Towards Autonomous Service Level Agreement Negotiation for Adaptive Service Composition

Jun Yan\textsuperscript{2,1}, Jianying Zhang\textsuperscript{1}, Jian Lin\textsuperscript{1}, Mohan B. Chhetri\textsuperscript{1}, Suk K. Goh\textsuperscript{1}, Ryszard Kowalczyk\textsuperscript{1}

\textsuperscript{1}Faculty of Information and Communication Technologies, Swinburne University of Technology, P.O. Box 218, Hawthorn, VIC, Australia 3122

\{jy.zhang, j.lin, m.chhetri, s.goh, r.kowalczyk\}@ict.swin.edu.au

\textsuperscript{2}School of Information Technology and Computer Science, University of Wollongong, NSW, Australia 2522

\textsuperscript{1}jyan@uow.edu.au

Abstract

This paper reports innovative research aiming at supporting autonomous establishment and maintenance of service level agreements in order to guarantee end-to-end quality of service requirements for service composition provision. In this research, a set of interrelated service level agreements is established and maintained for a service composition, through autonomous agent negotiation. To enable this, an innovative framework is proposed in which agents on behalf of the service requestor and the service providers can negotiate service level agreements in a coordinated way. This framework also enables adaptive service level agreement re-negotiation in the dynamic and ever-changing service environment.

Keywords: Agent, Autonomous Negotiation, Service Composition, Service Level Agreement.

1. Introduction

Service-Oriented Computing (SOC) is an emerging paradigm that utilises services as fundamental elements for developing distributed applications such as e-business processing [17]. SOC promises an approach to development in which the externally observable behaviours of the business applications are semantically described and advertised as services. Therefore, organisations are able to integrate their systems in a seamless manner by composing distributed business applications with little effort into a network of services, i.e., a service composition, to create dynamic business processes and agile applications. The most well-known integration platform for SOC is the Web Services framework which is based on a family of related XML-based standards, including the Web Services Description Language (WSDL), the Universal Description, Discovery and Integration (UDDI), and the Simple Object Access Protocol (SOAP).

The provision of a service composition implies consumption of a set of services that are dynamically purchased from other service providers. This buy and offer relationship is commonly governed by an agreement, known as Service Level Agreement (SLA), which regards the guarantees of a service provision [9]. A SLA is a contractual obligation between the service provider and the service consumer, which specifies mutually-agreed understandings and expectations of the provision of a service, in both functional and non-functional aspects. The functional aspects of a SLA are about what interactions need to be carried out to offer the service. The non-functional aspects describe a set of Quality of Service (QoS) constraints on how well the service should be offered. For example, a SLA may define the constraints on price and response time for the service provision.

SLA management has been the subject of intense research for several years. However, SLA management for the provision of service compositions, which may involve the buy and offer relationship between one service consumer and many service providers, is still at its infancy. More specifically, today’s SLA management largely relies on either the customer QoS requests on individual services or manual configuration to determine the QoS constraints on individual services in a service composition. There lacks frameworks and mechanisms to manage non-functional aspects of SLAs flexibly and autonomously, in the context of service composition provision.

In light of this situation, this paper reports innovative research on SLA management for service compositions with a focus on autonomously establishing and managing QoS aspects. To achieve this goal, this paper presents a framework for autonomous and coordinated SLA negotiation and re-negotiation using intelligent agents, for adaptive service composition provision. Based on this framework, a set of SLAs can be formed aiming at...
collectively fulfilling the end-to-end QoS requirements for the service composition provision.

The rest of the paper is organised as follows. The next section briefly introduces the major related work in the area of SLA management, and the background of this research. Section 3 presents a SLA negotiation framework for service composition provision. Based on this framework, Section 4 describes mechanisms for coordinated SLA negotiation. After that, Section 5 discusses adaptive SLA re-negotiation in dealing with exceptions. Finally, Section 6 concludes this document and outlines the future work.

2. Related work and background

Over the past a few years, active research on SLA management, mainly carried out in the context of single service offering, has covered various areas such as SLA specification and languages, SLA creation, operation, monitoring, termination, and so on. Just to name a few, [13] studied the importance of SLA management in SOA comprehensively. The Global Grid Forum (GGF) published the Web Service Agreement Specification (WS-Agreement) which is an XML language for specifying an agreement between a service provider and a consumer, and a protocol for creation of an agreement using agreement templates [1]. Keller and Ludwig presented a Web Service Level Agreement (WSLA) framework for defining and monitoring SLAs in inter-domain environments [9]. Debusmann et al. presented an approach for SLA-driven management of distributed systems using common information model (CIM) [5]. Demirkan et al. discussed an exploratory study to identify requirements on negotiation support for the processes associated with SLA development [6]. SNAP [4] defines a general framework within which a client can reserve and acquire any resource from resources providers in a uniform fashion.

The distinct research reported in this paper is carried out in the context of the Adaptive Service Agreement and Process Management (ASAPM) project which focuses on enabling flexible, dynamic and robust management of service-oriented application provision processes that are not available in the current generation of service environments. As shown in Figure 1, the overall architecture of ASAPM consists of four components:

- The Negotiation Management is responsible for autonomous SLA negotiation and re-negotiation aiming at fulfilling end-to-end QoS requirements of the service composition, as explained in this paper.
- The SLA Lifecycle Management and Dynamic Service Profiling manages SLA documents over their lifetime, and maintains up-to-date service profiles which may be used for SLA negotiation and re-negotiation.
- Workflow enactment, Monitoring and Visualisation is responsible for enactment of the service composition according to its definition, monitoring of non-functional parameters (actual QoS values), and visualisation of service composition enactment.
- Mediated Workflow Re-planning is responsible for service re-composition mediation, aiming at providing alternative plans satisfying the original user request when an unrecoverable error occurs.

Detailed information about the project, each component, and the interaction between components can be found in [18].

3. SLA negotiation framework

3.1. Solution overview

Service composition provision may involve services buy and offer between one service consumer and many service providers. Generally speaking, there are two approaches to managing this multilateral relationship. One approach is to have a single SLA, agreed by all the parties involved, specifying all the aspects of the service composition provision including the QoS constraints. Obviously, such an approach may result in an agreement that is too complex to be managed. In the case that the agreement needs to be modified as a response to an exception, even a very small change requires the endorsement of all the parties. Alternatively, this research adopts a more suitable approach by managing multiple SLAs for the service composition provision. Each SLA, agreed by the service consumer and one service provider, governs the provision of a single service. In such an approach, the non-functional aspects of each SLA define the particular QoS constraints on the provision of a single service. They collectively fulfill the end-to-end QoS requirements of the whole service composition. In an exceptional scenario, the SLA modification only involves relevant SLAs and parties. Other SLAs remain
unchanged as long as the overall requirements are still satisfied after modification.

Although the QoS constraints of each SLA can be manually and statically designated, techniques to provide flexible, autonomous and dynamic support are highly desirable. Such techniques should be able to flexibly handle QoS aggregation and trade-off between services to collectively fulfill the given overall QoS requirements. The agreement on QoS constraints can be achieved autonomously by the service consumer and the service provider through the use of software. Moreover, the pre-agreed QoS constraints can be dynamically modified during the enactment of the service composition in order to adapt to changes and exceptions.

The agent technology [16] which offers abilities of autonomous operations, interaction and cooperation has been recognized as a promising technology for SLA management in the SOC paradigm. In particular, the agent technology enables autonomous negotiation as a means of establishing service contracts [8]. In this context, the negotiation is normally driven by the operational policy of both the service consumer and the service provider. In addition, the agent technology offers ability of autonomous problem solvers that can act flexibly for QoS adaptation in dynamic and exceptional environments. Therefore, the agent technology is well positioned to address the key issues in this research.

3.2. Framework for SLA negotiation

SLA negotiation for a service composition involves two aspects. One aspect is the negotiation between the service consumer and one or many service providers for QoS constraints of a single service in the composition. The other aspect is the coordination of negotiation for multiple services to ensure end-to-end QoS. To address these two functional aspects, a comprehensive framework is designed, as shown in Figure 2.

![Figure 2. SLA negotiation framework](image)

In Figure 2, the negotiation capability of the service consumer is supported by a Multi-Agent System (MAS). This MAS consists of a Coordinator Agent (CA) and a set of Negotiator Agents (NA). The CA is responsible for the negotiation for the service composition as a whole. It interacts with the composition planner to receive the service composition definition and the overall user QoS requirements. A NA is then dynamically created to represent each service extracted from the composition. Each NA is responsible for SLA negotiation with one or many service providers for a particular service in the composition, under the guidance and control of the CA. In this framework, all the agents are supported by JADE (Java Agent Development Framework) which is a middleware that implements an agent platform and a development framework. The negotiation capabilities on the side of the service providers can be implemented in different ways, as long as they follow the same negotiation protocol and negotiation semantics. The easiest way is to have either centralized or distributed JADE agents to act on behalf of service providers. Alternatively, for the purpose of interopenability, the negotiation capabilities on the side of the service providers can be implemented as distributed Web services called Negotiation Web Services (NWS) which can be registered in and retrieved from the service registry. The communication between the JADE agents and the negotiation Web services has already been supported by a toolkit called WS2JADE [15] which enables run-time deployment and control of Web services with JADE agents and allows JADE agents to access and use Web services.

4. Coordinated SLA negotiation

4.1. QoS requirements conversion

The framework proposed in Section 3 supports coordinated SLA negotiation aiming at collectively fulfilling the given end-to-end QoS. To enable this, end-to-end QoS, i.e., the composition level requirements, need to be converted or decomposed into local QoS requirements, i.e., the service level requirements for individual services. Therefore, the negotiation objectives for each service and the coordination rules for the composition are established. In other words, the negotiation space for each service is determined.

Regarding QoS conversion, research has been done in the workflow area to calculate QoS values of a workflow process from QoS values of its constituent tasks. A well known approach called the Stochastic Workflow Reduction (SWR) algorithm is presented in [3]. The SWR algorithm repeatedly applies a set of reduction rules to a workflow until only one atomic task remains [10]. Each time a reduction rule is applied, the workflow structure changes. After several iterations only one task will remain. When this state is reached, the remaining task contains the QoS metrics corresponding to the workflow under analysis.
QoS conversion in this research is carried out by the CA in a reverse way to workflow reduction. The end-to-end QoS can be viewed as the requirements on a single composite service. While the composite service is gradually decomposed into a collection of atomic services, the overall QoS requirements are also decomposed into local requirements for these atomic services. Figure 3 depicts an example of workflow reduction and QoS conversion. For the purpose of simplicity, only two QoS parameter, i.e., cost and response time, are considered, represented as $C$ and $T$ in Figure 3. Using the workflow reduction algorithm, the workflow (i.e., the service composition) in Figure 3 which consists of services $S(i), S(j), S(k), S(m)$ and $S(n)$ can be converted into a single service $S(i,j,k,m,n)$ which has the cost requirement $C(i,j,k,m,n)$ and response time requirement $T(i,j,k,m,n)$ (i.e., the end-to-end QoS requirements). This overall requirement can be converted into requirements on $S(i,j), S(k,m)$ and $S(n)$ with specified dependencies. Finally, the requirements on individual services and the dependencies between them can be obtained. Note that the QoS requirements on a service can be specified in different formats such as the value range, e.g., cost between $\alpha$ and $\beta$, and the value condition, e.g., cost less than $\gamma$. During the QoS conversion, one important feature is that the profiling data of each service (type), which include historical performance such as maximum, minimum and average cost of using this service will be retrieved from the SLA Lifecycle Management and Dynamic Profiling (see Figure 1). These data provide estimation and expectation of the local QoS capabilities and serve as the basis for QoS conversion.

4.2. Single SLA negotiation

In the framework proposed in Section 3, single SLA negotiation refers to the process by which a NA and one or many service providers come to a mutually acceptable agreement on service QoS. In autonomous negotiation, four fundamental topics need to be taken into account. They are negotiation attributes, negotiation objectives, negotiation protocols, and decision making frameworks.

Negotiation attributes are objects to be negotiated. In this research, negotiation attributes are QoS parameters for provision of services, which may include cost of using the service, response time of the service, accessibility and reliability of the service, and so on.

The negotiation objectives are goals to be achieved through negotiation. The local QoS requirements obtained using QoS conversion are the objectives for negotiation in this research.

A negotiation protocol refers to a set of rules that describe the circumstances under which the interactions between agents take place [14]. They cover the permissible types of participants, the negotiation states (e.g. accepting offers, negotiation closed), the events that cause negotiation states to be changed (e.g. offer accepted), and the valid actions of the agents in particular states (e.g. which messages can be sent by whom, at what stage). In this research, the Iterated Contract Net Interaction Protocol (ICNIP) which is contributed by Foundation for Intelligent Physical Agents (FIPA) is exploited [7]. ICNIP supports recursive negotiation and allows for multi-round iterative negotiation to find a compromise. By exchanging modified proposals and counter-proposals, a real negotiation is carried out.

Decision making frameworks are employed by agents to act in accordance with the negotiation protocol in order to achieve their objectives. A decision making framework includes a negotiation strategy which refers to a sequence of actions (e.g., accept and refuse) for an agent to follow. The sophistication of the framework, as well as the range of decisions that have to be made, are influenced by the negotiation protocol, the nature of the negotiation attributes and the range of operations that can be performed on it. In this research, the decision making framework is based on the utility function which scores each point in the decision space of the negotiation object to represent the level of satisfaction [11][12][2]. In short, each service provider initially provides the NA a proposal specifying the service QoS it is willing to offer. The NA then calculates the utility value of each received proposal, selects the best proposal according to the utility value, and either accepts one or generates a counter proposal by making concession and/or trade-off. If a counter proposal is sent to service providers, a new negotiation iteration starts. Each service provider provides the NA a new proposal. This procedure is repeated until an agreement is achieved, i.e., a proposal is accepted by the NA or a counter proposal is accepted by one service provider, or the timeout.

4.3. Negotiation coordination

As indicated in Figure 4, negotiation coordination refers to the management of concurrent negotiation for
various services of the composition through the interaction between the CA and a set of NA.

![Negotiation protocol](image)

**Figure 4. Negotiation protocol**

The negotiation coordination may offer enhanced flexibility, as it can handle QoS aggregation and trade-off between services. As indicated earlier in Section 4.1, QoS requirements for individual services are interrelated so that they may influence each other. Therefore, if negotiation for one service has achieved a better result, which may be common especially for a service with many providers, the negotiation spaces for other relevant services can be increased if required. This can be demonstrated by the following example. Suppose that the total cost for service A and service B should not exceed $10. This requirement is converted into the requirement on service A, e.g., the cost of A is less than $4, and the requirement on service B, e.g., the cost of B is less than $6. If the negotiation for service A reaches a value $3, $1 is saved and the cost upper bound for service B can be changed to $7 if required.

The negotiation coordination may also provide additional information to NA for extra requirements or simply for better SLA negotiation. For example, if multiple services can be provided by the same provider, it is possible to have combined negotiation in order to gain aggregated concessions and preferences. In this case, the CA in Figure 4 coordinates to exchange information between the relevant NA and facilitates the decision making process.

In this research, the negotiation coordination is supported by extending the FIPA ICNIP. As shown in Figure 4, the most significant extension is on negotiation triggering and negotiation confirmation. CA triggers the NA to start SLA negotiation by sending it the reserve values defining negotiation attributes and objectives. The NA then negotiates with many providers using the standard FIPA ICNIP, followed by the confirmation of negotiation or the modification of the reserve values by the CA. Following the instructions, the NA accepts an offer or restarts negotiation, respectively.

5. **SLA re-negotiation**

The service composition will be enacted by the Workflow Enactment, Monitoring and Visualisation (see Figure 1) after all the SLA documents are successfully created. During the enactment, this component also monitors the actual QoS values of atomic service provision and detects breach of agreements. In this research, breach of a SLA means that the agreed QoS constraints are violated. For example, the response time of the service is longer than the promised interval or the cost of using the service is more than the agreed value. In this case, one or more SLAs need to be re-negotiated autonomously, aiming at bringing the service composition provision back to normal so that the original end-to-end QoS requirements can still be met. The ability of doing so will largely enhance the system’s adaptability and robustness in the dynamic service environment.

Like SLA negotiation, re-negotiation is carried out in a coordinated manner. When a SLA breach is detected, the CA is informed of the state of the service composition enactment, available SLAs, and the actual QoS values of the services that have been enacted. Based on this information, the CA will try to establish re-negotiation objectives for the service that encounters the SLA breach. The establishment of re-negotiation objectives will consider the fact that some time and money have been wasted and no results are generated. Therefore, the re-negotiation aims at collectively fulfilling the original end-to-end QoS with the actual QoS values of services that have been enacted, consumed time and cost, and agreed QoS values for unacted services. Under the coordination of the CA, re-negotiation will be carried out between the responsible NA and service providers to achieve a new agreement on QoS constraints. In more complex situations, if this re-negotiation cannot achieve an agreement, its negotiation space may need to be increased. As a result, the pre-agreed QoS constraints for some unacted services may need to be re-negotiated as well to make some compromises. Again, the CA will determine the re-negotiation objectives and coordinate the re-negotiation with service providers. Please note that SLA breach and SLA re-negotiation may lead to violation of business rules and consequently incur penalties, which are out of the scope of this research.

6. **Conclusions and future work**

Service level agreement management is a critical issue in the emerging paradigm of service-oriented computing. Adequate support for SLA management can guarantee both functional and non-functional aspects of the service provision. However, today’s research on SLA focuses on the scenario of single service provision. SLA management in the context of the service composition which consists of a set of interrelated services has not been addressed satisfactorily. In particular, support for management of non-functional aspects of SLA for individual services in a service composition in order to ensure end-to-end QoS is rather weak. There lacks mechanisms to autonomously and dynamically determine
and adjust QoS values for individual services in a composition.

This paper reports innovative research conducted in the context of the ASAPM project. In this research, SLAs for service compositions, especially those non-functional aspects of SLAs, are expected to be established and managed autonomously, dynamically and adaptively. To achieve this goal, the agent technology is exploited in this research and a novel framework for coordinated SLA negotiation is presented. In this framework, autonomous negotiation can be carried out in a coordinated way to determine QoS constraints for individual services that collectively fulfil end-to-end QoS. This framework also supports SLA re-negotiation in dealing with SLA breach during the service composition enactment.

The framework presented in this paper builds a solid foundation for the future research work. Based on this framework, the negotiation techniques will be extended to support more QoS attributes. More complicated decision-making mechanisms will be developed. The proof-of-concept protocol will be implemented and integrated with other components to provide a complete solution to adaptive service agreement and adaptive process management.

Acknowledgement

This work is jointly supported by the Adaptive Service Agreement and Process Management (ASAPM) in Services Grid project (AU-DEST-CC060081), the EU FP6 Integrated Project on Adaptive Services Grid (EU-IST-004617), and Australian Research Council under Discovery Grant DP0663841.

References


