Grant Agreement N° 215483

Title: *Validated set of adaptation and monitoring principles, techniques and methodologies considering context and HCI*

Authors: *CITY, FBK, POLIMI, SZTAKI, USTUTT, TUW, TILBURG, UNIDUE, UCBL*

Editor: *Gabor Kecskemeti (SZTAKI)*

Reviewers: *Harald Psaier (TUW)*
Claudia Di Napoli (CNR)
Annapaola Marconi (FBK)
Andreas Metzger (UNIDUE)

Identifier: *Deliverable # CD-JRA-1.2.7*

Type: *Deliverable*

Version: *1.0*

Date: *February 29, 2012*

Status: *Final*

Class: *External*

Management summary

This deliverable aims to present the research progress of the project partners in the project's last year. First, this deliverable puts special emphasis on the further refinements of the comprehensive adaptation and monitoring scenarios introduced in CD-JRA-1.2.5. Next, the research summarized in this document was focusing on the unexpected situations that could occur in cross-layer adaptation and monitoring techniques while they are maintaining context independent and HCI aware execution of large scale and heavily distributed service-based applications. The research results are presented through systematically listing and detailing the related research papers of the project partners. Finally, the deliverable offers an outlook on the future research directions in the field of adaptation and monitoring frameworks.

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 todays 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/`

Foreword

This deliverable, CD-JRA-1.2.7 “Validated set of adaptation and monitoring principles, techniques and methodologies considering context and HCI” aims to address the following goals:

- to continue, refine and consolidate the monitoring and adaptation scenarios defined in CD-JRA-1.2.5, so they could encompass a contextual changes and HCI awareness while also reveal opportunities for handling unexpected situations in SBAs.
- through the summaries of the joint papers, present the research results and experiences on the realization and experimentation efforts while concretizing the previously defined scenarios.
- to identify future research directions derived from the research experiences in the current research domains of the different S-Cube partners that lead us towards the vision of the project.

Contents

Chapter 1

Introduction

In highly dynamic service environments, where the service availability, quality and reliability is dependent on several application-independent factors, the behavior and user experience about Service-Based Applications (SBAs) are highly dependent on the monitoring and adaptation techniques utilized during application execution. The different layers (e.g. BPM, SCC) of the SBAs however can be monitored and adapted with different approaches. These layers earlier were individually addressed, and with our previous deliverable we have shown several adaptation and monitoring scenarios that address the locality issues of the adaptation strategies (e.g., adaptation on the SCC layer could break BPM layer behavior of the SBA). Therefore the work package has explored different and orthogonal directions of integration of adaptation methodologies. These methodologies were first introduced in CD-JRA-1.2.3, then integrated in CD-JRA-1.2.4, finally adapted for a set of integration show cases in CD-JRA-1.2.5. In alignment with T-JRA-1.2.2 ("*Integrated Adaptation Principles, Techniques and Methodologies*"), the current deliverable validates, consolidates and, if needed, refines the show case scenarios with jointly written research papers and it provides an overview of the future research directions that lead towards sustainable research among the work package members even after the end of the project.

In several service-based applications, e.g., in the telecommunications domains, services need to interact heavily with humans, and it is important to adapt the interactions to different kinds of users and to different situations. This kind of adaptation should take into account different types of users (from developers to end-users), different levels of expertise, as well as different contexts of usage (e.g., connectivity, available devices, etc.). The contributed research papers in the deliverable provide the foundations for cross-layer, context-aware and user-driven monitoring, i.e., theories, principles, methodologies, and techniques that use contextual information and information about the user of monitoring results to drive monitoring activities. The deliverable – completing research task T-JRA-1.2.3 ("*Comprehensive, Context-Aware Monitoring*") – also reveals how the project members adapted the show case scenarios in order to handle the previously mentioned dynamically changing contexts and HCI environments.

This deliverable also aims at revealing the findings of the project partners about the principles, techniques and methodologies for proactive adaptation, i.e., to timely anticipate the need for changes across different functional layers. This deliverable reviews those joint scientific efforts that discuss the homogenization of proactive adaptation mechanisms in the different service-based application layers that experience different degrees of autonomy. The proactive adaptation capabilities discussed in the contributed research papers are also considering techniques to detect and handle unexpected situations, context awareness and user-driven adaptation taking into account different context factors, including those related to human errors or quality aspects of the operational environment of the service-based application (a strongly related research issue with work package JRA-1.3).

1.1 Relations with the Integrated Research Framework

1.1.1 Contribution to WP Challenges

The scope and the results of the deliverable directly contribute to the research challenges of the WP-JRA-1.2 and the S-Cube Integrated Research Framework. Specifically, this deliverable addresses the following challenges:

Comprehensive and integrated adaptation and monitoring principles, techniques, and methodologies. The research results focused on the scenario "*Quality-driven Multilayer SBA Monitoring and Adaptation*" are related to this challenge in sections 2.2.2, 2.2.4, 2.2.5, 2.2.7 and 2.2.9.

Proactive Adaptation and Predictive Monitoring. The scenario titled "*Assumption-based Proactive Monitoring and Adaptation*" focuses on this challenge, and it is detailed in several novel research papers that are detailed in sections 2.2.6 and 2.2.9.

Context- and HCI-aware SBA monitoring and adaptation. Those newly introduced research papers that are related to the scenario titled "*Context-based Adaptation and Monitoring*" address this challenge. See sections 2.2.1, 2.2.3 and 2.2.7.

The scenarios referred in the challenges listing were defined in CD-JRA-1.2.5 and are shortly summarized in Section 2.1.

1.1.2 Relations with other work-packages

As with previous research results, the crosscutting nature of the topic of the work package is represented in several joint research opportunities that were exploited during the last year of the project. The deliverable in several cases presents joint research papers that were collaboratively produced among the various work packages of the project. In the following, we list the most prominent connections of the research results presented in this deliverable:

Connections with JRA-1.1 Work package JRA-1.1 is focusing on the software engineering aspects of SBAs. The HCI and context aware automated adaptation of the monitoring system represents a new software engineering challenge that was collaboratively investigated between the JRA-1.1 and JRA-1.2. These collaborative efforts are reflected in section 2.2.1. The analysis and then the classification of the various BPEL extensions in section 2.2.8 reveals the effects on methodologies considering adaptation-aware business process design.

Connections with JRA-1.3. This work package aims at the quality characteristics of the SBAs. First, in section 2.2.5, we present how service quality aware autonomous behavior of federated cloud infrastructures supporting service-based applications could reduce SLA violations. Later on, in section 2.2.9, we propose the use of fuzzy rules in the field of proactive adaptation and monitoring in order to predict and reduce QoS dissatisfaction in service-based applications.

Connections with JRA-2.1. This work package targets the issues that occur in the Business Processes Layer during the execution of service based applications. In sections 2.2.2, 2.2.4 we investigate various aspects of the cross layer behavior of monitoring and adaptation frameworks. These works present A&M approaches that not only allow the BPM layer to get notifications from layers below, but we also present methodologies and techniques that interactively involve the BPM layer in selecting and enacting efficient adaptation strategies meeting all the requirements of the involved service-based application layers.

Connections with JRA-2.2. The work package JRA-2.2 circles around the research questions of service compositions and coordination. In this deliverable, several contributed research papers address context aware adaptation of service compositions, thus they were developed with strong collaboration with JRA-2.2 (e.g. see sections 2.2.1 or 2.2.3). In section 2.2.7, we propose service compositions deployed and dynamically managed over the Amazon EC2 cloud infrastructure, this work has involved also JRA-2.3 along with JRA-2.2.

Connections with JRA-2.3. Self-* service infrastructures and various discovery mechanisms are the main concerns of this work package. Thus, the topic of utilizing service-based applications on infrastructure as a service cloud computing systems has been discussed in two sections: 2.2.5 and 2.2.7. Building on top of these solutions the CLAM (cross layer adaptation and monitoring) framework is proposed (see sections 2.2.2 and 2.2.4) allowing the business and composition layers of service-based applications to consider infrastructure level events and adaptation options in unexpected situations.

1.2 Deliverable Structure

This document aims to present our achievements regarding the validated set of adaptation and monitoring principles, techniques and methodologies considering context and HCI. This deliverable is structured as follows:

- First, chapter 2 circles around the scientific papers we have written in collaboration aiming at the deliverable's target objectives. These papers are attached to the deliverable, and therefore this deliverable only provides a structured view on them. This chapter is subdivided into sections as follows:
  - Afterwards, in section 2.1, we define the way the research results of the S-Cube partners will be presented. Throughout the section, we shortly summarize the relevant content of the previous deliverables as a result, reading the current deliverable will not need an extensive knowledge about past results.
  - Next, in section 2.2, we provide a listing of the various research papers contributed to this deliverable. This listing not only provides the tables describing the individual contibutions, but it also provides brief descriptions of the research ideas that initiate the integration of monitoring and adaptation principles across service-based application layers. We remark that the results presented in this chapter are based on the materials presented in a set of papers that are referred from and attached to the current deliverable only.
- Then, in chapter 3, concludes the deliverable and the last year of work of the S-Cube project in the scope of adaptation and monitoring approaches.
- Finally, in the Appendix, the contributed papers are attached to the document and reported as separate chapters.

Chapter 2

Individual contributions

The goal of this chapter is to provide an overall view of the research carried out during the last year of the work package. The chapter presents the results of the work package through the latest research papers that present the improved and consolidated validation results and methodologies even considering unexpected situations. The chapter first provides a schematic overview about how each research paper will be presented (see section 2.1). Then, based on this schema, in section 2.2 we will present and align the novel approaches developed by the S-Cube partners that aim at addressing and supporting the context and HCI aware integrated cross-layer adaptation and monitoring.

2.1 Template for presenting the results

In order to provide a synthetic overview of the contributions of the partners and to relate them to the cross-layer adaptation & monitoring principles, techniques, scenarios and methodologies presented in CD-JRA-1.2.5, a synthetic template is proposed (Table 2.2). Through the template we aim at presenting the research contributions of the various S-Cube members to the overall research results and future research directions of the project in relation with work package JRA-1.2.

For each contributed research paper, the template in Table 2.2 is filled out considering four major areas.

- First, it describes the paper in general.
- Then, it puts the paper in context of the research executed in the work package.
- Afterwards, it dives into the details of the contributed research paper.
- Finally, the template table is followed by the extended abstract of the paper that describes the concepts, the research questions and highlights the proposed solutions.

In the general description, first, the *title* and the *authors* of the article is specified. The authors field not only lists the researchers contributing to the particular paper but it also highlights if the paper was jointly developed by two or more project members. Next, the authors are followed by the *type* and a *short description* of the research presented in the filled in template. The type can indicate if a new *methodology* is introduced for a particular adaptation and monitoring related domain. Also, the type can reveal if a new *model* is proposed to describe the A&M behavior inside service-based applications. Then, the research paper could also contribute with a new *technique* applicable in cross-layer A&M that is also highlighted as a new type. In the type field, the template could also highlight the inclusion of *experimental evaluation* in the described research paper. Finally, the *classification* type of contribution reveals that the particular research carried out has provided a systematic evaluation and provided a taxonomic discussion of the research field.

The work package related contextual description in the template starts with the identification of the targeted *integration scenario*. The integration scenarios were introduced in the previous deliverable of this work package (titled "*CD-JRA-1.2.5 – Comprehensive, integrated adaptation and monitoring* *principles, techniques and methodologies across functional SBA layers considering context and HCI*") to provide a common reference for the adaptation and monitoring problem in hand, providing a common basis for the different research works developed within the project. In the following we provide a short overview of the three identified scenarios:

Quality-driven Multilayer SBA Monitoring and Adaptation. The SBA reference model in this scenario primarily targets applications implemented as long-running business processes and workflows. The primary adaptation and monitoring problem in this scenario considers the quality requirements expressed as KPI, PPM, SLA, and other metrics of the application, across its functional layers. The scenario exploits data mining techniques to perform the diagnosis of the problem that leads to the violation of the high-level quality requirements. Regarding the individual quality properties at different layers, the scenario associates different adaptation actions (also in that layer) that are expected to improve the specific quality factor and, therefore, contribute to an overall improvement in quality.

Assumption-based Proactive Monitoring and Adaptation. In this scenario we consider business processes that are realized on top of executable service compositions implemented in BPEL. The idea that distinguishes this approach in this integrated scenario is based on the use of assumptions. This scenario uses assumptions to relate the continuously monitored data to the SBA requirements. The scope of the scenario is to be able to anticipate the needs for adaptation and to provide the corresponding monitoring and adaptation support in order to enable proactive adaptation of service-based business processes in case of failures and requirement violations.

Context-based Adaptation and Monitoring. In this scenario we primarily focus on SBAs, in which the context of those SBA plays the key role in the various activities across the SBA life-cycle.

Title	the title of the research work
Authors	Authors of the contribution
Type	Type of contribution (Methodology / Model / Technique / Classification / Experimental Evaluation / ..)
Short description	Brief description of the contribution with respect to the research problem presented in the integration scenario
Targeted integration scenario	One or more scenarios from the previous deliverable, namely: (i) Quality-driven Multilayer SBA Monitoring and Adaptation, (ii) Assumption-based Proactive Monitoring and Adaptation, (iii) Context-based Adaptation and Monitoring.
Contribution to the adaptation problem	Specific adaptation problem addressed by the approach (if any)
Contribution to the monitoring problem	Specific monitoring problem addressed by the approach (if any)
Integrated SBA Layers for Monitoring	Any combination of monitoring across BPM, SCC, SI layers that the research work has provided a consolidated approach for.
Integrated SBA layers for adaptation	Any combination of adaptation solutions through BPM, SCC, SI layers that the contributed article has aimed at.
Cross-layer mechanisms	Mechanisms applied for interaction between SBA layers involved
Unexpected situations handled	List of unexpected situations identified and handled by the approach
Architecture elements	Specify and refine the relevant components of the architecture
Requirements/constraints	The important assumptions and limitations of the current approach. These might be referred in the future research directions (e.g. possible ways to reduce limitations)
Refined SBA Life Cycle activities	Which activities of the SBA lifecycle were affected and updated by the current research paper
Future research directions	List the the remaining research issues that needs to be addressed in the topic of the actual scientific result

Table 2.2: Contribution overview template

The distinguishing factor of the SBAs participating in this scenario refers to the fact that these SBAs operate in continuously changing environments. This integrated scenario relies on the use of templates that characterize the monitoring and adaptation activities in general settings, which are then instantiated in different way for the specific contexts.

The rest of the WP related contextual description is focused on the particular paper's relations and effects on the monitoring and adaptation problem: (i) *Contribution to the adaptation problem*, (ii) *Contribution to the monitoring problem*, (iii) *Integrated SBA Layers for Monitoring*, (iv) *Integrated SBA layers for adaptation*, (v) *Cross-layer mechanisms*.

Finally, the last four rows of the template detail the paper and underline its strongest contributions to the work package. Therefore, for each paper, we have described the *unexpected situations* that the research identified and proposed solutions for. In the following row, the template lists the *architecture elements* to provide insights to architectures that could handle the previously identified problems and unexpected behavior. Afterwards, we have listed those *requirements and/or constraints* that need to be met in order to allow the correct operation of the proposed approach. Next, the table summarizes the *SBA life-cycle elements and activities* (first defined by the work package JRA-1.1 in [3]) that were refined while carrying out the research described in the paper. In the last row, for every contributed paper, the table summarizes the future research directions that S-Cube researchers identified for sustainable and continued research in the area of monitoring and adaptation of service-based applications.

2.2 Summary of the individual contributions to the integrated cross-layer adaptation and monitoring principles

2.2.1 Identifying, Modifying, Creating, and Removing Monitor Rules for Service Oriented Computing

Title	Identifying, Modifying, Creating, and Removing Monitor Rules for Service Oriented Computing [4]
Authors	Ricardo Contreras (CITY), Andrea Zisman (CITY)
Type	Technique
Short description	Supports the adaptation of the monitor activity due to changes in the context of the user of the SBA (context HCI-aware)
Targeted integration scenario	Context-based Adaptation and Monitoring.
Contribution to the adaptation problem	The technique can be combined with techniques/approaches capable of dealing with SBA adaptation
Contribution to the monitoring problem	Automatic specification and deployment of monitor rules. This is performed based on the user context and the specification of the SBA
Integrated SBA Layers for Monitoring	Focus on SCC
Integrated SBA layers for adaptation	Focus on SCC
Cross-layer mechanisms	-
Unexpected situations handled	Optimisation: monitor rules repository kept to a minimum size
Architecture elements	**Rule Adaptor:** identifies, creates, modifies and removes monitor rules. **Path Identifier:** retrieves the part of the specification related to a user context type. **Rule Verifier:** whether a rule is valid for a SBA or not. Monitor: uses rules to verify the SBA

Requirements/constraints	(i) Uses event calculus (EC) for the specification of rules. (ii) Use of patterns (also specified in EC) for the different user context types. (iii) Assumes the existence of user context annotations to retrieve the part of the specification related to a particular context.
Refined SBA Life Cycle activities	**Operation and Management:** monitoring mechanisms for the identification of problems. **Requirements Engineering and Design:** context types present in the SBA (annotations)
Future research directions	To investigate new techniques dealing with the (semi-) automatic adaptation of the monitor as a reaction to changes/adaptation in the application.

Extended abstract

Monitoring of service-based systems is considered an important activity to support service-oriented computing. Monitoring can be used to verify the behaviour of a service-based system, and the quality and contextual aspects of the services participating in the system. Existing approaches for monitoring service-based systems assume that monitor rules are pre-defined and known in advance, which is not always the case. We have created a pattern-based HCI-aware monitor adaptation framework to support identification, modification, creation, and removal of monitor rules. In the framework, changes in the monitor rules are based on users interaction with a service-based system and different types of user context.

2.2.2 CLAM (Cross-layer Adaptation manager)

Title	CLAM (Cross-layer Adaptation manager) [5]
Authors	Annapaola Marconi (FBK), Marco Pistore (FBK), Asli Zengin (FBK), Luciano Baresi (Polimi)
Type	Methodology / Model / Technique
Short description	CLAM (i) analyzes the effects and consequences of an adaptation trigger for the whole service-based system, (ii) addresses the negative influences on the system through a gradual construction of adaptation strategies.
Targeted integration scenario	Quality-driven Multilayer SBA Monitoring and Adaptation
Contribution to the adaptation problem	Handling of cross-layer adaptations
Contribution to the monitoring problem	–
Integrated SBA Layers for Monitoring	–
Integrated SBA layers for adaptation	BPM, SCC, SI
Cross-layer mechanisms	State-of-the-art analysis and adaptation techniques are integrated at the CLAM platform. CLAM provides a cross-layer managing mechanism to coordinate these techniques.
Unexpected situations handled	Searching for existing approaches and finding the running adaptation tools is not trivial. We try to handle this issue by searching through what S-Cube partners can provide and by directly contacting the authors / developers of the tools.

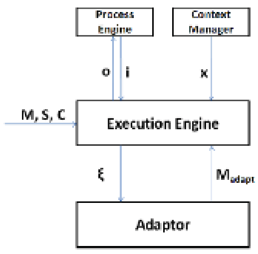
Architecture elements	Rule engine, model updater, tree constructor, ranker: • The core part of CLAM is the rule engine. It reasons on its predefined rules and contacts the appropriate tool at each step of its analysis. • The model updater is responsible for creating and keeping the updated SBS configurations each time an adaptation is proposed in the process. • The tree constructor keeps the proposed adaptation alternatives in a tree-form structure where each tree branch corresponds to an alternative adaptation strategy. • The ranker gets the complete tree and select the best path based on some pre-defined selection criteria.
Requirements/constraints	(i) High level cross-layer dependency model of the SBS should be given as input to the CLAM. (ii) Each time a new tool is plugged in the platform, cross-layer rules should be updated. (iii) The performance of CLAM depends on the integrated tools. The more tools CLAM has, the more comprehensive will be the analysis, but it will take more time as well.
Refined SBA Life Cycle activities	All parts of the SBA life cycle activities
Future research directions	In the current implementation we showed the CLAM approach on a case study. Future research will comprise the enhancement and evaluation of the approach through the trial of various case studies from different application scenarios. The use of CLAM in different adaptation cases and in different domains and possibly with different sets of tools will demonstrate the applicability, extensibility and flexibility of the approach.

Extended abstract

Adaptation is a cross- and multi-layer problem, and at each layer it could target different, possibly conflicting, system aspects. For example, when reorganizing the composition, one could privilege the application's price, its speed, or the compliance with some external regulations. Many existing solutions have addressed adaptation in a "local" way by only considering one system aspect at one layer; in contrast CLAM fosters a comprehensive approach able to address different layers and aspects concurrently, reason on the dependencies and consequences among them, and identify global solutions. These solutions must harmonize layers and system aspects, and provide an integrated adaptation plan based on local activities. CLAM relies on a comprehensive high-level model of the application and of the layers behind it. Each model element is associated with a set of analyzers to understand the problem, solvers, to identify possible solutions, and enactors, to apply them on the element. The coordinated operation of analyzers, solvers, and enactors is governed by predefined rules that identify the dependencies, and consequences, between the elements of the model and run the different tools. For each adaptation need, CLAM produces a tree of alternative adaptations, identifies the most convenient one, and applies it.

2.2.3 Adaptation of Service-based Business Processes by Context-Aware Replanning

Title	Adaptation of Service-based Business Processes by Context-Aware Replanning [6]
Authors	Antonio Bucchiarone (FBK), Raman Kazhamiakin (SayService), Marco Pistore (FBK), Heorhi Raik (FBK)
Type	Methodology / Technique
Short description	Adaptation of business processes to exogenous context changes and negative operation outcomes which are not handled by the process by the run-time and context-aware planning of the adaptation activities
Targeted integration scenario	Context-based Adaptation and Monitoring

Contribution to the adaptation problem	Run-time adaptation of business processes to exogenous context changes and negative operation outcomes unhandled by the process
Contribution to the monitoring problem	–
Integrated SBA Layers for Monitoring	–
Integrated SBA layers for adaptation	SCC
Cross-layer mechanisms	–
Unexpected situations handled	exogenous context changes and negative operation outcomes that are not handled by the process
Architecture elements	The architecture contains four elements communicating to each other: 1. Execution engine coordinates the work of other elements and tracks the execution of the process; 2. Process engine provides the execution of process activities; 3. Context manager tracks exogenous context changes; 4. Adaptor, given an adaptation problem, uses a planner to derive an adaptation procedure; 
Requirements/constraints	(i) Only one adaptation strategy is used (local adaptation that tries to change the problem locally, so that the process can be executed from the point where it has failed); (ii) The approach is supposed to be used as short-term adaptation, i.e. instance based adaptation versus process model evolution; (iii) The adaptation process optimality criterion is the just minimal number of execution steps;
Refined SBA Life Cycle activities	–
Future research directions	The approach can benefit from using different adaptation strategies (for example, jump forward or roll back and compensate strategies). The approach can be successfully used to refine abstract activities at run time. In this case some most volatile activities remain abstract and are refined only immediately before the execution, taking into account the current context. The adaptation history can be successfully used to bring corresponding changes to the process model (process evolution). While deriving the adaptation activities, different optimality criteria can be used.

Extended abstract

Service-based business processes are typically used by organizations to achieve business goals through the coordinated execution of a set of activities implemented as services and service compositions. Since they are executed in dynamic, open and non-deterministic environments, business processes often need to be adapted to exogenous context changes and execution problems. In this paper we provide an adaptation approach that can automatically adapt business processes to run-time context changes that impede achievement of a business goal. We define a formal framework that adopts planning techniques to automatically derive necessary adaptation activities on demand. The adaptation consists in identifying recovery activities that guarantee that the execution of a business process can be successfully resumed and, as a consequence, the business goals are achieved. The solution proposed is evaluated on a real-world scenario from the logistics domain.

2.2.4 Multi-layer Monitoring and Adaptation

Title	Multi-layer Monitoring and Adaptation [7]
Authors	Sam Gunea (Polimi), Gabor Kecskemeti (SZTAKI), Annapaola Marconi (FBK), and Branimir Wetzstein (USTUTT)
Type	Methodology / Technique
Short description	We propose a framework that integrates layer specific monitoring and adaptation techniques, and enables multi-layered control loops in service-based systems. 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.
Targeted integration scenario	Quality-driven Multilayer SBA Monitoring and Adaptation
Contribution to the adaptation problem	Multi-layer adaptation of service based systems: identification and enactment of holistic adaptation strategies coordinating layer-specific adaptation mechanisms.
Contribution to the monitoring problem	Multi-layer monitoring and analysis of service-based systems: correlation (as general and domain-specific metrics) of monitoring events captured by layer-specific sensors, and holistic identification of the adaptation need through the analysis of the aggregated data.
Integrated SBA Layers for Monitoring	BPM, SCC, SI
Integrated SBA layers for adaptation	BPM, SCC, SI
Cross-layer mechanisms	Cross-layer correlation of monitoring events, cross-layer analysis of adaptation needs, cross-layer identification of adaptation strategies, cross-layer adaptation enactment.
Unexpected situations handled	Integration of heterogeneous, layer-specific monitoring and adaptation techniques.
Architecture elements	The main architectural elements of the framework are: **Monitoring and Correlation Component:** obtains low-level data/events from the process or from the context of execution using Dynamo, or from the infrastructure using Laysi; aggregates the monitoring events using the event correlation capabilities provided by EcoWare. **Adaptation Needs Analysis Component:** the Influential Factor Analysis component identifies the relations between the set of metrics (potential influential factors) and the KPI category based on historical process instances; the Adaptation Needs Analysis component uses this information to identify the adaptation needs, i.e., what is to be adapted in order to improve the KPI. **Cross Layer Adaptation Manager Component:** identifies the application components that are affected by the adaptation actions, and computes an adaptation strategy that properly coordinates the layer-specific adaptation capabilities. **Adaptation Enactment component:** enacts the adaptation strategy properly coordinating software adaptations through DyBPEL, and infrastructure adaptations through Laysi.
Requirements/constraints	Business layer not fully covered (lack of monitoring/adaptation capabilities to be integrated in the framework).
Refined SBA Life Cycle activities	Operation & Management, Identify adaptation need, Identify adaptation strategy, Enact adaptation.

Future research directions	We will continue to evaluate the approach through new application scenarios, and through the addition of new adaptation capabilities and adaptation enacting techniques. We will also integrate additional kinds of layers, such as platforms, typically seen in cloud computing setups, and business layers. This will also require the development of new specialized monitors and adaptations. Finally, we will study the feasibility of managing different kinds of KPI constraints.

Extended abstract

We propose a framework that integrates software and infrastructure specific monitoring and adaptation techniques developed within S-Cube, enabling multi-layered control loops in service-based systems. All the steps in the control loop acknowledge the multi-faceted nature of the system, ensuring that we always reason holistically, and adapt the system in a coordinated fashion. In our prototype we have focused on the monitoring and adaptation of BPEL processes that are deployed onto a dynamic infrastructure.

Building upon our past experiences we have integrated process and infrastructure level monitoring with a correlation technique that makes use of complex event processing. The correlated data, combined with machine-learning techniques, allow us to pinpoint where the problems lie in the multi-layered system, and where it would be more convenient to adapt. We then build a complex adaptation strategy that may involve the software and/or the infrastructure layer, and enact it through appropriate effectors.

In the Monitoring and Correlation step, sensors deployed throughout the system capture run-time data about its software and infrastructural elements. The collected data are then aggregated and manipulated to produce higher-level correlated data under the form of general and domain-specific metrics. The main goal is to reveal correlations between what is being observed at the software and at the infrastructure layer to enable global system reasoning.

In the Analysis of Adaptation Needs step, the framework uses the correlated data to identify anomalous situations, and to pinpoint and formalize where it needs to adapt. It may be sufficient to adapt at the software or at the infrastructure layer, or we may have to adapt at both.

In the Identification of Multi-layer Adaptation Strategies step, the framework is aware of the adaptation capabilities that exist within the system. It uses this knowledge to define a multi-layer adaptation strategy as a set of software and/or infrastructure adaptation actions to enact. A strategy determines both the order of these actions and the data they need to exchange to accomplish their goals.

In the Adaptation Enactment step, different adaptation engines, both at the software and the infrastructure layer, enact their corresponding parts of the multi-layer strategy. Each engine typically contains a number of specific modules targeting different atomic adaptation capabilities.

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.

2.2.5 Facilitating self-adaptable Inter-Cloud management

Title	Facilitating self-adaptable Inter-Cloud management [8]
Authors	Gabor Kecskemeti (SZTAKI), Michael Maurer (TUW), Ivona Brandic (TUW), Attila Kertesz (SZTAKI), Zsolt Nemeth (SZTAKI), Schahram Dustdar (TUW)
Type	Methodology
Short description	This contribution introduces a methodology to autonomously operate cloud federations by controlling their behavior with the help of knowledge management systems. Such systems do not only suggest reactive actions to comply with established Service Level Agreements (SLA) between provider and consumer, but they also find a balance between the fulfillment of established SLAs and efficient energy consumption.

Targeted integration scenario	Quality-driven Multilayer SBA Monitoring and Adaptation
Contribution to the adaptation problem	This contribution offers three options to incorporate the concepts of knowledge management systems into the Federated Cloud Management architecture to perform adaptation: (i) local integration is applied on a per deployed component basis, e.g. every CloudBroker utilizes a separate KM system for its internal purposes; (ii) global integration is based on a single KM system that controls the autonomous behavior of the architectural components considering the available information from the entire cloud federation; and (iii) a hybrid KM system combining both global and local integration options.
Contribution to the monitoring problem	Within analyzing the various autonomous actions that the KM system can exercise, the authors investigated the monitoring system of the FCM architecture and the possible metrics it can collect to allow the identification of those cases when the architecture encounters unsatisfactory behavior. In the current system, they monitor and analyze the behavior of CloudBrokers, the FCM repository and individual service instances.
Integrated SBA Layers for Monitoring	SI.
Integrated SBA layers for adaptation	SI.
Cross-layer mechanisms	–
Unexpected situations handled	The proposed approach for Inter-Cloud management by FCM is capable of handling various unexpected situations based on its knowledge management solution with extendible adaptation rules. In this contribution, the following actions can be triggered for unexpected situations: Reschedule of service calls, Rearrange VM queues, Extend/Shrink VM Queue, Rearrange VA storage , Self-Instantiated Deployment.
Architecture elements	**Monitors:** Various metrics have been defined in the KM system that are used in the rules for triggering actions. **Adaptation strategy engine:** The actions of the KM represent the strategies in this contribution. **Adaptation enactment engine:** the Global Autonomous Manager incorporating the Knowledge Management System is responsible for executing the actions. **Adaptation capabilities:** These capabilities are predefined in the extensible set of rules within the KM system.
Requirements/constraints	The proposed Self-adaptable Inter-Cloud management solution adopting rule-based techniques for its knowledge management to federate clouds of multiple infrastructures. In the future additional rules will be defined to cover a wider range of system failures or malfunctions.
Refined SBA Life Cycle activities	**Construction:** Developers may provide additional adaptation actions to the Knowledge Management System of the Global Autonomous Manager **Operation, management and QA:** The Global Autonomous Manager continuously execute the predefined rules to perform autonomous control **Identify adaptation requirements:** The rules of the KM system specify this process. **Identify adaptation strategy:** The actions to be triggered according to the monitored events are predefined in the rules of the KM system **Deployment and provisioning:** Service deployments of the Self-adaptable Inter-Cloud management architecture are managed by the FCM components following the predefined rules of the KM system. The paper describes its operation by detailing the rule for removing VA from native repository of a specific Cloud due to high global costs.

Future research directions	In this contribution various metrics have been defined to indicate possible SLA violations in federations, and rules have been developed to trigger adaptation actions in the case of predicted violations. Regarding future works, the authors plan to investigate more the green aspects in the autonomous behavior of cloud federations. They will also aim at defining new rules for advanced action triggers and evaluate the applicability of case based reasoning. An experimental system will also be set up to investigate the effects of the autonomous behavior on the overall performance of the cloud federation.

Extended abstract

Cloud Computing represent a novel computing paradigm where computing resources are provided on demand following the rules established in form of Service Level Agreements (SLAs). SLAs represent the popular formats for the establishment of electronic contract between consumer and provider stating the terms of use, objectives and penalties to be paid in case objectives are violated. Thus, appropriate management of Cloud Computing infrastructures (such as Amazon, Rackspace, Eucalyptus, Opennebula) is the key issue for the success of Cloud Computing as the next generation ICT infrastructure. Thereby, the interaction of the system with humans should be minimized while established SLAs with the customers should not be violated. Since Cloud Computing infrastructures represent mega scale infrastructures comprising up to thousands of physical hosts, there is a high potential of energy waste by overprovisioning resources in order to keep the violation level of SLAs as low as possible. Federated cloud management systems offer a simplified use of these infrastructures by hiding their proprietary solutions. As the infrastructure becomes more complex underneath these systems, the situations (like system failures, handling of load peaks and slopes) that users cannot easily handle, occur more and more frequently. Therefore, federations need to manage these situations autonomously without user interactions.

This paper introduces a methodology to autonomously operate cloud federations by controlling their behavior with the help of knowledge management systems. Such systems do not only suggest reactive actions to comply with established Service Level Agreements (SLA) between provider and consumer, but they also find a balance between the fulfillment of established SLAs and efficient energy consumption. The paper adopts rule-based techniques as its knowledge management solution and provides an implementation of federated clouds on top of multiple simulated infrastructures. Using the FCM architecture as the basis of the investigations, the authors analyzed different approaches to integrate the knowledge management system within this architecture, and found a hybrid approach that incorporates fine-grained local adaptation operations with options for high-level override. This research carried out has pinpointed the adaptation actions and their possible effects on cloud federations. Finally, metrics have been developed that could indicate possible SLA violations in federations, and defined rules that could trigger adaptation actions in the case of predicted violations.

2.2.6 Web Service Interaction Adaptation using Complex Event Processing Patterns

Title	Web Service Interaction Adaptation using Complex Event Processing Patterns [9]
Authors	Yehia Taher (TILBURG), Michael Parkin (TILBURG), Mike P. Papazoglou (TILBURG), Willem-Jan van den Heuvel (TILBURG)
Type	Technique

Short description	Differences in Web Service interfaces can be classified into two types, signature and protocol incompatibilities, and techniques exist to resolve one or the other of these issues but rarely both. This paper describes an approach based on complex event processing to resolve both signature and protocol incompatibility problems that may exist between Web Service protocols. Our approach uses a small set of operators that can be applied to incoming messages individually or in combination to modify the structure, type and number of messages sent to the destination. The paper describes how CEP-based adapters, which are deployable in CEP engines, can be automatically generated from automata representations of the operators through a standard process and presents a proof-of-concept implementation.
Targeted integration scenario	Assumption-based Proactive Monitoring and Adaptation
Contribution to the adaptation problem	Resolution of both signature and protocol incompatibility problems that may exist between Web Service protocols.
Contribution to the monitoring problem	–
Integrated SBA Layers for Monitoring	–
Integrated SBA layers for adaptation	–
Cross-layer mechanisms	–
Unexpected situations handled	Situations regarding the structural and behavioural incompatibilities between web services.
Architecture elements	The architecture is composed of two integrated environments. (i) The Design Time Environment is used to instantiate the template operators. This phase models the adapter using operator automata through the use of an incompatibility detection process to produce a platform independent model. (ii) The run-time environment, on the other hand, contains a CEP platform, in which the transformation phase takes the platform independent model to produce the adapter as a CCQ (continuous computation query) for a CEP engine, i.e, a platform specific model. It consists of a continuous query engine and a set of SOAP message integration layers that allow the environment to send and receive messages to and from Web Services.
Requirements/constraints	The approach assumes services to be modeled using Automata and it does not involve incompatibilities regarding semantics or deadlocks, but handles those related only to structural and behavioral properties.
Refined SBA Life Cycle activities	The design-time activities relates to the construction, and deployment & provisioning phases of the SBA life-cycle; and the run-time activities defined in this study relates to all adaptation phases including the identification of adaptation needs and strategies, and their enactment as well as operation & management.
Future research directions	Ongoing work aims at extending the proposed solution toward two directions: (i) comparing our similarity measures to others and testing detection algorithm on real services; and (ii) assisting business process designers in determining how to address incompatibilities.

Extended abstract

Web services provide a solution to the integration of distributed software through the standardization of data format, interface definition language, transport mechanism and other interoperability aspects such as security and quality of service. The Web Service Description Language (WSDL) defines a Web Service interface as a document in XML format and a service as a set of endpoints that operate on messages containing either document-oriented or procedure-oriented information. The interface document provides a contract between the provider of a service and its users and allows some flexibility for the service provider as it hides the details of the implementation of the service from those using it.

Web Service interfaces (i.e., WSDL, BPEL, etc.) define the messages and protocol that should be used to communicate with the service. However, if two services wish to interact successfully, they must both support the same messages and protocol through the implementation of compatible WSDL and BPEL documents. Unfortunately, this is difficult to achieve in practice; Web Services are often developed independently and follow different standards or approaches in constructing their interfaces and Web Service compositions will often use of services in ways that were not foreseen in their original design and construction. Therefore, it is likely that most Web Services will be incompatible since many services will not support the same interface.

To solve this problem, one needs to generate adapters that can make two Web services collaborate even if they were not designed in that a way. The generation of adapters requires the elicitation of mismatches between services. The study describes an approach that makes use of complex event processing (CEP) to resolve both signature and protocol incompatibility problems that may exist between Web Service interfaces. The approach is oriented towards the use of a set of operators that can be applied to incoming messages individually or in combination to modify the structure, type and number of messages sent to the destination. By using a continuous query engine running within a CEP platform, we demonstrate how adapters can be automatically generated for a CEP engine and how signature and protocol adaptation between Web Services can be achieved practically in a proof-of-concept implementation. The adapters are capable of intercepting messages sent between services and can adapt the structure, type and number of incoming messages into the desired output message or messages.

2.2.7 Design for Self-adaptation in Service-oriented Systems in the Cloud

Title	Design for Self-adaptation in Service-oriented Systems in the Cloud [10]
Authors	A. Bucchiarone (FBK), C. Cappiello (POLIMI), E. Di Nitto (POLIMI), S. Gorlatch (MUENSTER), D. Meilander (MUENSTER), A. Metzger(UNIDUE)
Type	Methodology
Short description	In this work we focus on two main aspects, that is, the kinds of changes that trigger self-adaptation in a service-oriented system and the strategies that can be adopted to deal with adaptation. We provide a preliminary contribution to the systematic understanding of adaptation across layers (Infrastructure and service composition layers).
Targeted integration scenario	(i) Quality-driven Multilayer SBA Monitoring and Adaptation (ii) Context-based Adaptation and Monitoring.
Contribution to the adaptation problem	Distribution handling for the dynamic adaptation of running application sessions by adding/removing Cloud resources on demand using particular adaptation strategies
Contribution to the monitoring problem	Monitoring of application-specific data, e.g., update rate, number of entities, etc.
Integrated SBA Layers for Monitoring	SCC, SI
Integrated SBA layers for adaptation	SCC, SI
Cross-layer mechanisms	This work will not scrutinizes the aspect of cross-layer adaptation
Unexpected situations handled	• Change in QoS, e.g., caused by unreliable hoster resources; • Change in the machine, e.g., caused by increasing user interactions, making computation of state updates more expensive; • Change in the business context, e.g., more users connect to the application due to changing user preferences.

Architecture elements	In order to support Real-Time Online Interactive Applications (ROIA) development and adaptation on Clouds, we develop the RTF-RMS resource management system [2] on top of the Real-Time Framework (RTF – [1]). RTF-RMS implements the following mechanisms for ROIA development on Clouds: (i) *Monitoring* of application-specific data, e.g., update rate, number of entities, etc. (ii) *Distribution handling* for the dynamic adaptation of running application sessions by adding/removing Cloud resources on demand using particular adaptation strategies. (iii) *Application profiles* that allow developers to specify application-specific adaptation triggers. (iv) High-level *development support* for communication handling and application state distribution.
Requirements/constraints	• In its current implementation, RTF-RMS only supports the Amazon Elastic Compute Cloud (EC2) interface. • For the definition of application profiles, the application developer has to manually find concrete values and suitable thresholds for his application by conducting experiments in a Cloud environment. • Continuous adaptation during runtime eventually leads to an inefficient application architecture.
Refined SBA Life Cycle activities	All core life-cycle activities
Future research directions	Our work on ROIA development along the S-Cube Lifecycle Model identified the demand for balancing run-time adaptation and application re-design. Particularly, the continuous adaptation on Cloud resources may lead to an inefficient application structure which requires application redesign. In our future research, we will address this task with particular regard to the adaptation on Cloud resources that implies additional challenges, like unknown resource locations, heterogeneous resource performance, etc. We will study how to incorporate new design-for-adaptation activities into the software development process using RTF-RMS and the S-Cube Lifecycle Model, e.g., how to define suitable adaptation triggers and strategies.

Extended abstract

Service-oriented systems are able to offer complex and flexible functionalities in widely distributed environments by composing different types of services. These systems have to be adaptable to unforeseen changes in the functionality offered by component services and to their unavailability or decreasing performances. Furthermore, when systems are made available to a high number of potential users, they should also be able to dynamically adapt to the current context of use as well as to specific requirements and needs of the specific users. In order to address these issues, mechanisms that enable adaptation should be introduced in the life-cycle of systems, both in the design and in the runtime phases.

In this work we will go through the life-cycle of a service-oriented system highlighting those activities that are needed to support adaptation. The adaptation activities can be performed at various layers of the service-oriented system. In particular, they can concern the layer where services are composed together or the layer of the executing infrastructure, typically, a cloud system. To exemplify the various steps and activities we use an example from the domain of real-time online interactive applications.

2.2.8 A Classification of BPEL Extensions

Title	A Classification of BPEL Extensions [11]
Authors	Oliver Kopp (USTUTT), Katharina Grlach (USTUTT), Dimka Karastoyanova (USTUTT), Frank Leymann (USTUTT), Michael Reiter (USTUTT), David Schumm (USTUTT), Mirko Sonntag (USTUTT), Steve Strauch (USTUTT), Tobias Unger (USTUTT), Matthias Wieland (USTUTT)

Type	Classification
Short description	In this paper, we provide (i) a classification of existing BPEL extensions and (ii) guidelines to develop extensions.
Targeted integration scenario	–
Contribution to the adaptation problem	Some of the classified BPEL extensions aim at improving the adaptability of BPEL by means e.g. of aspect-oriented programming and run-time injection of additional behaviour.
Contribution to the monitoring problem	Some of the classified BPEL extensions aim at enabling monitoring of BPEL process, e.g. by representing the state of activities as resources that can be accessed at run-time, and by exposing hooks that trigger the evaluation of business rules.
Integrated SBA Layers for Monitoring	SCC
Integrated SBA layers for adaptation	SCC
Cross-layer mechanisms	–
Unexpected situations handled	Several of the classified BPEL extensions aim at enabling recovery of activity failures and increased reliability of BPEL processes.
Architecture elements	Service composition, Monitoring engine, Adaptation engine
Requirements/constraints	Depends on the particular BPEL extension
Refined SBA Life Cycle activities	Depends on the particular BPEL extension
Future research directions	The findings presented in this work on the design and implementation of BPEL extensions remains valid in the context of the Business Process Model and Notation (BPMN) language. As part of our future work, we will classify BPMN extensions according to the presented classification framework.

Extended abstract

The Business Process Execution Language (BPEL) has been designed for the implementation of business processes using Web service technology. Nowadays, BPEL is used for implementing business processes in numerous different scenarios such as the automation of scientific simulations, the provisioning Software as a Service (SaaS) applications and as exchange format for business processes. With the growth over time of the adoption of BPEL, it has been applied to scenarios and use-cases that were not originally envisoned, and for which its constructs are not sufficient. For instance, the modelling of sub-processes is a functionality that the BPEL specification and, as a consequence, standard-conforming implementations do not cover.

As a result, BPEL is frequently extended for supporting desired functionality that is not available in standard BPEL. Depending on the particular purpose, an extension may improve efficiency, increase flexibility, ensure better performance, or add more functionality. However, extensions have also disadvantages. The whole toolset that is used for business process management (BPM) needs to support the extension. Common components of this toolset are applications for modeling, adapting, executing, monitoring, and analyzing the processes. Moreover, if business partners exchange (parts of) their processes, their toolsets need to understand and support the extensions as well.

In this paper, we provide (i) a classification of existing BPEL extensions and (ii) guidelines to develop extensions. An interesting finding of this work is that only around half of the sixty-two classified extensions conform to the definition of BPEL extension set by the BPEL specification. In some cases, this is because the design of the extension has not carefully taken into account the limitations it had to conform to. However, it is also a symptom that the ways of extending BPEL allowed by the specification are limited, and, in retrospective, perhaps too limited.

2.2.9 A Penalty-Based Approach for QoS Dissatisfaction Using Fuzzy Rules

Title	A Penalty-Based Approach for QoS Dissatisfaction Using Fuzzy Rules [12]
Authors	Barbara Pernici (POLIMI), Seyed Hossein Siadat (POLIMI), Salima Benbernou (UPD/UCBL), Mourad Ouziri (UPD/UCBL)
Type	Model/Methodoly/Experimental Evaluation.
Short description	Quality of Service (QoS) guarantees are commonly defined in Service Level Agreements (SLAs) between provider and consumer of services. Such guarantees are often violated due to various reasons. QoS violation requires a service adaptation and penalties have to be associated when promises are not met. However, there is a lack of research in defining and assessing penalties according to the degree of violation. In this paper, we provide an approach based on fuzzy logic for modeling and measuring penalties with respect to the extent of QoS violation. Penalties are assigned by means of fuzzy rules.
Targeted integration scenario	Quality-driven Multilayer SBA Monitoring and Adaptation & Assumption-based Proactive Monitoring and Adaptation
Contribution to the adaptation problem	We are interested in changes of service quality and the associated penalties in case of non exact fulfilments of QoS stipulated in the SLA. We are aware of degree of penalties. For that, we take advantage of the fuzzy logic for measuring the overall penalties based on the QoS and selecting service adaptation strategies
Contribution to the monitoring problem	–
Integrated SBA Layers for Monitoring	In order to compose services, we need services fulfilling the best QoS.
Integrated SBA layers for adaptation	In order to compose services, we need services fulfilling the best QoS.
Cross-layer mechanisms	–
Unexpected situations handled	–
Architecture elements	–
Requirements/constraints	We assume that the relevant QoS and their associated penalties are collected from the monitoring phase. Our system requires human expertise to define the membership function for each parameters and determine the number of rules.
Refined SBA Life Cycle activities	**Requirements Engineering:** Penalties definition according to the QoS, fuzzy parameters. **Construction:** definition of if-then-else-fuzzy rules. **Identify adaptation needs:** quality factor analysis, importance of penalties. **Identify adaptation strategy:** Selection of adaptation strategies based on an inference approach and adaptation priority.
Future research directions	(i) Identify more varieties of penalties. (ii) Penalties on multi SLAs.

Extended abstract

QoS guarantees defined in contracts may be violated due to various reasons. This situation needs to be handled through applying adaptation techniques not to bring dissatisfaction. The concept of penalty has been used in SLAs to compensate the conditions under which guarantee terms are not met. Despite some works have been done on the description, negotiation and monitoring of SLAs, however there is not much work on the definition of penalty clauses. WS-Agreement specification has been studied to define penalties based on different types of violation. However, penalties are assigned to violation of a single property instead of assigning penalties to violation of overall QoS. Moreover, the approach introduces a method for measuring penalties which is for fixed predefined number of violations, instead of measuring the extent of violation and assigning penalties accordingly. One main issue is how to determine the appropriate amount of penalties as compensations from providers to satisfy customers. As quality parameters can be satisfied partially, the assessment of penalties can be based on the degree of quality violation. Understanding the violation degree is a prerequisite for assessing penalties. However, measuring such violation is yet an open research challenge. In addition, the influencing factors in defining penalties need to be identified. A static amount of penalty (manual approaches) does not reflect the extent of violation at runtime. The amount and level of penalties are related to the degree of quality violation provided from the provider side. On the other side, the customers characteristics may also affect the amount of penalties. For example a penalty to satisfy a gold/loyal customer is different with the one for an occasional customer. To the best of our knowledge, there is no formal relation between the assigned penalty and its influencing factors. Moreover, the extent and type of penalties are not clearly expressed in related work. However, understanding such relation and providing a mapping between them are complicated issues. We argue what is missing is a suitable mechanism for modelling penalties that takes into account both provider and consumer sides. Apart from the degree of violation, we also consider the state of customer and service provider with respect to their past history (e.g. whether the service has been penalised previously) in determining the right amount of penalties. However, as the relation between a given penalty and its infuencing factors is not linear, conventional mathematical techniques are not applicable for modelling penalties. The goal of this paper is to apply an inference technique using fuzzy logic as a solution to propose a penalty-based approach for compensating conditions in which quality guarantees are not respected. Fuzzy logic is well suited for describing QoS and measuring quality parameters. We demonstrate a penalty inference model with a rule based mechanism applying fuzzy set theory. Measuring an appropriate value for penalties with respect to the amount of violation is described in this work.

Chapter 3

Future Research challenges and Conclusions

3.1 Future Research Challenges

In this section, we present the research challenges on monitoring and adaptation identified during the S-Cube Research Roadmap Workshop (Barcelona, November 22, 2011), attended by 40 project members and associate members.

The overall objective was to identify research challenges that may become relevant after and beyond S-Cube (in 510 years) and which have the potential to radically challenge existing thinking (i.e., beyond incremental). The Research Roadmap Workshop was organized along four topical sessions:

- S1: “Service life-cycle and software engineering”,
- S2: “Service technology foundations”,
- S3: “Multi-layer and mixed-initiative monitoring and ?adaptation for service-oriented systems”,
- S4: “Online service quality prediction for proactive ?adaptation”.

In the following we briefly introduce the challenges emerged for S3, which are going to be presented in a paper to the 2012 ICSE workshop on “European Software Services and Systems Research – Results and Challenges”.

3.1.1 Challenge 1. Approaches for retrieving and analyzing context information to support individuals in performing the right adaptation decisions in user-centric systems.

(i) *User-driven monitoring:* when the user context changes, the way the SBA is monitored should reflect the changes, (ii) *User-driven adaptation:* react to changes in the user context and generate a flexible interaction protocol that allows the user to control and coordinate the execution, (iii) users as *spectators*: at the same time service consumers, service inventors, and data donors, (iv) provision of ways that *encourage and incentivize users* to create and share contents and data with the system and the other peers in the network.

3.1.2 Challenge 2. Decentralized models and techniques to monitor and predict service quality issues.

(i) Techniques to combine information from SOA layers with the one coming from the *network infrastructure*, (ii) definition of *cross-layer quality metrics* considering the whole service delivery chain, including communication networks, (iii) use of *decentralized strategies* for the monitoring and predicting models.

3.1.3 Challenge 3: Techniques for combining and cross-correlating observations, predictions and events from different sources and provided by different techniques.

(i) *Flexible and dynamic correlation* of useful information (observations, predictions, and events) from different sources, across the functional layers, and provided by different analysis, decision and adaptation mechanisms, (ii) *learning correlation rules automatically* to allow SBAs to adapt to unforeseen situations.

3.1.4 Challenge 4: Assurances for adaptation.

(i) Quality assurance techniques to *prevent* run-time design decisions/adaptations to lead to *inconsistent situations*, (ii) assurances of service-oriented systems in order to architect *resilience against unknown situations* and for dealing with rare events, (iii) concepts and techniques to avoid *unwanted co-adaptation, malicious adaptations*, as well as races and other anomalies in order to ensure trustworthy self-adaptation and evolution of open, service-oriented systems, (iv) concepts and techniques for *formally guaranteeing the correctness* of adaptations.

3.2 Conclusions

This deliverable summarized the research carried out by the members of the S-Cube project while investigating and solving problems in relation to the main challenges of the work package JRA-1.2. In this deliverable, we have presented the ways of consolidating our joint research on the validated set of adaptation and monitoring principles, techniques and methodologies with special focus in context and HCI awareness. This work has been presented through several jointly written scientific papers that were systematically analyzed in the context of the deliverable's main aims.

The future work and research directions of the sustainable research and collaboration are as diverse as the research results presented in this deliverable. The diversity of the future work is inherited from the three previously identified research scenarios of the deliverable titled CD-JRA-1.2.5. The research directions are detailed in the summary tables for each contributed paper, here we only provide a short overview on the future approaches S-Cube members identified for handling unexpected situations in context and HCI aware service-based applications. We provide the summaries along the research scenarios they are going to enhance:

Quality-driven Multilayer SBA Monitoring and Adaptation. Future research should comprise the enhancement and evaluation of the new layers incorporated into the SBA (such as platforms, typically seen in cloud computing setups) through the trial of various case studies from different application scenarios. These scenarios should consider situations such as unknown resource locations or heterogeneous resource performance. For the evaluation of the SBA novel analyzer hierarchies, specialized monitoring, tailor-made adaptation strategies and their selection criteria have to be defined to enable flexible selection of the enacted adaptation strategies on the system. Finally, new studies have to be initiated to target the feasibility of managing different kinds of KPI constraints.

Assumption-based Proactive Monitoring and Adaptation. The scenario should be extended towards two directions: (i) comparing the currently competing similarity measures and testing detection algorithms on services deployed in lifelike environments; and (ii) assisting business process designers in determining how to address incompatibilities upon service substitutions. Researchers need to identify new ways to penalize improper proactive changes in the SBA while considering multiple SLA constraints and violations.

Context- based Adaptation and Monitoring. Investigate new techniques dealing with the (semi-) automatic adaptation of the monitor as a reaction to context changes/adaptation in the application. The adaptation history should be used to bring corresponding changes to the process model (process evolution).

Bibliography

[1] The Real-Time Framework – www.real-time-framework.com, 2011.

[2] Meilaender, Dominik; Bucchiarone, Antonio; Cappiello, Cinzia, Di Nitto, Elisabetta; Gorlatch, Sergei. Using a Lifecycle Model for Developing and Executing Real-Time Online Applications on Clouds, In Proc. of 7th International Workshop on Engineering Service-Oriented Architectures (WESOA), 2011.

[3] Elisabetta di Nitto, editor. State of the art report on software engineering design knowledge and Survey of HCI and contextual Knowledge, July 2008. S-Cube project deliverable: PO-JRA-1.1.1. http://www.s- cube-network.eu/achievements-results/s-cube-deliverables.

[4] Ricardo Contreras and Andrea Zisman. 2011. Identifying, modifying, creating, and removing monitor rules for service oriented computing. In Proceedings of the 3rd International Workshop on Principles of Engineering Service-Oriented Systems (PESOS '11). ACM, New York, NY, USA, 43-49.

[5] Zengin, A., Marconi, A., & Pistore, M. (2011). CLAM: Cross-layer Adaptation Manager for Service-Based Applications. In Proceedings of the International Workshop on Quality Assurance for Service-Based Applications, Qasba 11 (pp. 2127). Acm.

[6] Bucchiarone, A., Pistore, M., Raik, H., & Kazhamiakin, R. (2011). Adaptation of Service-based Business Processes by Context-Aware Replanning. In Proc. SOCA 2011.

[7] S. Guinea, G. Kecskemeti, A. Marconi and B. Wetzstein. "Multi-layered Monitoring and Adaptation", in Service-Oriented Computing: 9th International Conference, ICSOC 2011, Paphos, Cyprus, December 5-8, 2011 Proceedings. ISBN: 978-3-642-25535-9.

[8] G. Kecskemeti, M. Maurer, I. Brandic, A. Kertesz, Zs. Nemeth and S. Dustdar: "Facilitating self-adaptable Inter-Cloud management", in Proceedings of the Cloud Computing for Compute and Data Intensive Applications special session of the 20th Euromicro conference on parallel, distributed and network-based processing. Leipniz Supercomputing Centre, Garching, February 15-17th 2012.

[9] Taher, Yehia; Parkin, Michael; Papazoglou, Mike P.; van den Heuvel, Willem-Jan. "Adaptation of Web Service Interactions using Complex Event Processing Patterns", in Service-Oriented Computing: 9th International Conference, ICSOC 2011, Paphos, Cyprus, December 5-8, 2011 Proceedings. ISBN: 978-3-642-25535-9.

[10] Bucchiarone, A., Cappiello, C., Nitto, E. D., Gorlatch, S., Mailander, D., & Metzger, A. (2011). Design for Self-adaptation in Service-oriented Systems in the Cloud. In D. Petcu, & J. L. V. Poletti (Eds.), European Research Activities in Cloud Computing. Cambridge Scholars Publishing.

[11] Kopp, Oliver; Grlach, Katharina; Karastoyanova, Dimka; Leymann, Frank; Reiter, Michael; Schumm, David; Sonntag, Mirko; Strauch, Steve; Unger, Tobias; Wieland, Matthias; Khalaf, Rania: A Classification of BPEL Extensions. In: Journal of Systems Integration. Vol. 2(4), Online, 2011

[12] Barbara Pernici, Seyed Hossein Siadat, Salima Benbernou, Mourad Ouziri: A Penalty-Based Approach for QoS Dissatisfaction Using Fuzzy Rules. ICSOC 2011: 574-581