University of Technology	Austria
Université Claude Bernard Lyon	France
University of Crete	Greece
Universidad Politécnica de Madrid	Spain
University of Stuttgart	Germany
University of Hamburg	Germany
Vrije Universiteit Amsterdam	Netherlands

Published S-Cube documents

All public S-Cube deliverables are available from the S-Cube Web Portal at the following URL:

http://www.s-cube-network.eu/results/deliverables/

The S-Cube Deliverable Series

Vision and Objectives of S-Cube

The Software Services and Systems Network (S-Cube) will establish a unified, multidisciplinary, vibrant research community which will enable Europe to lead the software-services revolution, helping shape the software-service based Internet which is the backbone of our future interactive society.

By integrating diverse research communities, S-Cube intends to achieve world-wide scientific excellence in a field that is critical for European competitiveness. S-Cube will accomplish its aims by meeting the following objectives:

- Re-aligning, re-shaping and integrating research agendas of key European players from diverse research areas and by synthesizing and integrating diversified knowledge, thereby establishing a long-lasting foundation for steering research and for achieving innovation at the highest level.
- Inaugurating a Europe-wide common program of education and training for researchers and industry thereby creating a common culture that will have a profound impact on the future of the field.
- Establishing a pro-active mobility plan to enable cross-fertilisation and thereby fostering the integration of research communities and the establishment of a common software services research culture.
- Establishing trust relationships with industry via European Technology Platforms (specifically NESSI) to achieve a catalytic effect in shaping European research, strengthening industrial competitiveness and addressing main societal challenges.
- Defining a broader research vision and perspective that will shape the software-service based Internet of the future and will accelerate economic growth and improve the living conditions of European citizens.

S-Cube will produce an integrated research community of international reputation and acclaim that will help define the future shape of the field of software services which is of critical for European competitiveness. S-Cube will provide service engineering methodologies which facilitate the development, deployment and adjustment of sophisticated hybrid service-based systems that cannot be addressed with today’s limited software engineering approaches. S-Cube will further introduce an advanced training program for researchers and practitioners. Finally, S-Cube intends to bring strategic added value to European industry by using industry best-practice models and by implementing research results into pilot business cases and prototype systems.

S-Cube materials are available from URL: http://www.s-cube-network.eu/

Contents

Acronyms

AI Artificial Intelligence. 58

ANT Actor-Network Theory. 62

APPM Autonomic Power and Performance Manager. 41

AST Abstract Syntax Tree. 36

AW Autonomic Workflow. 38

BAM Business Activity Monitoring. 100

BPEL Business Process Execution Language. 17, 24, 35, 39, 43, 44, 48, 54, 58, 68, 70–73, 80, 81, 89, 90

BPM Business Process Management. 48, 62, 65

BPMN v1.x Business Process Model and Notation v1.x. 61, 65, 66, 68, 71, 72, 80

BPMN v2.0 Business Process Model and Notation v2.0. 71, 72, 83

BTL Business Transaction Language. 65

CCS Calculus of Communicating Systems. 73

CEP Complex Event Processing. 44, 89, 90

CSCW Computer Supported Cooperative Work. 65

CTR Concurrent Transaction Logic. 68

CTT ConcurTaskTree. 24

DAG Direct Acyclic Graph. 58

DSOL Declarative Service Orchestration Language. 69

ECA Event-Condition-Action. 38, 39, 83

ESB Enterprise Service Bus. 47

EUD End User Development. 23

FCA Formal Concept Analysis. 80

GPP Granular Process Properties. 65

GSD Global Software Development. 20

HCI Human-Computer Interaction. 21–23

HTS Human-Interaction Service. 23, 24

IaaS Infrastructure as a Service. 19, 40, 41, 47, 98

KPI Key Performance Indicator. 48, 61, 67, 90

LTL Linear Temporal Logic. 66, 72

LTS Labeled Transition System. 73

NFC Non-Functional Characteristic. 83

NFP Non-Functional Property. 50

OM Organization Model. 39

PaaS Platform as a Service. 19

PPM Process Performance Metric. 67

QAttr Quality attribute. 41

QoC Quality of Context. 16

QoS Quality of Service. 16, 27, 35, 39, 43, 44, 48, 50–52, 54–58, 65, 67, 78, 79, 92–94

SaaS Software as a Service. 19, 20

SBA Service-Based Application. 11, 13, 15, 21–24, 27, 35, 38, 41–45, 47, 48, 50, 53, 55, 65, 89, 92–94, 98–100

SBS Service-Based System. 15, 35–37, 48

SBVR Semantics of Business Vocabulary and Rules. 69

SCC Service Computing. 48

SE Software Engineering. 65

SESE Single-Entry-Single-Exit. 68

SI Service Identification. 33, 48

SIM Service Identification Method. 33

SLA Service-Level Agreement. 14, 35, 36, 38, 39, 41, 42, 44, 48, 52–58, 78, 92, 96, 97, 100

SLO Service-Level Objective. 39, 52

SN Service Network. 65

SNA Social Network Analysis. 61

SOA Service Oriented Architecture. 19, 20, 22–24, 29, 30, 33, 39, 42, 47, 72, 83, 100

SOAP Simple Object Access Protocol. 22, 35, 36

SOC Service Oriented Computing. 21, 65

SVN Service Value Network. 62

SXM Stream X-Machines. 38

UDDI Universal Description, Discovery and Integration. 22

UI User Interface. 23

UM User Modeling. 22

UML Unified Modeling Language. 36, 62, 68, 69, 72, 74

VM Virtual Machine. 41, 97, 98

WS Web Service. 50

WS-CDL Web Services Choreography Description Language. 72, 90

WSCI Web Service Choreography Interface. 72

WSDL Web Service Description Language. 16, 17, 22, 71

WSDM Web Services Distributed Management. 89

WSPL Web Service Policy Language. 39

WSRF Web Services Resource Framework. 89

XML eXtensible Markup Language. 15, 22, 36, 50, 56

XPDL XML Process Definition Language. 89

Chapter 1

Introduction

The goal of this deliverable is to document the advancements in the State of the Art on the topics related to Service-Based Applications (SBAs) since 2008, year in which the NoE submitted a number of deliverables, namely [PO-08a, PO-08c, PO-08b, PO-08d, PO-08e, PO-08f, PO-08g], dedicated to the different research areas covered by S-Cube.

In the first round of State of the Art reports delivered in 2008, each JRA was required to produce a State of the Art for the research areas it covered. Results where then consolidated in [PO-08a], which provided a summary of the State of the Art reports produced by each JRA, and eventually resulted in the S-Cube book [PPPM10]. The present deliverable (CD-IA-1.1.7) presents the entire consolidated consolidated State of the Art considering both research results produced within and outside S-Cube. The far larger scope of CD-IA-1.1.7 required a different approach on organizing how the information is presented. First of all, Due to the significant alignment between research activities and the joint research efforts undertaking by S-Cube members within the S-Cube work programme, the deliverable is organized as a “flat” collection of topics (which, as could be expected, may have some limited overlap). Each topic is correlated to the interested JRAs by the “tags” at the beginning of each topic. Moreover, for the sake of readability, each topic (as well as the current section) is followed by a dedicated references list.

Secondly, to keep the size of the deliverable manageable and readable, it was decided to adopt a concise style when discussing the advancements of the State of the Art. Therefore, the topics do not necessarily cover in detail each and every contribution it mentions, but instead focus on *trends* of research that have emerged, e.g. matters of particular interests within a certain research area.

Finally, it should be noted that evaluating “how much” has S-Cube contributed to each topic falls outside of the scope of this deliverable. (Nevertheless, S-Cube results are highlighted by tagging the relative citations with the acronyms of the S-Cube beneficiaries that contributed them.) An evaluation of the cohesiveness of S-Cube research will be presented in the upcoming CD-IA-3.2.5 deliverable.

Bibliography

[PO-08a] Comprehensive overview of the state of the art on service-based systems. Deliverable, S-CUBE Consortium, 2008.

[PO-08b] State of the art report, gap analysis of knowledge on principles, techniques and methodologies for monitoring and adaptation of sbas. Deliverable, S-Cube Consortium, 2008.

[PO-08c] State of the art report on software engineering design knowledge and survey of HCI and contextual knowledge. Deliverable, S-Cube Consortium, 2008.

[PO-08d] Survey of quality related aspects relevant for service-based applications. Deliverable, S-Cube Consortium, 2008.

[PO-08e] Survey on business process management. Deliverable, S-Cube Consortium, 2008.

[PO-08f] Overview of the state of the art in composition and coordination of services. Deliverable, S-Cube Consortium, 2008.

[PO-08g] Use case description and state-of-the-art. Deliverable, S-Cube Consortium, 2008.

[PPPM10] Mike P. Papazoglou, Klaus Pohl, Michael Parkin, and Andreas Metzger, editors. *Service Research Challenges and Solutions for the Future Internet - S-Cube - Towards Engineering, Managing and Adapting Service-Based Systems*, volume 6500 of *Lecture Notes in Computer Science*. Springer, 2010.

Chapter 2

Update of the State of the Art on Service-Based Applications

Table 2.1 presented an outlook of the topics of the State of the Art covered in this chapter, together with the JRAs the topics are associated to. It should be noted that the research challenges addressed by each of those JRAs are discussed elsewhere, namely in the upcoming IA-3.1.7 and IA-3.1.8 deliverables.

Topics	**Related JRAs**
Service-Based Application Lifecycle	JRA-1.1
Context modeling	JRA-1.1, JRA-1.2
Evolution	JRA-1.1, JRA-1.2
Using the Cloud to Facilitate Global Software Development Challenges	JRA-1.1, JRA-2.3
Human-Computer Interaction & Task Models	JRA-1.1
Software Processes for Adaptable Systems	JRA-1.1, JRA-1.2
Service-Oriented Architecture Migration	JRA-1.1
Service Identification	JRA-1.1
Monitoring Service-Based Applications	JRA-1.2
Monitor Adaptation	JRA-1.2
Adaptable Cloud Infrastructures	JRA-1.2, JRA-2.3
Adaptation of Service Compositions	JRA-1.2, JRA-2.2
Cross-layer Monitoring and Adaptation	JRA-1.2, JRA-2.1, JRA-2.2, JRA-2.3
Quality Modeling	JRA-1.3
Proactively Negotiating and Agreeing On Service Level Agreements	JRA-1.2, JRA-1.3, JRA-2.2, JRA-2.3
Quality Prediction Techniques to Support Proactive Adaptation	JRA-1.2, JRA-1.3
Modeling, Design and Performance Analytics of Service Networks	JRA-1.2, JRA-2.1
Modeling, Analysis and Monitoring of Business Transactions	JRA-1.2, JRA-2.1
Modeling and Analysis of Service Orchestrations	JRA-2.2
Modeling and Analysis of Service Choreographies	JRA-2.2
Fragmentation of Service Compositions	JRA-2.2
Execution of Distributed Processes	JRA-2.2
Monitoring and Analysis of Distributed Processes Performance	JRA-1.2, JRA-2.2
Prediction and Prevention of Service Level Agreement Violations in Service Compositions	JRA-1.3, JRA-2.2
Chemical- and Nature-Inspired Techniques for Service Composition	JRA-2.3
Service Level Agreement-Based Virtualized Service Provision	JRA-1.3, JRA-2.3
Federated Cloud Management	JRA-2.3
Process Mining	JRA-2.3

Table 2.1: Topics covered in this deliverable and the relationship to the S-Cube JRAs

2.1 Service-Based Application Lifecycle

Involved Workpackages: WP-JRA-1.1

Many of the existing SBA development methodologies are based on the results carried out in the fields of classical software and system engineering and do not easily facilitate SBA adaptation [Rat03, PvdH06, AGA+08]. In fact, they mainly focus on the phases that precede the release of software and, even in the cases in which they focus on the operation phases, they usually do not consider the possibility for SBAs to adapt dynamically to new situations, contexts, requirement needs, service faults, etc. Some levels of adaptation are instead supported by other contributions for SBA development such as SOUP (Service Oriented Unified Process) [Mit09] or the approach by [LPRS07]; however, these approaches lack of explicit guidelines for the design phase. Other work discuss instead proactive adaptation techniques mainly in isolation and confined to individual phases of the service life-cycle such as [MSPR10] (UniDUE), [HKMP08] (UniDue, FBK), [GBK+10] (UniDue, FBK) and [LWR+09] (TUW, USTUTT).

Lane and Richardson [LR11] (Lero) carried out a systematic literature review of SBA development approaches; they identified 57 such approaches of which there were only 8 that specifically dealt with adaptation. Furthermore, only four of these 8 approaches were concerned with adaptation of SBAs, the others were concerned with the adaptation of services. Each of the four approaches shows interesting features, but even those that enable the definition of various adaptation strategies lack a coherent approach to support designers in this complex task. Moreover, they focus on the implementation process without considering what impact adaptation has on the rest of the development and operational lifecycle [VH08, WALK09]. They also tend to focus on particular types of adaptation so it is difficult to elicit generic adaptation mechanisms from them. In summary, each of these approaches focused on the analysis and design processes without consideration for any other development or runtime processes. Runtime issues are instead addressed in a recent and notable contribution in the development of adaptive SBA is the Vienna Runtime Environment for Service-oriented Computing (VRESCo) [HLM+10] (TUW), a framework implemented to address issues like dynamic selection, binding and invocation of services. The solution developed in the S-Cube project and proposed in [BCN+09, BKC+10] (FBK, Polimi) aims at considering both design time and runtime aspects and provide complete support for the SBA designer. In particular, in contrast to more traditional life-cycle models, this model considers the specifics of service-based systems, particularly concerning evolution and adaptation.

Nowadays, recent literature contributions started to address adaptation issues in the contexts of Grid or Cloud Computing. All the existing contributions focus on specific aspects. In [BMD09] authors considers cloud computing stating that such environment require adaptable services that can cope with system failures and environmental change minimizing human intervention. In the reminder, the focus is on models for service negotiation and Service-Level Agreement (SLA) mapping. [LBCP09] instead considers "Elastic cloud computing APIs" as the natural opportunity for designing controllers able to automate an adaptive service and resource provisioning, and many recent works have explored feedback control policies for a variety of network services under various assumptions. Some partners of the S-Cube project have shown that the S-Cube lifecycle can be also used in the cloud environment [MGC+10] (POLIMI, FBK).

Bibliography

[AGA+08] Ali Arsanjani, Shuvanker Ghosh, Abdul Allam, Tina Abdollah, Sella Ganapathy, and Kerrie Holley. Soma: A method for developing service-oriented solutions. *IBM Systems Journal*, 47(3):377–396, 2008.

[BCN+09] Antonio Bucchiarone, Cinzia Cappiello, Elisabetta Di Nitto, Raman Kazhamiakin, Valentina Mazza, and Marco Pistore. Design for adaptation of service-based applications: Main issues and requirements. In *ICSOC/ServiceWave Workshops*, pages 467–476, 2009.

[BKC+10] Antonio Bucchiarone, Raman Kazhamiakin, Cinzia Cappiello, Elisabetta di Nitto, and Valentina Mazza. A context-driven adaptation process for service-based applications. In *Proceedings of the 2nd International Workshop on Principles of Engineering Service-Oriented Systems*, PESOS '10, pages 50–56, New York, NY, USA, 2010. ACM.

[BMD09] Ivona Brandic, Dejan Music, and Schahram Dustdar. Service mediation and negotiation bootstrapping as first achievements towards self-adaptable grid and cloud services. In *Proceedings of the 6th international conference industry session on Grids meets autonomic computing*, GMAC '09, pages 1–8, New York, NY, USA, 2009. ACM.

[GBK+10] Andreas Gehlert, Antonio Bucchiarone, Raman Kazhamiakin, Andreas Metzger, Marco Pistore, and Klaus Pohl. Exploiting assumption-based verification for the adaptation of service-based applications. In *SAC*, pages 2430–2437, 2010.

[HKMP08] Julia Hielscher, Raman Kazhamiakin, Andreas Metzger, and Marco Pistore. A framework for proactive self-adaptation of service-based applications based on online testing. In Petri Mähönen, Klaus Pohl, and Thierry Priol, editors, *ServiceWave*, volume 5377 of *Lecture Notes in Computer Science*, pages 122–133. Springer, 2008.

[HLM+10] Waldemar Hummer, Philipp Leitner, Anton Michlmayr, Florian Rosenberg, and Schahram Dustdar. *VRESCo - Vienna Runtime Environment for Service-oriented Computing*, pages 299–324. Service Engineering. European Research Results. Springer, 2010.

[LBCP09] Harold C. Lim, Shivnath Babu, Jeffrey S. Chase, and Sujay S. Parekh. Automated control in cloud computing: challenges and opportunities. In *Proceedings of the 1st workshop on Automated control for datacenters and clouds*, ACDC '09, pages 13–18, New York, NY, USA, 2009. ACM.

[LPRS07] David Linner, Heiko Pfeffer, Ilja Radusch, and Stephan Steglich. Biology as inspiration towards a novel service life-cycle. In Bin Xiao, Laurence Tianruo Yang, Jianhua Ma, Christian Müller-Schloer, and Yu Hua, editors, *ATC*, volume 4610 of *Lecture Notes in Computer Science*, pages 94–102. Springer, 2007.

[LR11] Stephen Lane and Ita Richardson. Process models for service-based applications: A systematic literature review. *Information & Software Technology*, 53(5):424–439, 2011.

[LWR+09] Philipp Leitner, Branimir Wetzstein, Florian Rosenberg, Anton Michlmayr, Schahram Dustdar, and Frank Leymann. Runtime prediction of service level agreement violations for composite services. In *ICSOC/ServiceWave Workshops*, pages 176–186, 2009.

[MGC+10] Dominik Meiländer, Sergei Gorlatch, Cinzia Cappiello, Valentina Mazza, Raman Kazhamiakin, and Antonio Bucchiarone. Using a lifecycle model for developing and executing adaptable interactive distributed applications. In *ServiceWave*, pages 175–186, 2010.

[Mit09] K. Mittal. Service oriented unified process, 2009.

[MSPR10] Andreas Metzger, Osama Sammodi, Klaus Pohl, and Mark Rzepka. Towards pro-active adaptation with confidence: augmenting service monitoring with online testing. In *Proceedings of the 2010 ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems*, SEAMS '10, pages 20–28, New York, NY, USA, 2010. ACM.

[PvdH06] Mike P. Papazoglou and Willem-Jan van den Heuvel. Service-oriented design and development methodology. *Int. J. Web Eng. Technol.*, 2(4):412–442, 2006.

[Rat03] Rational. Rational unified process - best practices for software development teams. Rational Software White Paper TP026B, Rev 11/01, December 2003.

[VH08] Samyr Vale and Slimane Hammoudi. Model driven development of context-aware service oriented architecture. In *Proceedings of the 2008 11th IEEE International Conference on Computational Science and Engineering - Workshops*, pages 412–418, Washington, DC, USA, 2008. IEEE Computer Society.

[WALK09] Yves Wautelet, Youssef Achbany, Jean-Charles Lange, and Manuel Kolp. A process for developing adaptable and open service systems: Application in supply chain management. In Joaquim Filipe and José Cordeiro, editors, *ICEIS*, volume 24 of *Lecture Notes in Business Information Processing*, pages 564–576. Springer, 2009.

2.2 Context modeling

Involved Workpackages: WP-JRA-1.1, WP-JRA-1.2

Several definitions and characterizations for the term context can be found in the literature and, along with these definitions, different context models have been proposed. Context models aim to provide a better understanding of the properties, aspects, categories, or dimensions related to context. It has been noticed that most of the approaches dealing with context models or context modeling are general in order to cover different scenarios. The approaches described here address context modeling in SBAs. They propose new context models to cover issues ranging from requirements elicitation to the development of context aware applications.

In [BKC+10] (FBK, Polimi) a context model has been proposed for the formalization of the most relevant aspects characterizing a Service-Based System (SBS). It consists of an eXtensible Markup Language (XML) representation of the main context components for SBSs featuring six main dimensions that are used to describe the status of an application. These dimensions include: (i) time, referring to the information about the time in which the system is accessed; (ii) ambient, related to space factors (e.g. address) or environmental conditions of the user; (iii) user, concerned with the privileges, roles, and preferences of users; (iv) service, related to information about the services in the system; (v) business, which takes into account business application factors; and (vi) computing, which specifies the available software and/or hardware characteristics.

An ontology-based context framework is presented in [BWP09b, BWP09a]. The work in [BWP09b] aims at facilitating the development and deployment of context-based web service applications (see also Section 2.1). The work in [BWP09a] aims at dynamically integrating context model-based constraints into web service processes. The context model is composed of four context types, namely Functional, Quality of Service (QoS), Domain, and Platform context. Functional context describes the operational features of services in terms of i) syntax: input and output parameters; ii) effect: pre and post-conditions; and iii) protocol: rules and data flow. QoS context deals with the needs, explicitly declared by the user and requirements not known to users. It includes i) runtime attributes: measurable properties; ii) financial/business attributes: assessment of a service from a financial perspective; iii) security attributes: whether the service is compliant with security requirements; and iv) trust attributes: relationship between clients and providers. The Domain describes each application in its own context including i) semantics: concepts and properties; ii) linguistics: the language used; and iii) measures and standards. The Platform context category describes the technical environment. It involves i) devices: computer/hw platform, and ii) connectivity: network infrastructure.

An integrated context model for business process management is presented in [WNL11]. The approach aims for an integrated view of context data belonging to context-aware services, workflows, human tasks and their interrelations (see also Section 2.5). The model consists of three parts: workflow model, service model, and task model. Each part is represented in terms of classes with sub-classes in some cases. For each class there are attributes describing states and context. The workflow model is concerned with the control flow of the application, including state of execution, processed data, and generated tasks. The service model is concerned with services, including supported operations, access protocol, and performed operations. The task model is concerned with human tasks, including action, duration, origin, and destinations.

A framework for context information provisioning is proposed in [BE11]. The framework relies on context service deployment on the cloud and use of context brokers to mediate between context consumers and context services using a publish/subscribe model (see Section 2.4). A multi-attribute decision algorithm is used for the selection of potential context services that can fulfill contexts consumers' requests for context information. The selection is made based on the QoS and Quality of Context (QoC) information requirements expressed by the context consumer (Section 2.14). Context information is processed according to who, where, when, what, and why a service was invoked. This is enough to respond to different situational circumstances, e.g. the identity of client who invoked a service, the activity that the client is carrying out at the time it invokes a service, and the device a client is using to invoke the service.

A service-based approach to develop context-aware automotive telematics systems is presented in [TCH10]. Telematic systems are automotive technologies combining advanced communications and vehicle technologies. The approach utilizes services as a means to acquire context and assist telematics in adapting contexts. A four-layer-architecture enables the separation of development of telematics and management of context adaptation (see Section 2.13). The first layer supports the operation and integration between hardware and sensors. The second layer wraps functions (from the first layer) as physical context services that expose standard Web Service Description Language (WSDL) descriptions. This layer also manages discovery and binding of external services provided by a third party, e.g. location services. The third layer includes social context models that represent relationships and interaction constraints between entities, as well as how interactions are affected by physical context facts. Each social context is modeled using a role-oriented adaptive design composition consisting of functional roles, interactions constraints, and organizer role. A social context model is implemented as a service that exposes WSDL interfaces corresponding to its functional role. The fourth layer includes context-aware telematics that use context services (social and physical) to "sense" context and adapt themselves in response to changes of context.

In [SC11] a context-based model for access control in mobile web services is proposed. The work combines semantic web technologies with context-based access control mechanisms. An ontology is used for modeling and reasoning about context, and specifying access control policies. The context model consists of four different context types covering access control: subject contexts, object contexts, transaction contexts, and environment contexts. In the case of subject contexts, a subject is an entity that takes action on an object or resource. Subject contexts define the specific subject-related contexts that must be held by a subject to obtain rights to use an object or resource. In the case of object contexts, an object is an entity that is influenced by a subject. Object contexts are any object-related information that can be used for characterizing the situation in which an object was created and information about its current status. In the case of transaction contexts, a transaction involves the user, the mobile platform, the specific resource or service, and the physical environment. Transaction contexts specify particular actions to be performed in the system. In the case of environment contexts, an environment describes the operational, technical, and situational environment at the time a transaction takes place.

Bibliography

[BE11] Elarbi Badidi and Larbi Esmahi. A cloud-based approach for context information provisioning. *CoRR*, abs/1105.2213, 2011.

[BKC+10] Antonio Bucchiarone, Raman Kazhamiakin, Cinzia Cappiello, Elisabetta di Nitto, and Valentina Mazza. A context-driven adaptation process for service-based applications. In *Pro-*

ceedings of the 2nd International Workshop on Principles of Engineering Service-Oriented Systems, PESOS '10, pages 50–56, New York, NY, USA, 2010. ACM.

[BWP09a] Kosala Yapa Bandara, Ming Xue Wang, and Claus Pahl. Dynamic integration of context model constraints in web service processes. In *International Software Engineering Conference (SE 2009)*, 2009.

[BWP09b] Kosala Yapa Bandara, MingXue Wang, and Claus Pahl. Context modeling and constraints binding in web service business processes. In *Proceedings of the first international workshop on Context-aware software technology and applications*, CASTA '09, pages 29–32, New York, NY, USA, 2009. ACM.

[SC11] Haibo Shen and Yu Cheng. A semantic context-based model for mobile web services access control. *International Journal of Computer Network and Information Security (IJCNIS)*, 3(1), 2011.

[TCH10] Minh H. Tran, Alan W. Colman, and Jun Han. Service-based development of context-aware automotive telematics systems. In Radu Calinescu, Richard F. Paige, and Marta Z. Kwiatkowska, editors, *ICECCS*, pages 53–62. IEEE Computer Society, 2010.

[WNL11] Matthias Wieland, Daniela Nicklas, and Frank Leymann. Context model for representation of business process management artifacts. In Chun Hua Lin and Ming Zhang, editors, *International Proceedings of Economics Development and Research: IPEDR*, volume 9 of *Economics and Business Information*, pages 46–51. IACSIT PRESS, Mai 2011.

2.3 Evolution

Involved Workpackages: WP-JRA-1.1, WP-JRA-1.2

Works on service evolution can be categorized based on how they deal with service compatibility. More specifically, on the one end of the spectrum there are approaches that do not consider whether changes to a service break the consumers of the service, preferring to remain as neutral as possible. On the other end, there are approaches that aim to enforce non-breaking changes of services to the extent that versioning of the service description can be simply subsumed under one version, the active one (i.e. deployed and running). Recent examples of the former type are the version-aware extensions of WSDL and Business Process Execution Language (BPEL) discussed in [JSBR09] and [JSR09], respectively, and the VRESCo service environment with its service registry encoding explicitly versioning history [LMRD08] (TUW).

Approaches that attempt to preserve the compatibility of a service usually fall in one of the following categories [And10]:

Corrective: adaptation-based approaches that actively enforce the non-breaking of existing consumers by modifying the service and as such they are more appropriately discussed in JRA-1.2, and

Preventive: approaches that attempt to confine and forbid changes that would disrupt the consumers (instead of fixing them).

Since the beginning of S-Cube, few works have been concerned with preventive evolution outside of the context of the network; the most important of them are [JD08, PM08, Flu08, BLM$^+$08]. All these works basically follow the guidelines of [BE04] to define backward compatibility preserving changes as a limited set of allowed service description modifications and propose different versioning methods and strategies depending on their goals.

Within the confines of S-Cube, a series of publications have appeared dealing with the issue of compatibility in service evolution (in a preventive manner) [ABP08, ABP09, PAB11] (Tilburg, UCBL) and [And10]. This body of work establishes a robust theoretical foundation for dealing with the evolution of software services. The focus is on controlling the development of the service so as to constrain the effect of changes to the service consumers and avoid disruption, inconsistencies and spurious results when utilized by its clients. The contribution to the State of the Art is threefold: firstly, it introduces a language-independent, technology-agnostic notation for the representation of services that transcends existing formalisms and languages. As such, the notation is equally applicable to different types of software services, from enterprise-level business services, to Cloud-related services. Secondly, the work develops a rigorous formal framework, consistency criteria and algorithms which control and delimit the evolution of services. The theoretical framework presented is at the cross-section of programming language theories, service oriented computing and software engineering. At the same time it calls into question the existing standards and support technologies as regards the facilities they provide for service change. To this effect, this provides a series of recommendations for the improvement of service-related technologies towards the direction of supporting service evolution. Finally, the works introduce a novel model of managing the evolution of services using bilateral agreements called service contracts between service providers and consumers. Explicit contracts between providers and consumers allow for greater flexibility in evolving both parties, but also allow for the evolution of the agreements themselves under clearly defined conditions.

Bibliography

[ABP08] Vasilios Andrikopoulos, Salima Benbernou, and Mike P. Papazoglou. Managing the evolution of service specifications. In Zohra Bellahsene and Michel Léonard, editors, *CAiSE*, volume 5074 of *Lecture Notes in Computer Science*, pages 359–374. Springer, 2008.

[ABP09] Vasilios Andrikopoulos, Salima Benbernou, and Mike P. Papazoglou. Evolving services from a contractual perspective. In *CAiSE*, pages 290–304, 2009.

[And10] Vasilios Andrikopoulos. *A Theory and Model for the Evolution of Software Services*. Number 262 in CentER Dissertation Series. Tilburg University Press, 2010.

[BE04] Kyle Brown and Michael Ellis. Best practices for web services versioning, January 2004.

[BLM+08] Karin Becker, Andre Lopes, Dejan S. Milojicic, Jim Pruyne, and Sharad Singhal. Automatically determining compatibility of evolving services. In *ICWS*, pages 161–168, 2008.

[Flu08] Greg Flurry. Service versioning in SOA, October 2008.

[JD08] Kjell-Sverre Jerijærvi and Jean-Jacques Dubray. Contract versioning, compatibility and composability, December 2008.

[JSBR09] Matjaz B. Juric, Ana Sasa, Bostjan Brumen, and Ivan Rozman. WSDL and UDDI extensions for version support in web services. *Journal of Systems and Software*, 82(8):1326–1343, 2009.

[JSR09] Matjaz B. Juric, Ana Sasa, and Ivan Rozman. Ws-bpel extensions for versioning. *Information & Software Technology*, 51(8):1261–1274, 2009.

[LMRD08] Philipp Leitner, Anton Michlmayr, Florian Rosenberg, and Schahram Dustdar. End-to-end versioning support for web services. In *IEEE SCC (1)*, pages 59–66, 2008.

[PAB11] Mike P. Papazoglou, Vasilios Andrikopoulos, and Salima Benbernou. Managing evolving services. *IEEE Software*, 28(3):49–55, 2011.

[PM08] Deepti Parachuri and Sudeep Mallick. Service versioning in SOA, December 2008.

2.4 Using the Cloud to Facilitate Global Software Development Challenges

Involved Workpackages: WP-JRA-1.1, WP-JRA-2.3

The cloud model is composed of three service models (Infrastructure as a Service (IaaS), Software as a Service (SaaS), and Platform as a Service (PaaS)), five essential characteristics, and four deployment models [BGK+11]. So in order to investigate the state of the art of the existing research in the domain, it is important to comprehend its different representations; that is, IaaS, PaaS, and SaaS. As their names suggest, different cloud based resources are provided in the form of services which further establishes their involvement with the services research. In IaaS, users can scale up and scale down computing resources dynamically in the form of computing power, storage, and network connectivity. Examples include Amazon EC2 and Microsoft Windows Azure Platform. PaaS provides a development platform as a service to assist application design, development, testing, and deployment. It usually requires no software downloads or installations. Google App Engine, Microsoft Azure, and Amazon Simple Storage Service are some of the examples of this representation of the cloud. With the rapid development of the Internet, users' requirements for computing and data storage are realized through the World Wide Web [ZZCH10]. SaaS is a software delivery paradigm where software is hosted off premise and is delivered via web to large number of tenants. A single software installation can cover many user needs as it is provided to end users on demand in the form of service. This software can be shared by multiple users and further enhancements and features can be requested on demand. The following section highlights the contribution S-Cube makes to the cloud computing research, and state of the existing research on the topic is also presented.

In S-Cube, cloud computing research has been mainly focused on IaaS and SaaS. The first work on IaaS [KKB09] (SZTAKI, TUW) targets using SLAs for the provision of non-functional guarantees, and the second one [BEM+10] (TUW, SZTAKI) aims at self-manageable cloud infrastructures which could avoid SLA violations by reacting actively to the changing conditions. Another work [MKKK11] (SZTAKI) on this domain proposes deployment of cloud services on various cloud infrastructures by using an automatic service deployment methodology. On the whole, the focus of the IaaS research has been on configuring the infrastructure in an optimal way for optimal provision of cloud services to end users. In terms of SaaS, [GKMW11] (Polimi, SZTAKI, FBK, USTUTT) proposed a multilayer framework for monitoring the requirements and use cases of complex service-based systems. The usefulness of the research could come into play while building a SaaS layer on top of a cloud infrastructure. Another work [MMLP09] (UniDUE, USTUTT) on SaaS tends to focus on customization of software as a service to meet varying functional and quality requirements of cloud tenants. The proposed approach makes use of software product line engineering domain.

This software can be shared by multiple users and further enhancements and features can be requested on demand. In the following section, we briefly look into the state of the existing research on cloud computing and potential contribution to the S-Cube project.

Building application on a software delivery platform changes the way software is designed, developed, and delivered [ECM08]. This application software is not intrinsically service oriented in the high performance cloud computing [HZG+10]. Moreover, the use of cloud brings in different challenges as well [BGK+11, KM10, ZZCH10]. Security, privacy, and trust have been a research concern for services [Zha08] and researchers have targeted the same issues for the cloud as well. In addition, research on the cloud computing domain has focused on license types & payments systems [JST+09, MS08], intended users' groups (for example private, corporate, and public users), security, privacy, and trust [KM10, Mic09], standardization, and formal agreements [ZZCH10].

In service-oriented environments, the primary focus is not software product but rather service provided by the software. Since Service Oriented Architecture (SOA) is considered to be as one of the main technical foundations of the cloud, researchers have been trying to bridge both of these to address issues with the existing cloud paradigm. [TSB10] Has proposed a service oriented cloud computing architecture which allows application to run on different clouds and thus supports application migration from one cloud to another one and service deployment to different clouds. The idea is to separate the roles of service provider and the cloud provider to promote an open platform with open standards. But research in the area not only has been focusing on creating a link between SOA and the cloud but also has been addressing different challenges posed by the cloud paradigm. Software has become a vital component of almost every business [KM10]. Though SaaS leverages web services as a technology foundation [JST+09], it does bring up attributes like scalability and multi-tenancy which are not addressed by SOA. After a software application has been developed and deployed, subscription is a key step that connects a cloud service provider with a consumer. Therefore, the design of a subscription management is critical for a SaaS provider to appropriately decompose its application into a service. Since the cloud adopts a usage based charging model, the client would subscribe to the service first in order to use it. Considering the importance of the topic, researchers have proposed pricing and subscription management for the cloud services [JST+09, MS08]. Clouds provide capacity to serve a wide and constantly expanding range of information processing needs [RMSG10]. Software development activities, for example testing capabilities have been proposed providing as a service. This service can provide users with testing services such as auto generation of test cases, their execution, and evaluation of the results [YTC+10, YZX+09, CBZ10].

One of the main objectives of the cloud computing is to provide resources to users and there could be four types of such resources [ZZ09]: infrastructure, software, applications, and business processes. Provision of these resources can benefit by supporting collaboration and communication among the corporate users. By considering these characteristics, the authors propose using the cloud paradigm to support Global Software Development (GSD) in which teams are distributed across multiple geographical locations [HM01]. The characteristics and the architecture of the cloud model itself has the potential to fulfill the GSD task requirements. In [HCR+11] (VUA, LERO, Polimi), the authors discuss that cloud deployment models allow certain trusted partners (which could be GSD team members) to share resources among themselves. Service models may not only provide access to collaboration and productivity tools but also allow network access to computing resources and the use as you go feature is likely to reduce the overall project costs across multiple development sites as computing resources and infrastructure is not required up-front.

Bibliography

[BEM+10] Ivona Brandic, Vincent C. Emeakaroha, Michael Maurer, Schahram Dustdar, Sandor Acs, Attila Kertesz, and Gabor Kecskemeti. LAYSI: A layered approach for SLA-violation propagation in self-manageable cloud infrastructures. In *Proceedings of the 2010 IEEE 34th Annual Computer Software and Applications Conference Workshops*, COMPSACW '10, pages 365–370, Washington, DC, USA, 2010. IEEE Computer Society.

[BGK+11] Michael Behrendt, Bernard Glasner, Petra Kopp, Robert Dieckmann, Gerd Breiter, Stefan Pappe, Heather Kreger, and Ali Arsanjani. IBM cloud computing reference 2.0 introduction and overview architecture. Technical report, 2011.

[CBZ10] George Candea, Stefan Bucur, and Cristian Zamfir. Automated software testing as a service. In Joseph M. Hellerstein, Surajit Chaudhuri, and Mendel Rosenblum, editors, *SoCC*, pages 155–160. ACM, 2010.

[ECM08] Javier Espadas, David Concha, and Arturo Molina. Application development over software-as-a-service platforms. In *ICSEA*, pages 97–104. IEEE Computer Society, 2008.

[GKMW11] Sam Guinea, Gabor Kecskemeti, Annapaola Marconi, and Branimir Wetzstein. Multi-layered monitoring and adaptation. In *Proceedings of the Ninth International Conference* *on Service Oriented Computing (ICSOC 2011); Paphos, Cyprus, December 5-8, 2011*. Springer, Dezember 2011.

[HCR+11] Sajid Hashmi, Viktor Clerc, Maryam Razavian, Christina Manteli, Damian Andrew Tamburri, Patricia Lago, Elisabetta Di Nitto, and Ita Richardson. Using the cloud to facilitate global software development challenges. Helsinki, Finland, 08/2011 2011.

[HM01] James D. Herbsleb and Deependra Moitra. Guest editors' introduction: Global software development. *IEEE Software*, 18(2), 2001.

[HZG+10] Zhengxiong Hou, Xingshe Zhou, Jianhua Gu, Yunlan Wang, and Tianhai Zhao. Asaas: Application software as a service for high performance cloud computing. In *HPCC*, pages 156–163. IEEE, 2010.

[JST+09] Zhong Bo Jiang, Wei Sun, Kai Tang, Jane L. Snowdon, and Xin Zhang. A pattern-based design approach for subscription management of software as a service. In *SERVICES I*, pages 678–685, 2009.

[KKB09] A. Kertesz, G. Kecskemeti, and I. Brandic. Autonomic resource virtualization in cloud-like environments. Technical Report TUV-1841-2009-04, Vienna University of Technology, 2009.

[KM10] Khaled M. Khan and Qutaibah M. Malluhi. Establishing trust in cloud computing. *IT Professional*, 12(5):20–27, 2010.

[Mic09] Bret Michael. In clouds shall we trust? *IEEE Security & Privacy*, 7(5):3, 2009.

[MKKK11] Attila Csaba Marosi, Gabor Kecskemeti, Attila Kertesz, and P. Kacsuk. FCM: an architecture for integrating IaaS cloud systems. In *Proceedings of The Second International Conference on Cloud Computing, GRIDs, and Virtualization*, 2011.

[MMLP09] Ralph Mietzner, Andreas Metzger, Frank Leymann, and Klaus Pohl. Variability modeling to support customization and deployment of multi-tenant-aware software as a service applications. In *Proceedings of the 2009 ICSE Workshop on Principles of Engineering Service Oriented Systems*, PESOS '09, pages 18–25, Washington, DC, USA, 2009. IEEE Computer Society.

[MS08] Dan Ma and Abraham Seidmann. The pricing strategy analysis for the "software-as-a-service" business model. In Jörn Altmann, Dirk Neumann, and Thomas Fahringer, editors, *GECON*, volume 5206 of *Lecture Notes in Computer Science*, pages 103–112. Springer, 2008.

[RMSG10] M. Rodriguez-Martinez, J. Seguel, and M. Greer. Open source cloud computing tools: A case study with a weather application. In *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, pages 443 –449, july 2010.

[TSB10] Wei-Tek Tsai, Xin Sun, and Janaka Balasooriya. Service-oriented cloud computing architecture. In Shahram Latifi, editor, *ITNG*, pages 684–689. IEEE Computer Society, 2010.

[YTC+10] Lian Yu, Wei-Tek Tsai, Xiangji Chen, Linqing Liu, Yan Zhao, Liangjie Tang, and Wei Zhao. Testing as a service over cloud. In *Proceedings of the 2010 Fifth IEEE International Symposium on Service Oriented System Engineering*, SOSE '10, pages 181–188, Washington, DC, USA, 2010. IEEE Computer Society.

[YZX+09] Lian Yu, Le Zhang, Huiru Xiang, Yu Su, Wei Zhao, and Jun Zhu. A framework of testing as a service. In *Management and Service Science, 2009. MASS '09. International Conference on*, pages 1 –4, sept. 2009.

[Zha08] Liang-Jie (LJ) Zhang. Eic editorial: Introduction to the knowledge areas of services computing. *IEEE Transactions on Services Computing*, 1:62–74, 2008.

[ZZ09] Liang-Jie Zhang and Qun Zhou. Ccoa: Cloud computing open architecture. In *ICWS*, pages 607–616, 2009.

[ZZCH10] Shuai Zhang, Shufen Zhang, Xuebin Chen, and Xiuzhen Huo. Cloud computing research and development trend. In *Future Networks, 2010. ICFN '10. Second International Conference on*, pages 93 –97, jan. 2010.

2.5 Human-Computer Interaction & Task Models

Involved Workpackages: WP-JRA-1.1

This section reports updates on approaches to exploit knowledge from Human-Computer Interaction (HCI) in Service Oriented Computing (SOC).

The state-of-the-art presented in [PO-08] (Tilburg, FBK, Polimi, UniDue, CITY) highlighted promising HCI topics for the enhancement of services and SBAs. Notable advances have been made for some of these such as User Interface generation, Task Modeling and Social Networks, while others have known a markedly slower development. Overall however, it may be said in the light of the surveyed material that a greater awareness and consideration of HCI concerns can be identified in the literature than was possible only a couple of years ago.

Accessibility The state of the art presented in [PO-08] (Tilburg, FBK, Polimi, UniDue, CITY) outlined the economic, legal and social necessity for the accessibility of software services to users with varying abilities. Obstacles towards this goal include the lack of guidance for developers without specialized accessibility knowledge and skills to navigate, identify and apply relevant accessibility guidelines to their application.

[GVT+10, GTH11] propose a framework and tools to help developers overcome this problem. The authors re-iterate and motivate the need to make SBAs more accessible and define and characterize an accessible Web service. Further, they describe a Web service accessibility assessment framework comprising sets of categorized accessibility guidelines (core, basic and extended), and specific techniques to verify Web service compliance to each guideline. The authors also propose a web service assessment tool allowing among others automatic evaluation of a Web service's accessibility according to the above-mentioned guidelines.

User Modeling, Customization and Personalization Extensive research has been carried in S-Cube on the topic of adaptation (e.g. see Section 2.6, Section 2.11, Section 2.12). From an HCI perspective, user modeling can provide important context information supporting services adaptation and personalization to end-users. It is however challenging to acquire a significant amount of information about a user's characteristics with minimum disruption to that user, or without relying on their repeated, explicit participation to the data collection which is far from guaranteed. Sharing and reusing existing user models (including partial ones) across applications can overcome this problem, but has its own set of challenges including data incompatibility for communication and exchange from system to system.

[CF09] address the interoperability issue preventing feature-based, user-adaptive applications from exchanging User Modeling (UM) knowledge. The authors suggest that the loosely coupled structure and the standards underlying Web Services (Simple Object Access Protocol (SOAP), WSDL and Universal

Description, Discovery and Integration (UDDI)) provide a solution to the UM syntactic interoperability issue, and propose a framework supporting a discovery mechanism, a pattern of communication, and a dialogue model for the cooperation of user-adaptive systems. In this approach, user-adaptive systems make their UMs available as services and provide to each other a set of WSDL operations to access this data. A framework architecture uses among others a central UDDI registry enhanced with UM specific capabilities for the systems to publish their description.

In S-Cube, [CZ10] (CITY) address the modeling of user information and propose an XML format model and an ontology to represent different types of information about users that interact with service based systems (user context) - for instance user characteristics such as role, skills and preferences (direct user context types), and related user context types such as time, location, and environment that may influence user information. This work is relatable to the context modeling research outlined in Section 2.2 and Section 2.12, i.e. Bucchiarone et al. run-time context-aware adaptation approach [BPRK].

Perhaps unsurprisingly considering services' focus on reuse, no advances were registered on the topic of *customization* – the design and creation of content that meets a customer's specific needs. However, a new contribution to the literature on *personalization* in the context of SOA was sourced.

[Cas10] addresses the challenge of providing services that accommodate a user's specific context. The author envisages a Service Personalization ARchitecture (SPARK) incorporating ontology based service descriptions, probabilistic semantic approaches and machine learning techniques to enable machines to understand different services functionality, and consequently provide best-matching-service discovery, composition, and/or recommendation as fitting for a end user's specific preferences and goals.

User Interface Generation [PO-08] (Tilburg, FBK, Polimi, UniDue, CITY) presented motivations and challenges for providing high quality User Interfaces (UIs) for services. A trend towards End User Development (EUD) using graphical user interface-focused approaches to enable it seem to have developed since, and an explicit recourse to HCI guidelines for the development of usable UIs can also be noted.

[Cas11] reviews issues behind past failures of composition technologies (including service composition), and makes a case to empower end user to develop applications. The author suggests EUD as a key driver for improving composition platforms and tools, and hints at principles and proposals including more user-friendly composition paradigms to render EUD feasible for composition-based applications.

[NDP09] propose one such innovative paradigm for the automatic generation of service UIs and a simpler, "WYSIWYG" (What You See Is What You Get) composition of Web applications at the presentation layer. They outline a user-centric approach to graphically model and create simple SBAs using the concept of service annotations: reusable, formally defined information fragments that provide additional information about a Web service – here, UI-related visual and behavioural information. The ServFace Builder, a web-based design-time authoring tool, uses consolidated HCI user interface design guidelines, the service description, and UI-related service annotations to automatically generate a UI (a service front-end) for each operation that a user acting as a service composer adds to his application. The user can then compose services using these front-ends rather than the application logic or data, this without requiring technical knowledge about service composition [DFN+10, NDP09, FJN+09, Con].

[LSRH09] also address the challenge of EUD for unskilled users. The authors report current web-based service front-ends as inadequate due to a lack of formal engineering techniques, and stress the ubiquity of HCI as a key feature of an Internet of Services for anytime, anywhere access to flexible, user-friendly services. As a step towards a user-centric SOA conceived as an Internet of services enabling users with no programming skills to co-produce and share composed applications, they outline a SOA-oriented framework built upon a visual gadget programming environment (FAST), a mashup platform to build small enterprise applications (EzWeb), and design principles including context-aware, adaptive interaction and end-user knowledge exploitation to improve service front-ends.

In contrast, other work in the area addresses professional or skilled programmers and propose methods and tooling to support them engineering SBAs and/or UIs.

[PSS10a, PSS11] aim to address misunderstandings about service functionalities and interaction that arise due to the fact that SBA developers are often not the services developers. The authors discuss a selection of HCI models, their relations to services and how they can support SBA engineering, and propose an HCI-based environment to generate usable service front-ends. Here again they use service annotations, and present a method to implement a concrete interface from a task model and an associated tool supporting the management and use of Web services and their annotations for UI design and development.

[DST+10] also target SBA developers, more specifically for the engineering of applications that require service composition, process automation logic, and include human interaction using a complex UI tailored to support a specific task. They outline an approach to bring together UIs, Web services and people in a single orchestration logic and tool. The authors describe: a composition paradigm based on web-based UI components and services as building blocks; BPEL4UI, an extension to BPEL with UI-specific constructs to model UI orchestrations; and MarcoFlow, a system for the design and deployment of a UI orchestration.

Task Modeling Although many business processes involve human tasks, taking into account or modeling these interactions is still challenging using approaches currently in practice. Since the state of the art [PO-08] (Tilburg, FBK, Polimi, UniDue, CITY) proposed HCI task models as a way forward to address this issue, a few developments related to user tasks in SOA have been noted.

[ZLS+10] address the limitations of notations such as BPEL4People, and model human tasks as services using Human-Interaction Services (HTSs), a type of Web service handling presentation logics and coordinating the Web services with business functionalities. The approach uses standard BPEL processes, and automatically generates rich user interfaces for the HTSs using a web browser-based mashup toolkit implementing a component-based, on-the-fly mashup composition builder and runtime.

Previously referenced work by Paternó et al. [PSS10b, PSS] exploits task modeling and delivers an environment to support tasks and services matching using ConcurTaskTree (CTT) models to develop user interfaces.

[KP10] use user task model for service discovery. The authors make use of a domain ontology and designer-provided CTT user task models to produce service models specifying service combinations that realize application functionality. The approach generates service discovery queries from each system task description that are executed to discover, categorize and rank services based on a textual similarity measure.

As outlined in more details in Section 2.2, [WNL11] include a task model concerned with human tasks including action, duration, origin, and destinations as part of their integrated context model for business process management.

Finally, in S-Cube [CD-10] (CITY, LERO, Polimi, VUA) propose an approach to task-based service discovery that exploits reusable task structures comprising a user task and goal description, CTT task model, and class descriptions for corresponding service solution(s). The authors present a task-based service discovery algorithm for the discovery of best-fit services relevant to an end-user task: the algorithm essentially discover services by matching terms describing the user problem to related user task models that, in turn, are used to reformulate the service queries. Although this approach like [KP10] require non-automated processes (e.g. for the design of task models) which limits somehow their cost-effectiveness, the convergence of ideas and of chosen task modeling notations serves as an indicator of the potential of these approaches for enhancements to SBA engineering.

Social Networks The state of the art presented in [PO-08] (Tilburg, FBK, Polimi, UniDue, CITY) suggested that aspects of social networks may be applied to the distribution, discoverability and recommendations of services. Academic research carried since seem to be progressing in this direction as visible from the contributions identified in the literature. For instance, [MHH11] suggest that social networks, with their underlying principles and metrics, can offer innovative solutions to some of the issues services (and more specifically, Web services) face today, and propose incorporating social elements into web service operations to support SOA needs such as description, discovery and composition. The authors advocate consideration of past experiences (as opposed to information exclusively related to Web services), helping users become proactive, and stressing the interactions that occur between users, between Web services, and between users and Web services. Consequently, they propose to put in place a social network-based strategy for Web service management to offer better exposure, use, and follow-up of Web services.

[MB09] address the challenge of trust establishment for Web services. The authors propose RATEWeb, a Reputation Assessment framework, and develop a community ontology that serves as a template for describing communities and Web services. They also describe techniques based on a community model for ratings collection in order to support trust-based selection and composition of Web services: Web services share feedback ratings of service providers with their peers, collectively building a provider's reputation used to evaluate trust. This ultimately helps them determine to which extent they may trust other services to provide a required functionality before they interact with them.

[MFB+11] apply social networking concepts to enhance Web service discovery. The authors propose to group similar Web services into communities and discuss necessary set up, structuring and management to help service discovery. They report two behaviors taking place within a community: competition as the Web services in a community vie to participate in Web services compositions, and cooperation as they contribute together to the attractiveness and growth of their community – for instance, by maintaining a good reputation level and substituting for each other in case one of them fails. The authors propose guidelines and a community architecture to support these behaviors. They outline protocols for the establishment and dismantlement of communities, and introduce the concept of a master component (possibly implemented as a broker Web service) deployed to lead and oversee events (e.g. new service arrival, service departure) once the functionality of the community has been defined. Finally, the authors describe a reputation system to support competition between communities of web services offering the same functionality.

[EM11] also address service discovery; the authors suggest that the interactions among Web services construct a network of relationships leading to a social network of services, which they characterize. They describe a search algorithm for these networks in order to select an effective set of services that can collaborate to attain the main goal of the SBA at hand. Finally, [MHDC10] apply social network analysis to enhance service composition using the Social Composer (SoCo), a framework providing dynamic recommendations for services discovery and selection based on the users' interactions and information from his social networks.

Bibliography

[BPRK] A. Bucchiarone, M. Pistore, H. Raik, and R. Kazhamiakin. Adaptation of service-based business processes by context-aware replanning. Submitted to SOCA 2011.

[Cas10] Gilbert Cassar. Spark: A service personalisation architecture. In *PerCom Workshops*, pages 857–858. IEEE, 2010.

[Cas11] Fabio Casati. How end-user development will save composition technologies from their continuing failures. In Maria Francesca Costabile, Yvonne Dittrich, Gerhard Fischer, and Antonio Piccinno, editors, *IS-EUD*, volume 6654 of *Lecture Notes in Computer Science*, pages 4–6. Springer, 2011.

[CD-10] Analysis on how to exploit codified HCI and codified context knowledge for SBA engineering. Deliverable, S-Cube Consortium, 2010.

[CF09] Federica Cena and Roberto Furnari. A model for feature-based user model interoperability on the web. In Tsvi Kuflik, Shlomo Berkovsky, Francesca Carmagnola, Dominik Heckmann, and Antonio Krüger, editors, *Advances in Ubiquitous User Modelling*, volume 5830 of *Lecture Notes in Computer Science*, pages 37–54. Springer, 2009.

[Con] ServFace Consortium. Servface research project. http://www.servface.eu.

[CZ10] Ricardo Contreras and Andrea Zisman. A pattern-based approach for monitor adaptation. In *SwSTE*, pages 30–37. IEEE Computer Society, 2010.

[DFN+10] Lars Dannecker, Marius Feldmann, Tobias Nestler, Gerald Hübsch, Uwe Jugel, and Klemens Muthmann. Rapid development of composite applications using annotated web services. In Florian Daniel and Federico Michele Facca, editors, *ICWE Workshops*, volume 6385 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2010.

[DST+10] Florian Daniel, Stefano Soi, Stefano Tranquillini, Fabio Casati, Chang Heng, and Li Yan. From people to services to ui: Distributed orchestration of user interfaces. In Richard Hull, Jan Mendling, and Stefan Tai, editors, *BPM*, volume 6336 of *Lecture Notes in Computer Science*, pages 310–326. Springer, 2010.

[EM11] Said Elnaffar and Zakaria Maamar. Composite web services formation using a social network of web services: A preliminary investigation. *Procedia CS*, 5:466–471, 2011.

[FJN+09] Marius Feldmann, Jordan Janeiro, Tobias Nestler, Gerald Hübsch, Uwe Jugel, André Preußner, and Alexander Schill. An integrated approach for creating service-based interactive applications. In Tom Gross, Jan Gulliksen, Paula Kotzé, Lars Oestreicher, Philippe A. Palanque, Raquel Oliveira Prates, and Marco Winckler, editors, *INTERACT (2)*, volume 5727 of *Lecture Notes in Computer Science*, pages 896–899. Springer, 2009.

[GTH11] Dimitrios Giakoumis, Dimitrios Tzovaras, and George Hassapis. Enabling accessibility characteristics in the web services domain. In Constantine Stephanidis, editor, *HCI (5)*, volume 6765 of *Lecture Notes in Computer Science*, pages 177–186. Springer, 2011.

[GVT+10] D. Giakoumis, K. Votis, D. Tzovaras, S. Likothanassis, and G. Hassapis. Introducing accessibility in the web services domain. In *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*, volume 2, pages 18 –22, july 2010.

[KP10] Kyriakos Kritikos and Fabio Paternò. Task-driven service discovery and selection. In Santucci [San10], pages 89–92.

[LSRH09] David Lizcano, Javier Soriano, Marcos Reyes, and Juan José Hierro. A user-centric approach for developing and deploying service front-ends in the future internet of services. *IJWGS*, 5(2):155–191, 2009.

[MB09] Zaki Malik and Athman Bouguettaya. Rateweb: Reputation assessment for trust establishment among web services. *VLDB J.*, 18(4):885–911, 2009.

[MFB+11] Zakaria Maamar, Noura Faci, Youakim Badr, Leandro Krug Wives, Pédro Bispo dos Santos, Djamal Benslimane, and José Palazzo Moreira de Oliveira. Towards a framework for weaving social networks principles into web services discovery. In Rajendra Akerkar, editor, *WIMS*, page 51. ACM, 2011.

[MHDC10] Abderrahmane Maaradji, Hakim Hacid, Johann Daigremont, and Noël Crespi. Towards a social network based approach for services composition. In *ICC*, pages 1–5. IEEE, 2010.

[MHH11] Zakaria Maamar, Hakim Hacid, and Michael N. Huhns. Why web services need social networks. *IEEE Internet Computing*, 15(2):90–94, 2011.

[NDP09] Tobias Nestler, Lars Dannecker, and Andreas Pursche. User-centric composition of service front-ends at the presentation layer. In *ICSOC/ServiceWave Workshops*, pages 520–529, 2009.

[PO-08] State of the art report on software engineering design knowledge and survey of HCI and contextual knowledge. Deliverable, S-Cube Consortium, 2008.

[PSS] Fabio Paternò, Carmen Santoro, and Lucio Davide Spano. ConcurTaskTrees and MARIA languages for authoring service-based applications. http://www.w3.org/2010/02/mbui/soi/paterno-1.pdf.

[PSS10a] Fabio Paternò, Carmen Santoro, and Lucio Davide Spano. Exploiting web service annotations in model-based user interface development. In Noi Sukaviriya, Jean Vanderdonckt, and Michael Harrison, editors, *EICS*, pages 219–224. ACM, 2010.

[PSS10b] Fabio Paternò, Carmen Santoro, and Lucio Davide Spano. User task-based development of multi-device service-oriented applications. In Santucci [San10], page 407.

[PSS11] Fabio Paternò, Carmen Santoro, and Lucio Davide Spano. Engineering the authoring of usable service front ends. *Journal of Systems and Software*, 84(10):1806–1822, 2011.

[San10] Giuseppe Santucci, editor. *Proceedings of the International Conference on Advanced Visual Interfaces, AVI 2010, Roma, Italy, May 26-28, 2010*. ACM Press, 2010.

[WNL11] Matthias Wieland, Daniela Nicklas, and Frank Leymann. Context model for representation of business process management artifacts. In Chun Hua Lin and Ming Zhang, editors, *International Proceedings of Economics Development and Research: IPEDR*, volume 9 of *Economics and Business Information*, pages 46–51. IACSIT PRESS, Mai 2011.

[ZLS+10] Qi Zhao, Xuanzhe Liu, Dawei Sun, Tiancheng Liu, Ying Li, and Gang Huang. Mashing-up rich user interfaces for human-interaction in ws-bpel. In *ICWS*, pages 559–566. IEEE Computer Society, 2010.

2.6 Software Processes for Adaptable Systems

Involved Workpackages: WP-JRA-1.1, WP-JRA-1.2

Several approaches have been proposed to aid in the development of adaptable SBAs. Most of these approaches focus on a particular area of adaptation with few providing holistic processes.

Life-cycles that support adaptation include the BEA reference lifecycle [DGK+], it is a complete development life-cycle for SBAs that also supports some aspects of adaptation. It provides guidelines for the service monitoring and runtime correctness analysis. The Web Services Development Methodology (SLDC) [PvdH06] is another complete development life-cycle that somewhat supports adaptation. It is an iterative life-cycle with processes for Analysis, Design, Construction, Testing, Provisioning, Deployment, Execution and Monitoring. It has adaptation activities for QoS monitoring and for providing notifications of compliance failures.

Trainotti et al. [TPC+05] propose the ASTRO approach which supports the automated composition of business services. This is not a life-cycle, rather a process and tool-set for specific aspects of adaptation. ASTRO provides tools for generating compositions, monitoring the compositions and tools which display graphical representations of the executing services. The SeCSE methodology [ATO07] also provides tools for the adaptation of SBAs. It provides tools for service binding/rebinding, run-time service composition and recovery management.

There are several other processes reported in the literature that also facilitate adaptation in one way or another. In order to get a complete set of processes available Lane and Richardson [LR11] (Lero) conducted a systematic literature review using the method proposed in [KC07]. They identified 8 processes in their study that satisfied the criteria of facilitation adaptation.

Four of the approaches support adaptation through adaptation of the services themselves [ABC+07, kANK09, CK07, BVB09], while the others support adaptation at the application level [OYP05, VH08, WHA+09]. Four of them are also based on other models while four are bespoke. Finally, half of the models identified employed the model driven development approach while the others did not.

The S-Cube partners have built on the state of the art and created a holistic design time and runtime adaptation reference model [LBR11] (Lero, FBK). The model used the state of the art to construct a skeleton reference model that was enhanced with data collected during an empirical investigation. The empirical investigation consisted of interviews with 20 service practitioners during an expert opinion and a case study. The data from the interviews was transcribed and coded to identify process attributes that could be used to construct the model.

In order to validate the model it was compared to a related component-oriented adaptation approach in order to verify its capability to adapt an application. In order to see that the model is suitable for use in a real project scenario it was mapped to a development approach that has been widely used by service development practitioners. Future validation of the model will include a survey with an expert panel which will give a better view of the models generalisability.

Bibliography

[ABC+07] Marco Autili, Luca Berardinelli, Vittorio Cortellessa, Antinisca Di Marco, Davide Di Ruscio, Paola Inverardi, and Massimo Tivoli. A development process for self-adapting service oriented applications. In *ICSOC*, pages 442–448, 2007.

[ATO07] ATOS. SeCSE methodology, version 3, 2007.

[BCT05] Boualem Benatallah, Fabio Casati, and Paolo Traverso, editors. *Service-Oriented Computing - ICSOC 2005, Third International Conference, Amsterdam, The Netherlands, December 12-15, 2005, Proceedings*, volume 3826 of *Lecture Notes in Computer Science*. Springer, 2005.

[BVB09] Khouloud Boukadi, Lucien Vincent, and Patrick Burlat. Modeling adaptable business service for enterprise collaboration. In Luis M. Camarinha-Matos, Iraklis Paraskakis, and Hamideh Afsarmanesh, editors, *PRO-VE*, volume 307 of *IFIP Conference Proceedings*, pages 51–60. Springer, 2009.

[CK07] Soo Ho Chang and Soo Dong Kim. A service-oriented analysis and design approach to developing adaptable services. In *IEEE SCC*, pages 204–211. IEEE Computer Society, 2007.

[DGK+] Surekha Durvasula, Martin Guttmann, Ashok Kumar, Jeffery Lamb, Tom Mitchell, Burc Oral, Yogish Pai, Tom Sedlack, Harsh Sharma, and Sankar Ram Sundaresan. SOA practitioners' guide, part 3.

[kANK09] Adil kenzi, Bouchra El Asri, Mahmoud Nassar, and Abdelaziz Kriouile. A model driven framework for multiview service oriented system development. *Computer Systems and Applications, ACS/IEEE International Conference on*, 0:404–411, 2009.

[KC07] Barbara Kitchenham and Stuart Charters. Guidelines for performing systematic literature reviews in software engineering. Technical Report EBSE 2007-001, Keele University and Durham University Joint Report, 2007.

[LBR11] S. Lane, A. Bucchiarone, and I. Richardson. A process reference model for developing adaptable service-based applications, 2011. TBD.

[LR11] Stephen Lane and Ita Richardson. Process models for service-based applications: A systematic literature review. *Information & Software Technology*, 53(5):424–439, 2011.

[OYP05] Bart Orriëns, Jian Yang, and Mike P. Papazoglou. A rule driven approach for developing adaptive service oriented business collaboration. In Benatallah et al. [BCT05], pages 61–72.

[PvdH06] Mike P. Papazoglou and Willem-Jan van den Heuvel. Service-oriented design and development methodology. *Int. J. Web Eng. Technol.*, 2(4):412–442, 2006.

[TPC+05] Michele Trainotti, Marco Pistore, Gaetano Calabrese, Gabriele Zacco, Gigi Lucchese, Fabio Barbon, Piergiorgio Bertoli, and Paolo Traverso. ASTRO: Supporting composition and execution of web services. In Benatallah et al. [BCT05], pages 495–501.

[VH08] S. Vale and S. Hammoudi. Model driven development of context-aware service oriented architecture. In *Computational Science and Engineering Workshops, 2008. CSEWORKSHOPS '08. 11th IEEE International Conference on*, pages 412 –418, july 2008.

[WHA+09] Martin Wirsing, Matthias Hölzl, Lucia Acciai, Federico Banti, Allan Clark, Alessandro Fantechi, Stephen Gilmore, Stefania Gnesi, László Gönczy, Nora Koch, Alessandro Lapadula, Philip Mayer, Franco Mazzanti, Rosario Pugliese, Andreas Schroeder, Francesco Tiezzi, Mirco Tribastone, and Dániel Varró. Sensoria patterns: Augmenting service engineering with formal analysis, transformation and dynamicity. In Tiziana Margaria and Bernhard Steffen, editors, *Leveraging Applications of Formal Methods, Verification and Validation*, volume 17 of *Communications in Computer and Information Science*, pages 170–190. Springer Berlin Heidelberg, 2009.

2.7 Service-Oriented Architecture Migration

Involved Workpackages: WP-JRA-1.1

One of the key promises of service oriented paradigm is facilitating reuse of enterprise assets in legacy systems. Migration of legacy systems to service-based systems enables achieving advantages offered by SOA while still reusing the embedded capabilities in the legacy systems. To obtain the SOA Migration categories in the field, a systematic review that extracts migration categories existing in the field was conducted and reported in [RL10]. Each of these categories called SOA migration families are described in the following.

Code Translation: Migration, in code translation family, entails moving the legacy system as a whole to a service-oriented platform or technology, without decomposing the existing system. Approaches that fall in this category either translate the whole code to web services or wrap the whole application as a web service [BT03, ZBZ+08, Hua09, CMX10].

Service Identification: This family of approaches focuses on the identification of the candidate services in the existing legacy system using reverse engineering techniques [OSL05, CZL+09, SP08, LT08, IZH08, CWBP09, ZZYZ10].

Business Model Transformation: This family supports decision making on the migration process itself. Due to the orthogonal view of this category on the migration process, this category is recognized as a "meta-process" [LBLA04, LS06, HKW+07, UZ09, CAD07].

Design Element Transformation: Migration in this family is limited to reshaping the existing legacy elements to the service-based elements [APK+08, GPGF09, GGCY05, HA08, HTZZ08, LHYC07, LAN+09, Sne06, ZK00]. More precisely, a set of legacy elements, extracted using "code analysis" or simply known beforehand, are transferred to a set of services or service-based elements.

Forward Engineering: The main focus of this family is on development of service-based systems starting from the desired business processes [CAO+07, CLC05, MHV10, KMPV10, NGvD10]. Migration here entails Top-Down service-based development while locating the realization of the required business functionalities and transforming them to services.

Design and Composite Element Transformation: Migration in this family embraces recovering and refactoring of the legacy architecture to the service-oriented architecture as well as reshaping the legacy elements to service-based elements [CFFT08, CGDG08, HCM+08, WZZ08, ZYC06].

Pattern-based Composition Transformation: A common feature in this family is using "patterns" for transforming the existing architecture to service-based architecture [ATZ08, KS08, PB06].

Forward Engineering with Gap Analysis Family: What distinguishes this family from pure top-down service development approaches is that at different abstraction levels a comparison (a gap analysis) among the new and pre-existing artifacts occurs [NvdHP+09a, AZR10, WZWZ10, ZYZZ10]. This comparison serves to assess how the desired business services can be realized by exploiting pre-existing capabilities.

A chronological perspective on the distribution of the primary studies among families brings insight about the trends in the field. These trends are fundamentally two, treated in the following.

Generally families centering forward engineering are more recent. The "Forward Engineering with Gap Analysis Family" family appeared in early 2009, by an S-Cube contribution reported in [NvdHP+09a, NvdHP+09b]. By early 2011 this family has 4 members that indicates an increasing interest on this type of migration. In the same vein, the "Forward engineering family" had gained considerably more attention since early 2009. In our view, this shift form reverse engineering to forward engineering is in line with the shift in how SOA is perceived: from a technology enabler to a software engineering paradigm [Sho10]. More precisely, in the SOA hype organizations perceived SOA as a technology enabler that facilitates integrating their existing applications, while now SOA is perceived as a paradigm that guides the development of enterprise services from existing assets.

Migration, inherently, embraces a trade-off analysis between the leveraging the legacy assets (i.e. "As-Is state") and achieving the promises of SOA (i.e. "To-Be state"). Based on what drives the migration one could categorize the migration approaches to the following to types: the ones driven by "As-Is state" and the ones driven by "To-Be" state [MR11]. The analysis of migration approaches finds that the ones being published before 2009 were mainly driven by "As-Is state" (almost 80%). This indicates an important gap between research and practice as majority of the industrial approaches are "To-Be driven". From 2009 onwards, however, the "To-Be driven" approaches gained increasing focus (i.e. 40% of approaches). It should be noted that "To-Be driven" approaches has been acknowledged as a best practice that enables achieving the migration goals. This highlights promising steps for research to support industry-relevant research.

Bibliography

[APK+08] Kyeongrim Ahn, Sangpil Park, HeeSoo Kim, Jongryel Kim, and Jungchun Park. The design of soa-based business collaboration system. In Jinhwa Kim, Dursun Delen, Jinsoo Park, Franz Ko, and Yun Ji Na, editors, *NCM (2)*, pages 707–712. IEEE Computer Society, 2008.

[ATZ08] Francesca Arcelli, Christian Tosi, and Marco Zanoni. Can design pattern detection be useful for legacy system migration towards SOA? In *workshop on Systems development in SOA environments*, pages 63–68, 2008.

[AZR10] Saad Alahmari, Ed Zaluska, and David De Roure. A service identification framework for legacy system migration into SOA. In *IEEE SCC*, pages 614–617, 2010.

[BT03] Thierry Bodhuin and Maria Tortorella. Using grid technologies for web-enabling legacy systems. In *STEP*, pages 186–195. IEEE Computer Society, 2003.

[CAD07] Sam Chung, Joseph Byung Chul An, and Sergio Davalos. Service-oriented software reengineering: Sosr. In *HICSS*, page 172. IEEE Computer Society, 2007.

[CAO+07] Semih Cetin, N. Ilker Altintas, Halit Oguztuzun, Ali H. Dogru, Ozgur Tufekci, and Selma Suloglu. A mashup-based strategy for migration to service-oriented computing. In *Pervasive Services, IEEE International Conference on*, pages 169–172, 2007.

[CFFT08] Gerardo Canfora, Anna Rita Fasolino, Gianni Frattolillo, and Porfirio Tramontana. A wrapping approach for migrating legacy system interactive functionalities to service oriented architectures. *Journal of Systems and Software*, 81(4):463–480, 2008.

[CGDG08] Félix Cuadrado, Boni García, Juan C. Dueñas, and Hugo A. Parada G. A case study on software evolution towards service-oriented architecture. In *AINA Workshops*, pages 1399–1404. IEEE Computer Society, 2008.

[CLC05] Feng Chen, Shaoyun Li, and William Cheng-Chung Chu. Feature analysis for service-oriented reengineering. In *APSEC*, pages 201–208. IEEE Computer Society, 2005.

[CMX10] Guo Chenghao, Wang Min, and Zhou Xiaoming. A wrapping approach and tool for migrating legacy components to web services. In *Networking and Distributed Computing (ICNDC), 2010 First International Conference on*, pages 94–98, 2010.

[CWBP09] Sam Chung, Daehee Won, Seung-Ho Baeg, and Sangdeok Park. Service-oriented reverse reengineering: 5w1h model-driven re-documentation and candidate services identification. In *SOCA*, pages 1–6, 2009.

[CZL+09] Feng Chen, Zhuopeng Zhang, Jianzhi Li, Jian Kang, and Hongji Yang. Service identification via ontology mapping. In Sheikh Iqbal Ahamed, Elisa Bertino, Carl K. Chang, Vladimir Getov, Lin Liu, Hua Ming, and Rajesh Subramanyan, editors, *COMPSAC (1)*, pages 486–491. IEEE Computer Society, 2009.

[GGCY05] He Guo, Chunyan Guo, Feng Chen, and Hongji Yang. Wrapping client-server application to web services for internet computing. In *PDCAT*, pages 366–370. IEEE Computer Society, 2005.

[GPGF09] Miguel Ángel Conde González, Francisco José García Peñalvo, María José Casany Guerrero, and Marc Alier Forment. Adapting lms architecture to the soa: An architectural approach. In *ICIW*, pages 322–327, 2009.

[HA08] A.N. Haidar and A.E. Abdallah. Composition and customization of web services using wrappers: A formal approach based on CSP. In *Software Engineering Workshop, SEW '08.*, pages 187–194, 2008.

[HCM+08] Reiko Heckel, Rui Correia, Carlos M. P. Matos, Mohammad El-Ramly, Georgios Koutsoukos, and Luis Filipe Andrade. Architectural transformations: From legacy to three-tier and services. In Tom Mens and Serge Demeyer, editors, *Software Evolution*, pages 139–170. Springer, 2008.

[HKW+07] John Hutchinson, Gerald Kotonya, James Walkerdine, Peter Sawyer, Glen Dobson, and Victor Onditi. Evolving existing systems to service-oriented architectures: Perspective and challenges. In *ICWS*, pages 896–903, 2007.

[HTZZ08] He Yuan Huang, Huafang Tan, Jun Zhu, and Wei Zhao. A lightweight approach to partially reuse existing component-based system in service-oriented environment. In Hong Mei, editor, *ICSR*, volume 5030 of *Lecture Notes in Computer Science*, pages 245–256. Springer, 2008.

[Hua09] Gengjie Huang. Study and design of enterprise information-based system based on SOA. In *Computer Science & Education, ICCSE '09.*, pages 933–936, 2009.

[IZH08] N. Ilk, J.L. Zhao, and P. Hofmann. On reuse of source code components in modernizing enterprise systems. In *Advanced Management of Information for Globalized Enterprises*, pages 1–5, 2008.

[KMPV10] V. Koufi, F. Malamateniou, D. Papakonstantinou, and G. Vassilacopoulos. Using ESB and BPEl for evolving healthcare systems towards pervasive, grid-enabled SOA. In George A. Papadopoulos, Wita Wojtkowski, Gregory Wojtkowski, Stanislaw Wrycza, and JoÂ¿e Zupancic, editors, *Information Systems Development*, chapter 18, pages 167–175. Springer US, Boston, MA, 2010.

[KS08] Kalapriya Kannan and Biplav Srivastava. Promoting reuse via extraction of domain concepts and service abstractions from design diagrams. In *IEEE SCC (1)*, pages 265–272, 2008.

[LAN+09] Zhang Li, Xie Anming, Zhang Naiyue, Hu Jianbin, and Chen Zhong. A SOA modernization method based on tollgate model. In *Information Engineering and Electronic Commerce, 2009. IEEC '09. International Symposium on*, pages 285–289, 2009.

[LBLA04] Janet Lavery, Cornelia Boldyreff, Bin Ling, and Colin Allison. Modelling the evolution of legacy systems to web-based systems. *Journal of Software Maintenance*, 16(1-2):5–30, 2004.

[LHYC07] Shing-Han Li, Shi-Ming Huang, David C. Yen, and Cheng-Chun Chang. Migrating legacy information systems to web services architecture. *J. Database Manag.*, 18(4):1–25, 2007.

[LS06] Grace A. Lewis and Dennis B. Smith. Developing realistic approaches for the migration of legacy components to service-oriented architecture environments. In Dirk Draheim and Gerald Weber, editors, *TEAA*, volume 4473 of *Lecture Notes in Computer Science*, pages 226–240. Springer, 2006.

[LT08] Shimin Li and Ladan Tahvildari. E-bus: a toolkit for extracting business services from java software systems. In Wilhelm Schäfer, Matthew B. Dwyer, and Volker Gruhn, editors, *ICSE Companion*, pages 961–962. ACM, 2008.

[MHV10] JJ Mulcahy, Shihong Huang, and AB Veghte. Leveraging service-oriented architecture to extend a legacy commerce system. In *Systems Conference, 2010 4th Annual IEEE*, pages 243–248, 2010.

[MR11] Patricia Lago Maryam Razavian. A survey of SOA migration in industry. In *ICSOC 2011*, 2011.

[NGvD10] Khalid Adam Nasr, Hans-Gerhard Groß, and Arie van Deursen. Adopting and evaluating service oriented architecture in industry. In Rafael Capilla, Rudolf Ferenc, and Juan C. Dueñas, editors, *CSMR*, pages 11–20. IEEE, 2010.

[NvdHP+09a] Dinh Khoa Nguyen, Willem-Jan van den Heuvel, Mike P. Papazoglou, Valeria de Castro, and Esperanza Marcos. Gambuse: A gap analysis methodology for engineering soa-based applications. In Alexander Borgida, Vinay K. Chaudhri, Paolo Giorgini, and Eric S. K. Yu, editors, *Conceptual Modeling: Foundations and Applications*, volume 5600 of *Lecture Notes in Computer Science*, pages 293–318. Springer, 2009.

[NvdHP+09b] Dinh Khoa Nguyen, Willem-Jan van den Heuvel, Mike P. Papazoglou, Valeria de Castro, and Esperanza Marcos. Gap analysis methodology for business service engineering. In Birgit Hofreiter and Hannes Werthner, editors, *CEC*, pages 215–220. IEEE Computer Society, 2009.

[OSL05] Liam O'Brien, Dennis B. Smith, and Grace A. Lewis. Supporting migration to services using software architecture reconstruction. In Kostas Kontogiannis, Ying Zou, and Massimiliano Di Penta, editors, *STEP*, pages 81–91. IEEE Computer Society, 2005.

[PB06] Claus Pahl and Ronan Barrett. Layered patterns in modelling and transformation of service-based software architectures. In Volker Gruhn and Flávio Oquendo, editors, *EWSA*, volume 4344 of *Lecture Notes in Computer Science*, pages 144–158. Springer, 2006.

[RL10] Maryam Razavian and Patricia Lago. A frame of reference for soa migration. In *ServiceWave*, pages 150–162, 2010.

[Sho10] Daniel Sholler. Hype cycle for application architecture. Technical Report G00200962, Gartner, Inc., 2010.

[Sne06] Harry M. Sneed. Integrating legacy software into a service oriented architecture. In *CSMR*, pages 3–14. IEEE Computer Society, 2006.

[SP08] Renuka Sindhgatta and Karthikeyan Ponnalagu. Locating components realizing services in existing systems. In *IEEE SCC (1)*, pages 127–134, 2008.

[UZ09] Amjad Umar and Adalberto Zordan. Reengineering for service oriented architectures: A strategic decision model for integration versus migration. *Journal of Systems and Software*, 82(3):448–462, 2009.

[WZWZ10] Tao Wang, Xiaowei Zhou, Jun Wei, and Wenbo Zhang. Towards runtime plug-and-play software. In Ji Wang, W. K. Chan, and Fei-Ching Kuo, editors, *QSIC*, pages 365–368. IEEE Computer Society, 2010.

[WZZ08] Yan Liu 0011, Qingling Wang, Mingguang Zhuang, and Yunyun Zhu. Reengineering legacy systems with RESTful web service. In *COMPSAC*, pages 785–790. IEEE Computer Society, 2008.

[ZBZ+08] Bo Zhang, Liang Bao, Rumin Zhou, Shengming Hu, and Ping Chen. A black-box strategy to migrate GUI-based legacy systems to web services. In Jonathan Lee, Deron Liang, and Y. C. Cheng, editors, *SOSE*, pages 25–31. IEEE Computer Society, 2008.

[ZK00] Ying Zou and Kostas Kontogiannis. Web-based specification and integration of legacy services. In Stephen A. MacKay and J. Howard Johnson, editors, *CASCON*, page 17. IBM, 2000.

[ZYC06] Zhuopeng Zhang, Hongji Yang, and William C. Chu. Extracting reusable object-oriented legacy code segments with combined formal concept analysis and slicing techniques for service integration. In *QSIC*, pages 385–392. IEEE Computer Society, 2006.

[ZYZZ10] Zhuo Zhang, Hongji Yang, Dongdai Zhou, and Shaochun Zhong. A SOA based approach to user-oriented system migration. In *CIT*, pages 1486–1491. IEEE Computer Society, 2010.

[ZZYZ10] Zhuo Zhang, Dong-Dai Zhou, Hong-Ji Yang, and Shao-Chun Zhong. A service composition approach based on sequence mining for migrating e-learning legacy system to soa. *International Journal of Automation and Computing*, 7(4):584–595, November 2010.

2.8 Service Identification

Involved Workpackages: WP-JRA-1.1

A key factor that determines whether an enterprise really can benefit from adopting SOA is the design of services, including the scope of functionality a service exposes to meet business needs, and the boundaries between services to achieve maximum reusability and flexibility [Erl05].

In the design of services, Service Identification (SI) is a significant task aiming at determining (based on available resources) services that are appropriate for use in an SOA. Many Service Identification Methods (SIMs) have been proposed from both academia and industry.

The very first attempt to tackle the problem of identifying services is originated in 2004, in [JZC04]. This attempt is an algorithm-based approach to devise a list of services based on application domain knowledge. Given the relative new theme and approach, validation was put in place to ensure its credibility to the service community. Since this attempt a large number of approaches were proposed ranging from automated/pattern-based (e.g. [JSM08, CBFB07]) to manual/guidelines-based (e.g. [IB07, KKB07]).

However, none of the early attempts offered a comprehensive methodology delivering all service types (business services, data services, composite services, IT services, web services and partner services [CBFB07]). Most attempts were targeted to the identification of web-services and business services.

Until 2008, academic research worked exclusively on diversifying the types of approaches used, the types of inputs/outputs usable/obtainable. Finally, very few early attempts had been validated in practice. From the survey [GL10] (VUA), it resulted that only 2 attempts out of 12 had been validated in practice, namely [JZC04, ZY04].

After 2008 massive improvements were gradually introduced in the service identification methods practice. S-Cube directly and indirectly promoted industrial validation of research attempts on real-life case studies or action-research attempts. The service community recognized in this field too much industrial distance and lack of validation. As a consequence, more and more attempts worked on validating their work in industrial scenarios either with fine-grained case studies or evaluation in practice (e.g. action research). From the already-cited survey [GL10] (VUA), it resulted that over 50% of the different approaches found (18 in total) involved this serious validation effort.

Moreover, more complete and comprehensive approaches were introduced in the practice. Almost every approach identified after 2008, provided at least two service types as output (e.g. [Far08] produced Business, Data, IT and Composite services). Out of 27 approaches identified from 2008 to present, only 6 offered one type of output (e.g. [ACP08, BCAP09]).

Also, newer approaches tackled the difficulty of manual as well as automated tasks by using ontology-based approaches which envisioned applications in the Web 2.0 paradigm (e.g. [KSB08, YOCS09]). It is noticeable how the ontology based approaches are mostly the ones giving out only one or two types of outputs, probably because of their new and exploratory approaches.

With particular reference to this last finding, it may be that S-Cube positively influenced (either directly or indirectly) the usage of ontology-based approaches with its attempt of formalizing service knowledge (by usage of the knowledge model). Indeed, S-cube fostered the exploration of service identification methods and the application of these to practice.

This consolidated report can be referenced to Sections 2.1 and Section 2.7. In this respect, Service Identification practices can be used to develop (or help migrate to) Service-Based solutions.

Bibliography

[ACP08] Lerina Aversano, Luigi Cerulo, and C. Palumbo. Mining candidate web services from legacy code. In Chao Liu 0002 and Filippo Ricca, editors, *WSE*, pages 37–40. IEEE Computer Society, 2008.

[BCAP09] Devis Bianchini, Cinzia Cappiello, Valeria De Antonellis, and Barbara Pernici. P2s: A methodology to enable inter-organizational process design through web services. In Pascal van Eck, Jaap Gordijn, and Roel Wieringa, editors, *CAiSE*, volume 5565 of *Lecture Notes in Computer Science*, pages 334–348. Springer, 2009.

[CBFB07] Sodki Chaari, Frédérique Biennier, Joël Favrel, and C. Benamar. Towards a service-oriented enterprise based on business components identification. In Ricardo Jardim-Gonçalves, Jörg P. Müller, Kai Mertins, and Martin Zelm, editors, *IESA*, pages 495–506. Springer, 2007.

[Erl05] Thomas Erl. *Service-Oriented Architecture: Concepts, Technology, and Design*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2005.

[Far08] Nafise Fareghzadeh. Service identification approach to SOA development. In *Proceedings of World Academy of Science, Engineering and Technology*, 2008.

[GL10] Qing Gu and Patricia Lago. Service identification methods: A systematic literature review. In *ServiceWave*, pages 37–50, 2010.

[IB07] Srikanth Inaganti and Gopala Krishna Behara. Service identification: BPM and SOA handshake, 2007.

[JSM08] Pooyan Jamshidi, Mohsen Sharifi, and Saeed Mansour. To establish enterprise service model from enterprise business model. In *IEEE SCC (1)*, pages 93–100, 2008.

[JZC04] Hemant K. Jain, Huimin Zhao, and Nageswara Rao Chinta. A spanning tree based approach to identifying web services. *Int. J. Web Service Res.*, 1(1):1–20, 2004.

[KKB07] Karsten Klose, Ralf Knackstedt, and Daniel Beverungen. Identification of Services - A Stakeholder-Based Approach to SOA Development and its Application in the Area of Production Planning. In *Proceedings of the 15th European Conference on Information Systems (ECIS 2007), St. Gallen, Switzerland, June 7-9*, pages 1802–1814, 2007.

[KSB08] DongSu Kang, Chee-yang Song, and Doo-Kwon Baik. A method of service identification for product line. In *Proceedings of the 2008 Third International Conference on Convergence and Hybrid Information Technology - Volume 02*, pages 1040–1045, Washington, DC, USA, 2008. IEEE Computer Society.

[YOCS09] R. Yousef, M. Odeh, D. Coward, and A. Sharieh. BPAOntoSOA: A generic framework to derive software service oriented models from business process architectures. In *ICADIWT*, pages 50–55, 2009.

[ZY04] Zhuopeng Zhang and Hongji Yang. Incubating services in legacy systems for architectural migration. In *APSEC*, pages 196–203. IEEE Computer Society, 2004.

2.9 Monitoring Service-Based Applications

Involved Workpackages: WP-JRA-1.2

Monitoring is the process of collecting and reporting relevant information about the execution of a SBA. Since the state of the art presented in [PO-08] (UCBL, Polimi, FBK, SZTAKI, INRIA, CNR, UniDue, USTUTT), several of the research directions presented therein have continued to be developed and refined, while other completely new ones have been undertaken. In particular researchers have gradually concentrated more on the relationship between the notion of SLA and monitoring, as well as on the monitoring of multi-level SBSs.

In [PO-08] are discussed numerous approaches in which expected service behaviors were expressed through some sort of specification language, and then evaluated by an appropriate runtime environment. This trend has continued to evolve. In [BG11], Baresi and Guinea (Polimi) present a refinement of their WSCoL constraint language, a language for specifying a service's behavior in the context of a BPEL process composition. The main novelties consist of new meta-constraints defining "when" a monitoring specification should be considered and when it can be ignored, and how the approach deals with time-based properties such as constraints on response times.

In [MS07] Mahbub and Spanoudakis (CITY) have also continued to refine their extensive work in the field of monitoring SBSs. In particular they suggest an extension to the WS-Agreement standard as the main tool service designers should use when defining the properties they need the system to check during execution. This extension is based on EC-Assertion, an event-calculus language previously discussed in [PO-08]. Similarly, in [EMT07] Erradi et al. present a language that extends the WS-Policy standard, called WS-Poliscy4MASC, by defining new types of monitoring and control policy assertions. MASC stands for Manageable and Adaptive Service Compositions. The monitoring capabilities allow the execution environment to detect business exceptions and runtime faults. Through synchronous and asynchronous monitoring both at the SOAP message level and at the orchestration level.

In [MRD08] Moser et al. (TUW) have also continued to refine their work in the field. In this paper they present a monitoring extension to the ActiveBPEL execution environment. The extension is implemented using aspect oriented programming techniques. It intercepts the process' execution at the SOAP invocation level and collects runtime events. These events are persisted allowing for the evaluation of simple aggregate QoS metrics such as response time, availability, and accuracy.

In [SGC+09, SC10] Simmonds et al. present an approach that has clear differences with respect to the above approaches. Instead of introducing some sort of monitoring language, the authors suggest to specify the properties that need to be monitored using Unified Modeling Language (UML) Sequence Diagrams. In particular the authors identify a subset of Sequence Diagrams that are expressive enough to specify safety and liveness properties of Web service interactions. However they do not consider timeliness constraints. By transforming these diagrams to automata, they enable conformance checking of finite execution traces against the specification.

In [WWY+11] Wu et al. propose an efficient, non-intrusive online monitoring approach to dynamically analyze data-centric properties for service oriented applications involving multiltiple participants. They introduce a new monitoring specification language called Par-BCL, which stands for a Parametric Behavior Constraint Language. It concentrates on data-centric temporal properties. This allows their approach to explicitly consider the XML payloads traveling within a SOAP message. To avoid heavy runtime performance hits they statically analyze the monitoring properties to generate an optimized parameter state machine.

In [ZPG10] Zahoor et al. present a declarative event-oriented framework, called DISC, that serves as a unified framework to bridge the gap between the process design, verification and monitoring. The approach allows designers to enrich their composition design with data relationships and constraints. The approach is based on event calculus. At design time "abduction reasoning" can be used to find a set of plans, and at execution time "deduction reasoning" can be used to calculate the effect of run-time violations.

As previously stated, since [PO-08] researchers have also begun exploring new directions. In particular new works are now considering the relationship between monitoring and formal SLA, and the monitoring of multi-level SBSs, in a more in depth fashion. SLAs allow providers and clients to mitigate risks by associating penalty payments with poor service quality. However once they are in place their conditions need to be monitored.

In [RSE08] Raimondi et al. propose an approach in which common properties such as timeliness, reliability, and request throughput are translated into timed automata. The acceptance of a timed word by a timed automata can be decided in quadratic time. However, since the timed automata can operate while the message are being exchanged at runtime, the authors' experiments show that their approach introduces an overhead, measured as the percentage of time used for validation over the total duration of the experiment, of merely 0.6%. In this paper the authors also present a Eclipse plugin tool that can derive the monitors from SLAs.

In these years an important EU-funded IP project called SLA@SOI (Polimi, City, FBK) has also provided important contributions to the field of SLA monitoring. The project developed a runtime framework capable of managing SLAs throughout their entire lifecycle. A key aspect of managing an SLA is obviously being able to monitor it. A further complication was that the project focused on multi-layered systems. In particular, the project considered three layers: a business, a software, and an infrastructural. This meant that the monitoring part of the framework had to be flexible enough to cope with coordinated monitoring at all three levels. The way this is achieved is by allowing different kinds of monitors, at different levels, to work together. In [CS10] Comuzzi and Spanoudakis (CITY) present a framework for the dynamic assessment of the monitorability of a given SLA and its terms. The monitorability checks are based on comparisons between the SLA terms for specific services and the descriptions of the monitoring capabilities currently available in the system. Once a decision regarding monitorability has been made the approach can dynamically set up the monitoring framework through appropriate configurations. In [FS11] Foster and Spanoudakis extended the approach with algorithms for decomposing an SLA and representing it through an Abstract Syntax Tree (AST). The approach then uses the AST to search for the sensors and analyzers needed to effectively monitor the specific SLA terms. Finally, the approach generates a monitoring configuration.

In [BGNS10], Baresi et al. (Polimi, CITY) started to model the main conceptual notions that are commonly present in all state of the art monitoring approaches. The goal was to provide a common basis through which designers could enable the use of more than one existing monitoring approach at a time, in order to use the most appropriate monitoring solution for each functional and/or non-functional quality in the system. In particular the approach, named SECMOL, successfully integrated the monitoring capabilities of Dynamo, EC-Assertion, and SLANG. Although the focus was not yet on multi-level SBSs, thanks to its general approach it can easily be adapted to consider multiple service layers.

Bibliography

[BG11] Luciano Baresi and Sam Guinea. Self-supervising BPEL processes. *IEEE Trans. Software Eng.*, 37(2):247–263, 2011.

[BGNS10] Luciano Baresi, Sam Guinea, Olivier Nano, and George Spanoudakis. Comprehensive monitoring of bpel processes. *IEEE Internet Computing*, 14(3):50–57, 2010.

[CS10] Marco Comuzzi and George Spanoudakis. Dynamic set-up of monitoring infrastructures for service based systems. In *SAC*, pages 2414–2421, 2010.

[EMT07] Abdelkarim Erradi, Piyush Maheshwari, and Vladimir Tosic. Ws-policy based monitoring of composite web services. In *ECOWS*, pages 99–108. IEEE Computer Society, 2007.

[FS11] Howard Foster and George Spanoudakis. Advanced service monitoring configurations with SLA decomposition and selection. In William C. Chu, W. Eric Wong, Mathew J. Palakal, and Chih-Cheng Hung, editors, *SAC*, pages 1582–1589. ACM, 2011.

[MRD08] Oliver Moser, Florian Rosenberg, and Schahram Dustdar. Non-intrusive monitoring and service adaptation for ws-bpel. In Jinpeng Huai, Robin Chen, Hsiao-Wuen Hon, Yunhao Liu, Wei-Ying Ma, Andrew Tomkins, and Xiaodong Zhang, editors, *WWW*, pages 815–824. ACM, 2008.

[MS07] Khaled Mahbub and George Spanoudakis. Monitoring *s-agreement* s: An event calculus-based approach. In Luciano Baresi and Elisabetta Di Nitto, editors, *Test and Analysis of Web Services*, pages 265–306. Springer, 2007.

[PO-08] State of the art report, gap analysis of knowledge on principles, techniques and methodologies for monitoring and adaptation of sbas. Deliverable, S-Cube Consortium, 2008.

[RSE08] Franco Raimondi, James Skene, and Wolfgang Emmerich. Efficient online monitoring of web-service SLAs. In Mary Jean Harrold and Gail C. Murphy, editors, *SIGSOFT FSE*, pages 170–180. ACM, 2008.

[SC10] Jocelyn Simmonds and Marsha Chechik. RuMoR: monitoring and recovery for BPEL applications. In Charles Pecheur, Jamie Andrews, and Elisabetta Di Nitto, editors, *ASE*, pages 345–346. ACM, 2010.

[SGC+09] Jocelyn Simmonds, Yuan Gan, Marsha Chechik, Shiva Nejati, Bill O'Farrell, Elena Litani, and Julie Waterhouse. Runtime monitoring of web service conversations. *IEEE T. Services Computing*, 2(3):223–244, 2009.

[WWY+11] Guoquan Wu, Jun Wei, Chunyang Ye, Xiaozhe Shao, Hua Zhong, and Tao Huang. Runtime monitoring of data-centric temporal properties for web services. In *ICWS*, pages 161–170, 2011.

[ZPG10] Ehtesham Zahoor, Olivier Perrin, and Claude Godart. DISC: A declarative framework for self-healing web services composition. In *ICWS*, pages 25–33. IEEE Computer Society, 2010.

2.10 Monitor Adaptation

Involved Workpackages: WP-JRA-1.2

The majority of the existing approaches that tackle the concept of adaptation for SBAs focus on the adaptation of the application itself (see also Section 2.12). Several of these approaches make use of a monitor component to support the identification of problems in the service-based application that may trigger the need for adaptation (see also Section 2.9). These monitor components are responsible to verify the behavior of service-based applications with respect to some pre-defined properties and requirements. However, an important problem is concerned with the support for the adaptation of the monitor component itself. More specifically, the support for changes in the monitor properties, monitor rules, specifications to be monitored, or even changes of the monitor component to be used depending on the situation. In the following are described some approaches that complement the ones presented in WP-JRA-1.2 and are related to the concept of monitor adaptation.

A run-time monitor architecture for web services is presented in [BDS10]. The work aims to provide a holistic monitoring framework by enabling the integration of different verification tools. The architecture is capable of integrating different monitoring approaches and it was designed with the intention of being pluggable to support multiple concurrent monitors for different monitoring aspects. The work described in [BDS10] concentrates on the behaviour verification of services with respect to their advertised specification during run-time. The approach is based on Stream X-Machines (SXM) to represent the behavior of web services. SXMs are special instances of x-machines capable of representing both data and control of a system. An integrated tool in the monitoring architecture is used to process the requests/responses to the SXM. The outputs of both SXM models and web services are compared. If there is a match with the outputs, the service behaved as expected; otherwise a deviation occurred. In this work, monitor adaptation is concerned with SXM models since a new web service in the service composition implies the creation of a new SXM. However, it is not clear whether the creation of a SXM is a fully automated process in the approach.

In [RBS10] the authors present a component-based framework for monitoring and managing features of composite SOA applications. Their proposal relies on the use of components responsible for each activity, namely monitoring component, SLA analysis component, decision taken component, and execution of actions component. In the framework, the different components are attached to each service being managed in order to provide the required information. For example, the monitor component collects, stores, and filters information. The framework works by monitoring data from each individual service and calculating a set of metrics for them. The list of available metrics is exposed by the monitoring component and used by the SLA analysis component which can read the metrics and check if the specified conditions are being fulfilled or not. In the case in which a condition is not fulfilled, or has some risks of not being fulfilled, the decision component is activated and decides on the actions to be taken (Section 2.24). When a set of actions are identified, they are passed to the executor component to be executed over the managed service. The work supports the addition and removal of different components at runtime. For example, a service to which no monitoring information is required may not need the monitoring component and may only have an execution component to modify some parameter of the service. The monitor component can also be adapted with respect to the use of appropriate sensors depending on input data.

The approach in [TZ10] discusses the use of Autonomic Workflows (AWs) to self-manage processes based on service composition. An AW is an extended workflow that contains semantic information about the process to be executed, its objectives, and all related data and constraints that may be useful for the formulation of the process. The approach supports adaptation of autonomic workflows based on their lifecycle (i.e. inception, binding, and execution). In the work, the adaptation is based on the use of policies and reactions to process anomalies. Policies are pre-defined at a high level language as Event-Condition-Action (ECA) rules and are kept in a knowledge base. The approach manages the workflow during all its lifecycle by collecting and organizing the information from the operating environment. The collected information is used together with semantic descriptions of services for adapting, reacting, and improving the workflow at run-time using ECA rules. The approach also uses a Manager component to support some of the monitoring activities (e.g., checking execution, handling anomalies) of a workflow. When there are changes in the workflow, the Manager checks the new workflow using the ECA rules in the knowledge base.

In [CTNHA10] a high-level model for adaptation is proposed. It focuses on the adaptation of a service composition taking into consideration global constraints, e.g. the execution of a service-based application within a time constraint. The approach relies on the creation of an Organization Model (OM) based on a BPEL specification. The OM involves the use of agents (each agent is bounded to a different service), models of complex collaborations between different agents, and specification of rules triggering adaptation. In the work, agents are used to establish the correct execution of a service in a service composition and to trigger the adaptation of the service composition. This is done by checking the information of the agents and verifying whether rules are being satisfied. When a rule is violated the agent is replaced (i.e., the replacing agent is bounded to a different service). In the approach, agents also contain monitoring mechanisms that allow them to pro-actively decide to stop participating in a service composition, e.g. an agent can monitor its Service-Level Objective (SLO) and predict than an SLA will be violated (see also Section 2.24). When an agent decides not to participate in a service composition, a different agent is used to replace it. The approach assumes the existence of additional agents (bounded to services) for replacement. Monitoring in this approach uses information from the participating agents and the rules for triggering adaptation.

The work in [BBP+11] tackles automated evolution, repair, and tuning of services compositions. More specifically, it focuses on the automation of important aspects of a service-oriented application, including resource and service discovery, binding and composition, deployment, and monitoring. The authors propose a roadmap outlining selected research opportunities in the context of self-organizing SOAs. The work also suggests the concept of self-organizing service that is capable of managing its life-cycle (i.e. discovery, composition, and execution) in an autonomic way. The work uses an infrastructure to support self-organizing SOA to allow a user to specify desired QoS characteristics at the level of the service composition. These QoS characteristics can be automatically broken down into requirements for individual services.

An approach to achieve highly adaptable Web services through context-adaptable web service policies is presented in [YWM+11]. In this work, policies are sets of one or more monitoring rules. The work assumes that ,both policies and rules are adaptable based on context information. It extends the Web Service Policy Language (WSPL) to allow the specification of context at both policy and rule levels. The extended policies are woven into the service composition. The approach uses three operations for extending the policies, namely: (i) a context specification method for specifying policy context, (ii) a policy translator method for translating a policy to a required format, and (iii) a policy integration method for applying context-based policies to web services.

An event-based framework for specifying and reasoning about monitoring properties is presented in [ZPG11]. The approach tackles the problem associated with the definition of suitable monitoring properties at the design phase (i.e. properties known in advance) by allowing them to be defined during design or execution times. The approach builds upon an event-based declarative composition design that serves as a unified framework to bridge the gaps among process design, verification, and monitoring. The framework has four main stages including: (i) composition design, which involves the composition specification; (ii) instantiation and verification, which involves finding a solution (or identifying conflicts) in the composition; and (iii) execution of the process and composition monitoring, which involves verification and recovery. The framework is based on event calculus (a language based on first-order logic) that allows specifying and reasoning about monitoring properties in terms of events and fluents. The approach allows definition of functional and non-functional properties, identification of violations, and calculation of the effects that a violation may have on the overall process execution.

In [LKS+10] a business centric monitoring framework is proposed to bridge the gap between the business and service levels in complex business applications. The approach uses business information invariants (i.e. fields of information in composite applications that remain unchanged) to define one or more monitor sets. Monitor sets are defined as collections of attributes and mappings from each attribute to one or more business item attributes in order to associate the service activity with the business composition execution. The user selects the components to be monitored from a business centric view. This allows the user to select and specify monitor sets for a business composition by binding a monitor set with inputs/outputs of a business component. Monitor models check the execution of business compositions using events generated by the service components.

Bibliography

[BBP+11] Walter Binder, Daniele Bonetta, Cesare Pautasso, Achille Peternier, Diego Milano, Heiko Schuldt, Nenad Stojnic, Boi Faltings, and Immanuel Trummer. Towards self-organizing service-oriented architectures. In *SERVICES*, pages 115–121. IEEE Computer Society, 2011.

[BDS10] Konstantinos Bratanis, Dimitris Dranidis, and Anthony J. H. Simons. An extensible architecture for run-time monitoring of conversational web services. In *Proceedings of the 3rd International Workshop on Monitoring, Adaptation and Beyond*, MONA '10, pages 9–16, New York, NY, USA, 2010. ACM.

[CTNHA10] Mario Henrique Cruz Torres, Victor Noël, Tom Holvoet, and Jean-Paul Arcangeli. MAS organisations to adapt your composite service. In *Proceedings of the 3rd International Workshop on Monitoring, Adaptation and Beyond*, MONA '10, pages 33–39, New York, NY, USA, 2010. ACM.

[LKS+10] Geetika T. Lakshmanan, Paul T. Keyser, Aleksander Slominski, Francisco Curbera, and Rania Khalaf. A business centric end-to-end monitoring approach for service composites. In *IEEE SCC*, pages 409–416. IEEE Computer Society, 2010.

[RBS10] Cristian Ruz, Françoise Baude, and Bastien Sauvan. Component-based generic approach for reconfigurable management of component-based soa applications. In *Proceedings of the 3rd International Workshop on Monitoring, Adaptation and Beyond*, MONA '10, pages 25–32, New York, NY, USA, 2010. ACM.

[TZ10] Giancarlo Tretola and Eugenio Zimeo. Autonomic internet-scale workflows. In *Proceedings of the 3rd International Workshop on Monitoring, Adaptation and Beyond*, MONA '10, pages 48–56, New York, NY, USA, 2010. ACM.

[YWM+11] Hamdi Yahyaoui, L. Wang, Azzam Mourad, Mohammed Almulla, and Quan Z. Sheng. Towards context-adaptable web service policies. *Procedia CS*, 5:610–617, 2011.

[ZPG11] Ehtesham Zahoor, Olivier Perrin, and Claude Godart. An event-based reasoning approach to web services monitoring. In *ICWS*, pages 628–635. IEEE Computer Society, 2011.

2.11 Adaptable Cloud Infrastructures

Involved Workpackages: WP-JRA-1.2, WP-JRA-2.3

As our previous state of the art report [PO-08] (Tilburg, FBK, TUW, UniDue, SZTAKI, Polimi, US-TUTT) identified, service infrastructures were rigid and could hardly participate in cross-layer adaptation and monitoring strategies. Since that report, previously – mainly based on grid computing – service infrastructures have started to move towards a new paradigm: cloud computing, and in particular IaaS systems. Cloud computing can be defined as the convergence and evolution of several concepts from virtualization, distributed application design, Grid and enterprise IT management to enable a more flexible approach for deploying and scaling applications [RBE+11]. However, the basic services offered by IaaS cloud systems are really focused on virtual machine management functionalities only. Consequently, advanced SBA or infrastructure level techniques emerged that autonomously tackle the IaaS level problems to reduce the number of interventions needed by users or higher level system components (e.g. components of the business process layer [JDF09] or SBA level functionalities [CJZ+10]). The research summarized in the following paragraphs reveal how did these transitional service infrastructures become aware of their surroundings and not only offer adaptation options for the higher layers of the SBAs but also exhibit autonomous behavior in parallel.

Researchers have identified several options to apply autonomous behavior at the infrastructure provider. [KCH09] describes an approach that controls Virtual Machine (VM) allocation by using Kalman filters based on resource utilization (the principal resource used in their experiments was the CPU). The application of the Kalman filter avoids the transient fluctuations autonomous systems frequently face. However, this paper identifies that application aware VM allocation can further increase efficiency. Therefore, they reveal the extension of their allocation feedback controller with VM coupling requirements (e.g. a specific VM should be available in parallel to another).

[KHY08] focuses on the performance management issues of data centers, and proposes a solution that autonomously reduces the energy footprint of infrastructure providers. The paper proposes the concept of Autonomic Power and Performance Manager (APPM) that considers performance characteristics of the controlled data center to determine the optimal state of the system in accordance to power management. Their APPM component is capable to optimize power consumption of hierarchical architectures by utilizing the autonomous behavior of the smallest power aware components in the system (e.g. hosts, clusters, cpu, network, memory).

More focus was targeted on SLA related research that would allow the incorporation of higher layers of the SBAs. This research is also detailed in Section 2.15. First, in [NBY09], researchers discuss the challenges of ensuring Quality attributes (QAttrs) of cloud aware applications. They propose that the El Farol Bar problem [Art99] can be used as the modeling mechanism for cloud related QAttrs. Then they present a technique that applies SLAs and utility theory to autonomously optimize application execution.

S-Cube partners started SLA aware autonomous cloud infrastructure related research in [KKB09] (SZTAKI, TUW), where they have identified that an important characteristic of Cloud-based services is the provision of non-functional guarantees in the form of SLAs, such as guarantees on execution time or price. However, due to system malfunctions, changing workload conditions, hard- and software failures, established SLAs can be violated. In order to avoid costly SLA violations, flexible and adaptive SLA attainment strategies are needed. This paper investigated the application of autonomic computing to SLA-based resource virtualization considering a three-layered Cloud based infrastructure including agreement negotiation, service brokering and deployment using virtualization. For each layer in the infrastructure, the authors exemplify how the principles of autonomic computing can be applied to achieve component self-management. Finally, the paper summarizes the events (e.g. service instance overloading, broker failures) that lead to the execution of a predefined set of autonomic actions (e.g. self-initiated deployment, negotiation bootstrapping). SLA-aware infrastructure behavior is further detailed in Section 2.15.

As described in [EBMD10] researchers implemented a highly scalable framework for mapping Low Level Resource Metrics to High Level SLA Parameters (LoM2HiS framework) facilitating the exchange of large numbers of messages. They designed and implemented a communication model based on the Java Messaging Service (JMS) API, which is a Java Message Oriented Middleware (MOM) API for sending messages between two or more clients. Once possible SLA violation threats are detected, reactive actions can be taken in order to prevent real SLA violations.

Later, in [BEM$^+$10] (TUW, SZTAKI), the S-Cube project partners discussed how self-manageable Cloud infrastructures are required to achieve high level of flexibility on one hand, and compliance to users' requirements specified by means of SLAs on the other. Such infrastructures should automatically respond to changing component, workload, and environmental conditions minimizing user interactions with the system and preventing violations of agreed SLAs. However, identification of sources responsible for the possible SLA violation and the decision about the reactive actions necessary to prevent SLA violation is far from trivial. First, this paper presents a novel approach for mapping low-level resource metrics to SLA parameters necessary for the identification of failure sources. Second, the paper devises a layered Cloud architecture for the bottom-up propagation of failures to the layer, which can react to sensed SLA violation threats. Moreover, the authors present a communication model for the propagation of SLA violation threats to the appropriate layer of the Cloud infrastructure introduced in the previous paper [KKB09] (SZTAKI, TUW).

Finally, in [GKMW11] (Polimi, SZTAKI, FBK, USTUTT), S-Cube members examine the requirements and use cases of service-based applications that become increasingly multi-layered in nature (e.g. software as a service is frequently built on top of infrastructure as a service systems). Most existing SOA monitoring and adaptation techniques address layer-specific issues – these techniques are discussed in Sections 2.9 and 2.12. These techniques, if used in isolation, cannot deal with real-world domains, where changes in one layer often affect other layers, and information from multiple layers is essential in truly understanding problems and in developing comprehensive solutions. Most existing SOA monitoring and adaptation techniques address layer-specific issues. These techniques, if used in isolation, cannot deal with real-world domains, where changes in one layer often affect other layers, and information from multiple layers is essential in truly understanding problems and in developing comprehensive solutions. This paper proposes a framework that integrates layer specific monitoring and adaptation techniques, and enables multi-layered control loops in service-based systems. Among others, the paper investigates how the SLA violation propagation mechanism can be incorporated to the multi-layered control loop. The paper offers to aggregate infrastructure level events (such as resource availability, service instance rebinding or deployment) and incorporate them in novel adaptation strategies. These strategies now incorporate the definition of multi layer SLA constraints that not only influence the infrastructure level behavior but also consider the higher layers of the SBAs. Further cross-layer adaptation and monitoring challenges are listed in Section 2.13.

Bibliography

[Art99] W.B. Arthur. Complexity and the economy. *Science*, 284(5411):107, 1999.

[BEM+10] Ivona Brandic, Vincent C. Emeakaroha, Michael Maurer, Schahram Dustdar, Sandor Acs, Attila Kertesz, and Gabor Kecskemeti. LAYSI: A layered approach for SLA-violation propagation in self-manageable cloud infrastructures. In *Proceedings of the 2010 IEEE 34th Annual Computer Software and Applications Conference Workshops*, COMPSACW '10, pages 365–370, Washington, DC, USA, 2010. IEEE Computer Society.

[CJZ+10] Gang Chen, Hai Jin, Deqing Zou, Bing Bing Zhou, Weizhong Qiang, and Gang Hu. Shelp: Automatic self-healing for multiple application instances in a virtual machine environment. In *CLUSTER*, pages 97–106. IEEE, 2010.

[EBMD10] Vincent C. Emeakaroha, Ivona Brandic, Michael Maurer, and Schahram Dustdar. Low level metrics to high level slas - lom2his framework: Bridging the gap between monitored metrics and SLA parameters in cloud environments. In Waleed W. Smari and John P. McIntire, editors, *HPCS*, pages 48–54. IEEE, 2010.

[GKMW11] Sam Guinea, Gabor Kecskemeti, Annapaola Marconi, and Branimir Wetzstein. Multi-layered monitoring and adaptation. In *Proceedings of the Ninth International Conference on Service Oriented Computing (ICSOC 2011); Paphos, Cyprus, December 5-8, 2011*. Springer, Dezember 2011.

[JDF09] Ernst Juhnke, Tim Dörnemann, and Bernd Freisleben. Fault-tolerant bpel workflow execution via cloud-aware recovery policies. In *EUROMICRO-SEAA*, pages 31–38. IEEE Computer Society, 2009.

[KCH09] Evangelia Kalyvianaki, Themistoklis Charalambous, and Steven Hand. Self-adaptive and self-configured cpu resource provisioning for virtualized servers using kalman filters. In Simon A. Dobson, John Strassner, Manish Parashar, and Onn Shehory, editors, *ICAC*, pages 117–126. ACM, 2009.

[KHY08] Bithika Khargharia, Salim Hariri, and Mazin S. Yousif. Autonomic power and performance management for computing systems. *Cluster Computing*, 11(2):167–181, 2008.

[KKB09] A. Kertesz, G. Kecskemeti, and I. Brandic. Autonomic resource virtualization in cloud-like environments. Technical Report TUV-1841-2009-04, Vienna University of Technology, 2009.

[NBY09] Vivek Nallur, Rami Bahsoon, and Xin Yao. Self-optimizing architecture for ensuring quality attributes in the cloud. In *WICSA/ECSA*, pages 281–284. IEEE, 2009.

[PO-08] Comprehensive overview of the state of the art on service-based systems. Deliverable, S-CUBE Consortium, 2008.

[RBE+11] Benny Rochwerger, David Breitgand, Amir Epstein, David Hadas, Irit Loy, Kenneth Nagin, J. Tordsson, C. Ragusa, Massimo Villari, Stuart Clayman, Eliezer Levy, A. Maraschini, Philippe Massonet, H. Muñoz, and G. Tofetti. Reservoir - when one cloud is not enough. *IEEE Computer*, 44(3):44–51, 2011.

2.12 Adaptation of Service Compositions

Involved Workpackages: WP-JRA-1.2, WP-JRA-2.2

SBAs rely on the invocation of services and are deployed in dynamic and distributed settings by composing different services, possibly owned by different service providers. System developers exploit in the functionality offered by services without having them under their direct control. This introduces critical dependencies between SBAs and their component services that could change or be unavailable without to notice. Since the developers of SBA usually aims at guaranteeing continuity of service to its users, SBAs have to be equipped with adaptation capabilities to react dynamically and automatically to unavailability of the component services. The unavailability of component services is not the unique reason to adapt an SBAs, there are several situations that may trigger the need to adapt them at runtime. Example of these situations are: (i) changes in the context of the service composition environment, (ii) changes in functional and quality aspects of services participating in SBAs, (iii) failures or unavailability of services participating in SBAs, (iv) new available services, or even (v) new application requirements. The goal of this section is to give an overview of approaches and frameworks that try to solve the previous situations. These approaches are divided in two categories: Built-in vs Dynamic Adaptation and Context-aware adaptation approaches.

Built-in vs Dynamic Adaptation Adaptable systems change their behavior, reconfigure their structure and evolve over time reacting to changes in the operating conditions, so to always meet users' expectations. As suggested in Ardagna and Pernici [AP07], adaptation mechanisms can either be embedded in the description of the adaptable SBAs or implicit in its structure. Various frameworks can be found in the literature with the objective to support adaptation of SBAs, each of them addressing a specific issue. Most of them concern built-in adaptation, i.e. the adaptation logic is completely specified at design time. They concentrate on how to specify adaptation mechanisms and adaptable applications, exploiting different tools. For instance, the specification may be performed by extending standard notations (i.e., BPEL) with adaptation specific tools [KHC+05] using event-condition-action like rules [BGP07, CNM06], variability modeling [HBR07] or aspect-oriented approaches [KSPBC06].

In the literature there are, however, proposals of frameworks for dynamic adaptation, all featuring an adaptation manager separated from the application. Notably, all these approaches are in the service-oriented field and in most cases, they are reduced to monitoring services utilized by a composition (availability or QoS properties) and replacing them if violation is detected (e.g., [ZZL09, SZK05]).

In [SZK05] the authors consider the problem of adapting the application by replacing malfunctioning services at runtime. The adaptation rule is fixed at design time, but it is dynamically applied by a manager component that monitors functional and non-functional properties, creates queries for discovering malfunctioning services and replaces them with dynamically discovered replacements. Narendra et al. [NPKR07] propose an aspect-oriented approach for runtime optimization of non-functional QoS measures. The METEOR-S framework [VGS+05] supports dynamic reconfiguration of processes, based on constraints referring to several QoS dimensions. Reconfiguration is performed essentially at deployment-time.

PAWS [ACM+07] is a framework for flexible and adaptive execution of web service based applications. At design-time, flexibility is achieved through a number of mechanisms, i.e., identifying a set of candidate services for each process task, negotiating QoS, specifying quality constraints, and identifying mapping rules for invoking services with different interfaces. The runtime engine exploits the design-time mechanisms to support adaptation during process execution, in terms of selecting the best set of services to execute the process, reacting to a service failure, or preserving the execution when a context change occurs.

A considerable amount of work has been done within S-Cube investigating dynamic adaptation, namely [MRD08, HLM+10, PS11, BG11, TPPvdH11], [MZ09] (CITY), [ZBC10] (UCBL, UPM). In the following we describe the main contributions that significantly improved the state of the art in this field.

The Vienna Runtime Environment for Service-oriented Computing (VRESCo) [HLM+10] is a framework implemented to address issues like dynamic selection, binding and invocation of services.

Zemni et al. [ZBC10] (UCBL, UPM) propose a soft constraint-based framework to express QoS properties reflecting both customer preferences and penalties applied to unfitting situations. These properties are used to model SLAs and to decide how to rebuild service compositions which may not satisfy all system requirements, in order not to stop its execution.

Pernici et al. [PS11] define a fuzzy service based adaptation based on QoS satisfaction. They use a satisfaction degree calculated using a fuzzy inference system for the adaptation decision making and define and algorithm able to decide which adaptation strategy to take (i.e., service re-negotiation or replacement).

Mahub and Zisman [MZ09] (CITY) realize a service discovery framework able to support proactive identification of services in parallel to the execution of the system. Moreover they define a way to describe replacement policies describing what needs to be changed, and how and when the changes should be executed.

Taher et. al. [TPPvdH11] propose an approach to resolve differences between Web Services protocols through the use of Complex Event Processing (CEP) technology. Using a continuous query engine within a CEP platform, they show how different incompatibilities between Web Services (i.e., signature, protocol incompatibilities) can be solved. Some time, also if the service composition does not change, can happen that its behavior may evolve over time and become incorrect due to changes on their partner services.

The VieDAME framework [MRD08] uses an aspect-oriented approach to allow adaptation of SBAs for certain QoS criteria based on various alternative services. In the framework, a service participating in the system can be marked as replaceable to indicate the alternative services can be invoked instead of the original one, when necessary.

Finally, Baresi and Guinea [BG11], propose the idea to self-supervising BPEL processes. This is a special-purpose compositions that asses their behavior and react through user-defined rules. This supervision is based on monitoring and adaptation. The former checks the system's execution to see whether everything proceeding as planned, while the latter attempts to fix any anomalies. This approach is realized using two distinct languages for defining monitoring and adaptation and at the same time they enrich BPEL processes with self-supervision capabilities.

Context-awareness The adaptation of a service composition may depend also on the context in which component services are executed. The literature is rich of contributions focusing on the development of context-aware applications [BDR07]. In the following we present some of them used in the field of service-based applications.

Chaari et al. [CLC07] propose an architecture for a service based application trying to separate the aspects of context management and adaptation from the service application core. Authors propose to apply adaptation to three different levels in the service application: service (application behavior), data (content adaptation) and user interface (visualization). Service behavior can be adapted substituting the current version of a service with a one that could be more suitable for the measured context. The flexibility of such system is limited since all the versions of the services have to be pre-built depending on the different contexts. In [MCR06], context-based adaptation for mobile learning systems has been studied: suitable activities are suggested on the basis of the context in which the user is operating. Each user has preferences in terms of needs, interests, personal features and learning styles. The proposed adaptation mechanism is implemented in three steps; each of them is in charge of the analysis of the information and of the selection of the most suitable activity for a user, depending on his/her context. In [NLZ+11], an idea of a multi-granularity context model is presented, in particular, for location and time values, with an ontological approach to reason on the context and to compose services in the most appropriate way.

Within the S-Cube project, Bucchiarone et al. [BPRK] propose a context-aware adaptation approach for SBAs able to generate adaptations completely at runtime taking into account the current context and the available services, without hard-coding them in the adaptation logic. They realize an adaptation engine able to observe the SBA execution and trigger adaptation on demand. In other words, the execution engine, that using a planning algorithm, is able to generate an adaptation solution anytime that extraordinary situations from the execution context are detected.

Bibliography

[ACM+07] Danilo Ardagna, Marco Comuzzi, Enrico Mussi, Barbara Pernici, and Pierluigi Plebani. Paws: A framework for executing adaptive web-service processes. *IEEE Software*, 24(6):39–46, 2007.

[AP07] Danilo Ardagna and Barbara Pernici. Adaptive service composition in flexible processes. *IEEE Trans. Software Eng.*, 33(6):369–384, 2007.

[BDR07] Matthias Baldauf, Schahram Dustdar, and Florian Rosenberg. A survey on context-aware systems. *IJAHUC*, 2(4):263–277, 2007.

[BG11] Luciano Baresi and Sam Guinea. Self-supervising BPEL processes. *IEEE Trans. Software Eng.*, 37(2):247–263, 2011.

[BGP07] Luciano Baresi, Sam Guinea, and Liliana Pasquale. Self-healing bpel processes with dynamo and the jboss rule engine. In *International workshop on Engineering of software services for pervasive environments: in conjunction with the 6th ESEC/FSE joint meeting*, ESSPE '07, pages 11–20, New York, NY, USA, 2007. ACM.

[BPRK] A. Bucchiarone, M. Pistore, H. Raik, and R. Kazhamiakin. Adaptation of service-based business processes by context-aware replanning. Submitted to SOCA 2011.

[CLC07] Tarak Chaari, Frédérique Laforest, and Augusto Celentano. Adaptation in context-aware pervasive information systems: the secas project. *Int. J. Pervasive Computing and Communications*, 3(4):400–425, 2007.

[CNM06] Massimiliano Colombo, Elisabetta Di Nitto, and Marco Mauri. Scene: A service composition execution environment supporting dynamic changes disciplined through rules. In Dan and Lamersdorf [DL06], pages 191–202.

[DL06] Asit Dan and Winfried Lamersdorf, editors. *Service-Oriented Computing - ICSOC 2006, 4th International Conference, Chicago, IL, USA, December 4-7, 2006, Proceedings*, volume 4294 of *Lecture Notes in Computer Science*. Springer, 2006.

[HBR07] Alena Hallerbach, Thomas Bauer, and Manfred Reichert. Managing process variants in the process life cycle. Technical Report 414, University of Twente, 2007.

[HLM+10] Waldemar Hummer, Philipp Leitner, Anton Michlmayr, Florian Rosenberg, and Schahram Dustdar. *VRESCo - Vienna Runtime Environment for Service-oriented Computing*, pages 299–324. Service Engineering. European Research Results. Springer, 2010.

[KHC+05] Dimka Karastoyanova, Alejandro Houspanossian, Mariano Cilia, Frank Leymann, and Alejandro P. Buchmann. Extending bpel for run time adaptability. In *EDOC*, pages 15–26. IEEE Computer Society, 2005.

[KSPBC06] Woralak Kongdenfha, Régis Saint-Paul, Boualem Benatallah, and Fabio Casati. An aspect-oriented framework for service adaptation. In Dan and Lamersdorf [DL06], pages 15–26.

[MCR06] Estefanía Martín, Rosa M. Carro, and Pilar Rodríguez. A mechanism to support context-based adaptation in m-learning. In Wolfgang Nejdl and Klaus Tochtermann, editors, *EC-TEL*, volume 4227 of *Lecture Notes in Computer Science*, pages 302–315. Springer, 2006.

[MRD08] Oliver Moser, Florian Rosenberg, and Schahram Dustdar. Viedame - flexible and robust bpel processes through monitoring and adaptation. In *ICSE Companion*, pages 917–918, 2008.

[MZ09] Khaled Mahbub and Andrea Zisman. Replacement policies for service-based systems. In *ICSOC/ServiceWave Workshops*, pages 345–357, 2009.

[NLZ+11] Wenjia Niu, Gang Li, Zhijun Zhao, Hui Tang, and Zhongzhi Shi. Multi-granularity context model for dynamic web service composition. *J. Network and Computer Applications*, 34(1):312–326, 2011.

[NPKR07] N. C. Narendra, Karthikeyan Ponnalagu, Jayatheerthan Krishnamurthy, and R. Ramkumar. Run-time adaptation of non-functional properties of composite web services using aspect-oriented programming. In *Proceedings of the 5th international conference on Service-Oriented Computing*, ICSOC '07, pages 546–557, Berlin, Heidelberg, 2007. Springer-Verlag.

[PS11] Barbara Pernici and Seyed Hossein Siadat. A fuzzy service adaptation based on QoS satisfaction. In Haralambos Mouratidis and Colette Rolland, editors, *CAiSE*, volume 6741 of *Lecture Notes in Computer Science*, pages 48–61. Springer, 2011.

[SZK05] George Spanoudakis, Andrea Zisman, and Alexander Kozlenkov. A service discovery framework for service centric systems. In *IEEE SCC*, pages 251–259. IEEE Computer Society, 2005.

[TPPvdH11] Yehia Taher, Michael Parkin, Mike P. Papazoglou, and Willem-Jan van den Heuvel. Adaptation of web service interactions using complex event processing patterns. In *ICSOC 2011*, 2011.

[VGS+05] K. Verma, K. Gomadam, A.P. Sheth, J.A. Miller, and Z Wu. The Meteor-S approach for configuring and executing dynamic web processes. Technical report, University of Georgia, 2005.

[ZBC10] Mohamed Anis Zemni, Salima Benbernou, and Manuel Carro. A soft constraint-based approach to QoS-aware service selection. In *ICSOC*, pages 596–602, 2010.

[ZZL09] Yanlong Zhai, Jing Zhang, and Kwei-Jay Lin. Soa middleware support for service process reconfiguration with end-to-end qos constraints. In *Web Services, IEEE International Conference on*, pages 815–822, Los Alamitos, CA, USA, 2009. IEEE Computer Society.

2.13 Cross-layer Monitoring and Adaptation

Involved Workpackages: WP-JRA-1.2, WP-JRA-2.1, WP-JRA-2.2, WP-JRA-2.3

The operation of service-based applications is a complex task. These distributed systems must provide their functionality with the required/agreed qualities of services, cope with the unreliable network on which they operate, and also deal with the changes in the context in which they are executed, or in the partner services with which they interact. This means that the system must probe the execution to discover all these problems as soon as they materialize, and these applications must be able to adapt their behavior to cope with them. The service abstraction has become so pervasive that we are now building systems that are multi-layered in nature. Cloud-computing allows us to build software as a service on top of a dynamic infrastructure that is also provided as a service IaaS. This complicates the development of self-adaptive systems because the layers are intrinsically dependent one of the other.

Most existing SOA monitoring and adaptation techniques address one specific functional layer at a time. This makes them inadequate in real-world domains, where changes in one layer often affect others. If we do not consider the system as a whole we can run into different kinds of misjudgments. For example, if we witness an unexpected behavior at the software layer we may be inclined to adapt at that same layer, even though a more cost-effective solution might be found either at the infrastructure layer, or by combining adaptations at both layers. Even worse, a purely software adaptation might turn out to be useless due to infrastructural constraints we fail to consider. Similar considerations are made in case of the unexpected behavior at the infrastructure layer, or at both. Please refer to Section 2.9, Section 2.23 and Section 2.18 for a detailed survey on single-layer monitoring and adaptation techniques, respectively. In this section we focus on monitoring and adaptation approaches taking into account the cross-layered nature of SBAs.

Researchers that do consider cross-layered applications tend to concentrate either on monitoring them or on adapting them. Concerning cross-layer monitoring, Foster et al. [FS11] have proposed an extensible framework for monitoring business, software, and infrastructure services. The framework allows different kinds of reasoners, tailored to different kinds of services, to be integrated and to collaborate to monitor decomposable service level agreement terms and expressions. The framework automatically assigns the decomposed atomic terms to specific reasoners, yet the approach does not support the correlation of terms monitored at different layers. Mos et al. [Mos2009] propose a cross-layered monitoring approach that considers service and infrastructure level events produced by services deployed to a distributed Enterprise Service Bus (ESB). Basic computations can be performed on the events to produce aggregate information (e.g., averages) or complex event processing can be used for more complex correlations and verifications. The resulting data are analyzed by comparing them to thresholds, and the knowledge collected at the various levels are presented through appropriately differentiated user interfaces and visualization techniques. The approach does not correlate knowledge collected at the different levels

Regarding cross-level adaptation, Efstratiou et al. [ECDF01] present an approach for adapting multiple applications that share common resources. These applications are not composed, but rather single entities affected by the same contextual attributes. Since these applications live in the same space they need to coordinate how they manage the shared resources to avoid conflicts. However, they expect the users to perceive and model the conflicts manually. In Gjørven et al. [GRE08] propose a framework to support cross-layer self-adaption in SBAs. They present a technologically independent middleware named QUA to support coordinated cross-layer adaptations by integrating interface and application layer adaptation mechanisms. Popescu et al. [PSL+10] propose a framework for cross-layer adaptation of service-based systems comprised of organization, coordination and service layers. In this approach a designer needs to prepare a taxonomy of the adaptation mismatches, and then a set of adaptation templates, known as patterns, that define generic solutions for these mismatches. Kazhamiakin et al. [KWK+09] (FBK, USTUTT) analyze the dependencies of Key Performance Indicators (KPIs) on process quality factors from different functional levels of an SBA such as QoS parameters, and then an adaptation strategy is decided to improve all the negatively affected quality metrics in the SBA. This work proposes a set of adaptation solutions for KPI violations through the consideration of the non-functional layer dependencies. Zengin et al. [ZKP11] (FBK) propose an approach that introduces a Cross-Layer Adaptation Manager (CLAM). The approach relies on a cross-layer meta model of the SBSs and a set of predefined domain specific rules to integrate and coordinate existing analysis and adaptation tools. It assesses the impact of an initial adaptation trigger at different system levels, and if needed, proposes additional adaptations, consistent with the overall system. The proposed framework is implemented as a prototype tool and evaluated on a service-based application for taxi reservation. Similarly to [ZKP11], the work in [THF08, BDF+09] adopt a cross-layer representation of the application model. However, these works target a limited number of adaptation cases such as service replacement.

An approach that integrates cross-layered monitoring and adaptation of service-based systems is [GKMW11] (Polimi, SZTAKI, FBK, USTUTT). In this work Guinea et al. propose a framework that integrates software and infrastructure specific monitoring and adaptation techniques, enabling cross-layered control loops in SBSs. All the steps in the control loop acknowledge the multi-faceted nature of the system, ensuring an holistic reasoning, and adapt the system in a coordinated fashion. In the developed prototype they focus on the monitoring and adaptation of BPEL processes that are deployed onto a dynamic infrastructure. Building upon a set of existing solutions they integrate process and infrastructure level monitoring [BG11] (Polimi) and [KKB11] (SZTAKI, TUW) with a correlation technique that makes use of complex event processing [BCGG10] (Polimi). The correlated data, combined with machine-learning techniques, allow to pinpoint where the problems lie in the cross-layered system, and where it would be more convenient to adapt [KWK+09] (FBK, USTUTT) and [WLR+11] (USTUTT, TUW). They then build a complex adaptation strategy that may involve the software and/or the infrastructure layer [ZKP11], and enact it through appropriate effectors. The proposed approach is evaluated on a medical imaging procedure for Computed Tomography (CT) Scans, an e-Health scenario characterized by strong dependencies between the software layer and infrastructural resources. Similarly, the work done by Schmieders et al. [SMO+11] (UniDue, CITY, FBK) proposes a solution to avoid SLA violations by applying cross-layer-adaptation techniques. The novelty of the approach is the exploitation of all SBA layers (Business Process Management (BAM), Service Computing (SCC) and SI) for the prevention of SLA violations. The identification of adaptation needs is based on SLA prediction, which uses assumptions on the characteristics of the running execution context. Multiple adaptation mechanisms are available to react on the adaptation need, acting on different layers of the SBA. The adaptation strategy chooses the right adaptation mechanism, coordinated by a multi-agent community.

Bibliography

[BCGG10] Luciano Baresi, Mauro Caporuscio, Carlo Ghezzi, and Sam Guinea. Model-driven management of services. In Antonio Brogi, Cesare Pautasso, and George Angelos Papadopoulos, editors, *ECOWS*, pages 147–154. IEEE Computer Society, 2010.

[BDF+09] B. Burgstaller, D. Dhungana, X. Franch, P. Gruenbacher, L. López, J. Marco, and M. Oriol. Monitoring and adaptation of service-oriented systems with goal and variability models. Technical Report LSI-09-8-R, Universitat Politécnica de Catalunya, 2009.

[BG11] Luciano Baresi and Sam Guinea. Self-supervising BPEL processes. *IEEE Trans. Software Eng.*, 37(2):247–263, 2011.

[ECDF01] Christos Efstratiou, Keith Cheverst, Nigel Davies, and Adrian Friday. An architecture for the effective support of adaptive context-aware applications. In Kian-Lee Tan, Michael J. Franklin, and John C. S. Lui, editors, *Mobile Data Management*, volume 1987 of *Lecture Notes in Computer Science*, pages 15–26. Springer, 2001.

[FS11] Howard Foster and George Spanoudakis. Smart: a workbench for reporting the monitorability of services from slas. In *Proceeding of the 3rd international workshop on Principles of engineering service-oriented systems*, PESOS '11, pages 36–42, New York, NY, USA, 2011. ACM.

[GKMW11] Sam Guinea, Gabor Kecskemeti, Annapaola Marconi, and Branimir Wetzstein. Multi-layered monitoring and adaptation. In *Proceedings of the Ninth International Conference on Service Oriented Computing (ICSOC 2011); Paphos, Cyprus, December 5-8, 2011*. Springer, Dezember 2011.

[GRE08] Eli Gjørven, Romain Rouvoy, and Frank Eliassen. Cross-layer self-adaptation of service-oriented architectures. In Karl M. Göschka, Schahram Dustdar, Frank Leymann, and Vladimir Tosic, editors, *MW4SOC*, pages 37–42. ACM, 2008.

[KKB11] A. Kertesz, G. Kecskemeti, and I. Brandic. Autonomic SLA-aware service virtualization for distributed systems. In *Parallel, Distributed and Network-Based Processing (PDP), 2011 19th Euromicro International Conference on*, pages 503 –510, feb. 2011.

[KWK+09] Raman Kazhamiakin, Branimir Wetzstein, Dimka Karastoyanova, Marco Pistore, and Frank Leymann. Adaptation of service-based applications based on process quality factor analysis. In *ICSOC/ServiceWave Workshops*, pages 395–404, 2009.

[PSL+10] Razvan Popescu, Athanasios Staikopoulos, Peng Liu, Antonio Brogi, and Siobhán Clarke. Taxonomy-driven adaptation of multi-layer applications using templates. In *SASO*, pages 213–222. IEEE Computer Society, 2010.

[SMO+11] E. Schmieders, A. Micsik, M. Oriol, K. Mahbub, and R. Kazhamiakin. Combining SLA prediction and cross layer adaptation for preventing SLA violations. In *2nd Workshop on Software Services: Cloud Computing and Applications based on Software Services*, 2011.

[THF08] Uttam Kumar Tripathi, Knut Hinkelmann, and Daniela Feldkamp. Life cycle for change management in business processes using semantic technologies. *JCP*, 3(1):24–31, 2008.

[WLR+11] Branimir Wetzstein, Philipp Leitner, Florian Rosenberg, Schahram Dustdar, and Frank Leymann. Identifying influential factors of business process performance using dependency analysis. *Enterprise IS*, 5(1):79–98, 2011.

[ZKP11] Asli Zengin, Raman Kazhamiakin, and Marco Pistore. CLAM: Cross-layer management of adaptation decisions for service-based applications. In *ICWS*, pages 698–699, 2011.

2.14 Quality Modeling

Involved Workpackages: WP-JRA-1.3

In addition to the functional characteristics, services are characterized by a set of properties that are encompassed under the general term of QoS. According to the definition given in [KP09a] "QoS is a set of non-functional attributes of the entities used in the path from the Web Service (WS) to the client that bear on the WSs ability to satisfy stated or implied needs in an end-to-end fashion". This definition takes into account the usually neglected aspect that QoS must be seen in an end-to-end sense as it affects the end-to-end quality received by a WS client when requesting or invoking a particular WS. In this way, QoS characterizes any entity used in the path from the user to the WS, including the service infrastructure and its components. Thus, the QoS characteristics of all these entities constitute the QoS of a WS.

Starting from the work discussed in the [PO-08] (UCBL, TUW, UPM, Polimi, INRIA, UniDue, FBK, SZTAKI, UOC, CNR, USTUTT), in S-Cube a systematic review has been proposed in [KPP$^+$]. Here current approaches quality models, meta-models, and service level agreements are compared. Considering the result of this work, quality models include list of quality dimensions (a.k.a. attributes, characteristics) that are considered relevant for SBA. In addition to what presented in the previous deliverable on the state of the art [PO-08] (UCBL, TUW, UPM, Polimi, INRIA, UniDue, FBK, SZTAKI, UOC, CNR, USTUTT), three new relevant work have been proposed, namely [NES09, KP09b, MGI09] and [CKM$^+$08] (Polimi, UniDue, Tilburg, TUW). The first one deserves special attention, as it is the only model that adds new quality dimensions to the list of the usual dimensions that are proposed in the literature. More specifically, the authors proposed the following new dimensions: integrability, resource efficiency, reusability, security (isolation). With [CKM$^+$08] (Polimi, UniDue, Tilburg, TUW), S-Cube network proposes a quality model specifically designed to support the monitoring and the adaptation of SBAs.

Focusing on the meta-models, they are conceptualizations of the appropriate quality concepts and their relationships that can be used to capture and describe a quality model. For example, a typical meta-model will contain the concepts of QoS category, QoS attribute, and QoS metric and the relationships contains (from QoS categories to attributes) and measuredBy (from QoS attributes to metrics). Since the state of the art presented in [PO-08] (UCBL, TUW, UPM, Polimi, INRIA, UniDue, FBK, SZTAKI, UOC, CNR, USTUTT), most of the research focused on the use of semantic techniques to define quality meta-models. In particular, four new approaches gained the attention of the service community: PCM, WSQDL, QoS-MO, and WS-QoSOnto.

Policy-Centered Metamodel (PCM) PCM is proposed in [PPCM08] to achieve three main objectives: (i) the explicit distinction between the Non-Functional Property (NFP) offered by providers and requested by users; (ii) the concept of policy that aggregates NFP descriptions into a single entity with an applicability condition; (iii) a set of constraint operators, which is particularly relevant for NFP requests. The meta-model embracing the above perspective is formalized by an ontology that provides a good balance between expressiveness and complexity in the resulting NFP descriptions.

Web Service Quality Description Language (WSQDL) [MGI09] introduces WSQDL that provides a XML-based description method for standardizing the expression of QoS exchanged between services and their consumers. It defines the constructs needed to specify the quality factors, their taxonomy, the way they are assessed and the relationships between them. The QoS model underlying this language addresses QoS on an end-to-end basis by covering quality factors of the main elements acting in dynamic service environments: (i) network and hardware resources, (ii) application services and (iii) end-users. To this aim, four ontologies are proposed: the QoS core ontology, the infrastructure QoS ontology, the service QoS ontology, and the user QoS ontology.

Quality of Service Modeling Ontology (QoS-MO) QoS-MO is proposed by [TS08]. It is an upper level ontology that contains elements for the description of QoS Characteristics and Constraints of Web Services described in OWL-S. To instantiate a QoS description of a particular Web Service, one must create a new ontology that will import the QoS-MO ontology, extend its classes as needed and create the required individuals. The functional and non-functional (QoS) specifications of a Web Service may be described on a single ontology or on two separated ones. The proposed upper level ontology include four main concepts. First of all the *QoS Characteristic*: quantifiable features of a service as latency, response time. Then, the *QoS Category* a way for grouping different QoS characteristics, and *QoS Dimension* how a QoS Characteristic can be measured. Finally, the *QoS Context* that defines the dependencies between several QoS Characteristics.

WS-QoSOnto The ontology introduced in [Tra08] has been developed to support the service discovery. For this reason, it includes both the functional and non functional aspects about a Web service. This work is also interesting as it includes a comparison of the different ontologies proposed for the QoS with respect to criteria such as: flexibility, metric, unit, and dynamism.

Bibliography

[CKM+08] C. Cappiello, K. Kritikos, A. Metzger, M. Parkin, B. Pernici, P. Plebani, and M. Treiber. A quality model for service monitoring and adaptation. In *Proceedings of Workshop on Monitoring, Adaptation and Beyond (MONA+) at ServiceWave Conference*, 2008.

[KP09a] Kyriakos Kritikos and Dimitris Plexousakis. Requirements for qos-based web service description and discovery. *IEEE Trans. Serv. Comput.*, 2:320–337, October 2009.

[KP09b] Kyriakos Kritikos and Dimitris Plexousakis. Requirements for qos-based web service description and discovery. *IEEE T. Services Computing*, 2(4):320–337, 2009.

[KPP+] K. Kyriakos, B. Pernici, P. Plebani, C. Cappiello, M. Comuzzi, S. Benbernou, I. Brandic, A. Kertesz, M. Parkin, and M. Carro. A survey on service quality description. *ACM Computing Surveys. Submitted to*.

[MGI09] Nebil Ben Mabrouk, Nikolaos Georgantas, and Valerie Issarny. A semantic end-to-end qos model for dynamic service oriented environments. In *Proceedings of the 2009 ICSE Workshop on Principles of Engineering Service Oriented Systems*, PESOS '09, pages 34–41, Washington, DC, USA, 2009. IEEE Computer Society.

[NES09] NESSI Open Framework - Reference Architecture. Deliverable D4.1: Non-functional aspects. Technical report, NESSI, 2009.

[PO-08] Survey of quality related aspects relevant for service-based applications. Deliverable, S-Cube Consortium, 2008.

[PPCM08] Flavio De Paoli, Matteo Palmonari, Marco Comerio, and Andrea Maurino. A meta-model for non-functional property descriptions of web services. In *ICWS*, pages 393–400. IEEE Computer Society, 2008.

[Tra08] Vuong Xuan Tran. Ws-qosonto: A qos ontology for web services. In *SOSE*, pages 233–238, 2008.

[TS08] Gustavo Fortes Tondello and Frank Siqueira. The qos-mo ontology for semantic qos modeling. In Roger L. Wainwright and Hisham Haddad, editors, *SAC*, pages 2336–2340. ACM, 2008.

2.15 Proactively Negotiating and Agreeing On Service Level Agreements

Involved Workpackages: WP-JRA-1.2, WP-JRA-1.3, WP-JRA-2.2, WP-JRA-2.3

The previous state of the art report related to SLAs [PO-08] (UCBL, TUW, UPM, Polimi, INRIA, UniDue, FBK, SZTAKI, UOC, CNR, USTUTT) covered QoS and SLA definition, QoS negotiation and quality assurance for service based application (Section 2.14). However, it did not cover the study of proactive SLA negotiation or runtime management of agreed SLA. In this section first we very briefly present what was the state of the art of proactive SLA negotiation and SLA violation management when the last report was produced and then continue to discuss what advancement has been made in proactive SLA negotiation and SLA violation management since the last state of the art report was produced.

Runtime SLA re-negotiation has been suggested in [OW06, MTV07, SY05] to manage SLA violations. In [MTV07] SLOs are revised and renegotiated at runtime and deployed services are adjusted to dynamically agreed service level objectives (Section 2.12). A similar approach allowing the change of service level objectives whilst keeping the existing SLA is described in [SY05]. The provision of compensation in case of violation of SLA is also the focus of [RWQ$^+$07]. This approach claims that the penalty clauses in the SLA should not only specify the monetary penalties or impact on potential future agreements between the parties; rather the penalty clauses should include several other issues such as which jurisdiction will be applied in case a conflict between the provider and the client arise, the impact of the penalty clauses on the choice of service level objectives. All of these approaches are reactive in nature, i.e. renegotiation starts only after an existing SLA is violated. The outcome of renegotiation is either a revised set of service level objectives allowing the acceptance of a service from an existing provider or a new SLA for a new service provider terminating the existing SLA. All these approaches either affect the quality of the delivered service or fail to guarantee uninterrupted service.

An agent based framework for SLA management is presented in [HYK$^+$09]. In this framework, an initiator agent from the service consumer's side and a responder agent from the service provider's side take part in the negotiation process. The responder agent advertises the service level capabilities and the initiator agent fetches these advertisements and initializes the SLA negotiation process. Different stages of SLA life cycle e.g. formation, enforcement and recovery is performed through the autonomous interactions among these agents. In the case of an SLA violation, the initiator agent may either claim compensation and renegotiate with the service provider or select a new service provider. In [Par08] a renegotiation protocol is described that allows the service consumer or service provider to initiate renegotiation while the existing SLA is still in force when this becomes necessary for service providers or consumers for different reasons (e.g., changes in the business requirements of a party). However these approaches are also reactive and do not guarantee uninterrupted service.

Proactive approaches to prevent SLA violations are increasingly appearing in the literature [LMRD10] (TUW) and [Lin11] (Section 2.24). Proactive management of SLA has been advocated in [Lin11]. In this approach a negotiated SLA is monitored at runtime to predict SLA violation by applying machine learning algorithms. If potential violations are detected, predefined policies are applied to take actions in order to prevent the SLA violation. Very similar approach is described in [LMRD10] (TUW), where SLA violation is predicted by using regression from monitored runtime data and then the prediction is used to trigger adaptations in the service composition to improve the performance of the composition instance such that violations are prevented.

Several approaches are also proposed by S-Cube community for proactive SLA negotiation and proactive adaptation of service based applications to prevent SLA violations. The authors in [MS10a, MS10b] (City) propose a framework for proactive SLA negotiation that integrates this process with dynamic service discovery and, hence, can provide integrated runtime support for both these key activities which are necessary in order to achieve the runtime operation of service based applications with minimized interruptions. More specifically the authors argue that an alternative service provider should be selected and an SLA should be negotiated by the participating parties prior to a foreseen problem and when the existing SLA is violated the faulty service provider should be replaced by the newly selected service provider. The authors of [SMO+11] (UniDue, SZTAKI, UPC, CITY, FBK) describe a framework which prevents SLA violations of SBAs, by applying cross-layer adaptation underpinned by monitoring and adaptation facilities. The framework identifies adaptation needs based on SLA prediction, which is supported by the monitoring of assumptions on the characteristics of the running execution context (Sections 2.13, 2.16, 2.24). The framework supports multiple adaptation mechanisms (e.g. service replacement, SLA renegotiation, system infrastructure adaptation) to react on adaptation needs, acting on different layers of the SBA (e.g. the composition layer, the infrastructure layer).

Bibliography

[HYK+09] Qiang He, Jun Yan, Ryszard Kowalczyk, Hai Jin, and Yun Yang. Lifetime service level agreement management with autonomous agents for services provision. *Inf. Sci.*, 179(15):2591–2605, 2009.

[Lin11] Tong Lin. Proactive service level agreement in cloud computing, 2011.

[LMRD10] Philipp Leitner, Anton Michlmayr, Florian Rosenberg, and Schahram Dustdar. Monitoring, prediction and prevention of SLA violations in composite services. In *ICWS*, pages 369–376. IEEE Computer Society, 2010.

[MS10a] Khaled Mahbub and George Spanoudakis. A framework for proactive SLA negotiation. In José A. Moinhos Cordeiro, Maria Virvou, and Boris Shishkov, editors, *ICSOFT (2)*, pages 251–256. SciTePress, 2010.

[MS10b] Khaled Mahbub and George Spanoudakis. Proactive SLA negotiation for service based systems. In *SERVICES*, pages 519–526. IEEE Computer Society, 2010.

[MTV07] Giuseppe Di Modica, Orazio Tomarchio, and Lorenzo Vita. A framework for the management of dynamic slas in composite service scenarios. In *ICSOC Workshops*, pages 139–150, 2007.

[OW06] O.Waeldrich and W.Ziegler. A ws-agreement based negotiation protocol. Technical report, Fraunhofer Institute SCAI, 2006.

[Par08] M Parkin. An SLA re-negotiation protocol. In *2nd Non Functional Properties and Service Level Agreements in Service Oriented Computing Workshop*, 2008.

[PO-08] Survey of quality related aspects relevant for service-based applications. Deliverable, S-Cube Consortium, 2008.

[RWQ+07] Omer Rana, Martijn Warnier, Thomas B. Quillinan, Frances Brazier, and Dana Cojocarasu. Managing the violations in service level agreements. In *the Proceedings of the Usage of Service Level Agreements in Grids Workshop*, September 2007.

[SMO+11] E. Schmieders, A. Micsik, M. Oriol, K. Mahbub, and R. Kazhamiakin. Combining SLA prediction and cross layer adaptation for preventing SLA violations, June 2011.

[SY05] Rizos Sakellariou and Viktor Yarmolenko. On the flexibility of WS-agreement for job submission. In *Proceedings of the 3rd international workshop on Middleware for grid computing*, MGC ’05, pages 1–6, New York, NY, USA, 2005. ACM.

2.16 Quality Prediction Techniques to Support Proactive Adaptation

Involved Workpackages: WP-JRA-1.2, WP-JRA-1.3

Quality prediction techniques to support proactive adaptation were identified as one of the research challenges in the package WP-JRA-1.3, and were treated in the deliverables CD-JRA-1.3.4 [CD-10] (UniDue, Tilburg, FBK, INRIA, LERO, Polimi, TUW, UPM, USTUTT) and CD-JRA-1.3.5 [CD-11] (UniDue, TUW, CITY, Polimi, INRIA, LERO, USTUTT, UPM, SZTAKI, UCBL, FBK, UOC, UPC, SEERC). Many S-Cube contributions in this subsection were also closely related to runtime quality assurance, as well as to QoS-aware service compositions (WP-JRA-2.2). In this overview, we list only those that are specifically related to the context of proactive adaptation. An introduction and overview of different approaches can be found in the book chapter on integrating perfective and corrective adaptation of service-based applications by Gehlert et al. [GMK+11], as well as in the book chapter on analytical quality prediction by Metzger et al. [MBC+10].

Machine Learning Based Prediction Based on their previous work on post-mortem analysis of SLA violations [WLR+09], Wetzstein et al. (USTUTT and TUW) studied applications of machine learning techniques on prediction of SLA violations, based on data mining [LWR+09] (TUW, USTUTT) of instance data. The approach monitors an orchestration instance, recording different metrics (such as sizes of internal data items, execution times for component services, and the messages exchanged between them), and performing predictions at a set of predefined check-points in service execution. The prediction relies on a machine learning model that uses the recorded data as well as data and parameters that are not known at the checkpoint, but can be estimated from the history of past instance executions. The prediction model, which is based on regression, can be retrained to take into account new data and thus to accumulate knowledge from one execution to the next. The approach was illustrated by instrumenting the Apache ODE [Apa11] BPEL engine and using the Weka data mining suite [HFH+09] for prediction model development.

In a follow-up work, Leitner et al. [LMRD10] (TUW) discussed the application of different adaptation mechanisms on top of the machine learning-based prediction of SLA violations. The basic adaptation actions presented in this work fall into two categories. The first includes data manipulation, which does not change the control structure of the orchestration, but rather changes the data used in the instance, to affect future behavior. Another type of adaptation action, more involved, is service rebinding, where a combination of 1:1 rebinding between the invocations encoded in the orchestration and the target component services can be chosen and activated at run-time.

Other, more complex adaptation actions that affect the logic of a running instance were studied in a separate work by Leitner et al. [LWK+10] (TUW, USTUTT). In their approach, the concepts of pointcut and aspect from aspect-oriented programming are used to define locations in a running instance where a composition fragment can be executed. Composition fragments are modeled as separate entities that implement a part of business logic. When an SLA violation is predicted, the alternative composition fragments are evaluated and the one deemed best suited to avoid violation is selected and executed when a special, “virtual” activity is encountered in the course of instance execution.

Another approach to QoS prediction based on machine learning was discussed in the work by Ejarque et al. [EMS+10] (Barcelona Supercomputing Center, SZTAKI), which focuses on cloud environments. The approach is based on dynamic allocation of cloud resources using scheduling policies negotiated between users and providers as a part of SLA. The proposed framework supports the entire life-cycle for service executions, including monitoring, collection and evaluation of historic data, identification of adaptation needs, and the adaptation of running instances itself. The prediction is based on semantically described data, in a dynamic multi-agent semantic resource allocation process, where the participants negotiate elements of their SLA with the provider, to satisfy policies that define scheduling rules that suit their particular needs.

Prediction Based on Online Testing The work by Hielscher et al. [HKMP08] (UniDue and FBK) discussed the opportunities for employing online testing to predict possible failures and SLA violations, and thus trigger the necessary adaptation mechanisms in advance. That kind of testing takes place during instance executions, by following data obtained from monitoring. The authors present a proactive self-adaptation framework named PROSA. The question of confidence with which the predictions based on online testing can be made was discussed by Metzger et al. [MSPR10] (UniDue). The approach in [MSPR10] (UniDue) is different of other uses of online testing [DDS03, WQY04, CCL07, BCS07], since it has shown how to employ online testing for triggering pro-active adaptation. In particular, they addressed the concern of avoiding unnecessary adaptations, because these can be costly and can, in some cases, end in a failure themselves. The proposed approach suggests complementing the usual statistical inference over historical data with online testing that is performed in parallel to the normal operation of a service-based system. By choosing the test cases carefully, online testing can complement the historic information in a way that promises more confident failure predictions than those made exclusively on the basis of historic data analysis.

One of the key questions related to online testing of service-based system is when, how, and how much to test. These questions were explored by Sammodi et al. [SMF+11] (UniDue). One major consideration that was identified in that respect is that services need to be tested and re-tested periodically, because third-party services can change dynamically and even without notice. In standard software testing, the tester is aware of changes in software components, and does not need to repeat the tests if there are not changes. Besides, in the standard case the tester is, in principle, informed (or in control) of all relevant environmental factors, and can therefore assume that the tests will yield the same results for the same inputs. However, these assumptions do not hold in a dynamic service environment, where the component services are typically controlled by third parties, and often visible to clients only through their external interfaces. Another key consideration is that testing is constrained by economic and technical limitations. Besides the performance hit it imposes on the run-time system, in service-based systems testing is also limited by availability of third-party components. Also, the use of those components for testing purposes can be expensive in terms of communication and money. The authors proposed a framework that applies online testing based on data on usage of services. The framework tries to use online testing to achieve as much coverage of service executions as possible, while relying on a limited number of test executions.

The work by Gehlert et al. [GMK+11] (UniDue, USTUTT and FBK) discussed the interaction between perfective and corrective adaptation, especially from the point of view of several concurrent adaptation goals, which may partly conflict with each other. The authors suggest using online testing to trigger corrective adaptation, i.e., such that it is exercised ahead of time to correct an impending SLA violation or failure, for instance by replacing the failing service with an alternative one. Requirements engineering is suggested as a basis for perfective adaptation, which takes place when a new service candidate enters the realm of a service-based system, which can be better suited for some users than the existing ones. To facilitate the integrated perfective-corrective approach, authors recommend the use of an enterprise service registry that keeps track of all internal and external services used in an enterprise SBA.

The application of online testing for proactive adaptation of conversational services was discussed by Dranidis et al. [DMK10] (SEERC, UniDue). Conversational services perform multiple message exchanges with their partners, and their failure can potentially be very expensive, because the ongoing conversations need to be rerouted to a replacement service, together with the relevant parts of the failed service state. This brings an additional level of complexity to both prediction and adaptation of conversational services. The prediction, in particular, needs to include not only the aspect of QoS compliance, but also message exchange protocol compliance. The authors presented the formal underpinnings for proactive adaptation of conversational services based on modeling the system behavior with X-Stream Machines. A new way for contextualizing tests was proposed, where the number of test cases is reduced to those relevant for a particular SBA environment. A framework is proposed which is a combination of the PROSA framework (mentioned above) and the JXSM suite for generating and executing service test cases.

Data-Aware Quality of Service Prediction The work by Ivanović, Carro and Hermenegildo [ICH09, ICH10] (UPM) discussed the possibility of predicting service composition QoS based on computation cost analysis. In their approach, a QoS of a service composition is expressed as a function of input data that is fed to the composition on invocation. That makes the analysis more complex, but leads to a higher level of precision when predicting QoS for an individual running instance that is a subject of monitoring and/or a candidate for adaptation, because composition behavior may, in general, significantly vary for different cases of input data. The complex XML data structures that carry messages are taken into account by abstracting from their components data measures such as scalar values, lengths of character strings, or the number of elements in lists and other data collections. Compositions are analyzed to obtain computational complexity functions that depend on data sizes and express the number of events (activities, service invocations, loop iterations, etc.). These computational complexity functions are then coupled with statistical estimates of infrastructure-level parameters, such as the system load, the atomic activity timing, network latency, and the degree of parallelism, to produce the QoS prediction as a function of input data.

The complexity functions (and the resulting QoS prediction functions) express, as functions of the data size, upper and lower bounds of the predicted quantities. While the complexity bounds are safe (the actual value is guaranteed to fall within them), the accuracy of QoS bounding functions depends on the accuracy of statistical estimates of the infrastructure-level parameters. In that way, the effect of the composition structure and the execution environment on the final QoS value are separately modeled. The effect of the execution environment can be based on up-to-date measurements made during composition execution, rather than on historic data that may correspond to different system load regimes, network conditions, and availability situations.

The authors illustrated the approach in two cases. In the first case, it was used to predict QoS (running time) for service orchestrations to detect, under the current infrastructure conditions, whether a running instance will meet the deadline. The prediction is performed at a series of points in the execution by calculating QoS bound functions use actual input data metrics as parameters and using the current infrastructure parameters. If the upper predicted QoS bound falls below the deadline, a failure is not predicted. If the lower predicted QoS bound falls above the deadline, an imminent violation is predicted. And in the third case, where the deadline falls between the bounds, a warning of possible violation is issued.

Another application of the approach was illustrated for the case of run-time adaptation of composition instances by choosing component services that have the lowest computational complexity for given input data. In the data-aware case, the bounds on computational cost are used as predictors of the overall QoS, and the components are chosen to minimize the overall complexity of the composition for a given input. Different complexity functions can be applied, depending on what events are measured as relevant: steps in execution, invocations to a particular external service, internal transactional operations, etc. Service descriptions are assumed to include the description of their computational cost in terms of input data in order to perform an informed selection which depends on invocation data.

Dynamic Simulation Models An approach to predicting QoS values for service compositions based on dynamic simulation models was discussed in the work by Ivanović et al. [ITCD10] (UPM, TUW). The authors start from the assumption that the QoS of a service composition that executes on a shared provider's infrastructure depends, on average, on both the structure of the composition and the capabilities of the infrastructure. Therefore, scaling up the infrastructure to meet the demands can improve the average QoS, even if no adaptation takes place on the instance or the workflow definition level in the classical sense. However, for the end-user the effects on the two are hardly distinguishable. Therefore, a simulation model for the composition with the provision resources can help providers to infer reasonable SLA levels under the assumed load levels and infrastructural resource capabilities, as well as to plan and test the policies for scaling these resources under different load change scenarios.

Starting from a Petri net model of a service workflow, a dynamic stock and flow continuous time model is derived that can be used for simulating the concurrent execution of a family of running instances on a service provider's infrastructure. That model is combined with a model of load and usage of computational resources on the provider's side, such as threads or virtual machines. The resource model may include demand-driven up- and down-scaling policies for the resources that depend on the load. The overall model is then simulated under different load regimes, including typical, corner, and extreme cases, to discover the average QoS for all executing instances under these conditions, as well as to study effects of different policies for managing provision resources.

Constraint-Based Quality of Service Prediction Using constraints to reason about possible SLA violations in service compositions was addressed by Ivanović, Carro and Hermenegildo [ICH11] (UPM). Instead of producing probabilistic estimates, this approach relies on formulating mathematical relations (equalities and inequalities) that constrain values of different parameters (such as component execution times, availability or costs) in the remainder of a composition execution for the case of SLA compliance and violation, respectively. Assumptions about possible value ranges for these parameters can also be included, if available. Structural parameters, such as branch conditions and the number of loop iterations, are also included into the analysis. Solving the constraints for the cases of successful execution and violation produces intervals of values for the parameters which are used to model the system. If the constraints cannot be solved because they are inconsistent, the corresponding case cannot be realized (i.e., it is impossible to have a successful / unsuccessful execution). The use of structural parameters makes it possible for a process monitor to foresee a violation if the execution takes a branch or starts a loop iteration outside the range inferred by solving constraints for the case of SLA conformance. The approach can be combined with the data-aware complexity functions to obtain a higher level of precision.

In principle, the constraint-based QoS prediction can be applied at any moment during execution of a composition instance as long as the current control state of execution is known. The authors suggest an architecture where the constraint-based QoS predictor receives information explicitly from the orchestration engine, in the form of a continuation that describes what remains to be executed in the orchestration from that point onwards. The predictor is then able to mechanically derive and solve constraints from the continuation. The predictions are published by the predictor and can be used by the adaptation layer to trigger the respective actions, taking into account the predicted value ranges for the cases of SLA conformance and violation.

Artificial Intelligence-planning based proactive adaptation. Run-time adaptability is a key feature of dynamic business environments, where the processes need to be constantly refined and restructured to deal with exceptional situations and changing requirements. The execution of such a system results in a set of adapted process variants instantiated on the same process model but dynamically restructured to handle specific contexts.

In such a dynamic environment, the process models cannot remain unchanged; the short-term adaptations applied to process instances should be used to pro-actively derive long-term changes in the process models.

Bucchiarone et al. [BMPS11, BMPR11] (FBK) propose an adaptation and evolution framework that exploits the information on process variants to identify the best performing recurring adaptations and adopt them as general solutions in the process model.

Since process variants are strictly related to specific execution contexts and cannot be adopted as general solutions, the framework supports context-aware evolution: instead of looking for recurring adaptations to be embedded in the model, this solution looks for recurring adaptation needs (i.e., process instances with the same context constraint violation and system configuration).

Based on the analysis of adapted instances, the framework automatically constructs and ranks corrective evolution variants which can handle the problematic context.

At the same time, exploiting sophisticated Artificial Intelligence (AI)-planning techniques, preventive evolution variants are identified by synthesizing process variants which can prevent the adaptation need. A prototype framework has been implemented (CAptEvo) [BMPR11] and the solution has been evaluated on a car logistics scenario.

Static and Run-Time Quality of Service Analysis for Orchestrations Mukherjee et al. [MJN08] studied the problem of determining QoS for BPEL processes. Their approach takes the information on QoS of composite services (the partner services in BPEL), and uses the semantics of control dependencies between BPEL activities to build an activity graph, which is then analyzed to produce the estimate of running time, reliability or cost for the given process. The approach treats a subset of BPEL that is well structured and does not include loops in the activity graph. Designers can use the approach to experiment with advanced fault handling techniques. The authors developed a tool that can be used to calculate QoS for BPEL v1.1 processes.

Calculation of QoS for composite services, taking into account real-world, complex processes, such as those written in BPEL, was studied by Dumas et al. [DGBP+10]. Their approach supports sequential and parallel constructs that can be expressed with Direct Acyclic Graphs (DAGs), as well as single-entry-multiple-exit loops. The derivation of QoS proceeds in two steps: first the control structure that links different orchestration components is analyzed, and then the QoS is aggregated in a bottom-up manner. The QoS values handled by the approach are understood as averages from the appropriate statistical distributions.

Provision Adaptation Driven-by Service Level Agreements Cardellini et al. [CCGP10] proposed a brokering service for the adaptive management of composite services. The goal of this broker is to dynamically adapt at runtime the composite service configuration, to fulfill the SLAs, despite variations of the operating environment. The authors specifically consider SLAs that specify upper bounds on the percentile of the service response time, which is expected to capture user perceived QoS with greater precision. The approach is based on a service selection scheme that minimizes the service broker cost while guaranteeing the negotiated QoS to the different service classes. The optimal service selection is determined by means of a scalable linear programming problem that can be efficiently solved.

Reliable execution time driven execution of service workflows was discussed by Stein et al. [SPJ11]. The authors proposed a novel algorithm that allows service consumers to execute business processes (or workflows) of interdependent services in a dependable manner within tight time constraints. The focus was on large inter-organizational service-oriented systems, where services are offered by external organizations that demand financial remuneration and where their use has to be negotiated in advance using explicit SLAs. Here, different providers often offer the same type of service at varying levels of quality and price, and some providers may be less trustworthy than others, possibly failing to meet their agreements. To control this unreliability, a system for selection of most suitable providers is proposed, based on advance reservation of services. The reservation introduces redundancy (which is minimized) in order to retain flexibility when failures occur.

Bibliography

[Apa11] Apache Software Foundation. *Apache ODE Project*, 2011.

[BCS07] Xiaoying Bai, Yinong Chen, and Zhongkui Shao. Adaptive web services testing. In *COMPSAC (2)*, pages 233–236, 2007.

[BMPR11] Antonio Bucchiarone, Annapaola Marconi, Marco Pistore, and Heorhi Raik. CAptEvo: Context-aware adaptation and evolution of business processes. In *ICSOC 2011, Demo Session*, December 2011.

[BMPS11] Antonio Bucchiarone, Annapaola Marconi, Marco Pistore, and Adina Sirbu. A context-aware framework for business processes evolution. In *EVL-BP 2011*, 2011.

[CCGP10] Valeria Cardellini, Emiliano Casalicchio, Vincenzo Grassi, and Francesco Lo Presti. Adaptive management of composite services under percentile-based service level agreements. In *ICSOC*, pages 381–395, 2010.

[CCL07] W. Chan, S. Cheung, and K. Leung. A metamorphic testing approach for online testing of service-oriented software applications. *Int. J. Web Service Res.*, 4(2):61–81, 2007.

[CD-10] Initial set of principles, techniques and methodologies for assuring end-to-end quality and monitoring of SLAs. Deliverable, S-Cube Consortium, 2010.

[CD-11] Integrated principles, techniques and methodologies for specifying end-to-end quality and negotiation SLAs and for assuring end-to-end quality provision and SLA conformance (incl. proactiveness). Deliverable, S-Cube Consortium, 2011.

[DBL10] *IEEE International Conference on Web Services, ICWS 2010, Miami, Florida, USA, July 5-10, 2010*. IEEE Computer Society, 2010.

[DDS03] Peter H. Deussen, George Din, and Ina Schieferdecker. A ttcn-3 based online test and validation platform for internet services. In *ISADS*, pages 177–184. IEEE Computer Society, 2003.

[DGBP+10] Marlon Dumas, Luciano García-Bañuelos, Artem Polyvyanyy, Yong Yang, and Liang Zhang. Aggregate quality of service computation for composite services. In *ICSOC*, pages 213–227, 2010.

[DMK10] Dimitris Dranidis, Andreas Metzger, and Dimitrios Kourtesis. Enabling proactive adaptation through just-in-time testing of conversational services. In *ServiceWave*, pages 63–75, 2010.

[EMS+10] J. Ejarque, A. Micsik, R. Sirvent, P. Pallinger, L. Kovacs, and R. M. Badia. Semantic resource allocation with historical data based predictions. In *The First International Conference on Cloud Computing, GRIDs, and Virtualization, CLOUD COMPUTING 2010*, 2010.

[GMK+11] A. Gehlert, A. Metzger, D. Karastoyanova, R. Kazhamiakin, K. Pohl, F. Leymann, and M. Pistore. *Service Engineering: European Research Results*, chapter Integrating perfective and corrective adaptation of service-based applications. Springer-Verlag/Wien, 2011.

[HFH+09] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: an update. *SIGKDD Explorations*, 11(1):10–18, 2009.

[HKMP08] Julia Hielscher, Raman Kazhamiakin, Andreas Metzger, and Marco Pistore. A framework for proactive self-adaptation of service-based applications based on online testing. In Petri Mähönen, Klaus Pohl, and Thierry Priol, editors, *ServiceWave*, volume 5377 of *Lecture Notes in Computer Science*, pages 122–133. Springer, 2008.

[ICH09] Dragan Ivanovic, Manuel Carro, and Manuel V. Hermenegildo. An initial proposal for data-aware resource analysis of orchestrations with applications to predictive monitoring. In *ICSOC/ServiceWave Workshops*, pages 414–424, 2009.

[ICH10] Dragan Ivanovic, Manuel Carro, and Manuel V. Hermenegildo. Towards data-aware qos-driven adaptation for service orchestrations. In *ICWS* [DBL10], pages 107–114.

[ICH11] D. Ivanović, M. Carro, and M. Hermenegildo. Constraint-Based Runtime Prediction of SLA Violations in Service Orchestrations. In Gerti Kappel, Hamid Motahari, and Zakaria Maamar, editors, *Service-Oriented Computing – ICSOC 2011*, number TBD in LNCS. Springer Verlag, December 2011.

[ITCD10] Dragan Ivanovic, Martin Treiber, Manuel Carro, and Schahram Dustdar. Building dynamic models of service compositions with simulation of provision resources. In Jeffrey Parsons, Motoshi Saeki, Peretz Shoval, Carson C. Woo, and Yair Wand, editors, *ER*, volume 6412 of *Lecture Notes in Computer Science*, pages 288–301. Springer, 2010.

[LMRD10] Philipp Leitner, Anton Michlmayr, Florian Rosenberg, and Schahram Dustdar. Monitoring, prediction and prevention of SLA violations in composite services. In *ICWS* [DBL10], pages 369–376.

[LWK+10] Philipp Leitner, Branimir Wetzstein, Dimka Karastoyanova, Waldemar Hummer, Schahram Dustdar, and Frank Leymann. Preventing SLA violations in service compositions using aspect-based fragment substitution. In *ICSOC*, pages 365–380, 2010.

[LWR+09] Philipp Leitner, Branimir Wetzstein, Florian Rosenberg, Anton Michlmayr, Schahram Dustdar, and Frank Leymann. Runtime prediction of service level agreement violations for composite services. In *ICSOC/ServiceWave Workshops*, pages 176–186, 2009.

[MBC+10] Andreas Metzger, Salima Benbernou, Manuel Carro, Maha Driss, Gabor Kecskemeti, Raman Kazhamiakin, Kyriakos Kritikos, Andrea Mocci, Elisabetta Di Nitto, Branimir Wetzstein, and Fabrizio Silvestri. Analytical quality assurance. In *S-CUBE Book*, pages 209–270, 2010.

[MJN08] Debdoot Mukherjee, Pankaj Jalote, and Mangala Gowri Nanda. Determining qos of ws-bpel compositions. In *ICSOC*, pages 378–393, 2008.

[MSPR10] Andreas Metzger, Osama Sammodi, Klaus Pohl, and Mark Rzepka. Towards pro-active adaptation with confidence: augmenting service monitoring with online testing. In *Proceedings of the 2010 ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems*, SEAMS '10, pages 20–28, New York, NY, USA, 2010. ACM.

[SMF+11] O. Sammodi, A. Metzger, X. Franch, M. Oriol, J. Marco, and K. Pohl. Usage-based online testing for proactive adaptation of service-based applications. In *COMPSAC 2011 – The Computed World: Software Beyond the Digital Society*. IEEE Computer Society, 2011.

[SPJ11] Sebastian Stein, Terry R. Payne, and Nicholas R. Jennings. Robust execution of service workflows using redundancy and advance reservations. *IEEE T. Services Computing*, 4(2):125–139, 2011.

[WLR+09] Branimir Wetzstein, Philipp Leitner, Florian Rosenberg, Ivona Brandic, Schahram Dustdar, and Frank Leymann. Monitoring and analyzing influential factors of business process performance. In *EDOC*, pages 141–150. IEEE Computer Society, 2009.

[WQY04] Qianxiang Wang, Lining Quan, and Fuchen Ying. Online testing of web-based applications. In *COMPSAC Workshops*, pages 166–169. IEEE Computer Society, 2004.

2.17 Modeling, Design and Performance Analytics of Service Networks

Involved Workpackages: WP-JRA-1.2, WP-JRA-2.1

There have been substantial research efforts within the S-Cube project and across the wider research community to develop understandings on employing various methods to analyze service networks and performance analytics. This section summarizes some of these efforts to provide a state-of-the-art account of service networks analysis research developments. The design, management and delivery of complex service systems suggests that we must develop a scientific understanding regarding the configuration of resources to deliver service excellence. In recent years, there have been growing interests in examining the relational infrastructure of service environments. Moving from a classical business-centric view of services to a scientific view of service environment saw the emergence of the "service science" discipline. Many of the initial works in service science sought the need to examine services from alternative perspectives, for example, educational, engineering, management, and design. We provide an overview of these research developments.

Many research efforts were carried out on service networks within the S-Cube project. For example, [ZNP+09] (CITY, UOC) examine the consumption of services and consumers value preferences by conducting value analysis of service networks and quality of service requirements. This work draws one's attention towards the viability and adaptability of the dynamics and evolving service ecosystems, for example cloud environments and service adaptation. In addition, [DKL10] (USTUTT) examines how organizations rely on the consumption and provision of services to survive within a business environment. This work highlights the importance of sustaining and managing relationships within the service networks. To build on the importance of modeling service relationships, [DKL10] (USTUTT) suggests that there is a need to introduce models to examine service networks and they focus on business relationships which are mapped with service networks modeling formalism to Business Process Model and Notation v1.x (BPMN v1.x). In addition, this work provides some overlap with context modelling which examines the underlying properties and dynamics of service networks. Thus, mapping services is a critical issue highlighted with the S-Cube project and even more important nowadays with the increasing digitization of service infrastructures. The Internet has a significant influence on service networks as [BDvdH+08] (UOC, USTUTT, Tiburg) highlights the transformation of service networks fostering the co-production of value through highly distributed networks of autonomous trading partners. [BDvdH+08] (UOC, USTUTT, Tiburg) are also concerned about mapping service networks to business processes and introduce a semi-automatic model transformation approach for creating the abstract business. Therefore, these approaches offer an opportunity to examine service network evolution and its influence on service performance.

Over the past few years, the concept of GSD provides a relational infrastructure which models the socio-technical dynamics of service delivery. In addition, [WDL+09] (USTUTT, UOC, Tiburg) examine the co-creation relationship among service network actors. They report on the need to calculate value across service networks through the use of KPIs. This suggests the need to introduce a new approach on service metrics [CWR10a, CWR10b] (LERO). KPIs are used to measure the performance of underlying cross-organizational business processes. Developing this notion a little further, the importance of a service relational structure is also highlighted by [CWR10a] (LERO). Here, they examine the need to address the significant gap in the manager's ability to value the contributory interaction of service networks in organizational performance through network dynamics metrics. [CWR10a] (LERO) explore how service (re)configuration is utilized to optimize network performance. They posit that managers must transform information on network activity and infrastructural capabilities into strategic knowledge and discuss how Social Network Analysis (SNA) can be a powerful tool for managers to understand organizational network performance and service interaction. Combining these efforts, [CWR10b] (LERO) also highlight the significance of understanding KPIs across service networks. They explain how the unprecedented growth in service-based business processes over a short period of time has underscored the need for understanding the mechanisms and theorizing the business models and business process management adopted across services. [CWR10b] (LERO) suggest that SNA provides a method to enhance a manager's ability to monitor KPIs while improving business process restructuring practices while incorporating a new approach to service analytics, i.e. shifting attention from financial to service relational structures. This is particularly true within an adaptive service system, such as the cloud, which demonstrates an agile service infrastructure. Understanding the service structure of cloud environments provides insights on SLA negotiations and agreements across the cloud infrastructure.

While remaining focused on the importance of analyzing the service network relational structures, [CRW11] (LERO) offers an alternative approach to the traditional service modeling approaches through Actor-Network Theory (ANT). [CRW11] (LERO) highlight the need to explore the socio-technical nature of service networks to investigate how service networks are formed, stabilizes, or even dissolved. While adopting ANT, [CRW11] (LERO) proposes the need to examine the intertwining dynamics are and how they are influenced by technology. Understanding these dynamics is vital to determine how IT-enabled service networks shape the social environment and how the social environment shapes the technological factors within a service network environment. Therefore, it is critical to examine the impact of implementing IT on service network dynamics. ANT is often described as a systematic approach to explore the infrastructure which supports the "scientific and technological achievements" within a network making it a more radical approach to researching service networks. ANT suggests that the world is made up of complex networks which are comprised of many complex interactions (local and global) which constantly reconfigure itself on a regular basis [Law99].

[Dub] examines service network analysis within BAM, with particular attention on discovering effective and efficient ways to adapt business processes to rapid changing environments, requirements, and collaborating entities. The focus of [Dub] work is on the notion that one must shift the focus of service network analysis away from a "value chain" concept and towards a "value network" concept. There have been some interesting developments on the notion of value networks. For example, [RZGLA11] presents a frameworks to examine service bundling and the exchange of outcomes between customers, brokers, and suppliers. The emphasis here is to develop an understanding on a cluster-based bundling process which is demonstrated through a case study. In addition [RZLG11] explains that the delivery of services continue to face new challenges. They suggest that Web 2.0 developments alter the concept of service interactions and consequently challenge the notion of "value co-creation" in services. [RZLG11] suggest that customers can contribute towards the bundling of services through a brokerage service matching customer's needs with supplier offerings.

On a similar research direction, [GLRZ11] discuss how enterprises jointly participate in "service value webs", in order to benefit fro the co-production of service which they could not produce on an individual basis. Following on from the development of the e3value methodology, they propose another approach to semi-automatically generate such service value networks. The rationale here is that service value webs may be created through predefined service profiles which match consumer needs. [RZG09] draws one's attention towards the VALUE-IT project which addresses the need to develop service adaptation configuration techniques through the e3value framework. The motivation for this work centers on the need to match consumer needs through an automated service configuration. These efforts are expanded upon my [RZCG$^+$10] as they presents an interesting approach to automatically generate e3value instance models, based on skeletal design techniques which consist of three main phases employed to generate value and matching service capabilities of service providers and service needs. The novel approach presented by [RZCG$^+$10] is the reuse of a set of value skeletons for covering an industry sector from specific business cases.

As outlined above, understanding service behavior is a critical factor in service composition and analysis. For example, modeling service "value" is a complex task. [BCW10] discuss the emergence of Service Value Networks (SVNs) which attempt to address factors surrounding service composition and a rising demand for customized services. They discuss how distributed contribute to an overall value proposition of a SVN. However [BCW10] highlights that although there is a need to introduce ad-hoc composition of services that satisfies individual customers' demands, there is a lack of purposeful design of incentives in service composition. To address this problem, [BCW10] introduce an auction to coordinate value creation in SVNs and increase a network's degree of interoperability through an agent-based simulation method. Modeling the aligning of business value with IT is therefore a critical task within service environments. This is also highlighted by [DGT08] as they examine methods to introduce innovative e-customs procedures though the alignment process in a multi-perspective way. One of the critical issues highlighted by [DGT08] in adopting this approach is scalability within a service environment. To address this issue, they translate and fit business-driven requirements with system capabilities to enroll e-customs procedures. In an attempt to develop this concept, [DGT08] develop a model-based approach using e3value and UML modeling techniques to support the alignment process.

Modeling service value plays a central role on one's ability to understand service provision. Service analytics is also highlighted by [dKG08] as they discuss the provision of services via the Internet which are increasingly offered by a multi-supplier bundle of services. One question posed by [dKG08] however, is how can one "automatically compose these e-service bundles". They examine how a computer-processable theory, i.e., an ontology, can represent customer needs while also facilitating computer-supported elicitation of needs though Web-based software. Much attention has been placed on the digitization of services and examining the concept of "value". [GdKW08] review the provision of services online. [GdKW08] examine the problem of composing e-service bundles to match consumer's needs through an economically viable model. They suggest that this problem is primarily concerned with "requirements engineering", i.e. the ability to meet consumer requirements. [GdKW08] propose a multi-supplier bundle of commercial e-services which may be used by suppliers to build a service catalog that describes their offerings. Thus, this research development shifts the concept of value from the service to the consumer.

To extend this notion [PG07] discusses the need to examine "networked value constellations" to enable value for the organization, technology, and the customers. [PG07] discuss that although modeling approaches for networked constellations, such as e3value, have been developed in the past, they extend these business modeling approaches to understand the strategic rationale behind business models and the motivations behind various value-creating actions.

To support these research efforts, S-Cube continues to highlight the importance of mapping representation of service network value co-creation behavior [Dub] (Polimi). In practice this work is critical as managers realize that the key to continued success is within their understanding of how workflows and business processes can be optimized and generate higher value mechanisms. Thus, service networks accommodate well-defined, collaborative end-to-end processes directed towards value co-creation. Research efforts continue to examine modeling and simulation concepts and mechanisms that span from logical end-to-end processes in service networks.

Bibliography

[BCW10] Benjamin Blau, Tobias Conte, and Christof Weinhardt. Incentives in Service Value Networks – On Truthfulness, Sustainability, and Interoperability. In *ICIS 2010 Proceedings*, Saint Louis, Missouri, USA, 12 2010. Paper 8.

[BDvdH+08] Marina Bitsaki, Olha Danylevych, Willem-Jan van den Heuvel, George Koutras, Frank Leymann, Michele Mancioppi, Christos Nikolaou, and Mike P. Papazoglou. Model transformations to leverage service networks. In George Feuerlicht and Winfried Lamersdorf, editors, *ICSOC Workshops*, volume 5472 of *Lecture Notes in Computer Science*, pages 103–117. Springer, 2008.

[CRW11] Noel Carroll, Ita Richardson, and Eoin Whelan. Service science: Introducing the need for performance analytics for service networks evolution. 05/2011 2011.

[CWR10a] Noel Carroll, Eoin Whelan, and Ita Richardson. Applying social network analysis to discover service innovation within agile service networks. *Service Science*, 2:pp 225–244, November 2010.

[CWR10b] Noel Carroll, Eoin Whelan, and Ita Richardson. The discovery of agile service networks through the use of social network analysis. In *Proceedings of the 2010 International Conference on Service Sciences*, ICSS '10, pages 131–137, Washington, DC, USA, 2010. IEEE Computer Society.

[DGT08] Zsofia Derzsi, Jaap Gordijn, and Yao-Hua Tan. Towards model-based assessment of business-it alignment in e-service networks from multiple perspectives. In *Proceedings of 16th European Conference on Information Systems*, 2008.

[dKG08] Sybren de Kinderen and Jaap Gordijn. E3service: A model-based approach for generating needs-driven e-service bundles in a networked enterprise. In *Proceedings of 16th European Conference on Information Systems*, 2008.

[DKL10] Olha Danylevych, Dimka Karastoyanova, and Frank Leymann. Service networks modelling: An soa & bpm standpoint. *J. UCS*, 16(13):1668–1693, 2010.

[Dub] Daniel J. Dubois. An approach for improving business process management in agile service networks. Minor research report, Politecnico di Milano (POLIMI).

[GdKW08] Jaap Gordijn, Sybren de Kinderen, and Roel Wieringa. Value-driven service matching. In *RE*, pages 67–70. IEEE Computer Society, 2008.

[GLRZ11] Jaap Gordijn, Pieter De Leenheer, and Iván S. Razo-Zapata. Generating service value webs by hierarchical configuration: A case in intellectual property rights clearing. In *HICSS*, pages 1–10. IEEE Computer Society, 2011.

[Law99] J. Law. After ANT: Complexity, naming and topology. In J. Law and J. Hassard, editors, *Actor network theory and after*, pages 1–14. Blackwell/Sociological Review, Malden, MA, 1999.

[PG07] Vincent Pijpers and Jaap Gordijn. E3forces: Analyzing strategic considerations of actors in networked value constellations. Technical report, Vrije Universiteit Amsterdam, 2007.

[RZCG+10] Ivan Razo-Zapata, Ania Chmielowiec, Jaap Gordijn, Maarten Van Steen, and Pieter De Leenheer. Generating value models using skeletal design techniques. In *In Proceedings of the 5th international BUSITAL workshop*, 2010.

[RZG09] Ivan Razo-Zapata and Jaap Gordijn. Automatic service configuration under e3value approach. In Hans Weigand and Sjaak Brinkkemper, editors, *Doctoral Consortium CAiSE 2009*, volume 479 of *Workshop proceedings*. CEUR, 2009.

[RZGLA11] Ivan S. Razo-Zapata, Jaap Gordijn, Pieter De Leenheer, and Hans Akkermans. Dynamic cluster-based service bundling: A value-oriented framework. In *Commerce and Enterprise Computing (CEC), 2011 IEEE 13th Conference on*, 2011.

[RZLG11] Ivan S. Razo-Zapata, Pieter De Leenheer, and Jaap Gordijn. Value-based service bundling: a customer-supplier approach. In *In Proc. of Service-oriented Enterprise Architecture for Enterprise Engineering (SoEA4EE) Scientific Workshop (EDOC 2011)*, 2011.

[WDL+09] Branimir Wetzstein, Olha Danylevych, Frank Leymann, Marina Bitsaki, Christos Nikolaou, Willem-Jan van den Heuvel, and Mike Papazoglou. Towards monitoring of key performance indicators across partners in service networks. In *International Workshop Series on Monitoring, Adaptation and Beyond (MONA+), collocated with ICSOC/ServiceWave 2009, Stockholm, Sweden*, Berlin–Heidelberg, February 2009. Springer.

[ZNP+09] K. Zachos, C. Nikolaou, P. Petridis, G. Stratakis, M. Voskakis, and E. Papathanasiou. Enhancing service network analysis and service selection using requirements-based service discovery. In *1st International Conferences on Advanced Service Computing, Service Computation 2009*, Athens, Greece, 2009.

2.18 Modeling, Analysis and Monitoring of Business Transactions

Involved Workpackages: WP-JRA-1.2, WP-JRA-2.1

This research line draws upon knowledge, concepts and techniques from various disciplines including SOC, Computer Supported Cooperative Work (CSCW), BAM and Software Engineering (SE). In particular, it defines how a business transaction should be viewed, which incorporates the process-level approach with the more conventional applications-level view.

Over the past two years, novel research results on business transactions have emerged focusing on a multi-modal transaction processing scheme for reliable business transactions that span from front-end SBAs to back-end system-level transaction support and beyond to organizations that are part of a Service Network (SN). In particular, the latest research promotes rich features and QoS similar to those offered by transaction processing monitors but at the level of application management and application control logic, known as business transactions.

In particular, advances in the state-of-the-art in modeling, analysis and monitoring of business transactions include the definition of the concept of “business transactions” living in agile SNs [CD-09] (Tilburg, UCBL, LERO, UOC, USTUTT). The business transaction then becomes the framework for expressing detailed operational business semantics. Conventional approaches to business transactions, such as Open EDI, UMM and ebXML, focus only on the documents exchanged between partners, rather than coupling their application interfaces, which inevitably differ. [CD-09] (Tilburg, UCBL, LERO, UOC, USTUTT) has targeted the concept of a business transaction and explored how process fragments, and particularly transactional process fragments, fit in the context of a running scenario which possesses transaction properties. Conventional (ACID) and unconventional (application-based) types of atomicity were introduced, including contract and delivery atomicity, in the frame of a business transaction model. The transaction model provides a comprehensive set of concepts and several standard primitives and conventions that can be utilized to develop complex SBAs involving transactional process fragments.

Modeling business transaction covers two different aspects: operational and structural. Much of the recent research works have been focused on operational aspect of business transaction. The operational aspect entails transaction property specification, guarantee correctness, recovery mechanism, and execution of transaction operation. A collection of developments inside and outside of the S-Cube project has been found revolving around the operational aspect of business transaction since the project started. The developments include language, framework, and theories for modeling business transaction. We briefly outline some of them in this section. We also summarize few significant developments within business transaction monitoring and analysis.

Transactions in networked business environment need to correlate the collaboration aspects such as mutual obligations of business processes. The close correlation of these aspects with transactions provides an opportunity to model, operate, and manage transactions from business perspective. Realizing this requirement, a business transaction model incorporating collaboration aspects has been developed in [HPvdH+10] (UCBL, Tilburg, LERO, USTUTT, UOC, UniHH). The collaboration aspects are called Granular Process Properties (GPP) determine the transactional properties *atomicity* and *consistency* for business activities that compose complex SBAs. Additionally, the authors propose Business Transaction Language (BTL) to model customizable business transaction for SBAs. It is a declarative language that facilitates to specify the activities and granular process properties. In [HHP10] (UCBL, Tilburg), the formal model of business transaction language is introduced. This research work is at preliminary stage where only the basic semantics of the language has been defined formally. The authors provide formal definition of vital transactions and transactional and temporal patterns. In addition, some of the granular process properties have also been formalized. The next stage of this research is expected to deliver a complete formal model of BTL.

BPMN v1.x is a model-driven approach that facilitates modeling business transaction. BPMN v1.x provides notations including *transaction*, *intermediate cancel event*, and *compensate activity* for modeling business transaction. A transactional business process can be assembled (modeled) using these constructs. The semantics of BPMN v1.x transaction notation has been formalized using Petri net in [Tak08]. The authors develop a mapper using a block of Petri net which produces formal semantics for transactions. However, BPMN v1.x transaction-related constructs are very basic, and do not facilitate the specification of transactional attributes. In [BGST10], the authors extend BPMN v1.x to facilitate annotating transactional attributes in a business process. In addition, a flow type that subsumes control and message flow is introduced to link two or more transaction participants within the same scope. The main advantage of this new artifact is that it enables reconfiguring (remodeling) business transaction dynamically.

Flexibility with respect to behavior is a strong requirement for business transaction in service based application. The semantics of classical atomicity and isolation is extended in [CHWR11] (LERO) with the aim of supporting flexible business transactions. The authors introduce *eventual failure atomicity* and suggest *relaxed isolation* properties – know since the early 80's – to model flexible behavior of business transactions. The eventual failure atomicity prevents a business transaction from being aborted entirely upon the failure of a component transaction, i.e. a *subtransaction*. The transaction manager forces (if needed) other component transaction to continue and simultaneously invokes suitable repairing actions for failed transaction depending on the failure type(e.g., system failure). The compensation transaction, i.e. to *undo the effects of completed component transactions*, functions if the failed transaction cannot be repaired. As for relaxed isolation, the authors suggest eliminating the notion of hierarchical sphere from the business transaction tree and introduce *collaboration sphere*. This implies that there is no control of the parent transaction over its child transactions in terms of the visualization of transaction effects. Therefore, the effects are immediately propagated to parent and siblings after that a child transaction has committed. More importantly, the effects propagate immediately after the commit to all other participant transactions within the same collaboration sphere that was constructed through a mutual agreement among the participants.

Furthermore, to increase flexibility, a constraint based recovery mechanism for business transaction is introduced in [CZML10]. The authors propose a set of actions including ignore, retry, and substitution to support forward recovery of business transactions. Upon failure of a component transaction, one of these actions is performed depending on some conditions. The same business transaction recovery mode (forward) with different mechanism is suggested in [SDN08] to increase flexibility in business transactions. The authors propose a framework based on the concept of abstract service, which acts as a mediator for compensation transactions that adopt forward recovery. Unlike the traditional compensation actions, this framework offers strategic compensation actions within business transactions.

Guaranteeing business transaction correctness in networked business environment is a huge challenge because business transactions in service oriented environment are controlled by various policies, e.g. business policy, compliance policy and other obligations. Conflict of policies during transaction processing may result in the abortion of a business transaction. In [HH11] (UCBL), the authors develop a theoretical solution to detect the compliance policy conflict between transactional activities. The authors provide conflict-driven tableau depth first search for Linear Temporal Logic (LTL) that is translated from a high-level compliance specification. This approach guarantees the correctness by detecting compliance policy inconsistencies in the business transaction. Additionally, in [SY07], a transaction model has been developed to support consistent business transactions in service oriented networked business environments. The model is developed with the aim of guaranteeing reliable business transactions by concentrating on ensuring the correctness of various facets of messages that are exchanged between transaction participants. The model separates control flow from message flow so that the correctness of the message types can be verified and also control flow can direct message flow at runtime based on message properties, organization policies, and business rules.

Transaction monitors are primarily responsible for collecting real-time information about transaction status and performance at runtime. In [WLR+09] (USTUTT, TUW), the authors provide a three-layer framework for monitoring and analysing factors that influence the performances of business activities. The framework integrates two monitoring components: Process Performance Metrics (PPMs) monitor and QoS monitor. PPM monitor measures the performance of process elements specifically, the events of business process at runtime. QoS monitor is an external component measures the availability of a process engine and partner web services. The information collected by both monitors is used to measure the performance of business transactions as a whole. Significant developments have also been booked by the software industry in the last few years. *OpTier* is a solution for business transaction management [Ana10]. OpTier has integrated a component in its recent version called *CoreFirst* to monitor cross-organizational business transactions. It monitors transaction response time, sequence, processing time, and latency, which are all metrics used to measure performance of business transactions. Unfortunately, only limited technical details of this solution has been made public to date.

Lastly, [vdHP10] (Tilburg) proposes a novel business transaction model for smart(er) service networks, which will leverage pooling real-time information from software services and sensor networks or mobile devices to more effectively manage fine-grained tenets of service based applications, such as critical business data, local and aggregated QoS, and associated KPIs. This will guarantee a continuous and cohesive information flow, correlation of end-to-end process properties, and termination and accuracy of interacting business processes driven by service application integration logic.

Bibliography

[Ana10] Clabby Analytics. Following transactions through the cloud: Better put OpTier in your budget. Technical report, September 2010.

[BGST10] Laura Bocchi, Roberto Guanciale, Daniele Strollo, and Emilio Tuosto. Bpmn modelling of services with dynamically reconfigurable transactions. In Paul P. Maglio, Mathias Weske, Jian Yang, and Marcelo Fantinato, editors, *ICSOC*, volume 6470 of *Lecture Notes in Computer Science*, pages 396–410, 2010.

[CD-09] Business transaction language. Deliverable, S-Cube Consortium, 2009.

[CHWR11] Noel Carroll, Rafiqul Haque, Eoin Whelan, and Ita Richardson. Modelling business transactions across service supply chain networks. In *Proceedings of the 20th International Conference on Information Systems Development*. Springer, 2011.

[CZML10] Jiuxin Cao, Biao Zhang, Bo Mao, and Bo Liu. Constraint rules-based recovery for business transaction. In *GCC*, pages 282–289. IEEE Computer Society, 2010.

[HH11] FranÃ§ois Hantry and Mohand-Said Hacid. Handling conflicts in depth-first-search for LTL tableau to debug compliance based languages. In EPTCS, editor, *FLACOS*, September 2011.

[HHP10] François Hantry, Mohand-Said Hacid, and Mike P. Papazoglou. Formal model for business-aware transaction management. Technical Report RR-LIRIS-2010-017, LIRIS UMR 5205 CNRS/INSA de Lyon/Universitè Claude Bernard Lyon 1/Universitè Lumière Lyon 2/École Centrale de Lyon, September 2010.

[HPvdH+10] Francois Hantry, Mike P. Papazoglou, Willem-Jan van den Heuvel, Rafique Haque, Eoin Whelan, Noel Carroll, Dimka Karastoyanova, Frank Leymann, Christos Nikolaou, Winfried Lamersdorf, and Mohand-Said Hacid. Business process management. In *S-CUBE Book*, pages 27–54, 2010.

[SDN08] Michael Schäfer, Peter Dolog, and Wolfgang Nejdl. An environment for flexible advanced compensations of web service transactions. *TWEB*, 2(2), 2008.

[SY07] Haiyang Sun and Jian Yang. CoBTx-Net: A model for reliability verification of collaborative business transaction. In *Business Process Management Workshops*, pages 220–231, 2007.

[Tak08] Tsukasa Takemura. Formal semantics and verification of bpmn transaction and compensation. In *APSCC*, pages 284–290, 2008.

[vdHP10] Willem-Jan van den Heuvel and Mike P. Papazoglou. Toward business transaction management in smart service networks. *IEEE Internet Computing*, 14(4):71–75, 2010.

[WLR+09] Branimir Wetzstein, Philipp Leitner, Florian Rosenberg, Ivona Brandic, Schahram Dustdar, and Frank Leymann. Monitoring and analyzing influential factors of business process performance. In *EDOC*, pages 141–150. IEEE Computer Society, 2009.

2.19 Modeling and Analysis of Service Orchestrations

Involved Workpackages: WP-JRA-2.2

Extending the classical service control flow with transactional aspects encapsulating a set of recovery mechanisms is widely accepted as a method towards ensuring service reliability. In [GRGH07] (INRIA), the authors present a transactional orchestration model based on transactional services modeled using state models. In order to describe the order of their invocation, and the conditions under which these services are invoked, dependencies between services are specified, showing how the behavior of a service influences the behavior of other services. Dependencies can either deal with the normal execution of an orchestration or specify the recovery mechanisms following service failures, thus allowing for a combined control and transactional flow representation. The approach relies on process monitoring techniques such as the ones analyzed in Section 2.23, in order to gather information about service behavior. An extension of traditional workflow patterns is conducted, in order to support transactional flow. In a similar sense, [WWD08] models service orchestrations using Concurrent Transaction Logic (CTR). A translation process is provided, resulting in a CTR formula for each distinct BPEL activity type. Bocchi et al. [BGST10] enrich BPMN v1.x with transactional attributes in order to support long running transactions in business process modeling. Similarly to the goals of service evolution approaches analyzed in Section 2.3, the authors allow for dynamically reconfiguring BPMN v1.x processes based on exploiting these transactional attributes.

Model-driven approaches to service orchestration have also been well-researched in recent years. In [MSK08], the authors extend UML in order to support service orchestration modeling. The extensions mainly focus on supporting event and compensation handling. The authors also describe the translation process from those extended models to BPEL code by employing a depth-first rule-based approach to the conversion, which uses a partitioning algorithm to group UML activity diagram nodes for implementation by a certain BPEL structured activity. Zhang and Duan [ZD08] take a different approach for translating UML activity diagrams to BPEL. They first decompose the diagrams into fragments called Single-Entry-Single-Exit (SESE) regions, where there is exactly one entry edge and exactly one exit edge. Three generic types of such regions are identified and a relevant BPEL code is associated. Hence, the translation process involves decomposing the diagrams into regions and generating the associated BPEL code.

On the other hand, Brambilla et al. [BDF09] use BPMN v1.x to model service orchestrations that also support human-executed tasks, then transform these models into a WebML representation, covering both the service interactions in the orchestration and the user interface front-end of the application. Daniel et al. [DST+10] focus exclusively on modeling user interface orchestrations, using an extension of BPEL called BPEL4UI. [MGWD09] takes a different direction, extending BPMN v1.x in order to support vocabularies and rules, which allow for the externalization of business logic. The rules are expressed using the REWERSE Rule Markup Language.

A significant amount of research has focused on declarative modeling of service orchestrations. Goedertier et al. [GV07] first explored the definition of a vocabulary for declarative process modeling, using the Semantics of Business Vocabulary and Rules (SBVR) language to describe service capabilities and service instances. They also employed Colored Petri Nets for the execution of such declarative process models, providing a complete service lifecycle model in terms of generic state transition. For more details on lifecycle modeling, please refer to Section 2.1. The more recent work of Pinto [Pin11] presents Declarative Service Orchestration Language (DSOL), which separates orchestration modeling into different aspects such as the orchestration interface, the orchestration goal, a set of facts that are true at invocation time, as well as a set of abstract actions that constitute the orchestration, written in a simple logic-based language, which are then translated to concrete actions (e.g. service invocations) written in Java.

A separate research direction in service orchestration modeling involves employing Semantic Web description frameworks such as OWL-S and WSMO. Research in S-Cube has focused on exploring problems in employing such semantic specifications in orchestration modeling. In [BP09] (UOC), the authors identify the existence of the frame problem in Web service specifications and devise a solution scheme that involves shifting from a procedure-oriented perspective of a service specification, to a state-oriented one. Change axioms are introduced, that declare, for each element of a service specification, which services may result in changing them. To facilitate the production of change axioms, an algorithm was devised that automatically generates change axioms for services and service orchestrations.

As far as service orchestration analysis is concerned, there has been significant effort devoted by S-Cube partners in the direction of creating dynamic orchestration models to assist in adapting the behavior of a service provision chain. Section 2.26 offers a detailed state-of-the-art on SLA-based service provision approaches. In [ITCD10] (UPM, TUW), the authors assume Petri-net representations of orchestrations, that are then converted to ordinary differential equations, with equation variables corresponding to places and transitions. The analysis of such continuous-time orchestration models, along with a computation resource models, offers insight into the selection of different resource management strategies.

In [SM08], the authors present an analysis method for service orchestrations based on UML. Orchestrations are modeled using a custom UML profile called UML4SOA, employing activity, structure and class diagrams. The intended behavior of a service is described using UML protocol state machines. The proposed approach uses the notion of model refinement to check whether an orchestration conforms to the behavior protocol. If refinement leads to violations, the analysis results in a graphical output that displays the reason for non-conformance.

Bibliography

[BDF09] Marco Brambilla, Matteo Dosmi, and Piero Fraternali. Model-driven engineering of service orchestrations. In *SERVICES I*, pages 562–569. IEEE Computer Society, 2009.

[BGST10] Laura Bocchi, Roberto Guanciale, Daniele Strollo, and Emilio Tuosto. Bpmn modelling of services with dynamically reconfigurable transactions. In Paul P. Maglio, Mathias Weske, Jian Yang, and Marcelo Fantinato, editors, *ICSOC*, volume 6470 of *Lecture Notes in Computer Science*, pages 396–410, 2010.

[BP09] George Baryannis and Dimitris Plexousakis. The frame problem in web service specifications. In *Proceedings of the 2009 ICSE Workshop on Principles of Engineering Service*

Oriented Systems, PESOS '09, pages 9–12, Washington, DC, USA, 2009. IEEE Computer Society.

[DST+10] Florian Daniel, Stefano Soi, Stefano Tranquillini, Fabio Casati, Chang Heng, and Li Yan. From people to services to ui: Distributed orchestration of user interfaces. In Richard Hull, Jan Mendling, and Stefan Tai, editors, *BPM*, volume 6336 of *Lecture Notes in Computer Science*, pages 310–326. Springer, 2010.

[GRGH07] Walid Gaaloul, Mohsen Rouached, Claude Godart, and Manfred Hauswirth. Verifying composite service transactional behavior using event calculus. In Robert Meersman and Zahir Tari, editors, *OTM Conferences (1)*, volume 4803 of *Lecture Notes in Computer Science*, pages 353–370. Springer, 2007.

[GV07] Stijn Goedertier and Jan Vanthienen. A vocabulary and execution model for declarative service orchestration. In Arthur H. M. ter Hofstede, Boualem Benatallah, and Hye-Young Paik, editors, *Business Process Management Workshops*, volume 4928 of *Lecture Notes in Computer Science*, pages 496–501. Springer, 2007.

[ITCD10] Dragan Ivanovic, Martin Treiber, Manuel Carro, and Schahram Dustdar. Building dynamic models of service compositions with simulation of provision resources. In Jeffrey Parsons, Motoshi Saeki, Peretz Shoval, Carson C. Woo, and Yair Wand, editors, *ER*, volume 6412 of *Lecture Notes in Computer Science*, pages 288–301. Springer, 2010.

[MGWD09] Milan Milanovic, Dragan Gasevic, Gerd Wagner, and Vladan Devedzic. Modeling service orchestrations with a rule-enhanced business process language. In Patrick Martin, Anatol W. Kark, and Darlene A. Stewart, editors, *CASCON*, pages 70–85. ACM, 2009.

[MSK08] Philip Mayer, Andreas Schroeder, and Nora Koch. A model-driven approach to service orchestration. In *IEEE SCC (2)*, pages 533–536, 2008.

[Pin11] Leandro Sales Pinto. A declarative approach to enable flexible and dynamic service compositions. In Richard N. Taylor, Harald Gall, and Nenad Medvidovic, editors, *ICSE*, pages 1130–1131. ACM, 2011.

[SM08] Andreas Schroeder and Philip Mayer. Verifying interaction protocol compliance of service orchestrations. In *ICSOC*, pages 545–550, 2008.

[WWD08] Yong Wang, Li Wang, and Guiping Dai. A web service orchestration model based on concurrent transaction logic. In *Proceedings of the 2008 Seventh International Conference on Grid and Cooperative Computing*, pages 475–482, Washington, DC, USA, 2008. IEEE Computer Society.

[ZD08] Man Zhang and Zhenhua Duan. From business process models to web services orchestration: The case of uml 2.0 activity diagram to bpel. In *ICSOC*, pages 505–510, 2008.

2.20 Modeling and Analysis of Service Choreographies

Involved Workpackages: WP-JRA-2.2

Service choreography is a form of service composition that focuses on the message-based interactions among multiple partner services called *participants*. The behavior that is mandated by the choreography to one of its participants is called *role*. The actual behaviors of the participants in a choreography, e.g. a BPEL process implementing one participant's business logic, are called *participant implementations*.

Paradigms of Service Choreography Modeling

Since the state-of-the-art presented in [PO-08] (UOC, UPM, USTUTT, TUW), a lot of research has been devoted on modeling languages for service choreographies. Similarly to the case of programming languages, modeling languages for service choreographies can be classified in paradigms, i.e. "styles" of modeling such as interaction and interconnection [DKB08]. In this section we classify the novel contributions on modeling choreographies according to their paradigm.

Interaction Choreographies The Business Process Model and Notation v2.0 (BPMN v2.0) specification [OMG09] provides a meta-model and a graphical notation for modeling interaction choreographies. Using workflow-like constructs such as gateways, sequence flows and events that are shared with the orchestration-part of the specifications, a choreography is modeled as a sequence of tasks, each representing a one-way or request-reply interaction between two participants.

BPEL^{gold} [KEvL^{+}11] is a BPEL extension that supports the modeling interaction choreographies by replacing WSDL-specific constructs like *partnerLinks*, *invoke*, *request* and *reply* with more abstract one such as *participantSets* and *interactions*.

Interaction modeling of choreographies has also been investigated on the basis of well-established formalisms. For example, [DW07] proposes a family of colored Petri-Nets for modeling interaction choreographies, in which each transition in the Petri-Net represents an interaction between two participants. Alternative approaches based on Petri-Nets are [DLC^{+}07, HW07, MDY07, VCDM09, CUSRE11]. Message sequence chart grammars are used in [BH10] to describe a choreography specification. The authors introduce dynamic communicating automata as a formal grounding which extend finite-state machines by adding support for dynamic processes creation and asynchronous communication between processes. A similar approach based on Collaboration Diagrams is discussed in [BF08a]. Additionally, interaction choreographies have been investigated using process algebras [CZ08, YZCQ07, CHY07, YCP^{+}08, BS08, Car09], Lightweight Coordination Calculus [BB09] and Signal Calculus [CFGS10].

Interconnection Choreographies BPMN v1.x has been one of the first languages to be used to model interconnection choreographies. However, the authors [DP07] claimed it had serious limitations, such as the impossibility to specify correlations between messages and particular instances of the participants that enact the choreography, as well as not providing constructs for specifying "multiple" participants, i.e. participants that do not directly represent an entity, but instead a selection among a number of alternative concrete participants. This work presents additional constructs that increase the expressiveness of BPMN v1.x in order to solve its shortcomings.

Many of the extensions of BPMN v1.x proposed in [DP07] for modeling interconnection choreographies have find their way in BPMN v2.0. Additionally to the possibly of specifying choreographies using the interaction paradigm, the BPMN v2.0 specification [OMG09] provides a meta-model and a graphical notation for modeling interconnection choreographies called Collaboration diagrams. These diagrams consist of pools, each representing one of the involved participants. A pool specifies the control-flow between the activities that are performed by the represented participant. Similarly to the case of BPMN v2.0 interaction choreographies, the control flow of collaborations is specified using gateways and events. Activities within separate pools exchange information through messages dispatched via message flows.

BPMN v1.x is not the only "industrial" language that has been extended to model interconnection choreographies. In [DKLW07] BPEL4Chor is introduced as an extension to BPEL for modeling interconnection choreographies. A BPEL4Chor choreography model consists of the specification of the roles of a number of participants, each represented as a separate BPEL processes. The processes of the various participants are "wired" together by message links that represent the message exchanges. The technical implementation (WSDL interfaces, etc.), called *grounding*, is separated from the choreography model and specified in a separate document to allow for alternative technical implementations. Similarly to the case of BPEL^{gold}, the result of the separation of grounding and business logic in the choreography entails that the BPEL process models comprised in the interaction choreography contain no longer WSDL-specific artifacts.

The different design phases of BPMN v2.0 interconnection choreography modeling are discussed in [DKL+08]. Additionally, this work proposes a mapping from BPMN v2.0 interconnection choreographies to BPEL4Chor to use the latter as an interchange format.

An *operating guideline* for a service defines the set of all partner services the can interact with this service. An approach to determine the operating guideline for a service more efficiently than to the computation algorithms presented in [LMW07]. The same authors propose in [LW11] a more compact representation of operating guidelines.

The research on bridging between choreography modeling paradigms, i.e. how to map choreographies specified with one paradigm to another, has exclusively focused on the relationships between interconnection and interaction choreographies. For example, [KL10] investigates how BPMN v1.x interconnection models that assume synchronous communication among the participants can be mapped to an iBPMN [DKL+08] interaction model by using interaction Petri nets as intermediary format.

Artifact-centric Choreographies Artifact-centric choreographies [HNN09, LW10] are an emerging choreography modeling paradigm that focuses on data objects (e.g. documents), the states of which are modified by the message interactions that occur among the participants. In contrast to [HNN09], where artifact-hubs are described as a centralized, "hub-and-spoke" architecture, in [LW10] the artifacts are distributed on several agents. Another approach to model UML-based interaction choreographies that are driven by the artifact or business entity state is proposed in [HZLS07]. Artifact-centric choreographies built on top of Active XML [ABM08] are proposed in [HB10]. In a nutshell, service interfaces are wrapped around existing Active XML objects, thus enabling remote calls between different participants that are modeled as agents.

Declarative Choreographies Declarative choreographies are specified as a set of message interactions that must occur among the participants, and a set of rules that constraint the order in which these message interactions take place. The declarative workflow language DecSerFlow is re-purposed for modeling declarative choreographies in [MPvdA+10]. The verification of declarative DecSerFlow choreographies, e.g. in terms of consistency checking, is achieved by mapping them to LTL predicates

Another approach to declarative choreographies is presented in [CMM+10, CMMT11]. In this case, however, instead of using LTL as formal underpinnings, the authors build on the agent-oriented framework CIFF.

Evaluation of Choreography Specification Languages Despite the large amount of notations and formalisms for specifying interaction choreographies that have been presented over the years, there is still limited understanding of what is a *good* choreography specification language.

[WRSC08] defines requirements and properties that should be met by a choreography language. The requirements have been elicited on the basis of a case study analyzing the SAP SOA infrastructure.

So far, most of the evaluations of choreography specification languages focus on their *expressiveness*, i.e. the richness of the constructs that are provided to the choreography designers. For example, in [DOZ06] the authors relate and compare Web Services Choreography Description Language (WS-CDL) with π-calculus and investigate to which extent WS-CDL supports workflow patterns [vdAtHKB03] and service interaction patterns [BDtH05]. Moreover, an overview is given which patterns are supported by the WS-CDL predecessor Web Service Choreography Interface (WSCI) [AAF+02] and BPEL. A similar work is provided in [KLW11], where the authors compare the expressiveness of BPMN v2.0, both Choreography and Collaboration diagrams, BPEL4Chor and BPEL^{gold}.

Analysis of Interaction Choreographies

In the past few years, most of the efforts on the analysis of service choreographies have been devoted to the interaction paradigm. The amount of research results that have been produced in the past four years is such that, in this deliverable, we must restrict ourselves to two key problems, namely conformance and realizability. An overview of other problems connected to service choreographies can be found in [SBFZ07].

Choreography Conformance The problem called *conformance* is the verification of the compliance in terms of messaging behavior of the participant implementations with respect to the roles mandated to them by a choreography. Since service orchestration is a way to specify one service's behavior, it is natural that the conformance of service orchestrations with respect to (interaction) service choreographies has attracted a large amount of research in the past few years.

The conformance of services whose behavior is described in π-calculus to an interaction choreography specification whose properties are formulated in spatial logics can be verified with the approach presented in [CSV07].

In the already mentioned [CSV07], π-calculus is used to model interaction choreography specifications. Assuming the participants to be implemented as Web services whose behavior is expressed in terms of spacial logics, it is possible to check the conformance between them and the interaction choreography. The technique is implemented in a tool that performs the verification automatically.

An approach to check the conformance of a black-box Web service to a interaction choreography specified as a Labeled Transition System (LTS) is described in [PASB11]. The implementation of the Web services realizing the participants is inferred iteratively by a learning algorithm. In each iteration, the learning algorithm creates a Mealy machine that represents a participant implementation. The Mealy machine is then transformed to a LTS and is checked for conformance against the LTS representing the choreography. Inconsistencies that result in lack of conformance originate either in an invalid Mealy machine projected as a role, or in an incorrect implementation of that role by the Web service. If the problem originates in the Mealy machine, a new one is created during the iteration by using the counter-example that causes the inconsistency. Then, the new LTS of the Web service is checked again against the choreography specification until (1) no inconsistencies can be found anymore or (2) it can be proved that the Web service implementation of a participant does not conform with the choreography.

An approach to compliance of orchestrations with choreographies modeled as LTSs is presented in [BZ07] (assuming synchronous communication among the participants), and [BZ08] (assuming asynchronous messaging using message queues).

[GSW11] investigates the conformance between ebBP-ST choreographies specified using the ebXML Business Process Specification Schema and participant implemented as BPEL process models. The authors propose to transform the choreography and the process model to the Calculus of Communicating Systems (CCS) algebra to perform bi-simulation equivalence model checking. This approach is limited to two communicating partners, i.e., to two process models.

The work in [BZ07, BB11] discusses the undecidability of the conformance in the case of asynchronous communication. To circumvent this issue the authors describe an approach to transform a class of asynchronous interactions to synchronous interactions.

Conformance is usually a statically-verified property. However, is it desirable to be able to keep orchestrations conforming to choreographies during their evolution. A conceptual approach to automatically adapt participants implemented as orchestrations to an evolving choreography is described in [HAD10]. The adaptation is done via change scripts containing rules that specify how the orchestrations are changed.

In [ET08], the authors present a technique to model choreographies with temporal constraints and they provide an algorithm for verifying if such temporal constraints are satisfied during the enactment of the choreography.

Choreography Realizability An interaction choreography is realizable is it is possible to derive from it, e.g. using projection, participant implementations that, once composed, can enact all and only the traces specified by the original choreography. In this sense, realizability is the opposite of conformance. On one hand, conformance checks that, given participant implementations, their composition is behaviorally equivalent to the original choreography. Realizability, on the other hand, is the problem of checking if such participants implementations can be derived from the choreography.

There is an entire spectrum of realizability notions that depend on (1) the notations adopted to specify the choreographies,(2) the notion of behavioral equivalence adopted and (3) the messaging model assumed (synchronous or asynchrnous with finite/infinite queues, etc.) [KP06]. In this section are surveyed some of the most recent research results concerning realizability.

[DW07] presents algorithms to check realizability of interaction choreographies modeled as colored Petri-Nets.

[MCvdHP08] (Tilburg, UPM) proposes a formal framework to describing multi-party choreography described using Deterministic Finite Automata with temporal properties and to verify their soundness (participant and time soundness), which are aspects of realizability.

In [BF08b] a formalization of the realizability problem for UML communication diagrams is given and realizability conditions are defined. In [SB09], the authors propose an approach to check realizability of a choreography that is described by an UML communication diagram. To perform the realizability check the choreography is encoded into the LOTOS process algebra and by using projection techniques a LOTUS process is generated for each peer. By using a LOTUS tool-set named CADP the behavioral equivalence between the generated peers and the choreography is checked in order to verify the realizability of the choreography. Thereby, the realizability checks can be performed for synchronous and bounded asynchronous communication scenarios.

Different realizability dimensions for interaction choreographies are also defined in [Dec09]. Role projections are used to check if the realizability dimensions were met by the choreography.

The authors of [LW09] present different realizability classes. In contrast to the other approaches, they suggest a technique to verify the realizability of a choreography by relating it to the controllability problem, i.e. if there exists a compatible partner process for the choreography.

In [BH10] the authors describe how the realizability of the message sequence chart grammars can be verified by using a dynamic communicating automata.

How to check the realizability of propositional state-chart representations is described in [DDC09]. Moreover, the authors propose an approach to describe state reachability within the propositional state-chart representations to verify different properties such as liveness, safety etc.

In many of the approaches listed above, the realizability analysis is performed by trying to "create" participants implementation that, once composed, are behaviorally equivalent to the original choreography. This, of course, implies that such participant implementations *can* be created. In most cases, the participant implementations are created using a process known as *projection*, i.e. the "extraction of the point of view of one participant" on the overall choreography specification. Examples of projection techniques can be found in e.g. [YZCQ07, Car09, KB10].

Bibliography

[AAF+02] Assaf Arkin, Sid Askary, Scott Fordin, Wolfgang Jekeli, Kohsuke Kawaguchi, David Orchard, Stefano Pogliani, Karsten Riemer, Susan Struble, Pal T. Nagy, Ivana Trickovic, and Sinisa Zimek. Web service choreography interface (WSCI) 1.0. Technical Report NOTE-wsci10-20020808, 2002.

[ABM08] Serge Abiteboul, Omar Benjelloun, and Tova Milo. The active xml project: an overview. *VLDB J.*, 17(5):1019–1040, 2008.

[BB09] Paolo Besana and Adam Barker. An executable calculus for service choreography. In *OTM Conferences (1)*, pages 373–380, 2009.

[BB11] Samik Basu and Tevfik Bultan. Choreography conformance via synchronizability. In Sadagopan Srinivasan, Krithi Ramamritham, Arun Kumar, M. P. Ravindra, Elisa Bertino, and Ravi Kumar, editors, *WWW*, pages 795–804. ACM, 2011.

[BDtH05] Alistair P. Barros, Marlon Dumas, and Arthur H. M. ter Hofstede. Service interaction patterns. In *Business Process Management*, pages 302–318, 2005.

[BF08a] Tevfik Bultan and Xiang Fu. Choreography modeling and analysis with collaboration diagrams. *IEEE Data Eng. Bull.*, 31(3):27–30, 2008.

[BF08b] Tevfik Bultan and Xiang Fu. Specification of realizable service conversations using collaboration diagrams. *Service Oriented Computing and Applications*, 2(1):27–39, 2008.

[BH10] Benedikt Bollig and Loïc Hélouët. Realizability of dynamic msc languages. In *CSR*, pages 48–59, 2010.

[BS08] A. K. Bhattacharjee and R. K. Shyamasundar. Scriptorc: A specification language for web service choreography. In *APSCC*, pages 1089–1096. IEEE, 2008.

[BZ07] Mario Bravetti and Gianluigi Zavattaro. Towards a unifying theory for choreography conformance and contract compliance. In Markus Lumpe and Wim Vanderperren, editors, *Software Composition*, volume 4829 of *Lecture Notes in Computer Science*, pages 34–50. Springer, 2007.

[BZ08] Mario Bravetti and Gianluigi Zavattaro. Contract compliance and choreography conformance in the presence of message queues. In Roberto Bruni and Karsten Wolf, editors, *WS-FM*, volume 5387 of *Lecture Notes in Computer Science*, pages 37–54. Springer, 2008.

[Car09] Marco Carbone. Session-based choreography with exceptions. *Electr. Notes Theor. Comput. Sci.*, 241:35–55, 2009.

[CFGS10] Vincenzo Ciancia, Gian Luigi Ferrari, Roberto Guanciale, and Daniele Strollo. Event based choreography. *Sci. Comput. Program.*, 75(10):848–878, 2010.

[CHY07] Marco Carbone, Kohei Honda, and Nobuko Yoshida. A calculus of global interaction based on session types. *Electr. Notes Theor. Comput. Sci.*, 171(3):127–151, 2007.

[CMM+10] Federico Chesani, Paola Mello, Marco Montali, Sergio Storari, and Paolo Torroni. On the integration of declarative choreographies and commitment-based agent societies into the sciff logic programming framework. *Multiagent and Grid Systems*, 6(2):165–190, 2010.

[CMMT11] Federico Chesani, Paola Mello, Marco Montali, and Paolo Torroni. Modeling and verifying business processes and choreographies through the abductive proof procedure sciff and its extensions. *Intelligenza Artificiale*, 5(1):101–105, 2011.

[CSV07] Luis Caires, David T. Sousa, and Hugo T. Vieira. Checking choreography conformance using SLMC. Technical Report 2007-TR-03, Universidade Nova de Lisboa, 2007.

[CUSRE11] Enrique Caliz, Karthikeyan Umapathy, Arturo J. Sánchez-Ruíz, and Sherif A. Elfayoumy. Analyzing web service choreography specifications using colored petri nets. In Hemant K. Jain, Atish P. Sinha, and Padmal Vitharana, editors, *DESRIST*, volume 6629 of *Lecture Notes in Computer Science*, pages 412–426. Springer, 2011.

[CZ08] Cai Chao and Qiu Zongyan. An approach to check choreography with channel passing in ws-cdl. In *Proceedings of the 2008 IEEE International Conference on Web Services*, pages 700–707, Washington, DC, USA, 2008. IEEE Computer Society.

[DDC09] Hywel R. Dunn-Davies and Jim Cunningham. Verifying realizability and reachability in recursive interaction protocol specifications. In Carles Sierra, Cristiano Castelfranchi, Keith S. Decker, and Jaime Simão Sichman, editors, *AAMAS (2)*, pages 1233–1234. IFAAMAS, 2009.

[Dec09] Gero Decker. Realizability of interaction models. In Oliver Kopp and Niels Lohmann, editors, *ZEUS*, volume 438 of *CEUR Workshop Proceedings*, pages 55–60. CEUR-WS.org, 2009.

[DKB08] Gero Decker, Oliver Kopp, and Alistair P. Barros. An introduction to service choreographies (servicechoreographien - eine einführung). *it - Information Technology*, 50(2):122–127, 2008.

[DKL+08] Gero Decker, Oliver Kopp, Frank Leymann, Kerstin Pfitzner, and Mathias Weske. Modeling service choreographies using bpmn and bpel4chor. In *CAiSE*, pages 79–93, 2008.

[DKLW07] Gero Decker, Oliver Kopp, Frank Leymann, and Mathias Weske. Bpel4chor: Extending bpel for modeling choreographies. In *ICWS*, pages 296–303. IEEE Computer Society, 2007.

[DLC+07] Xinguo Deng, Ziyu Lin, Weiqing Chen, Ruliang Xiao, Lina Fang, and Ling Li. Modeling web service choreography and orchestration with colored petri nets. In Wenying Feng and Feng Gao, editors, *SNPD (2)*, pages 838–843. IEEE Computer Society, 2007.

[DOZ06] Gero Decker, Hagen Overdick, and Johannes Maria Zaha. On the suitability of ws-cdl for choreography modeling. In Mathias Weske and Markus Nüttgens, editors, *EMISA*, volume 95 of *LNI*, pages 21–33. GI, 2006.

[DP07] Gero Decker and Frank Puhlmann. Extending bpmn for modeling complex choreographies. In *OTM Conferences (1)*, pages 24–40, 2007.

[DW07] Gero Decker and Mathias Weske. Local enforceability in interaction petri nets. In *BPM*, pages 305–319, 2007.

[ET08] Johann Eder and Amirreza Tahamtan. Temporal conformance of federated choreographies. In Sourav S. Bhowmick, Josef Küng, and Roland Wagner, editors, *DEXA*, volume 5181 of *Lecture Notes in Computer Science*, pages 668–675. Springer, 2008.

[GSW11] Matthias Geiger, Andreas Schönberger, and Guido Wirtz. A proposal for checking the conformance of ebbp-st choreographies and ws-bpel orchestrations. In Daniel Eichhorn, Agnes Koschmider, and Huayu Zhang, editors, *ZEUS*, volume 705 of *CEUR Workshop Proceedings*, pages 24–25. CEUR-WS.org, 2011.

[HAD10] Marcel Hiel, Huib Aldewereld, and Frank Dignum. Ensuring conformance in an evolving choreography. In *SOCA*, pages 1–4, 2010.

[HB10] Loïc Hélouët and Albert Benveniste. Document based modeling of web services choreographies using active xml. In *ICWS*, pages 291–298, 2010.

[HNN09] Richard Hull, Nanjangud C. Narendra, and Anil Nigam. Facilitating workflow interoperation using artifact-centric hubs. In *ICSOC/ServiceWave*, pages 1–18, 2009.

[HW07] Yu Huang and Hanpin Wang. A petri net semantics for web service choreography. In Yookun Cho, Roger L. Wainwright, Hisham Haddad, Sung Y. Shin, and Yong Wan Koo, editors, *SAC*, pages 1689–1690. ACM, 2007.

[HZLS07] Christian Huemer, Marco Zapletal, Philipp Liegl, and Rainer Schuster. Modeling business entity state centric choreographies. In *CEC/EEE*, pages 393–400, 2007.

[KB10] Surya Bahadur Kathayat and Rolv Bræk. From flow-global choreography to component types. In Frank Alexander Kraemer and Peter Herrmann, editors, *SAM*, volume 6598 of *Lecture Notes in Computer Science*, pages 36–55. Springer, 2010.

[KEvL+11] Oliver Kopp, Lasse Engler, Tammo van Lessen, Frank Leymann, and Jörg Nitzsche. Interaction Choreography Models in BPEL: Choreographies on the Enterprise Service Bus. In A. Fleischmann, W. Schmidt, D. Seese, and R. Singer, editors, *Subject-Orientation as Enabler for the Next Generation of BPM Tools and Methods - Second International Conference S-BPM ONE 2010*, volume 138 of *Communications in Computer and Information Science*. Springer, Januar 2011.

[KL10] Oliver Kopp, Frank Leymann, and Fei Wu 0002. Mapping interconnection choreography models to interaction choreography models. In *ZEUS*, pages 81–88, 2010.

[KLW11] Oliver Kopp, Frank Leymann, and Sebastian Wagner. Modeling choreographies: Bpmn 2.0 versus bpel-based approaches. In Markus Nüttgens, Oliver Thomas, and Barbara Weber, editors, *EMISA*, volume 190 of *LNI*, pages 225–230. GI, 2011.

[KP06] Raman Kazhamiakin and Marco Pistore. Analysis of realizability conditions for web service choreographies. In Elie Najm, Jean-François Pradat-Peyre, and Véronique Donzeau-Gouge, editors, *FORTE*, volume 4229 of *Lecture Notes in Computer Science*, pages 61–76. Springer, 2006.

[KY07] Jetty Kleijn and Alexandre Yakovlev, editors. *Petri Nets and Other Models of Concurrency - ICATPN 2007, 28th International Conference on Applications and Theory of Petri Nets and Other Models of Concurrency, ICATPN 2007, Siedlce, Poland, June 25-29, 2007, Proceedings*, volume 4546 of *Lecture Notes in Computer Science*. Springer, 2007.

[LMW07] Niels Lohmann, Peter Massuthe, and Karsten Wolf. Operating guidelines for finite-state services. In Kleijn and Yakovlev [KY07], pages 321–341.

[LW09] Niels Lohmann and Karsten Wolf. Realizability is controllability. In Cosimo Laneve and Jianwen Su, editors, *WS-FM*, volume 6194 of *Lecture Notes in Computer Science*, pages 110–127. Springer, 2009.

[LW10] Niels Lohmann and Karsten Wolf. Artifact-centric choreographies. In *ICSOC*, pages 32–46, 2010.

[LW11] Niels Lohmann and Karsten Wolf. Compact representations and efficient algorithms for operating guidelines. *Fundam. Inform.*, 108(1-2):43–62, 2011.

[MCvdHP08] Michele Mancioppi, Manuel Carro, Willem-Jan van den Heuvel, and Mike P. Papazoglou. Sound multi-party business protocols for service networks. In *ICSOC*, pages 302–316, 2008.

[MDY07] Peng Men, Zhenhua Duan, and Bin Yu. Utilizing fuzzy petri net for choreography based semantic web services discovery. In Kleijn and Yakovlev [KY07], pages 362–380.

[MPvdA+10] Marco Montali, Maja Pesic, Wil M. P. van der Aalst, Federico Chesani, Paola Mello, and Sergio Storari. Declarative specification and verification of service choreographiess. *TWEB*, 4(1), 2010.

[OMG09] OMG. Business Process Model and Notation Version 2.0. OMG Specification, Beta 1, Object Management Group, August 2009.

[PASB11] Warawoot Pacharoen, Toshiaki Aoki, Athasit Surarerks, and Pattarasinee Bhattarakosol. Conformance verification between web service choreography and implementation using learning and model checking. In *ICWS*, pages 722–723, 2011.

[PO-08] Overview of the state of the art in composition and coordination of services. Deliverable, S-Cube Consortium, 2008.

[SB09] Gwen Salaün and Tevfik Bultan. Realizability of choreographies using process algebra encodings. In Michael Leuschel and Heike Wehrheim, editors, *IFM*, volume 5423 of *Lecture Notes in Computer Science*, pages 167–182. Springer, 2009.

[SBFZ07] Jianwen Su, Tevfik Bultan, Xiang Fu, and Xiangpeng Zhao. Towards a theory of web service choreographies. In Marlon Dumas and Reiko Heckel, editors, *WS-FM*, volume 4937 of *Lecture Notes in Computer Science*, pages 1–16. Springer, 2007.

[VCDM09] Valentin Valero, María-Emilia Cambronero, Gregorio Díaz, and Hermenegilda Macià. A petri net approach for the design and analysis of web services choreographies. *J. Log. Algebr. Program.*, 78(5):359–380, 2009.

[vdAtHKB03] Wil M. P. van der Aalst, Arthur H. M. ter Hofstede, Bartek Kiepuszewski, and Alistair P. Barros. Workflow patterns. *Distributed and Parallel Databases*, 14(1):5–51, 2003.

[WRSC08] Sebastian Wieczorek, Andreas Roth, Alin Stefanescu, and Anis Charfi. Precise steps for choreography modeling for soa validation and verification. In *SOSE*, pages 148–153, 2008.

[YCP+08] Hongli Yang, Chao Cai, Liyang Peng, Xiangpeng Zhao, and Zongyan Qiu. Reasoning about channel passing in choreography. In *TASE*, pages 135–142. IEEE Computer Society, 2008.

[YZCQ07] Hongli Yang, Xiangpeng Zhao, Chao Cai, and Zongyan Qiu. Exploring the connection of choreography and orchestration with exception handling and finalization/compensation. In John Derrick and Jüri Vain, editors, *FORTE*, volume 4574 of *Lecture Notes in Computer Science*, pages 81–96. Springer, 2007.

2.21 Fragmentation of Service Compositions

Involved Workpackages: WP-JRA-2.2

Within the S-Cube workpackage WP-JRA-2.2 (Adaptable Coordinated Service Compositions), deliverable CD-JRA-2.2.3 [CD-09] (UniHH, UPM, USTUTT) was specifically dedicated to algorithms and techniques for service composition splitting (fragmentation) and merging. Aside from presenting the contributions by the S-Cube partners to the subject, this deliverable also surveyed the terminology and proposed operational definitions for the key notions of process fragments, fragmentation and merging operators. The deliverables that followed in the series, CD-JRA-2.2.4 [CD-10] (TUW, UCBL, UMAN, UniHH, UOC, UPM, USTUTT, VUA) – on models, mechanisms and protocols for coordinated service compositions – and CD-JRA-2.2.5 [CD-11] (TUW, UniHH, UPM, USTUTT, UniDuE, FBK) – on mechanisms for derivation of QoS and SLA specifications – also included contributions related to fragmentation and merging.

A comprehensive attempt at surveying process fragmentation techniques, goals, and criteria appears in the work by Mancioppi et al. [MDKL11] (USTUTT). This work is an evolution of the technical report presented by the authors in the S-Cube deliverable CD-JRA-2.2.3 [CD-09] (UniHH, UPM, USTUTT), where it was used as a basis to define and classify different fragmentation approaches. The survey was motivated by the lack of a common terminology and classification criteria, and by the large discrepancy in the characteristics that are covered when presenting novel fragmentation techniques. This issue is addressed by investigating classification criteria for process fragmentation techniques based on the "seven Ws", namely Why, What, When, Where, Who, Which, and hoW. The presented classification criteria are applied to some of the process fragmentation approaches available in the literature. In addition to enabling the classification of fragmentation techniques, the classification criteria here presented form a "check-list" for authors of future works in the field of process fragmentation.

Gehlert et al. [GDK09] (UniDue and USTUTT) have conducted a literature survey and explored how merging techniques can be applied to use case scenarios and service composition models, thus narrowing the gap between the fields of requirements engineering and business process modeling, which have evolved mostly separately. The work was motivated by similarities between use case scenarios and business process models, both of which specify steps that need to be taken towards accomplishment of a goal. One approach, dubbed "merge first, transform later" suggest starting from a collection of use-case scenarios, merging them and then transforming the merged scenario into a service composition. The other approach, "transform first, merge later" suggests converting each use-case scenario into a separate composition, and merging the services together later.

Fragmentation in the context of coordinated execution of distributed transactional services was studied in the work by Danylevich et al. [DKL09] (USTUTT). It focused on cases of acyclic compositions made of transactional services, which are described with properties such as execution time, concurrency, invocation, recovery, and time violation costs. Data and control dependencies within such a composition are represented in the form of a graph that features simple AND-type splits and joins. Absence of cycles in the dependency graph guarantees deadlock-free execution. In that case, the global transaction, given as a single composition, can be split into fragments that are executed in a distributed manner. Each fragment corresponds to a strata in the dependency graph. Several stratification schemes may exist for a given composition, and the choice between them is made on the basis of optimality in terms of performance and cost.

Logical, rather than physical fragmentation of business processes was studied in the work by Zaplata et al. [ZKML09] (UniHH). Instead of sending different parts of a process to different execution sites, in logical fragmentation the entire process is migrated from one site to another, and each site is in charge of executing a certain portion of it. Therefore, the fragmentation applies to responsibilities of the participants (execution engines) at run-time, which may be useful in situations where the participants enter and leave the system dynamically, or QoS parameters unexpectedly fluctuate. The proposed approach relies on the cryptographic infrastructure for encrypting (masking) parts of process definitions and data that are available to authorized participants only. The cryptographic infrastructure is also used to verify authenticity of process descriptions and the accompanying migration meta-data.

In their subsequent publication [ZHKL10] (UniHH), the authors have expanded their work by proposing a process migration meta-model that includes recording of process states, activity execution lifecycle events, as well as of cryptographic resources used for enforcing migration security policies. The meta-model also takes into account different possibilities for assigning (logical) process fragments to different participants at run-time, ranging from fixed roles, over custom algorithm and dependency on process-level data, to QoS and context sensitive choice. Since the process migration data needs to be shared and kept up to date between the participants, the work specifically treats the problem of doing so in parallel flows. The authors also propose an overall architecture for runtime migration support, which includes, alongside the migration-enabled participating execution engines, a migration manager and a security manager who perform the tasks of fragment-to-participant assignment and process content access protection, respectively.

The work by Bouhini et al. [BLMB10] (Manchester) discusses the discovery and selection of fragments with respect to a given composition goal to be achieved. The discovery is based on semantic characterization of fragments in Description Logic, and is performed by applying different levels of semantic matchmaking between fragments and goals descriptions. On top of the discovery process, the approach introduces a multi-agent system that selects the most relevant fragments based on non-functional criteria (such as their execution price), cohesion and overlap (inferred from the fragmentation technique) and finally the quality of semantic matchmaking between fragments and goals. The selection process is based on a agents-oriented negotiation where each agent is offers a semantically coherent list of fragments. To reduce the number of candidate fragments for selection, the approach aims at minimizing their overlap and, at the same time, at maximizing the number of goals they contribute to.

Fragmentation based on characteristics of data received, sent and processed by a service workflow was discussed in the work by Ivanović, Carro and Hermenegildo [ICH10] (UPM). Their approach is based on the notion of sharing between workflow activities, based on how these activities handle data and other resources (that represent messages, workflow-level variables, data sources, etc.), as well as external participants. The workflow is translated into a form of logic program (a set of Horn clauses [Llo87]), where data items, resources and activities are represented as logical variables, and subject to sharing analysis [MH92] to detect the sharing patterns between the variables. Sharing analysis in presence of complex control structures (such as loops) is involved, and it is taken care of by using program analysis-based techniques relying on abstract interpretation. The inputs to the analysis are given in the form of a lattice that describes workflow inputs, internal resources and partners, typically in terms of access rights or confidentiality restrictions. The results from the analysis are also a lattice that can be used directly to determine which activities belong to which fragments, by comparing them to the given policy levels. In effect, the policy level for an activity is determined on the basis of its input data flow.

In a follow-up publication [ICH11] (UPM), the authors extended the sharing-based approach to allow workflow designers to work with simple object-attribute Boolean matrices and avoid setting up and interpreting lattices directly. That was achieved by employing notions and techniques from Formal Concept Analysis (FCA) [GSW05] that establish the correspondence between object-attribute matrices and the underlying lattices. The users specify the relevant properties of workflow inputs, and the final result assigns those attributes to activities and intermediate data items in the workflow based on sharing. The fragments are then formed by grouping together activities whose attributes satisfy some (propositional) logic formula.

Naturally, advances in process fragmentation techniques have been booked also outside S-Cube. Process fragmentation was addressed on an abstract level in the work by Tan and Fan [TF07], from the viewpoint of distributed process enactment. The goal of the approach was to take a process workflow described in a Petri Net notation, and execute it in a way that maximizes the distribution of process activities among nodes of a grid or cluster of process execution (enactment) engines. That included developing a rigorous criteria for deciding which transitions and places in the given Petri Net can be detached into a separate fragment at each process execution step, so that the the original Petri net semantics is respected by the behavior of the fragmented process. The proposed approach is a form of run-time fragmentation that is transparent to the user/designer and is meant to be automatically applied by the nodes of the distributed process execution engine.

Some of the work on process fragmentation was motivated by the problem of information flow control, which is common in scientific computing, as well as in cross-organizational business processes. The work by Yildiz, Fdhila and Godart [YG07, FYG09] concentrates on cases of acyclic workflows where individual activities are essentially calls to external services that are hosted in different business domains. The data items that are passed between the external services have different security or confidentiality levels, and therefore need to be protected from unauthorized lookup in different domains. Their approach splits the original process (which is abstractly described using the BPMN v1.x notation) into fragments that correspond to the business domains, and then derives fragment interfaces and messaging patterns that ensure that the information flow in such distributed setting respects the given privacy/confidentiality constraints.

Another line of work was related to practical applications of fragmentation in real-world service composition languages, most notably BPEL. A technical report by Khalaf [Kha07] aimed at describing the syntactical details of splitting business processes written in a subset of BPEL into fragments that correctly handle data and control dependencies, based on an *a priori* assignment of activities to different fragments. The criteria for the assignment of activities to fragments is left open: the paper deals with how to correctly transform the BPEL process assuming that some fragmentation was defined. However, one possibility, based on the notion of roles of process participants, was presented in an earlier publication on business process decomposition by Khalaf and Leymann [KL06]. A detailed overview of splitting BPEL business processes into fragments while maintaining the original operational semantics is given in Khalaf's PhD thesis [Kha08].

An alternative approach to BPEL process fragmentation was presented in the work by Martin et al. [MWL08]. There, the splitting of a BPEL workflow into fragments and their distributed enactment was motivated by the need to attain the overall business goals of the process described with the workflow, rather than by technical capabilities of the underlying execution infrastructure. The approach starts by deriving a process model in a special Petri Net dialect from the executable BPEL specification, which is then segmented (divided into fragments) at design time. The resulting Petri Net fragments are then executed by separate execution engines. The correctness of the distributed execution with respect to the original BPEL specification is ensured by the soundness of the translation scheme for each BPEL construct into the Petri Net notation.

An approach to design-time specification of BPEL process fragments as units for process building and reuse was discussed in the work by Ma and Leymann [ML09]. The idea behind the approach is that design-time reusability can be significantly boosted by going below the level of entire BPEL processes. The BPEL fragments need to support the normal repertoire of BPEL atomic and complex constructs, with flexible choice of granularity (from activity to scope to process level). Fragments should additionally include information on the surrounding process context which is necessary for them to function (such as variables and partner services), and define a number of variability points that can be used for gluing together different fragments. For merging fragments into a process the criteria relies on unbroken control flow. The approach includes the proposal of syntactic and semantic extensions to BPEL for fragment definitions.

Fragmentation from the point of view of local knowledge of different fragment designers was discussed in the work by Eberle et al. [EUL09]. The motivating assumption was that process knowledge is distributed and not completely known just by one person (the designer). This approach defines fragment as a building block for constructing service compositions (either at design- or at run-time) based on a formal representation of fragment modeler's local knowledge, which is used for reasoning about possible fragment compositions.

Bibliography

[BLMB10] Chahrazed Bouhini, Freddy Lécué, Nikolay Mehandjiev, and Olivier Boissier. Discovery and selection of web services fragments for re-composition. In *SOCA*, pages 1–4. IEEE, 2010.

[CD-09] Algorithms and techniques for splitting and merging service compositions. Deliverable, S-Cube Consortium, 2009.

[CD-10] Models, mechanisms and protocols for coordinated service compositions. Deliverable, S-Cube Consortium, 2010.

[CD-11] Derivation of QoS and SLA specifications. Deliverable, S-Cube Consortium, 2011.

[DKL09] Olha Danylevych, Dimka Karastoyanova, and Frank Leymann. Optimal stratification of transactions. In *ICIW*, pages 493–498, 2009.

[EUL09] Hanna Eberle, Tobias Unger, and Frank Leymann. Process fragments. In Robert Meersman, Tharam S. Dillon, and Pilar Herrero, editors, *OTM Conferences (1)*, volume 5870 of *Lecture Notes in Computer Science*, pages 398–405. Springer, 2009.

[FYG09] Walid Fdhila, Ustun Yildiz, and Claude Godart. A flexible approach for automatic process decentralization using dependency tables. In *ICWS*, pages 847–855, 2009.

[GDK09] Andreas Gehlert, Olha Danylevych, and Dimka Karastoyanova. From requirements to executable processes: A literature study. In Stefanie Rinderle-Ma, Shazia Wasim Sadiq, and Frank Leymann, editors, *Business Process Management Workshops*, volume 43 of *Lecture Notes in Business Information Processing*, pages 42–52. Springer, 2009.

[GSW05] Bernhard Ganter, Gerd Stumme, and Rudolf Wille, editors. *Formal Concept Analysis, Foundations and Applications*, volume 3626 of *Lecture Notes in Computer Science*. Springer, 2005.

[ICH10] Dragan Ivanovic, Manuel Carro, and Manuel V. Hermenegildo. Automatic fragment identification in workflows based on sharing analysis. In *ICSOC*, pages 350–364, 2010.

[ICH11] D. Ivanović, M. Carro, and M. Hermenegildo. Automated attribute inference in complex service workflows based on sharing analysis. In *Proceedings of the 8th IEEE Conference on Services Computing SCC 2011*. IEEE Press, July 2011.

[Kha07] Rania Khalaf. Note on Syntactic Details of Split BPEL-D Business Processes. Technical Report 2007/2, IAAS, U. Stuttgart, July 2007.

[Kha08] Rania Khalaf. *Supporting Business Process Fragmentation While Maintaining Operational Semantics*. PhD thesis, University of Stuttgart, Faculty of Computer Science, Electrical Engineering and Information Technology, March 2008.

[KL06] Rania Khalaf and Frank Leymann. Role-based decomposition of business processes using BPEL. In *ICWS*, pages 770–780. IEEE Computer Society, 2006.

[Llo87] John W. Lloyd. *Foundations of Logic Programming, 2nd Edition*. Springer, 1987.

[MDKL11] M. Mancioppi, O. Danylevych, D. Karastoyanova, and F. Leymann. Towards classification criteria for process fragmentation techniques. In *BDP2011, colocated with BPM'11*, 2011.

[MH92] Kalyan Muthukumar and Manuel V. Hermenegildo. Compile-time derivation of variable dependency using abstract interpretation. *J. Log. Program.*, 13(2&3):315–347, 1992.

[ML09] Zhilei Ma and Frank Leymann. Bpel fragments for modularized reuse in modeling BPEL processes. In *ICNS*, pages 63–68, 2009.

[MWL08] Daniel Martin, Daniel Wutke, and Frank Leymann. A novel approach to decentralized workflow enactment. In *EDOC*, pages 127–136, 2008.

[TF07] Wei Tan and Yushun Fan. Dynamic workflow model fragmentation for distributed execution. *Computers in Industry*, 58(5):381–391, 2007.

[YG07] Ustun Yildiz and Claude Godart. Information flow control with decentralized service compositions. In *ICWS*, pages 9–17, 2007.

[ZHKL10] Sonja Zaplata, Kristof Hamann, Kristian Kottke, and Winfried Lamersdorf. Flexible execution of distributed business processes based on process instance migration. *Journal of Systems Integration (JSI)*, 1(3):3–16, 7 2010.

[ZKML09] Sonja Zaplata, Kristian Kottke, Matthias Meiners, and Winfried Lamersdorf. Towards runtime migration of ws-bpel processes. In *ICSOC/ServiceWave Workshops*, pages 477–487, 2009.

2.22 Execution of Distributed Processes

Involved Workpackages: WP-JRA-2.2

In the execution phase, distributed processes are enacted by a process engine, which is responsible for the execution of process models (e.g. the newly introduced BPMN v2.0; cp. also Section 2.19 and Section 2.20). This involves the navigation of control flow and the execution of the process' activities. In the last years, research on execution of distributed processes focused on flexible execution of distributed processes with different emphasis. Besides efforts on adaptation of service compositions (see Section 2.12) approaches have been proposed aiming at a process execution in a more distributed manner and/or a process execution with consideration of information from the outside of the process, i.e. the process' context. This section gives a brief overview of respecting research results in the area of execution of distributed processes.

Resources (i.e. services) used by processes in SOAs are usually distributed (*resource-oriented distribution*). Resulting challenges addressed in the past regard advanced semantics in order to select services based on their functional properties [FFST11, KFS09, KKZ10, SVGMBFB09, WVM09]. However, more and more approaches consider also Non-Functional Characteristics (NFCs) of single services [SBS+07, RMYT07, GGK+11, BABG08, PPCM08, SCM10, Wan09, HS10] or for service compositions [MCD10, LSYF09, MCT08, AJ10, WSO08] as well as the reputation and trust of the service provider [DA09, ZWCC11, MB09, LL09, CIM+09] and [BBDG08] (Polimi) or propose market-based service selection mechanisms [BNW10, LCLS08]. Work has also been done in order to distribute resources in a cloud environment [HM10, Dee10, Cza10, GMEL08]. In S-Cube, novel approaches were proposed in order to provide a semantically [CKP08, CKP09] (Polimi), market-based [VHL+11] (UniHH), NFC-driven [ZNP+09] (UOC) and [MRLD10b, MRLD10a] (TUW), capable with dynamic environments [HSZ11] (UniHH), driven by user-based experiences [LMRD09] (TUW) and chemical-oriented [DNGNT09] (CNR, SZTAKI) service selection with respect to composition fragments selection [BLMB10] (Manchester).

Beside the distribution of resources used by processes, recent research investigates the distribution of the process engine itself (*system-oriented distribution*) in order to address possible performance bottlenecks of centralized process engines. Current approaches use complex event processing or event-like subscribe/publish messages in order to execute a process decentralized [HMLJ08, GYJ11, CWWQ08], utilize multi agent systems in order to provide a flexible and robust infrastructure for process execution [LMJ10, BPJ+10]. Other approaches transform the process model into a set of ECA rules in order to be executed by a rule engine [BFMD10], use a tuple-space infrastructure which executes a transformed process model [MWL08, WML08, MWL10], investigate the application of decentralized workflow enactment in the scientific domain [SGK+10] or in the cloud [CZZX08, ALMS09, EHT07, HXH09, WBP10]. Chemical-based approaches contributed by the S-Cube consortium are discussed in Section 2.25.

Another trend is to distribute the process execution itself, i.e. multiple process engines are involved in executing a process (*execution-oriented distribution*). In the current research, such distributed process execution is achieved by splitting the process model into fragments executed by different process engines. Most papers use a physical fragmentation approach (cp. Section 2.21). However, in S-Cube, another proposed approach is the process instance migration, where the current state of a running process instance is transfered to another workflow engine which then continues the execution [ZKML09, ZHKL10] (UniHH). Corresponding work include ad-hoc management capabilities for such distributed business processes [ZBHL10] (UniHH).

Research in the last years concerning the execution of distributed processes has been also done in the area of mobile and ubiquitous computing. Therefore, several process engines or execution mechanisms for mobile devices were introduced [SRG08, PTKR10, GC11, KF09, dLM10] and [ZKL09] (UniHH). Since in mobile environments, the dynamic context plays an important role, there has been also work on context-based execution of distributed processes [DSKO09, GCFP10, WLS+10]. In S-Cube, a method for a generic structured context prediction [MZL10] (UniHH), a mechanism for using context in service mashup platforms [TKS+10] (TUW), methods for context modeling (cp. Section 2.2) and using context for adaptation (cp. Section 2.12) were proposed. Furthermore, a solution in order to provide process execution as a service for, e.g., mobile participants was introduced [ZL10] (UniHH).

Bibliography

[AJ10] Vikas Agarwal and Pankaj Jalote. From specification to adaptation: An integrated qos-driven approach for dynamic adaptation of web service compositions. In *ICWS*, pages 275–282. IEEE Computer Society, 2010.

[ALMS09] Tobias Anstett, Frank Leymann, Ralph Mietzner, and Steve Strauch. Towards bpel in the cloud: Exploiting different delivery models for the execution of business processes. In *SERVICES I*, pages 670–677, 2009.

[BABG08] Youakim Badr, Ajith Abraham, Frédérique Biennier, and Crina Grosan. Enhancing web service selection by user preferences of non-functional features. In *Proceedings of the 2008 4th International Conference on Next Generation Web Services Practices*, pages 60–65, Washington, DC, USA, 2008. IEEE Computer Society.

[BBDG08] Domenico Bianculli, Walter Binder, Mauro Luigi Drago, and Carlo Ghezzi. Transparent reputation management for composite web services. In *ICWS* [DBL08], pages 621–628.

[BFMD10] Jose Felipe Mejia Bernal, Paolo Falcarin, Maurizio Morisio, and Jia Dai. Dynamic context-aware business process: a rule-based approach supported by pattern identification. In *SAC*, pages 470–474, 2010.

[BLMB10] Chahrazed Bouhini, Freddy Lécué, Nikolay Mehandjiev, and Olivier Boissier. Discovery and selection of web services fragments for re-composition. In *SOCA*, pages 1–4. IEEE, 2010.

[BNW10] Nikolay Borissov, Dirk Neumann, and Christof Weinhardt. Automated bidding in computational markets: an application in market-based allocation of computing services. *Autonomous Agents and Multi-Agent Systems*, 21(2):115–142, 2010.

[BPJ+10] Lars Braubach, Alexander Pokahr, Kai Jander, Winfried Lamersdorf, and Birgit Burmeister. Go4flex: Goal-oriented process modelling. In *Proc. 4th International Symposium on Intelligent Distributed Computing (IDC2010)*, pages 77–87. Springer-Verlag, 9 2010.

[CIM+09] William Conner, Arun Iyengar, Thomas A. Mikalsen, Isabelle Rouvellou, and Klara Nahrstedt. A trust management framework for service-oriented environments. In Juan Quemada, Gonzalo León, Yoëlle S. Maarek, and Wolfgang Nejdl, editors, *WWW*, pages 891–900. ACM, 2009.

[CKP08] Marco Comuzzi, Kyriakos Kritikos, and Pierluigi Plebani. Semantic-aware service quality negotiation. In *ServiceWave*, pages 312–323, 2008.

[CKP09] Marco Comuzzi, Kyriakos Kritikos, and Pierluigi Plebani. A semantic based framework for supporting negotiation in service oriented architectures. In *CEC*, pages 137–145, 2009.

[CWWQ08] Wei Chen, Jun Wei, Guoquan Wu, and Xiaoqiang Qiao. Developing a concurrent service orchestration engine based on event-driven architecture. In *OTM Conferences (1)*, pages 675–690, 2008.

[Cza10] Pawel Czarnul. Modeling, run-time optimization and execution of distributed workflow applications in the jee-based beesycluster environment. *The Journal of Supercomputing*, pages 1–26, 2010.

[CZZX08] Ding Changsong, Hu Zhoujun, Hu Zhigang, and Li Xi. A distributed workflow management system model and its scheduling algorithm. In *Proceedings of the 2008 Japan-China Joint Workshop on Frontier of Computer Science and Technology*, pages 51–58. IEEE Computer Society, 2008.

[DA09] Mehrnoosh Divsalar and Mohammad Abdollahi Azgomi. A computational trust model for e-commerce systems. In Luigi Telesca, Katarina Stanoevska-Slabeva, and Veselin Rakocevic, editors, *DigiBiz*, volume 21 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 9–18. Springer, 2009.

[DBL08] *2008 IEEE International Conference on Web Services (ICWS 2008), September 23-26, 2008, Beijing, China*. IEEE Computer Society, 2008.

[Dee10] Ewa Deelman. Grids and clouds: Making workflow applications work in heterogeneous distributed environments. *IJHPCA*, 24(3):284–298, 2010.

[dLM10] Massimiliano de Leoni and Massimo Mecella. Mobile process management through web services. In *IEEE SCC*, pages 378–385, 2010.

[DNGNT09] C. Di Napoli, M. Giordano, Z. NÃ©meth, and N. Tonellotto. A chemical metaphor to model service selection for composition of services. In *Proceedings of the Second International Workshop on Parallel, Architectures and Bioinspired Algorithms*, pages 11–19, 2009.

[DSKO09] Michael Decker, Peter Stürzel, Stefan Klink, and Andreas Oberweis. Location constraints for mobile workflows. In Juan Enrique Agudo, Cherif Branki, Brian Cross, Gregorio Díaz, Frank-Dieter Dorloff, Guadalupe Ortiz, Key Pousttchi, Martin Randles, Mercedes Rico, Héctor Sánchez, A. Taleb-Bendiab, Frank Teuteberg, and Rainer Unland, editors, *TAMoCo*, volume 201 of *Frontiers in Artificial Intelligence and Applications*, pages 93–102. IOS Press, 2009.

[EHT07] Erik Elmroth, Francisco Hernández, and Johan Tordsson. A light-weight grid workflow execution engine enabling client and middleware independence. In Roman Wyrzykowski, Jack Dongarra, Konrad Karczewski, and Jerzy Wasniewski, editors, *PPAM*, volume 4967 of *Lecture Notes in Computer Science*, pages 754–761. Springer, 2007.

[FFST11] Dieter Fensel, Federico Michele Facca, Elena Paslaru Bontas Simperl, and Ioan Toma. *Semantic Web Services*. Springer, 2011.

[GC11] Christin Groba and Siobhán Clarke. Opportunistic composition of sequentially-connected services in mobile computing environments. In *ICWS*, pages 17–24, 2011.

[GCFP10] Pau Giner, Carlos Cetina, Joan Fons, and Vicente Pelechano. Developing mobile workflow support in the internet of things. *IEEE Pervasive Computing*, 9(2):18–26, 2010.

[GGK$^+$11] Stephen Gilmore, László Gönczy, Nora Koch, Philip Mayer, Mirco Tribastone, and Dániel Varró. Non-functional properties in the model-driven development of service-oriented systems. *Software and System Modeling*, 10(3):287–311, 2011.

[GMEL08] Tristan Glatard, Johan Montagnat, David Emsellem, and Diane Lingrand. A service-oriented architecture enabling dynamic service grouping for optimizing distributed workflow execution. *Future Generation Comp. Syst.*, 24(7):720–730, 2008.

[GYJ11] Siddarth Ganesan, Young Yoon, and Hans-Arno Jacobsen. Niños take five: the management infrastructure for distributed event-driven workflows. In David M. Eyers, Opher Etzion, Avigdor Gal, Stanley B. Zdonik, and Paul Vincent, editors, *DEBS*, pages 195–206. ACM, 2011.

[HM10] Tram Truong Huu and Johan Montagnat. Virtual resources allocation for workflow-based applications distribution on a cloud infrastructure. In *CCGRID*, pages 612–617, 2010.

[HMLJ08] Songlin Hu, Vinod Muthusamy, Guoli Li, and Hans-Arno Jacobsen. Distributed automatic service composition in large-scale systems. In Roberto Baldoni, editor, *DEBS*, volume 332 of *ACM International Conference Proceeding Series*, pages 233–244. ACM, 2008.

[HS10] Chung-Wei Hang and Munindar P. Singh. From quality to utility: Adaptive service selection framework. In *ICSOC*, pages 456–470, 2010.

[HSZ11] Kristof Hamann, Sebastian Steenbuck, and Sonja Zaplata. Towards nfc-aware process execution for dynamic environments. *ECEASST*, 37, 2011.

[HXH09] Xu Huibin, Liu Xinran, and Zhang Hong. A grid workflow process engine: Architecture and simulation. In *Information Technology and Applications, 2009. IFITA '09. International Forum on*, volume 1, pages 245 –248, 2009.

[KF09] Bill Karakostas and George Fakas. An architecture for reliable mobile workflow in a grid environment. In *ICIW*, pages 499–504, 2009.

[KFS09] Matthias Klusch, Benedikt Fries, and Katia P. Sycara. Owls-mx: A hybrid semantic web service matchmaker for owl-s services. *J. Web Sem.*, 7(2):121–133, 2009.

[KHC$^+$09] Markus Kirchberg, Patrick C. K. Hung, Barbara Carminati, Chi-Hung Chi, Rajaraman Kanagasabai, Emanuele Della Valle, Kun-Chan Lan, and Ling-Jyh Chen, editors. *4th IEEE Asia-Pacific Services Computing Conference, IEEE APSCC 2009, Singapore, December 7-11 2009, Proceedings*. IEEE, 2009.

[KKZ10] Matthias Klusch, Patrick Kapahnke, and Ingo Zinnikus. Adaptive hybrid semantic selection of sawsdl services with sawsdl-mx2. *Int. J. Semantic Web Inf. Syst.*, 6(4):1–26, 2010.

[LCLS08] Szu-Yin Lin, Bo-Yuan Chen, Chun-Chieh Liu, and Von-Wun Soo. Web service allocations based on combinatorial auctions and market-based mechanisms. In *CSCWD*, pages 452–458. IEEE, 2008.

[LL09] Lei Li, Yan Wang 0002, and Ee-Peng Lim. Trust-oriented composite service selection and discovery. In *ICSOC/ServiceWave*, pages 50–67, 2009.

[LMJ10] Guoli Li, Vinod Muthusamy, and Hans-Arno Jacobsen. A distributed service-oriented architecture for business process execution. *TWEB*, 4(1), 2010.

[LMRD09] Philipp Leitner, Anton Michlmayr, Florian Rosenberg, and Schahram Dustdar. Selecting web services based on past user experiences. In Kirchberg et al. [KHC+09], pages 205–212.

[LSYF09] Dongmei Liu, Zhiqing Shao, Caizhu Yu, and Guisheng Fan. A heuristic qos-aware service selection approach to web service composition. In Huaikou Miao and Gongzhu Hu, editors, *ACIS-ICIS*, pages 1184–1189. IEEE Computer Society, 2009.

[MB09] Zaki Malik and Athman Bouguettaya. Rateweb: Reputation assessment for trust establishment among web services. *VLDB J.*, 18(4):885–911, 2009.

[MCD10] Daniel A. Menascé, Emiliano Casalicchio, and Vinod K. Dubey. On optimal service selection in service oriented architectures. *Perform. Eval.*, 67(8):659–675, 2010.

[MCT08] Lijun Mei, Wing Kwong Chan, and T. H. Tse. An adaptive service selection approach to service composition. In *ICWS* [DBL08], pages 70–77.

[MRLD10a] Anton Michlmayr, Florian Rosenberg, Philipp Leitner, and Schahram Dustdar. End-to-end support for qos-aware service selection, binding, and mediation in vresco. *IEEE T. Services Computing*, 3(3):193–205, 2010.

[MRLD10b] Anton Michlmayr, Florian Rosenberg, Philipp Leitner, and Schahram Dustdar. Selective service provenance in the vresco runtime. *Int. J. Web Service Res.*, 7(2):65–86, 2010.

[MWL08] Daniel Martin, Daniel Wutke, and Frank Leymann. A novel approach to decentralized workflow enactment. In *EDOC*, pages 127–136, 2008.

[MWL10] Daniel Martin, Daniel Wutke, and Frank Leymann. Tuplespace middleware for petri net-based workflow execution. *IJWGS*, 6(1):35–57, 2010.

[MZL10] Matthias Meiners, Sonja Zaplata, and Winfried Lamersdorf. Structured context prediction: A generic approach. In Frank Eliassen and Rüdiger Kapitza, editors, *DAIS*, volume 6115 of *Lecture Notes in Computer Science*, pages 84–97, 2010.

[NR09] Elisabetta Di Nitto and Matei Ripeanu, editors. *Service-Oriented Computing - ICSOC 2007 Workshops, International Workshops, Vienna, Austria, September 17, 2007, Revised Selected Papers*, volume 4907 of *Lecture Notes in Computer Science*. Springer, 2009.

[PPCM08] Flavio De Paoli, Matteo Palmonari, Marco Comerio, and Andrea Maurino. A meta-model for non-functional property descriptions of web services. In *ICWS* [DBL08], pages 393–400.

[PTKR10] Rüdiger Pryss, Julian Tiedeken, Ulrich Kreher, and Manfred Reichert. Towards flexible process support on mobile devices. In Pnina Soffer and Erik Proper, editors, *CAiSE Forum*, volume 72 of *Lecture Notes in Business Information Processing*, pages 150–165. Springer, 2010.

[RMYT07] Stephan Reiff-Marganiec, HongQing Yu, and Marcel Tilly. Service selection based on non-functional properties. In Nitto and Ripeanu [NR09], pages 128–138.

[SBS+07] Christian Schröpfer, Maxim Binshtok, Solomon Eyal Shimony, Aviram Dayan, Ronen I. Brafman, Philipp Offermann, and Oliver Holschke. Introducing preferences over nfps into service selection in soa. In Nitto and Ripeanu [NR09], pages 68–79.

[SCM10] Benjamin Schmeling, Anis Charfi, and Mira Mezini. Non-functional concerns in web services: requirements and state of the art analysis. In Gabriele Kotsis, David Taniar, Eric Pardede, Imad Saleh, and Ismail Khalil, editors, *iiWAS*, pages 67–74. ACM, 2010.

[SGK+10] Mirko Sonntag, Katharina Görlach, Dimka Karastoyanova, Frank Leymann, and Michael Reiter. Process space-based scientific workflow enactment. *International Journal of Business Process Integration and Management IJBPIM Special Issue on Scientific Workflows Vol 5 No 1 pp 3244*, pages 32–44, 2010.

[SRG08] Rohan Sen, Gruia-Catalin Roman, and Christopher D. Gill. Cian: A workflow engine for manets. In Doug Lea and Gianluigi Zavattaro, editors, *COORDINATION*, volume 5052 of *Lecture Notes in Computer Science*, pages 280–295. Springer, 2008.

[SVGMBFB09] Francisco García Sánchez, Rafael Valencia-García, Rodrigo Martínez-Béjar, and Jesualdo Tomás Fernández-Breis. An ontology, intelligent agent-based framework for the provision of semantic web services. *Expert Syst. Appl.*, 36(2):3167–3187, 2009.

[TKS+10] Martin Treiber, Kyriakos Kritikos, Daniel Schall, Schahram Dustdar, and Dimitris Plexousakis. Modeling context-aware and socially-enriched mashups. In *Proceedings of the 3rd and 4th International Workshop on Web APIs and Services Mashups (Mashups '09/'10)*, pages 2:1–2:8, New York, NY, USA, 2010. ACM.

[VHL+11] Ante Vilenica, Kristof Hamann, Winfried Lamersdorf, Jan Sudeikat, and Wolfgang Renz. An extensible framework for dynamic market-based service selection and business process execution. In Pascal Felber and Romain Rouvoy, editors, *DAIS*, volume 6723 of *Lecture Notes in Computer Science*, pages 150–164, 2011.

[Wan09] Ping Wang. Qos-aware web services selection with intuitionistic fuzzy set under consumer's vague perception. *Expert Syst. Appl.*, 36(3):4460–4466, 2009.

[WBP10] MingXue Wang, Kosala Yapa Bandara, and Claus Pahl. Process as a service. In *IEEE SCC*, pages 578–585, 2010.

[WLS+10] Matthias Wieland, Frank Leymann, Michael Schäfer, Dominik Lucke, Carmen Constantinescu, and Engelbert Westkämper. Using Context-aware Workflows for Failure Management in a Smart Factory. In *Proceedings of the Fourth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies: UBICOMM 2010*, pages 379–384. IARIA, 2010.

[WML08] Daniel Wutke, Daniel Martin, and Frank Leymann. Facilitating complex web service interactions through a tuplespace binding. In René Meier and Sotirios Terzis, editors, *DAIS*, volume 5053 of *Lecture Notes in Computer Science*, pages 275–280, 2008.

[WSO08] Hiroshi Wada, Junichi Suzuki, and Katsuya Oba. Early aspects for non-functional properties in service oriented business processes. In *SERVICES I*, pages 231–238. IEEE Computer Society, 2008.

[WVM09] Mark D. Wilkinson, Benjamin P. Vandervalk, and E. Luke McCarthy. Sadi semantic web services – 'cause you can't always get what you want! In Kirchberg et al. [KHC+09], pages 13–18.

[ZBHL10] Sonja Zaplata, Dirk Bade, Kristof Hamann, and Winfried Lamersdorf. Ad-hoc management capabilities for distributed business processes. In Witold Abramowicz, Rainer Alt, Klaus-Peter Fähnrich, Bogdan Franczyk, and Leszek A. Maciaszek, editors, *ISSS/BPSC*, volume 177 of *LNI*, pages 139–152. GI, 2010.

[ZHKL10] Sonja Zaplata, Kristof Hamann, Kristian Kottke, and Winfried Lamersdorf. Flexible execution of distributed business processes based on process instance migration. *Journal of Systems Integration (JSI)*, 1(3):3–16, 7 2010.

[ZKL09] Sonja Zaplata, Christian P. Kunze, and Winfried Lamersdorf. Context-based cooperation in mobile business environments. *Business & Information Systems Engineering*, 1(4):301–314, 2009.

[ZKML09] Sonja Zaplata, Kristian Kottke, Matthias Meiners, and Winfried Lamersdorf. Towards runtime migration of ws-bpel processes. In *ICSOC/ServiceWave Workshops*, pages 477–487, 2009.

[ZL10] Sonja Zaplata and Winfried Lamersdorf. Towards mobile process as a service. In *SAC*, pages 372–379, 2010.

[ZNP+09] K. Zachos, C. Nikolaou, P. Petridis, G. Stratakis, M. Voskakis, and E. Papathanasiou. Enhancing service network analysis and service selection using requirements-based service discovery. In *1st International Conferences on Advanced Service Computing, Service Computation 2009*, Athens, Greece, 2009.

[ZWCC11] Shenghui Zhao, Guoxin Wu, Guilin Chen, and Haibao Chen. Reputation-aware service selection based on qos similarity. *Journal of Networks*, 6(7):950–957, Jul 2011.

2.23 Monitoring and Analysis of Distributed Processes Performance

Involved Workpackages: WP-JRA-1.2, WP-JRA-2.2

Apart from monitoring of SBAs (Section 2.9) and cross-layer monitoring (Section 2.13), another recent and related research direction is cross-organizational monitoring of business processes. Thereby, one deals with monitoring scenarios, mechanisms and techniques which go beyond monitoring of standalone processes implemented as service orchestrations.

The approaches presented in this section focus on the following aspects: (i) exposing of process models and process instances as manageable Web service resources to clients; (ii) correlation of monitoring data stemming from different processes and process execution environments; (iii) service bus extensions needed for conformance checking of message interactions in service choreographies; (iv) definition of monitoring contracts between participants in service choreographies; (v) monitoring in evolving business networks. Some of the approaches (in particular those dealing with aspects (i) and (ii)) do not explicitly focus on cross-organizational monitoring, however, they are presented here anyway, as they describe related monitoring mechanisms and techniques which can be used as basis for implementing such monitoring solutions.

In [vLLM+08], BPEL processes are exposed as a set of resources. The approach explains in detail how process models and process instances including activities, variables and other BPEL entities are mapped to resources which can be accessed using the Web Services Resource Framework (WSRF) via events (push mode) or operations (pull mode). Clients can thus access monitoring information (such as the state of a running process instance) on running processes at the service provider in a standardized way by using WSRF. The approach does not explicitly cope with definition of metrics or rules as part of monitoring contracts. The approach described in [ZBHL10] (UniHH) which has been developed in S-Cube also maps the process model and process instances to resources. It uses its own process meta-model derived from XML Process Definition Language (XPDL) and exposed it as managed resources using the Web Services Distributed Management (WSDM) set of standards. In addition it supports the definition of customized events and processing rules based on CEP statements implemented using the ESPER framework.

[KSK07] deals with monitoring of BPEL processes which are deployed on different types of BPEL engines. Thereby, a common audit format is presented which allows processing and correlating events across different BPEL engines. [LKS+10] deals with end-to-end monitoring in service-based applications. The underlying model is that of a business composite which realizes a business function by grouping a set of service components which can implement different business processes spread across the company. In order to monitor such a business composite end-to-end, the approach uses information invariants across service components to correlate service instances (and corresponding events) to instances of business composites. The approach is evaluated based on the IBM WebSphere tooling infrastructure. Both approaches deal with correlation issues across processes but do not cover monitoring contracts.

Service choreographies model at design time how participants should interact in terms of message exchanges. At runtime, monitoring can be used for conformance checking, i.e. for validating whether participant interactions comply to the choreography model. [RR09] presents a choreography monitoring infrastructure needed as a prerequisite for conformance checking. The approach assumes that choreographies are modeled in WS-CDL and deals in detail with message correlation and logging mechanisms and how they can be realized as part of a service bus. The conformance checking itself and reactions to protocol violations are not dealt with in the approach. [KEvL+11] introduces a new choreography language, BPELgold, which supports modeling of interaction choreography models based on BPEL. The paper shows how based on BPELgold a choreography-aware service bus can be used to ensure that executed message exchanges comply with a pre-defined choreography. In case a protocol violation is detected, different types of exception handling mechanisms are discussed in the context of choreographies, e.g., dropping of the wrong message, notifying the sender of the violation, triggering of the default exception handling or a pre-defined exception handling, and stopping of the choreography.

In S-Cube, choreography models have been used as the basis for defining monitoring contracts (alt. monitoring agreements) between partners for exchanging of process state information and process metrics. [WDL+09] (USTUTT, UOC, Tiburg) discusses the need for cross-organizational process monitoring in service networks. The concept of a monitoring agreement is introduced which defines the basic monitoring events and composite events each participant has to provide to the other participants in the network. Composite events are used for calculation of KPIs. In [WKK+10] (USTUTT), the choreography language BPEL4Chor is used as a basis for defining monitoring agreements between participants. The monitoring agreement specifies (i) resource events each participant has to provide for its own public process and (ii) composite events for defining metrics and rules. Resource events are specified based on state models of BPEL resources [KHK+11], while composite events are defined using CEP statements over resource events and other composite events. The monitoring agreement is deployed on the monitoring infrastructures of the participants who then exchange monitoring events as specified when the choreography instances are executed.

Monitoring of cross-organizational processes executed within business networks in the context of virtual enterprises [GEM+09] has typically considered monitoring only in the network formation phase, since network establishment determines what can be monitored during process execution. However when business networks and their processes evolve [DCS09], monitoring requirements change and thus resulting monitoring contracts also have to evolve. In that context, [CVG11] discusses monitoring in evolving business networks. For different types of business network evolution situations, mechanisms for preserving the monitorability of processes are specified. Preservation of monitorability includes taking into account dependencies between already established contracts and then updating the monitoring infrastructure in order to satisfy the monitoring requirements that arise after network evolution.

Bibliography

[CVG11] Marco Comuzzi, Jochem Vonk, and Paul Grefen. Measures and mechanisms for process monitoring in evolving business networks. *Data & Knowledge Engineering*, (0):–, 2011.

[DCS09] Nirmit Desai, Amit K. Chopra, and Munindar P. Singh. Amoeba: A methodology for modeling and evolving cross-organizational business processes. *ACM Trans. Softw. Eng. Methodol.*, 19(2), 2009.

[GEM+09] Paul W. P. J. Grefen, Rik Eshuis, Nikolay Mehandjiev, Giorgos Kouvas, and Georg Weichhart. Internet-based support for process-oriented instant virtual enterprises. *IEEE Internet Computing*, 13(6):65–73, 2009.

[KEvL+11] Oliver Kopp, Lasse Engler, Tammo van Lessen, Frank Leymann, and Jörg Nitzsche. Interaction Choreography Models in BPEL: Choreographies on the Enterprise Service Bus. In A. Fleischmann, W. Schmidt, D. Seese, and R. Singer, editors, *Subject-Orientation as Enabler for the Next Generation of BPM Tools and Methods - Second International Conference S-BPM ONE 2010*, volume 138 of *Communications in Computer and Information Science*. Springer, Januar 2011.

[KHK+11] Oliver Kopp, Sebastian Henke, Dimka Karastoyanova, Rania Khalaf, Frank Leymann, Mirko Sonntag, Thomas Steinmetz, Tobias Unger, and Branimir Wetzstein. An Event Model for WS-BPEL 2.0. Technischer Bericht Informatik 2011/07, Universität Stuttgart, Fakultät Informatik, Elektrotechnik und Informationstechnik, Germany, Universität Stuttgart, Institut für Architektur von Anwendungssystemen, September 2011.

[KSK07] Shinji Kikuchi, Hisashi Shimamura, and Yoshihiro Kanna. Monitoring method of cross-sites' processes executed by multiple ws-bpel processors. In *CEC/EEE*, pages 55–64. IEEE Computer Society, 2007.

[LKS+10] Geetika T. Lakshmanan, Paul T. Keyser, Aleksander Slominski, Francisco Curbera, and Rania Khalaf. A business centric end-to-end monitoring approach for service composites. In *IEEE SCC*, pages 409–416. IEEE Computer Society, 2010.

[RR09] Michael Von Riegen and Norbert Ritter. Reliable monitoring for runtime validation of choreographies. In Mark Perry, Hideyasu Sasaki, Matthias Ehmann, Guadalupe Ortiz Bellot, and Oana Dini, editors, *ICIW*, pages 310–315. IEEE Computer Society, 2009.

[vLLM+08] Tammo van Lessen, Frank Leymann, Ralph Mietzner, Jörg Nitzsche, and Daniel Schleicher. A management framework for ws-bpel. In Claus Pahl, Siobhán Clarke, and Rik Eshuis, editors, *ECOWS*, pages 187–196. IEEE Computer Society, 2008.

[WDL+09] Branimir Wetzstein, Olha Danylevych, Frank Leymann, Marina Bitsaki, Christos Nikolaou, Willem-Jan van den Heuvel, and Mike Papazoglou. Towards monitoring of key performance indicators across partners in service networks. In *International Workshop Series on Monitoring, Adaptation and Beyond (MONA+), collocated with ICSOC/ServiceWave 2009, Stockholm, Sweden*, Berlin–Heidelberg, February 2009. Springer.

[WKK+10] Branimir Wetzstein, Dimka Karastoyanova, Oliver Kopp, Frank Leymann, and Daniel Zwink. Cross-organizational process monitoring based on service choreographies. In Sung Y. Shin, Sascha Ossowski, Michael Schumacher, Mathew J. Palakal, and Chih-Cheng Hung, editors, *SAC*, pages 2485–2490. ACM, 2010.

[ZBHL10] Sonja Zaplata, Dirk Bade, Kristof Hamann, and Winfried Lamersdorf. Ad-hoc management capabilities for distributed business processes. In Witold Abramowicz, Rainer Alt, Klaus-Peter Fähnrich, Bogdan Franczyk, and Leszek A. Maciaszek, editors, *ISSS/BPSC*, volume 177 of *LNI*, pages 139–152. GI, 2010.

2.24 Prediction and Prevention of Service Level Agreement Violations in Service Compositions

Involved Workpackages: WP-JRA-1.3, WP-JRA-2.2

The current section is relatively concise because most of the works related to this topic have already been covered in Section 2.16.

Prediction of Service Level Agreement Violations Since the initial state-of-the-art survey [PO-08] (UOC, UPM, USTUTT, TUW), significant progress has been made within and outside of S-Cube. Generally speaking, research work in this area can be split into approaches which *explain* observed violations, and approaches which *predict* future violations. In the category of explaining approaches, some important work has been conducted by Bodenstaff et al. [BWRJ08, BWRJ09]. Furthermore, S-Cube partners have developed the idea of dependency analysis to discover the main factors that influence SLA conformance [WLR+09, WLR+11] (USTUTT, TUW). In the category of predictive approaches, Zeng et al. have proposed an event-based QoS prediction approach [ZLLC08]. This idea has been picked up and refined by S-Cube members in [LWR+09] (TUW, USTUTT). These approaches are both based on the idea of machine learning based prediction. Recently presented alternatives include prediction using runtime verification [GBK+10] (UniDue, FBK) or based on static analysis of service compositions [ICH09] (UPM).

Prevention of Service Level Agreement Violations Preventing SLA violations after they have been predicted by the means discussed above usually boils down to *adapting* the SBA. Primitively, this adaptation can happen on two levels: (1) firstly, by adapting the base services used, or by adapting these services, and (2) secondly, by structurally adapting the SBA. In the former problem space, important work has existed prior to S-Cube, and, to the best of our knowledge, little groundbreaking work has been produced since then. For the latter problem, the research community seems to have converged to the idea of using the notion of aspect-oriented programming to adapt e.g., service compositions, as indicated by researchers from within (e.g., [LWK+10] (TUW, USTUTT) and [KL09] (USTUTT)) and outside of S-Cube (e.g., [NLF+09]) having adopted this general idea. In addition, some important progress has been made on the foundations of adaptable SBAs, discussed e.g., in [NGM+08] (Polimi, UniDue, Tilburg) and more recently in [YYS+09]. Finally, important new insights into the development of adaptable SBAs have been presented by Lambers et al. in [LMEP08].

Bibliography

[BWRJ08] Lianne Bodenstaff, Andreas Wombacher, Manfred Reichert, and Michael C. Jaeger. Monitoring dependencies for slas: The mode4sla approach. In *IEEE SCC (1)*, pages 21–29. IEEE Computer Society, 2008.

[BWRJ09] Lianne Bodenstaff, Andreas Wombacher, Manfred Reichert, and Michael C. Jaeger. Analyzing impact factors on composite services. In *IEEE SCC*, pages 218–226. IEEE Computer Society, 2009.

[GBK+10] Andreas Gehlert, Antonio Bucchiarone, Raman Kazhamiakin, Andreas Metzger, Marco Pistore, and Klaus Pohl. Exploiting assumption-based verification for the adaptation of service-based applications. In *SAC*, pages 2430–2437, 2010.

[ICH09] Dragan Ivanovic, Manuel Carro, and Manuel V. Hermenegildo. An initial proposal for data-aware resource analysis of orchestrations with applications to predictive monitoring. In *ICSOC/ServiceWave Workshops*, pages 414–424, 2009.

[KL09] Dimka Karastoyanova and Frank Leymann. Bpel'n'aspects: Adapting service orchestration logic. In *ICWS*, pages 222–229. IEEE, 2009.

[LMEP08] Leen Lambers, Leonardo Mariani, Hartmut Ehrig, and Mauro Pezzè. A formal framework for developing adaptable service-based applications. In José Luiz Fiadeiro and Paola Inverardi, editors, *FASE*, volume 4961 of *Lecture Notes in Computer Science*, pages 392–406. Springer, 2008.

[LWK+10] Philipp Leitner, Branimir Wetzstein, Dimka Karastoyanova, Waldemar Hummer, Schahram Dustdar, and Frank Leymann. Preventing SLA violations in service compositions using aspect-based fragment substitution. In *ICSOC*, pages 365–380, 2010.

[LWR+09] Philipp Leitner, Branimir Wetzstein, Florian Rosenberg, Anton Michlmayr, Schahram Dustdar, and Frank Leymann. Runtime prediction of service level agreement violations for composite services. In *ICSOC/ServiceWave Workshops*, pages 176–186, 2009.

[NGM+08] Elisabetta Di Nitto, Carlo Ghezzi, Andreas Metzger, Mike P. Papazoglou, and Klaus Pohl. A journey to highly dynamic, self-adaptive service-based applications. *Autom. Softw. Eng.*, 15(3-4):313–341, 2008.

[NLF+09] Jörg Niemöller, Roman Levenshteyn, Eugen Freiter, Konstantinos Vandikas, Raphaël Quinet, and Ioannis Fikouras. Aspect orientation for composite services in the telecommunication domain. In *ICSOC/ServiceWave*, pages 19–33, 2009.

[PO-08] Overview of the state of the art in composition and coordination of services. Deliverable, S-Cube Consortium, 2008.

[WLR+09] Branimir Wetzstein, Philipp Leitner, Florian Rosenberg, Ivona Brandic, Schahram Dustdar, and Frank Leymann. Monitoring and analyzing influential factors of business process performance. In *EDOC*, pages 141–150. IEEE Computer Society, 2009.

[WLR+11] Branimir Wetzstein, Philipp Leitner, Florian Rosenberg, Schahram Dustdar, and Frank Leymann. Identifying influential factors of business process performance using dependency analysis. *Enterprise IS*, 5(1):79–98, 2011.

[YYS+09] Stephen S. Yau, Nong Ye, Hessam S. Sarjoughian, Dazhi Huang, A. Roontiva, M. Baydogan, and M. A. Muqsith. Toward development of adaptive service-based software systems. *IEEE T. Services Computing*, 2(3):247–260, 2009.

[ZLLC08] Liangzhao Zeng, Christoph Lingenfelder, Hui Lei, and Henry Chang. Event-driven quality of service prediction. In Athman Bouguettaya, Ingolf Krüger, and Tiziana Margaria, editors, *ICSOC*, volume 5364 of *Lecture Notes in Computer Science*, pages 147–161, 2008.

2.25 Chemical- and Nature-Inspired Techniques for Service Composition

Involved Workpackages: WP-JRA-2.3

Service Based Applications are realized by composing third-party services accessible through Internet, and as such they are managed and operated in a completely decentralized way. Following the recent trends in Service Oriented applications, it is very likely that the real-life provision of such services will increasingly follow the market-oriented approaches that regulate the demand and supply of services [BYV$^+$09]. Users will require SBAs with end-to-end quality requirements and third-party service providers will provide services with QoS. Both user requirements and service quality attributes, and also the number of available services may change in time due to the dynamic nature of service-based environments. In this scenario, the QoS-aware service composition cannot be just the binding of each SBA activity to one discovered service, since it is necessary to take into account that multiple offers for the same service may be available and/or may become available during the binding process.

To deal with this dynamicity, S-Cube contributes to the state of the art in the context of nature inspired computing approaches for service composition, by proposing a dynamic binding mechanism (see Section 2.12) based on a chemical metaphor [NGNT10] (CNR, SZTAKI) and [NGPW10] (CNR, INRIA). The use of a chemical approach allows the modeling of the QoS-aware mapping (see Section 2.14) between SBA activities and services as an evolving and autonomic mechanism that is distributed and incremental, i.e. a complete mapping is built by aggregating smaller mappings. Furthermore, environmental changes, in terms of new provider availability, or changes in the provided QoS, are taken into account as chemical system perturbations so that new mappings can be computed because new chemical reactions may take place in a way that simulate an adaptation of the system to different configurations not planned in advance. Research in S-Cube also involved workflow formalization and enactment within the chemical metaphor with a special emphasis on its dynamic aspects is described by Caeiro et al. [CNP11] (INRIA, SZTAKI).

Nature based approaches in the scope of composition and coordination are extensively studied. Service composition involves many aspects of optimization, scheduling, coordination and has attracted a lot of interest in the field of nature inspired methods. Early works include a neuro-endocrine framework for web service composition in an emergent and self-evolving way [DSH07, SD07], genetic algorithms [CPEV05] and ant colonies [HAES09]. Later, genetic-algorithm-based approach is proposed to compose services (in a utility computing, grid-like manner) in cloud computing [YZB11]. Dynamic service composition mimicking the slime mold Physarum polycephalum – a single-cell organism – that can form a veined network that explores the available space and connects food sources in the absence of central control mechanisms [ZZWQ11]. A tree-coded [GCC07] and a multi-objective SPEA2 [LCOZ10] / NSGA-II [LYO$^+$10] genetic algorithm was proposed for QoS-aware composition. Workflow management is closely related to service composition (in fact, it is a predecessor to service composition) and attracted a large deal of nature based solutions. Scheduling workflows on clouds by particle swarm optimization that takes into consideration both computational and communication aspects [PWGB10]. The idea of chemical workflow management [NPP06] was developed further in [FP10].

As a general trend, nature inspired models and techniques gain more attention in the scope of service management where apart from composition investigated in S-Cube, optimization, deployment, coordination and other problems are solved by these metaphors [ABD$^+$10]. This trend may reach a state of holistic approaches: some works do not aim specifically at a particular problem of service management rather, envision the whole service ecosystem as a nature metaphor and try to realize all related activities within this nature framework. Viroli et al. [VZ10] designed a conceptual architecture for clarifying the concepts of possible nature-inspired metaphors that could be adopted. The spatial distribution of pervasive services is treated as an ecological model in [VRVZ08]. Furthermore, the same spatial coordination problem is solved by a chemical model in [VCMZ11]. A survey of techniques for self-organizing, self-adapting pervasive and mobile services classifies physical, chemical, biological, social and ecological approaches [VZ11]. Services are naturally accessed or provided by mobile devices, which obviously raises various issues of adaptation, a genetic based adaptation is proposed in [VRB08].

Bibliography

[ABD+10] Françoise André, Ivona Brandic, Erwan Daubert, Guillaume Gauvrit, Maurizio Giordano, Gabor Kecskemeti, Attila Kertész, Claudia Di Napoli, Zsolt Németh, Jean-Louis Pazat, Harald Psaier, Wolfgang Renz, and Jan Sudeikat. Architectures & infrastructure. In Mike P. Papazoglou, Klaus Pohl, Michael Parkin, and Andreas Metzger, editors, *S-CUBE Book*, volume 6500 of *Lecture Notes in Computer Science*, pages 85–116. Springer, 2010.

[BYV+09] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic. Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Comp. Syst.*, 25(6):599–616, 2009.

[CNP11] Manuel Caeiro, Zsolt Németh, and Thierry Priol. A chemical model for dynamic workflow coordination. In Yiannis Cotronis, Marco Danelutto, and George Angelos Papadopoulos, editors, *PDP*, pages 215–222. IEEE Computer Society, 2011.

[CPEV05] Gerardo Canfora, Massimiliano Di Penta, Raffaele Esposito, and Maria Luisa Villani. An approach for QoS-aware service composition based on genetic algorithms. In Hans-Georg Beyer and Una-May O'Reilly, editors, *GECCO*, pages 1069–1075. ACM, 2005.

[DSH07] Yongsheng Ding, Hongbin Sun, and Kuangrong Hao. A bio-inspired emergent system for intelligent web service composition and management. *Knowl.-Based Syst.*, 20(5):457–465, 2007.

[FP10] Marc Eduard Frîncu and Dana Petcu. OSyRIS: a nature inspired workflow engine for service oriented environments. *j-SCPE*, 11(1):81–97, March 2010.

[GCC07] Chunming Gao, Meiling Cai, and Huowang Chen. Qos-aware service composition based on tree-coded genetic algorithm. In *COMPSAC (1)*, pages 361–367. IEEE Computer Society, 2007.

[HAES09] M. Shamim Hossain, Atif Alamri, and Abdulmotaleb El-Saddik. A biologically inspired framework for multimedia service management in a ubiquitous environment. *Concurrency and Computation: Practice and Experience*, 21(11):1450–1466, 2009.

[LCOZ10] Li Li, Peng Cheng, Ling Ou, and Zili Zhang. Applying multi-objective evolutionary algorithms to qos-aware web service composition. In Longbing Cao, Jiang Zhong, and Yong Feng, editors, *ADMA (2)*, volume 6441 of *Lecture Notes in Computer Science*, pages 270–281. Springer, 2010.

[LYO+10] Li Li, Pengyi Yang, Ling Ou, Zili Zhang, and Peng Cheng. Genetic algorithm-based multi-objective optimisation for qos-aware web services composition. In Yaxin Bi and Mary-Anne Williams, editors, *KSEM*, volume 6291 of *Lecture Notes in Computer Science*, pages 549–554. Springer, 2010.

[NGNT10] Claudia Di Napoli, Maurizio Giordano, Zsolt Németh, and Nicola Tonellotto. Adaptive instantiation of service workflows using a chemical approach. In Mario R. Guarracino, Frédéric Vivien, Jesper Larsson Träff, Mario Cannatoro, Marco Danelutto, Anders Hast, Francesca Perla, Andreas Knüpfer, Beniamino Di Martino, and Michael Alexander, editors, *Euro-Par Workshops*, volume 6586 of *Lecture Notes in Computer Science*, pages 247–255. Springer, 2010.

[NGPW10] Claudia Di Napoli, Maurizio Giordano, Jean-Louis Pazat, and Chen Wang. A chemical based middleware for workflow instantiation and execution. In Elisabetta Di Nitto and Ramin Yahyapour, editors, *ServiceWave*, volume 6481 of *Lecture Notes in Computer Science*, pages 100–111. Springer, 2010.

[NPP06] Z. Nemeth, C. Perez, and T. Priol. Distributed workflow coordination: molecules and reactions. In *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*, page 8 pp., april 2006.

[PWGB10] Suraj Pandey, Linlin Wu, Siddeswara Mayura Guru, and Rajkumar Buyya. A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments. In *AINA*, pages 400–407. IEEE Computer Society, 2010.

[SD07] Hongbin Sun and Yongsheng Ding. E-service emergence: A bio-inspired method of composition. In *CISIS*, pages 53–59. IEEE Computer Society, 2007.

[VCMZ11] Mirko Viroli, Matteo Casadei, Sara Montagna, and Franco Zambonelli. Spatial coordination of pervasive services through chemical-inspired tuple spaces. *TAAS*, 6(2):14, 2011.

[VRB08] Yves Vanrompay, Peter Rigole, and Yolande Berbers. Genetic algorithm-based optimization of service composition and deployment. In *Proceedings of the 3rd international workshop on Services integration in pervasive environments*, SIPE '08, pages 13–18, New York, NY, USA, 2008. ACM.

[VRVZ08] Cynthia Villalba, Alberto Rosi, Mirko Viroli, and Franco Zambonelli. Nature-inspired spatial metaphors for pervasive service ecosystems. In *SASO Workshops*, pages 332–337. IEEE Computer Society, 2008.

[VZ10] Mirko Viroli and Franco Zambonelli. A biochemical approach to adaptive service ecosystems. *Inf. Sci.*, 180(10):1876–1892, 2010.

[VZ11] Cynthia Villalba and Franco Zambonelli. Towards nature-inspired pervasive service ecosystems: Concepts and simulation experiences. *J. Network and Computer Applications*, 34(2):589–602, 2011.

[YZB11] Zhen Ye, Xiaofang Zhou, and Athman Bouguettaya. Genetic algorithm based qos-aware service compositions in cloud computing. In Jeffrey Xu Yu, Myoung-Ho Kim, and Rainer Unland, editors, *DASFAA (2)*, volume 6588 of *Lecture Notes in Computer Science*, pages 321–334. Springer, 2011.

[ZZWQ11] Huan Zhou, Zili Zhang, Yuheng Wu, and Tao Qian. Bio-inspired dynamic composition and reconfiguration of service-oriented internetware systems. In Ying Tan, Yuhui Shi, Yi Chai, and Guoyin Wang, editors, *ICSI (1)*, volume 6728 of *Lecture Notes in Computer Science*, pages 364–373. Springer, 2011.

2.26 Service Level Agreement-Based Virtualized Service Provision

Involved Workpackages: WP-JRA-1.3, WP-JRA-2.3

Solutions interoperating different service infrastructures use well-established and widely used technologies that enable the collaboration of different parties to fulfill service executions required by users. The recently emerged demands of users and researchers called for expanding the service model with business-oriented utilization (agreement handling) and support for human-provided and computation-intensive Grid services. Most of related works consider either virtualization approaches [ITF+08, PS09] without taking care of SLAs or concentrates on SLA management neglecting the appropriate resource virtualizations [VCB08, QA09]. To the best of our knowledge, the work of S-Cube partners on SLA-based resource virtualization [KKB09] (SZTAKI, TUW) was the first attempt to combine SLA-based resource negotiations with virtualized resources in terms of on-demand service provision resulting in a holistic virtualization approach.

Managing different heterogeneous and distributed environments requires sophisticated interoperation of adaptive coordinating components. Cloud Computing [BYV+09] is a recently emerged research paradigm, which is highly studied and a large body of work has been done trying to define and envision the boundaries of this new area. Despite its business-orientation, the applicability of SLAs in the Cloud middleware has been rarely studied [YG09]. Regarding self-* properties, the work by Van et. al. [NVDTM09] studied the applicability of the autonomic computing to Cloud-like systems, but they almost exclusively focus on virtualization issues like VM packing and placement. Interoperating Grid and Cloud infrastructures has been addressed by many papers [OPF09, BHML09, MSJ09, ABHM10], but they all lack autonomic management and incorporated agreement negotiation. In order to avoid costly SLA violations, flexible and adaptive SLA attainment strategies are used with a failure propagation approach in the SLA-aware Service Virtualization solution [KKB11] proposed by S-Cube partners. This system incorporates an autonomic manager, which is an abstract component that specifies how self-management is carried out; for the identification of failures they use case-based reasoning with a knowledge database.

Bibliography

[ABHM10] Brian Amedro, Françoise Baude, Fabrice Huet, and Elton N. Mathias. Combining grid and cloud resources by use of middleware for spmd applications. In *CloudCom*, pages 177–184. IEEE, 2010.

[BHML09] Constantino Vázquez Blanco, Eduardo Huedo, Rubén S. Montero, and Ignacio Martín Llorente. Dynamic provision of computing resources from grid infrastructures and cloud providers. In *GPC Workshops* [DBL09], pages 113–120.

[BYV+09] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic. Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Comp. Syst.*, 25(6):599–616, 2009.

[DBL09] *Workshops at the Grid and Pervasive Computing Conference, GPC 2009, Geneva, Switzerland, May 4-8, 2009*. IEEE Computer Society, 2009.

[ITF+08] Alexandru Iosup, Todd Tannenbaum, Matthew Farrellee, Dick H. J. Epema, and Miron Livny. Inter-operating grids through delegated matchmaking. *Scientific Programming*, 16(2-3):233–253, 2008.

[KKB09] Attila Kertesz, Gabor Kecskemeti, and Ivona Brandic. An SLA-based resource virtualization approach for on-demand service provision. In *Proceedings of the 3rd international workshop on Virtualization technologies in distributed computing*, VTDC '09, pages 27–34, New York, NY, USA, 2009. ACM.

[KKB11] A. Kertesz, G. Kecskemeti, and I. Brandic. Autonomic SLA-aware service virtualization for distributed systems. In *Parallel, Distributed and Network-Based Processing (PDP), 2011 19th Euromicro International Conference on*, pages 503 –510, feb. 2011.

[MSJ09] André Merzky, Katerina Stamou, and Shantenu Jha. Application level interoperability between clouds and grids. In *GPC Workshops* [DBL09], pages 143–150.

[NVDTM09] Hien Nguyen Van, Frederic Dang Tran, and Jean-Marc Menaud. Autonomic virtual resource management for service hosting platforms. In *Proceedings of the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing*, CLOUD '09, pages 1–8, Washington, DC, USA, 2009. IEEE Computer Society.

[OPF09] Simon Ostermann, Radu Prodan, and Thomas Fahringer. Extending grids with cloud resource management for scientific computing. In *GRID*, pages 42–49. IEEE, 2009.

[PS09] Young R. Park and Srinarayan Sharma. Providing service using a virtualization infrastructure. *Journal of Service Science*, 2(2):17–22, 2009.

[QA09] Dang Minh Quan and Jörn Altmann. Resource allocation algorithm for light communication grid-based workflows within an SLA context. *IJPEDS*, 24(1):31–48, 2009.

[VCB08] Srikumar Venugopal, Xingchen Chu, and Rajkumar Buyya. A negotiation mechanism for advance resource reservations using the alternate offers protocol. In Hans van den Berg and Gunnar Karlsson, editors, *IWQoS*, pages 40–49. IEEE, 2008.

[YG09] C. A. Yfoulis and A. Gounaris. Honoring SLAs on cloud computing services: a control perspective. In *Proceedings of the European Control Conference*, 2009.

2.27 Federated Cloud Management

Involved Workpackages: WP-JRA-2.3

Cloud Computing [BYV+09] is a novel research area that offers simple and cost effective outsourcing in dynamic service environments and allows the construction of SBAs extensible with the latest achievements of diverse research areas. IaaS cloud systems provide access to remote computing infrastructures by allowing their users to instantiate virtual appliances on their virtualized resources as VMs. Further discussions regarding the usability of Cloud infrastructures for software can be found in Section 2.4.

Managing different Cloud systems in a federation is a "hot" topic, which is currently addressed by many research papers e.g. [RBL+09, Clo09, FHT+12]. Efficiently managing such systems require many sophisticated approaches ranging from virtual machine distribution strategies to scheduling among suitable providers for user needs. Matthias Schmidt et al. [SFSF10] investigate different strategies for distributing virtual machine images within a data center: uni-cast, multi-cast, binary tree distribution and peer-to-peer distribution based on BitTorrent. Paul Marshall et al. [MKF10] describe an approach for developing an "elastic site" model where batch schedulers, storage and web services can utilize such resources. They concentrate on dynamically increasing and decreasing the number of resources, but rely on third party logic for balancing load among the allocated resources. Constantino Vázquez et al. [VHML11] are building complex grid infrastructures on top of IaaS cloud systems that allow them to adjust the number of grid resources dynamically.

S-Cube partners proposed a Federated Cloud Management architecture [MKKK11] (SZTAKI) that acts as an entry point to cloud federations and incorporates the concepts of meta-brokering, cloud brokering and on-demand service deployment. In order to fast track the virtual machine instantiation, this architecture uses the automatic service deployment component that is capable of optimizing service delivery by encapsulating services as virtual appliances in order to allow their decomposition and replication among the various IaaS cloud infrastructures.

Bibliography

[BYV+09] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic. Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Comp. Syst.*, 25(6):599–616, 2009.

[Clo09] Interoperable clouds: A white paper from the open cloud standards incubator. DMTF Informational DSP-IS0101, Distributed Management Task Force, Inc. (DMTF), 2009.

[FHT+12] Ana Juan Ferrer, Francisco Hernández, Johan Tordsson, Erik Elmroth, Ahmed Ali-Eldin, Csilla Zsigri, Raúl Sirvent, Jordi Guitart, Rosa M. Badia, Karim Djemame, Wolfgang Ziegler, Theo Dimitrakos, Srijith K. Nair, George Kousiouris, Kleopatra Konstanteli, Theodora A. Varvarigou, Benoit Hudzia, Alexander Kipp, Stefan Wesner, Marcelo Corrales, and et al. Optimis: A holistic approach to cloud service provisioning. *Future Generation Comp. Syst.*, 28(1):66–77, 2012.

[MKF10] Paul Marshall, Kate Keahey, and Timothy Freeman. Elastic site: Using clouds to elastically extend site resources. In *CCGRID*, pages 43–52. IEEE, 2010.

[MKKK11] Attila Csaba Marosi, Gabor Kecskemeti, Attila Kertesz, and P. Kacsuk. FCM: an architecture for integrating IaaS cloud systems. In *Proceedings of The Second International Conference on Cloud Computing, GRIDs, and Virtualization*, 2011.

[RBL+09] Benny Rochwerger, David Breitgand, Eliezer Levy, Alex Galis, Kenneth Nagin, Ignacio Martín Llorente, Rubén S. Montero, Yaron Wolfsthal, Erik Elmroth, Juan A. Cáceres, Muli Ben-Yehuda, Wolfgang Emmerich, and Fermín Galán. The reservoir model and architecture for open federated cloud computing. *IBM Journal of Research and Development*, 53(4):4, 2009.

[SFSF10] Matthias Schmidt, Niels Fallenbeck, Matthew Smith, and Bernd Freisleben. Efficient distribution of virtual machines for cloud computing. In Marco Danelutto, Julien Bourgeois, and Tom Gross, editors, *PDP*, pages 567–574. IEEE Computer Society, 2010.

[VHML11] Constantino Vázquez, Eduardo Huedo, Rubén S. Montero, and Ignacio Martín Llorente. On the use of clouds for grid resource provisioning. *Future Generation Comp. Syst.*, 27(5):600–605, 2011.

2.28 Process Mining

Involved Workpackages: WP-JRA-2.3

Since the mid-nineties several groups have been working on techniques for process mining [AGL98, CW98, Dat98, WvdA03, vdAWM04, GvdA07, vdWvDHS09, vdARV+10]. In [vdAvDH+03] an overview is given of the early work in this domain. In a nutshell, the goal of process mining is to build models without a priori knowledge, i.e., to discover structured process models derived from specific patterns that are present in actual traces of service executions.

Process mining is an active research area in business process analysis and it has been discussed in many papers [vdAW04, zMR00, GGMS05, TTM08]. Most of the works focus on different kinds of analyses and benefits that can be achieved. These approaches assume that process trace data are already available. However, extracting process trace data is a crucial challenge as discussed in [vdAW04, vdA07]. To this end, in the last years many efforts have been spent on this issue thus resulting in several approaches proposed to address this specific challenge.

Attempts have been made to extract process data from ERP systems log files, e.g. SAP, PeopleSoft. In [vG04], the author tries to extract process trace data from SAP R/3 log files, but it was not fully satisfying because process execution information is too detailed due to the usage at the transaction level and scattered in many tables. In [IG07], authors build a tool for constructing process instances from business process executions in SAP systems. They use SQL statements to identify used business objects in

SAP tables and correlate them with events to construct instances. However, this "table-based" approach provides less data for process mining analysis, as not all the data are stored with business objects.

Another approach based on data warehousing technologies was initially introduced in [CS02]. There, the authors exploit HP Process Manager (HPPM) system logs, which however made this approach not generic. This work is further extended as architecture for analysis and monitoring [GCC+04]. Information from process logs are extracted by querying log tables and stored in a process data warehouse.

More recently, most work have tried to exploit the huge collections of events generated by today SBAs, and traditional enterprise systems in general, (e.g., messages being sent and received between service components), which are recorded to proper log files, i.e., event logs. These files represent a huge and valuable source of knowledge that may be extracted by well-known data mining algorithms and techniques. In [zMR00, vDvdA05], meta models are proposed to correlate elements of event log files. These approaches mostly focus on extracting a small part of process trace data from log files, e.g. case id, task, timestamps, thus for discovering processes, and do not address that challenge in heterogeneous environments. In [Kas07], data from log files are used to analyze the interaction behavior of users with applications.

Starting from 2008, research studies conducted on process mining and discovery follow the established concept of Business Activity Monitoring (BAM). BAM considers the event-driven governance of business processes, and is, hence, mostly a term from the business domain. Technically, BAM is enabled by monitoring runtime of services and their interactions within company SBAs. To this end, event-based monitoring approaches produce a steady stream of low-level lifecycle events [ZLC07, BGPT09] and [WSL09] (USTUTT). These low-level events need to be aggregated so that real business information can be gained from them. Existing techniques to do this include SLA aggregation [ULMS08] or event-based SLA monitoring [SMS+02] [MRLD09] (TUW).

Within S-Cube, the research has included work related to the ideas of BAM, which considers event-based monitoring of business compliance [MZD09]. That differs from other approaches that, instead, aim at specifically mining for invocation sequences that lead to failed service invocations, which is complementary to BAM. Indeed, while BAM is mostly concerned with discovering failures, those works can be used to identify or predict them in advance.

In the literature, events emitted by service-based applications have found various other uses. For instance, another work produced in the scope of S-Cube project [MRLD10] (TUW), uses event information (along with various other data sources) to generate visualizations of the past behavior and quality of Web services, mostly to ease management and selection of services for other goals. Moreover, as already described in Section 2.16, other studies explore how SOA events might be used to generate predictions of SLA violations [ZLLC08, LWR+09] (TUW, USTUTT). Basically, this is done by training machine learning models from the collected event data, and using runtime event information as input to those models.

However, the vast majority of relevant contributions in the field of process mining come from van der Aalst [vdAW04, vdAWM04, vdAvDG+09]. In one of his latest work [vdAPS10], a generic approach to "operational support" in the context of the ProM framework is presented. Here, the author shows that process mining is not restricted to the "past", but also relevant for the "present" (i.e., recommendation and real-time conformance checking), and the "future" (i.e., prediction).

Pursuing the same goal within the S-Cube research framework, authors of [LNT+11] (TUW, CNR) focus on process mining by a different perspective. In particular, authors try to generate recommendations about what combinations of services have in the past been used together (indicating that it might make sense to use them in combination). In order to do that, they detect frequent sequential patterns, thus considering process mining as a specific instance of the more general sequential pattern mining problem. Furthermore, authors apply two sequential pattern mining algorithms to a real event log provided by the Vienna Runtime Environment for Service-oriented Computing (VRESCo). Authors show to be able to find services that are frequently invoked together within the same sequence. Also, they propose to exploit such knowledge at "design-time", when service-based application developers could be provided with "service recommendation tools" that are able to predict and thus to suggest next services that should be included in the current service composition.

Finally, in [vdASS11] authors present an explicit process model used for predictions. In particular, the degree of abstraction of the derived explicit model can be adjusted based on the volume of historic information. Of course, the more data are available, the more fine-grained and accurate is the ultimate process model. The approach allows for better diagnostics (unlike, for example, the non-parametric regression approach presented in [vDCvdA08]). Authors show that their approach has a better performance compared to competitors based on simulation or regression (both in terms of quality of prediction and computational cost).

Bibliography

[AGL98] Rakesh Agrawal, Dimitrios Gunopulos, and Frank Leymann. Mining process models from workflow logs. In Hans-Jörg Schek, Fèlix Saltor, Isidro Ramos, and Gustavo Alonso, editors, *EDBT*, volume 1377 of *Lecture Notes in Computer Science*, pages 469–483. Springer, 1998.

[BGPT09] Luciano Baresi, Sam Guinea, Marco Pistore, and Michele Trainotti. Dynamo + astro: An integrated approach for bpel monitoring. In *ICWS*, pages 230–237, 2009.

[CS02] Fabio Casati and Ming-Chien Shan. Semantic analysis of business process executions. In Christian S. Jensen, Keith G. Jeffery, Jaroslav Pokorný, Simonas Saltenis, Elisa Bertino, Klemens Böhm, and Matthias Jarke, editors, *EDBT*, volume 2287 of *Lecture Notes in Computer Science*, pages 287–296. Springer, 2002.

[CW98] Jonathan E. Cook and Alexander L. Wolf. Discovering models of software processes from event-based data. *ACM Trans. Softw. Eng. Methodol.*, 7(3):215–249, 1998.

[Dat98] Anindya Datta. Automating the discovery of as-is business process models: Probabilistic and algorithmic approaches. *Information Systems Research*, 9(3):275–301, 1998.

[GCC$^+$04] Daniela Grigori, Fabio Casati, Malu Castellanos, Umeshwar Dayal, Mehmet Sayal, and Ming-Chien Shan. Business process intelligence. *Comput. Ind.*, 53:321–343, April 2004.

[GGMS05] Gianluigi Greco, Antonella Guzzo, Giuseppe Manco, and Domenico Saccà. Mining and reasoning on workflows. *IEEE Trans. Knowl. Data Eng.*, 17(4):519–534, 2005.

[GvdA07] Christian W. Günther and Wil M. P. van der Aalst. Fuzzy mining - adaptive process simplification based on multi-perspective metrics. In Gustavo Alonso, Peter Dadam, and Michael Rosemann, editors, *BPM*, volume 4714 of *Lecture Notes in Computer Science*, pages 328–343. Springer, 2007.

[IG07] Jon Espen Ingvaldsen and Jon Atle Gulla. Preprocessing support for large scale process mining of sap transactions. In *Business Process Management Workshops*, pages 30–41, 2007.

[Kas07] G. Kassem. *Application Usage Mining: Grundlagen und Verfahren*. PhD thesis, School of Computer Science, University of Magdeburg, 2007.

[LNT$^+$11] Philipp Leitner, Franco Maria Nardini, Gabriele Tolomei, Schahram Dustdar, and Fabrizio Silvestri. Mining lifecycle event logs for enhancing service-based applications. Technical report, ISTI-CNR, May 2011.

[LWR+09] Philipp Leitner, Branimir Wetzstein, Florian Rosenberg, Anton Michlmayr, Schahram Dustdar, and Frank Leymann. Runtime prediction of service level agreement violations for composite services. In *ICSOC/ServiceWave Workshops*, pages 176–186, 2009.

[MRLD09] Anton Michlmayr, Florian Rosenberg, Philipp Leitner, and Schahram Dustdar. Comprehensive QoS monitoring of web services and event-based SLA violation detection. In *Proceedings of the 4th International Workshop on Middleware for Service Oriented Computing*, MWSOC '09, pages 1–6, New York, NY, USA, 2009. ACM.

[MRLD10] Anton Michlmayr, Florian Rosenberg, Philipp Leitner, and Schahram Dustdar. Selective service provenance in the vresco runtime. *Int. J. Web Service Res.*, 7(2):65–86, 2010.

[MZD09] Emmanuel Mulo, Uwe Zdun, and Schahram Dustdar. Monitoring web service event trails for business compliance. In *SOCA*, pages 1–8. IEEE, 2009.

[SMS+02] Akhil Sahai, Vijay Machiraju, Mehmet Sayal, Aad P. A. van Moorsel, and Fabio Casati. Automated SLA monitoring for web services. In Metin Feridun, Peter G. Kropf, and Gilbert Babin, editors, *DSOM*, volume 2506 of *Lecture Notes in Computer Science*, pages 28–41. Springer, 2002.

[TTM08] A. Tiwari, C. J. Turner, and B. Majeed. A review of business process mining: state-of-the-art and future trends. *Business Process Management Journal*, 14(1):5–22, 2008.

[ULMS08] Tobias Unger, Frank Leymann, Stephanie Mauchart, and Thorsten Scheibler. Aggregation of service level agreements in the context of business processes. In *EDOC*, pages 43–52. IEEE Computer Society, 2008.

[vdA07] Wil M. P. van der Aalst. Challenges in business process analysis. In Joaquim Filipe, José Cordeiro, and Jorge Cardoso, editors, *ICEIS (Selected Papers)*, volume 12 of *Lecture Notes in Business Information Processing*, pages 27–42. Springer, 2007.

[vdAPS10] Wil M. P. van der Aalst, Maja Pesic, and Minseok Song. Beyond process mining: From the past to present and future. In Barbara Pernici, editor, *CAiSE*, volume 6051 of *Lecture Notes in Computer Science*, pages 38–52. Springer, 2010.

[vdARV+10] Wil M. P. van der Aalst, Vladimir Rubin, H. M. W. Verbeek, Boudewijn F. van Dongen, Ekkart Kindler, and Christian W. Günther. Process mining: a two-step approach to balance between underfitting and overfitting. *Software and System Modeling*, 9(1):87–111, 2010.

[vdASS11] Wil M. P. van der Aalst, M. H. Schonenberg, and Minseok Song. Time prediction based on process mining. *Inf. Syst.*, 36(2):450–475, 2011.

[vdAvDG+09] Wil M. P. van der Aalst, Boudewijn F. van Dongen, Christian W. Günther, Anne Rozinat, Eric Verbeek, and Ton Weijters. Prom: The process mining toolkit. In Ana Karla A. de Medeiros and Barbara Weber, editors, *BPM (Demos)*, volume 489 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2009.

[vdAvDH+03] Wil M. P. van der Aalst, Boudewijn F. van Dongen, Joachim Herbst, Laura Maruster, Guido Schimm, and A. J. M. M. Weijters. Workflow mining: A survey of issues and approaches. *Data Knowl. Eng.*, 47(2):237–267, 2003.

[vdAW04] W. M. P. van der Aalst and A. J. M. M. Weijters. Process mining: a research agenda. *Comput. Ind.*, 53:231–244, April 2004.

[vdAWM04] Wil M. P. van der Aalst, Ton Weijters, and Laura Maruster. Workflow mining: Discovering process models from event logs. *IEEE Trans. Knowl. Data Eng.*, 16(9):1128–1142, 2004.

[vDCvdA08] Boudewijn F. van Dongen, R. A. Crooy, and Wil M. P. van der Aalst. Cycle time prediction: When will this case finally be finished? In Robert Meersman and Zahir Tari, editors, *OTM Conferences (1)*, volume 5331 of *Lecture Notes in Computer Science*, pages 319–336. Springer, 2008.

[vDvdA05] Boudewijn F. van Dongen and Wil M. P. van der Aalst. A meta model for process mining data. In Michele Missikoff and Antonio De Nicola, editors, *EMOI-INTEROP*, volume 160 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2005.

[vdWvDHS09] Jan Martijn E. M. van der Werf, Boudewijn F. van Dongen, Cor A. J. Hurkens, and Alexander Serebrenik. Process discovery using integer linear programming. *Fundam. Inform.*, 94(3-4):387–412, 2009.

[vG04] M. van Giessel. Process mining in SAP R/3: A method for applying process mining to SAP R/3. Master's thesis, Technische Universiteit Eindhoven, 2004.

[WSL09] Branimir Wetzstein, Steve Strauch, and Frank Leymann. Measuring performance metrics of WS-BPEL service compositions. In Jaime Lloret Mauri, Vicente Casares Giner, Rafael Tomas, Tomeu Serra, and Oana Dini, editors, *ICNS*, pages 49–56. IEEE Computer Society, 2009.

[WvdA03] A. J. M. M. Weijters and W. M. P. van der Aalst. Rediscovering workflow models from event-based data using little thumb. *Integr. Comput.-Aided Eng.*, 10:151–162, April 2003.

[ZLC07] Liangzhao Zeng, Hui Lei, and Henry Chang. Monitoring the qos for web services. In Bernd J. Krämer, Kwei-Jay Lin, and Priya Narasimhan, editors, *ICSOC*, volume 4749 of *Lecture Notes in Computer Science*, pages 132–144. Springer, 2007.

[ZLLC08] Liangzhao Zeng, Christoph Lingenfelder, Hui Lei, and Henry Chang. Event-driven quality of service prediction. In Athman Bouguettaya, Ingolf Krüger, and Tiziana Margaria, editors, *ICSOC*, volume 5364 of *Lecture Notes in Computer Science*, pages 147–161, 2008.

[zMR00] Michael zur Muehlen and Michael Rosemann. Workflow-based process monitoring and controlling - technical and organizational issues. In *HICSS*, 2000.

Chapter 3

Conclusions

In this deliverable, the S-Cube partners have presented the "delta" in the state of the art that has occurred in the topics covered by the NoE since year 2008. The updates take into consideration research contributions resulting from S-Cube as well as the research community at large.

As S-Cube approaches its completion, the partners are working together to envision the future research beyond the NoE. The upcoming IA-1.1.8 deliverable will present new research challenges that build on top of what S-Cube has achieved. Moreover, S-Cube partners are organizing a workshop entitled "S-Cube European Software Services and Systems Research – Results and Challenges." Co-located with the 34^{th} International Conference in Software Engineering (ICSE 2012), the workshop will take place on June 5, 2012 with the goal of presenting the findings of the S-Cube NoE and discussing the future challenges that the S-Cube research community will address in the coming years.