Cross-layer Service Adaptation
State-of-the-Art, Shortcoming Analysis, and Future Research Directions

Ameni Meskini[1], Yehia Taher[2], Rafiqul Haque[3] and Yahya Slimani[1]

[1]*University of Carthage, INSAT, LISI research Laboratory, Tunis, Tunisia*

[2]*Laboratoire PRiSM, Universite de Versailles/Saint-Quentin-en-Yvelines, Versailles, France*

[3]*Laboratoire d'InfoRmatique en Image et Systèmes d'information, Université Claude Bernard Lyon 1, Lyon, France*

Keywords: Web Service, Service based Application, Service Adaptation, Cross-layer Adaptation.

Abstract: In the past few years several cross-layer monitoring and adaptation technologies have been proposed. Although these are cross-layer adaptation technologies, however, in practice they focus on a particular layer. Some solutions involves two layers, yet none of the existing solutions do not consider all the layers during adaptation process. Furthermore, cross-layer adaptation approaches generate incompatibility problems. This is an adaptation coordination problem. Incompatibility refers to the situations where the adaptation is performed in a layer is not compatible with the constraints exposed by the other layers. This survey aims at studying and analyzing current approaches for web services adaptation, discussing their shortcomings and proposing research directions on cross-layer web service adaptation.

1 INTRODUCTION

Service adaptation has drawn enormous research interests in the area of Service Oriented Computing (SOC) (Geihs et al, 2009). Adaptation from the functional point of view can be defined as an ability of a Service Based Application (SBA) (Bucchiarone et al, 2009) to adapt changes or requirements that are needed to guarantee fault-tolerance or to optimize system performances.

While developing an SBA, it may not be possible to capture all functional and nonfunctional requirements because many times the requirements evolve at runtime. Typically, an application is able to carry out the operations (at runtime) that are studied and documented during requirement analysis. The unprecedented requirements that are evolved at runtime may lead to failure. In other words, applications are unable to perform the operations that have not been realized.

This limitation gave rise to the notion of service adaptation. Many new techniques such as service replacement or adding new service have been proposed to build adaptive SBAs. However, there are several challenges that cannot be dealt with efficiently by the existing techniques. One major challenge is handling the impact of adaptation operations. For instance, service replacement at different layers of SBAs. It is worth noting that an SBA has different layers (Papazoglou et al, 2008). The notion of multi-layer SBAs relies on the Service Oriented Architecture (SOA) (Liu et al, 2011) paradigm. These layers interact with each other.

Thus, if a service is adapted in one layer (e.g., BPM layer), it may affect the other layers (e.g., orchestration layer). This promotes the notion of cross-layer adaptation which is the main focus of this paper. Cross-layer adaptation is a process of adapting a service in different layers of SBAs. It promotes configuration challenges. Several techniques have been proposed to tackle these challenges. This paper aims to investigate all existing technologies, methodologies, and techniques related to cross-layer adaptation. It presents a comprehensive review of the state-of-the-art, summarizes their strengths and weaknesses, and identifies future research direction in this area.

This paper is organized as follows. In Section 2, we present the review of the state-of-the art. We discuss our findings in Section 3. A conclusion is drawn in Section 4.

Meskini A., Taher Y., Haque R. and Slimani Y..
Cross-layer Service Adaptation - State-of-the-Art, Shortcoming Analysis, and Future Research Directions.
DOI: 10.5220/0005435702600267
In *Proceedings of the 5th International Conference on Cloud Computing and Services Science* (CLOSER-2015), pages 260-267
ISBN: 978-989-758-104-5

2 SERVICE ADAPTATION APPROACHES

Although our focus in this paper is cross-layer adaptation, we cover all the existing technologies related to service adaptation. The purpose of this study is to provide a comprehensive understanding of the strength of service adaptation and also to outline the limitations why traditional adaptation technologies are unable to assist in cross-layer adaptation. It is worth noting that although our study mainly covers the service based systems, we try to cover adaptation in the agent based systems as well.

2.1 Interaction-based Adaptation Approaches

Interaction-based adaptation approaches deals with interactions between web services such as, actions required to mediate communications between web services, and adapting the compositions of web services in case of failure of a component. Unlike the approaches that focus on QoS adaptation (of instance composition), interaction-based adaptation approaches are concerned with the changes and the adaptation of interactions within a service composition.

For example, re-engineering services to ensure that they can integrate new components by guaranteeing interoperability. Sometime adaptation operations such as substitute to repair or to optimize QoS are not sufficient for efficient adaptation. The main reason is the emergence of new requirements and additional constraints which may not be possible to handle efficiently by he selected services. For instance, a service may not be able to meet user needs or may fail to handle heterogeneity of the interfaces between services or communication protocols. The objective of adaptation in this case is not only to manage the QoS adaptation, but also to ensure that the adaptation measures do not lead to interaction failures. The composition and the mediation are the most common solutions in such situations.

In interactions-based adaptation, existing approaches realize exchanging messages in service compositions based on pre-defined policies such as the policies proposed in (Baresi et al, 2007). This involves the business processes which are essentially composition of services. We found several services composition-based adaptation approaches which we discuss in this section. Baresi et al. (Baresi et al, 2007) address the problem of substitution of services and the dynamic binding of the service providers in order to repair failures. Their work targets the adaptation of the workflows (defined using BPEL (IBM, BEA Systems, Microsoft, SAP AG, Siebel Systems, 2003) at runtime to select between available alternatives based on nonfunctional requirements, or to retry a service following in the first choice. To enable the deployment and the reconfiguration of service compositions during its execution, the authors used a specification of BPEL process which is enriched by a set of rules and constraints for the discovery or dynamic service binding until the time to execute.

The choice depends on the criteria defined by the user during the establishment process. The proposed framework can also exchange services based on the events collected during the monitoring phase. It relies on three actors: the registry service (DIRE) (Baresi et al, 2007) that can be distributed between the service providers, the runtime environment (SCENE) (Baresi et al, 2007) with the rules of discovery and binding, and monitoring features (Dynamo) (Baresi et al, 2005) and (Baresi et al, 2007) that produce events to reconfigure the processes.

The main limitation of this approach is the web service composition language. The authors in (Ardagna et al, 2007) propose an implicit approach for adaptive composition of services within the flexible processes. This approach is implemented in business process management layer. The main objective is to select the best set of services available at run-time by taking the constraints of business process, users preferences, and execution contexts into account.

The authors introduce a new approach to model the problem of service selection. This approach is effective for large process and in the case of QoS constraints are at extreme. In the proposed model, the problem of service selection is formalized as a mixed problem of linear programming, the loop peeling is adopted in optimization, and the constraints posed by the stateful web services are considered.

2.2 Mediation-based Adaptation Approaches

While composing interactions, services may encounter heterogeneity problems. For instance, interaction types can be different, incompatible communication protocols; different semantics of interactions promote the heterogeneity problem.

These problems may occur in different steps of composition. Also, the problems may occur while adaptation actions carried out such as, during substitution of a WS by another WS.

The solution of the heterogeneity problem is called mediation which is critical to achieve adaptation and composition of services. However, additional mechanisms are needed for successful interactions between web services and to perform different adaptation actions (e.g., substitution, re-selection, composition, etc.). Adaptation (in this case "mediation) is an important functionality which enables integration of business services. Generally speaking, mediation resolves conflicts between two actors. In the context of web services, mediation aims to resolve heterogeneity between web services in order to enable successful interactions (Chafle et al., 2006).

One needs to generate a service that ensures interactions between the two services with two signatures, different protocols or interfaces, in order to guarantee interoperability. The requirements of an adaptation in these approaches stem from two sources: (i) the level of heterogeneity in the upper stack of interoperability (e.g., business level, infrastructure protocols.), and (ii) the diversity of customers, each one of them supports different protocols and interfaces. Mediation can be automatic (Williams et al., 2006) or semi-automatic (Reza et al, 2007).

Taher et al, 2009 (Taher et al, 2009) propose a multilayer software architecture. They propose a framework for transparent and flexible substitution of a service provider by another with respect to an existed consumer. A framework for automatic generation of adapters and service interfaces modelling using automata was adopted to solve the problem of incompatibility in the interaction between two services: a consumer and a new provider. If incompatibilities between these services are detected, an adapter is generated automatically based on the incompatibilities. The generation of the adapter relies on the automata model. The generated adapter contains a sufficient detail of the projected technology called CEP (Complex Event Processing) engine (Luckham et al, 2001).

However, unfortunately, the complex incompatibilities were not considered in this tool. For example, the implementation of several different operations of customer service and a supplier service is not possible by this tool.

The solution proposed in (Hau, 2003) uses OWL (Dean, 2002) to annotate interfaces too. Both solutions (proposed by Syu and Hau) have an abstraction layer called meta-data space. Semantic annotation is used to describe the methods of services.

Meta-services use these annotations to find appropriate matches between needs and implementations. These solutions differ from the other adaptation approaches. Two distinguished aspects of these approaches are as follows:

- Their locations are dependent on architectures in which they are embedded, and an adoption concerning with the interfaces of web service is often the responsibility of the service provider.
- These approaches are platform dependent such as they are dependent on languages and composition engines

2.3 Cross-layer Adaptation Approaches

The cross-layer adaptation refers to a process of adapting a general system consisting of several layers, where the technology and processes of each layer are integrated and controlled by the same adapter frame. In the context of SOA, this denotes a consistent adaptation through the service interface of different layers and applying a SOA system while maintaining the characteristics such as loose coupling and service autonomy.

The problem of monitoring and adaptation of different types of software systems has gained interests in both the research community and industry. In recent years, these issues have promoted interest in the area of SOA. However, the results and directions are still insufficient. One of the key issues here is that the proposed approaches are very fragmented. They deal only with the problems which are specific to a particular aspect of web service and a particular functional layer, such as business process management layer, service composition and coordination layer, or service infrastructure layer. However, the implementation of various layers of web service can be nested in different artifacts. A layer may contain objects that reside in another layer. However, such cases are ignored by traditional monitoring and adaptation solutions.

Consequently, there is a possibility that these solutions will detect the problems incorrectly which will lead to inaccurate decisions concerning adaptation. This shortcoming of existing solution promotes the need of cross-layer adaptation. In this section, we study the most recent solutions which have been proposed to provide a monitoring and adaptation tools that covers multiple layers. We found that in these solutions, controlling and adaptation are developed by using various techniques such as, monitoring and event logging, detecting the patterns of events, and correlation and mapping between events and appropriate adaptation strategies, etc.. The solutions proposed in (Gjrven et al, 2008), (Popescu et al., 2010), (Popescu et al., 2012), (Zengin et al., 2011), (Zengin et al., 2011) and (Zeginis et al. 2011) are based on the situation-action mechanism. The situations correspond to a set of events and disparities while the actions are defined as templates for adaptation. These approaches combine the taxonomies of adaptation problems and mechanisms based on the events for guiding the selection process of the adaptation models based on the degree of correspondence between events and disparities of adaptation.

In (Gjrven et al, 2008), a middleware called QuA is presented that provides a multilayer adaptation coordinated by incorporating multiple mechanisms of adaptation in the interface and application layers. However, the proposed middleware is lacking theflexibility because the adaptation logic is predefined and static. A multi-layer adaptation framework is proposed in (Popescu et al., 2010). The authors use taxonomy and adaptation models (patterns) which are created during the design phase to represent the possible solutions to adaptation problems. In this framework, they designed adaptive predefined templates to provide a means for dynamic multi-layers adaptation.

These models define the behaviour of the adaptation processes. However, this approach does not consider the infrastructure layer and the authors do not provide the mechanism for detection disparities. An adaptation manager called CLAM is proposed in (Zengin et al., 2011) to handle adaptive inter-layer and multilayer problems. The authors have classified a group of adaptation paths of an adaptation tree which can be built in any layer of SBAs. The limitation of these approaches is the execution control which is performed in an isolated manner. This does not allow an effective analysis of monitoring data and detected events because events are analysed and processed independently of each other and the critical information are not propagated between layers. This can lead to an incorrect identification of the original source of the problems. Also, some approaches do not realize monitoring in all the layers which affects the final step of adaptations. For example, the actual problem can occur in the infrastructure layer, while it is detected in the composition layer and therefore, it cannot be properly diagnosed.

Additionally, in (Guinea et al, 2011), it is also argued that monitoring of the web services is not sufficient to allow proper and effective adaptation at runtime. The authors present a framework which uses various techniques for monitoring different layers. Also, it uses a centralized agent of adaptation to collect the events and analyse the violations of KPIs.

Although the cross-layer adaptation approaches designed to identify the sources of problems through analysis and diagnosis that take several layers into account, the works presented in this section have some limitations. Based on our analysis, these approaches can be improved to be more efficient. For instance, since the adaptation approaches do not consider the characteristics and requirements of all the layers of SBAs rather they focus on a specific layer, the activities of adaptation may fail to achieve the desired effects. Furthermore, these approaches may lead to incompatibility problems.

2.4 Adaptation in Agent based Systems

From architectural point of view, there is a similarity between agent and service based systems. This is one of the main reasons we studied the adaptation solutions proposed in this domain. The notion of agent based system is relatively new. We found a few research works on adaptive agent based system.

Qureshi and Perini (Nauman et al., 2008) proposed a methodology called TProcess for seamless self-adaptation in agent based system. The methodology is shaped a triangle that includes three elements include requirement-time, design-time time, and runtime. The authors argue that adaptation should be built on the top of these mutual dependent elements. The critical components of TProcess are goal models which are defined at requirement-time step. The goal models contain QoS parameters, their values and conditions. These are mapped to the implementation platform in the design-time step.

In (Bernon et al., 2003), the author proposed a methodology called ADELFE to guide developers to develop adaptive multi-agent systems. The methodology is based on object-oriented methodologies, follows rational unified process and uses Unified Modeling. In (Ibrahim, 2004), the author proposed a framework for developing intelligent adaptive agents.

In the proposed framework, the agents are defined as systems or machines that utilize inferential or complex computational methodologies to modify or change control parameters, knowledge bases, task plans, problem-solving, methodologies, course of actions, or other objects in order to successfully accomplish a set of tasks that are of interest to the user. The intelligent adaptive agents are classified into three based on the agent's capabilities on performing external and internal. These categories are listed below:

- Internal adaptation: In this criterion, the internal systems of the agent are adaptive; however, its external actions do not reflect adaptive behaviour.
- External adaptation: It is simply the opposite of internal adaptation. In this the internal systems of agents do not reflect adaptive behaviour.
- Complete adaptation: Internal systems are adaptive and external actions reflect adaptive behavior.

There are a few significant differences between adaptive SBAs and adaptive agents. In SBAs, adaptations are performed in different layers, as these applications rely on multilayer architecture.

However, multilayer adaptation is of the scope of agent based systems. Additionally, none of the adaptive agent based solutions is aware of cross-layer adaption. However, evidently, the service based systems can be benefited by using the approaches used in adaptive agent based systems. Particularly, the notion of context-awareness and self-adaption can be efficacious for adaptive service based systems.

3 ANALYSIS AND RESEARCH DIRECTIONS

In this section, we summarise our findings and propose a few potential extensions specifically in the area of cross-layer adaptation. We studied various solutions published in the literature. It is worth noting that in this section we limit our discussion in the context of service based systems which is the main focus of this study.

3.1 Analysis

We studied different research initiatives that focus on adaptation problems concerning service interaction in the service composition layer. Specifically, we studied the heterogeneity problems regarding interactions which can be found in the service interface layer. The heterogeneity problem may lead to inconsistency with respect to data exchanged between the services. We found that the main reason for heterogeneity problem is different formats of the messages exchanged between services. For an effective and adaptation heterogeneity between web services must be dealt with efficiently.

We found mediation-based adaptation approaches deals with heterogeneity. They enable exchanging consistent data between Web services. However, these approaches have limitations. They lack of flexibility and the automation needs to be efficient for a complete and effective adaptation. Moreover, they are limited to technical and structural aspects of a system. They do not cover other aspects. In addition, due to the highly dynamic and evolving nature of the environment and different requirements of service users (infrastructure protocols, and behavior), a manual intervention is required, especially to define the management tasks to handle disparities or to specify or adjust the composition diagram. This is certainly a limitation to carry out adaptation operations efficiently.

In addition, the adaptation mechanisms are not rich enough and deals only with the specific adaptation situations and actions, which does not cover multiple anomalies that may occur in execution environments. The cross-layer adaptation approaches are fragmented and isolated. They do not consider the effects of changes and modifications on all the functional layers of the SBAs. The existing cross-layer, adaptation solutions are designed to adapt a particular functional layer, namely, the business layer, the service composition layer, or infrastructure layer. The realization of different layers of web service can be nested such as different artifacts of a layer can refer to the same objects reside in another layer, while these relationships are ignored by the current monitoring and cross-layer adaptation solutions.

Also, these mechanisms are designed to support quality assurance for adaptation. They deal with the analysis of adaptation activities against the system model, and adaptation measures. Table 1 presents a synthetic summary of the cross-layer adaptation solutions which we studied in this paper. We consider three factors, defined by (Reza et al.2007) adaptation objectives that involves adaptation requirements (repair, optimization, mediation, etc..), adaptation methodology, and the layers covered by the solutions. Also, these mechanisms are designed to support quality assurance for adaptation.

They deal with the analysis of adaptation activities against the system model, adaptation measures, and other adaptations.

Table 1 presents a summary of the approaches found in the literature. We consider three factors

Table 1: Classification of cross-layer adaptation approaches.

Approach	Adaptation Objectives	Methodology of adaptation	Layer affected
(Reza et al, 2007)	Fault tolerance	Proactive	BPM, SCC
(Popescu et al., 2010)	Mediation	Reactive	BPM
(Popescu et al., 2011)	Reparation	Reactive	BPM
(Guinea et al, 2011)	Reparation	Reactive	BPM, SI
(Mos et al. 2009)	Monitoring	Reactive	SI
(Schmieders et al., 2011)	Reparation	Reactive	SCC, SI
(Vidackovic et al., 2009)	Optimistaion	Reactive	BPM
(Gjrven et al, 2008)	Configuration	Proactive	BPM, SCC
(Syu et al, 2004)	Mediation	Reactive	SI

defined by (Reza et al. 2007): (i) adaptation objectives involves adaptation requirements (repair, optimization, mediation, etc.., (ii) Adaptation methodology, and (iii) affected SBA layers which concerns with the change of locations and adaptation progress. From the comparison (shown in the above table) we conclude that none of the current approaches cover all the layers of service based systems. The solutions proposed by Reza et al., Guinea et al., Schmieders et al., Gjerven et al. are relatively more efficient as they cover two layers.

However, cross-layer adaptation solution must cover all three layers of SBAs to deal with various runtime challenges efficient that evolve in current service based system such as cloud service based applications. Remarkably, most of these approaches cover BPM layer, however, to the best of our understanding if an event adapted in the BPM layer, yet it the adaptation has not been propagated to the bottom layers implies that the adaptation has not be realized automatically and may not have done efficiently. This is an important limitation. The current solutions focus on specific layers (e.g., infrastructure layer or Business Process Management layer). One might think of building a hybrid solution which can combine two or more of the existing solutions. However, it will promote a huge complexity. Developing a hybrid solution needs a list of complex tasks include the following:

- Analysis of the affected layer,
- Identification of adaptation actions,
- Aggregation of these actions to check their effects on different layers,
- Launching a coordination system to coordinate adaptation actions,
- Checking whether the adjustment performed at one layer is compatible with the constraints posed by other layers, etc..) which can be costly in terms of response time.

3.2 Research Directions

We identified four critical aspects: context awareness, self-adaptation, completeness, performance, which should be focused in the topic of cross layer adaptation.

Context aware adaptation and self-adaptation have already been studied in agent oriented system. It is worth noting that context awareness and self-adaptation are complementary because self-adaptive system should be aware of the context. Otherwise, self-adaptation can be difficult.

The Table 1 shown in the previous section unearthed a very important shortcoming of cross layer service adaptation technologies. Although these technologies are known as cross-layer adaptation solution, to the best of our understanding, these solutions are complete. These approaches lack the ability to trace incompatibilities that can be triggered through adaptation. Therefore, a solution is needed which can create adaptation loop which runs adaptation process until new requirements or changes are adapted by resolving incompatibilities or conflicts. Adaptation promotes performance challenge. In other words, the system performance can be challenged enormously by adaptation. We found literature reported trade-off between adaptation and performance. An extensive research is necessary to develop a solution that can process adaptation by guaranteeing high efficiency (with respect to processing time).

We plan to develop an intelligent and fault-tolerant solution for cross-layer adaptation that can address the requirements discussed in the above. The proposed solution will enable to perform adaptation process by guaranteeing efficiency and effectiveness. It will be able to perform adaptation in all the layers of SBAs without any incompatibilities or conflicts. The solution will be context-aware and will support self-adaptiveness. This will ensure the autonomic execution of adaptation operations across the SBA layers.

We strongly believe that the genetic algorithms are potential for our solution especially to optimize the adaptation process. Genetic algorithms are widely used to handle cases such as requirement evolution and performance optimization which are the two most critical issues.

4 CONCLUSIONS

In this paper we studied adaptation technologies particularly the cross-layer adaptation technologies. We discussed the outcomes of our analysis. In particular, we discussed the limitations of different approaches of cross-layer service adaptations.

The major limitation we found is the lack of coordination between adaptation activities that may lead to conflicts or incompatibilities. According to our study, the current solutions do not consider the fact that adaptation in a layer may affect adversely the other layers of service based systems. According to our study, current cross-layer adaptation approaches lack efficient coordination which leads to conflict and incompatibilities. We believe that these problem must be addressed for an efficient cross-layer service adaptation. We presented the results of a brief study on adaptive agent based systems. We found in our study that the agent based adaptive systems have some advanced , features such as context-awareness, self-adaptation, etc.. The adaptive SBAs can be benefited by these features especially, the service based adaptive systems can be more intelligent and autonomous.

Additionally, based on our understanding we presented some research directions in the area of cross layer service adaptations. We strongly believe that the research in this area should focus on context awareness, self-adaptation, and performance etc. to develop highly high-performance solutions. We also presented a proposal of a solution which are currently working on.

There are a few limitations of our study. Firstly, this is merely a literature review. However, the state of the art could be better reviewed or understood by benchmarking the existing solutions. A comparison of adaption technologies in different contexts can be done by following a set of rigorous protocols. This paper is missing such an comparison. In our future work, we plan to conduct an empirical study with the current cross-layer adaptation technologies. Also, we plan to conduct a study by covering more contexts.

REFERENCES

Ardagna, D. , Pernici, B.(2007). Adaptive Service Composition in Flexible Processes, IEEE Transactions on Software Engineering, vol. 33, no. 6, pp. 369-384, June 2007.

Baresi , L. , Guinea, S. , (2005), Dynamo: Dynamic Monitoring of WS-BPEL Processes. In 5th International Conference on Service Oriented Computing, pages 478–483, 2005.

Baresi, L., Di Nitto, E. , Ghezzi, C., and Guinea, S. , (2007) . A Framework for the Deployment of Adaptable Web Service Compositions, Service Oriented Computing and Applications, vol. 1, no. 1, pp.75-91, 2007.

Bernon, C., Gleizes, M. P., Peyruqueou, S., Picard, G. (2003). ADELFE: a methodology for adaptive multi-agent systems engineering. In Engineering Societies in the Agents World III (pp. 156-169). Springer Berlin Heidelberg.

Bucchiarone, A., Cappiello, C., Di Nitto, E., Kazhamiakin, R., Mazza, V., and Pistore, M., (2009). Design for Adaptation of Service-Based Applications: Main Issues and Requirements ,ICSOC/ServiceWave in page 467–476.

Chafle, G., Dasgupta, K. , Kumar, A. , Mittal, S.,and Srivastava, B. , (2006). Adaptation in Web Service Composition and Execution, *Proc. IEEE Int'l Conf. Web Services (ICWS '06),* pp. 549-557, 2006.

Dean, M., Connolly, D., Harmelen, F., Hendler, J., Horrocks, I., Debo-rah L., McGuinness, Peter F. Patel-Schneider, Andrea Stein, L., (2002). Web ontology language (OWL) reference version 1.0. Technical report, www.w3c.org, 2002.

Geihs, K., Reichle, R., Wagner, M., Khan ,M., (2009). Service-Oriented Adaptation in Ubiquitous Computing Environments , International Conference on Computational Science and Engineering.

Gjrven, E., Rouvoy, R., Eliassen, F. , (2008). Cross-layer self-adaptation of service-oriented architectures, Proceedings of the 3rd workshop on Middleware for service oriented computing 2008.

Guinea, S. , Kecskemeti, G. , Marconi, A. , Wetzstein, B. , (2011). Multi-layered monitoring and adaptation,

Hau, J., Lee, W. , Newhouse, S. , (2003) . The ICENI Semantic Service Adaptation Framework. In: UK e-Science All Hands Meeting (2003).

Ibrahim F. Imam: Adaptive applications of intelligent agents. ISCC 2004: 7-12.

IBM, BEA Systems, Microsoft, SAP AG, Siebel Systems, (2003). Business Process Execution Language for Web Services version 1.1. http://www.ibm.com/developerworks/library/specification/wsbpel/. 2004 Imam, I. F. (2004, June). Adaptive applications of intelligent agents. In Computers and Communications, 2004. Proceedings. ISCC 2004. *Ninth International Symposium on (Vol. 1, pp. 7-12). IEEE.*

Liu, F., Tong, J., Mao, J., Bohn, R., Messina, J., Badger, L., and Leaf, D., (2011). NIST Cloud Computing

Reference Architecture: Recommendations of the National Institute of Standards and Technology. NIST Special Publication 500-292. pp. 10.

Luckham, D., (2001), The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems. Addison-Wesley Longman (2001).

Nauman,A., Qureshi, Anna Perini: An Agent-Based Middleware for Adaptive Systems. QSIC 2008: 423-428.

Papazoglou, M. P., (2008) . Web Services - Principles and Technology. Prentice Hall. ISBN 978-0-321-15555-9. pp. 1-752.

Popescu, R. , Staikopoulos, A. , Liu, P., Brogi, A. , Clarke, S. , (2010) . Taxonomy-Driven Adaptation of Multilayer Applications Using Templates," saso, pp.213-222, 2010 Fourth IEEE International Conference on Self-Adaptive and Self-Organizing Systems, 2010.

Popescu, R., Staikopoulos, A. , Liu, P. , Brogi, A. , Clarke. S. (2011). A Formalised, TaxonomyDriven Approach to Cross-Layer Application Adaptation ACM Transactions on Autonomous and Adaptive Systems, ICSOC.

Popescu, R., Staikopoulos, A. , Liu, P. , Brogi, A. , Clarke. S., 2012. A Formalised, Taxonomy-Driven Approach to Cross-Layer Application Adaptation" in ACM Transactions on Autonomous and Adaptive Systems, Proceedings of the 9th international conference on Service-Oriented Computing, December 05-08, 2011, Paphos, Cyprus.

Reza, H. , Nezhad , M. , Benatallah , B. , Martens , A. , Curbera , F. , Casati, F. , (2007). Semi-automated adaptation of service interactions, Proceedings of the 16th international conference on World Wide Web, May 08-12, 2007, Banff, Alberta, Canada.

Schmieders, E., Micsik, A., Oriol, M., Mahbub, K., and Kazhamiakin R., "Combining SLA prediction and cross layer adaptation for preventing SLA violations". In Proceedings of the 2nd Workshop on Software Services: Cloud Computing and Applications based on Software Services, Timisoara, Romania, June 2011.

Syu, J.-Y., (2004). An Ontology-Based Approach to Automatic Adaptation of Web Services", Department of Information Management National Taiwan University, 2004. (http://www.im.ntu.edu.tw/ IM/Theses/r92/R91725051.pdf).

Taher, C., Aït-Bachir, .A, Fauvet, .M, Benslimane, .M,:Diagnosing Incompatibilities in Web Service Interactions for Automatic Generation of Adapters. AINA 2009: 652-659.

Vidackovic, K., Weiner, N., Kett, H., Renner, T. "Towards business-oriented monitoring and adaptation of distributed service-based applications from a process owner's viewpoint". In: ICSOC/ServiceWave Workshops. pp. 385394, 2009.

Williams , S.K., Battle , S. A., Cuadrado, J. E. , (2006). Protocol mediation for adaptation in semantic web services, Proceedings of the 3rd European conference on The Semantic Web: research and applications, June 11-14, 2006, Budva, Montenegro.

Zeginis, C., Konsolaki, K. , Kritikos, K. , and Plexousakis, D. , (2011). Ecmaf: An event-based cross-layer service monitoring and adaptation framework, In 5th Workshop on Non-Functional Properties and SLA Management in Service-Oriented Computing (NFPSLAM-SOC'11) co-located with ICSOC 2011. Springer, 2011.

Zeginis, C., Plexousakis, D., (2010). Web Service Adaptation: State of the art and Research Challenges, Technical Report 410, ICS-FORTH, October 2010.

Zengin, A. , Marconi, A. , Baresi, L. , Pistore, M. , (2011). CLAM: Managing cross-layer adaptation in service based systems, soca, pp.1-8, 2011 IEEE International Conference on Service-Oriented Computing.

Zengin, A., Kazhamiakin, R. , Pistore, M.: "CLAM: Cross-Layer Management of Adaptation Decisions for Service-Based Applications," icws, pp.698-699, 2011 IEEE International Conference on Web Services, 2011.