TOWARDS DEPENDENCY MANAGEMENT IN SERVICE COMPOSITIONS

Matthias Winkler
SAP Research CEC Dresden, SAP AG, Dresden, Germany

Alexander Schill
Chair for Computer Networks, TU Dresden, Dresden, Germany

Keywords: Service dependency, Business service, Service composition, Service monitoring, SLA negotiation.

Abstract: Business services are a valuable asset to be traded on internet service marketplaces. While they are offered via the internet their execution often involves manual steps. The provisioning of services is regulated by service level agreements (SLA). The composition of business services enables the creation of innovative business processes which can be offered as services again. Selling these service compositions brings along challenges for handling two important tasks, namely the monitoring of SLA violations and SLA renegotiation. Both tasks are challenging because they affect not only a single service but multiple stakeholders of the composition. In this paper we discuss the problem of dependencies between services in compositions based on a scenario from the logistics domain. Dependencies between services are problematic because they lead to situations where the SLA violation of one service affects the provisioning of other services. Similarly, the renegotiation of the SLA of one service has effects on the SLAs of other services. We present a conceptual architecture and an approach towards a solution for handling service dependencies.[a]

[a]The project was funded by means of the German Federal Ministry of Economy and Technology under the promotional reference "01MQ07012". The authors take the responsibility for the contents.

1 INTRODUCTION

Within the TEXO project (Pressebuero, 2008) the foundations for making business services tradable on service marketplaces are established. Business services are any kind of business activity offered by a service provider to create value for a consumer. They are offered via the internet but may be provided involving manual tasks executed by humans or machines. The provisioning and consumption of services is regulated by service level agreements (SLA). Monitoring SLAs during service provisioning is important to ensure efficient provisioning of the service and to build up trust between the stakeholders, i.e. the providers of atomic services, the providers of composite services, and the service consumers (Winkler et al., 2008). In order to better distinguish the composite service from its contained services, we call the services, which are the building blocks of a composition, atomic services, knowing that they may be composite services themselves.

When atomic services are composed to service compositions dependencies regarding their execution may exist. Problems during the execution of one atomic service may hinder the execution of other services. Problems are detected as service level objective (SLO) violations. SLOs are the single measurable elements of an SLA. In cases where services are affected by problems, the service composition may be adapted (e.g. exchange services, renegotiate SLA). Also, there are situations where one stakeholder of the composite service wants to renegotiate its SLA and thus affects the other SLAs of the composition. Although these dependencies have an impact on the execution of the composition, they are currently neither directly described nor managed. Information regarding dependencies is indirectly available, though, in the different SLAs negotiated between the composite service provider, the atomic service providers, and the consumers of the composition as well as in the underlying process of the composite service.

Based on a use-case from the logistics domain (section 2), we will discuss the problem of service dependencies in service compositions (section 3), outline our envisioned solution and present the results of first steps towards this solution (section 4). Finally,

Winkler M. and Schill A. (2009).
TOWARDS DEPENDENCY MANAGEMENT IN SERVICE COMPOSITIONS.
In *Proceedings of the International Conference on e-Business*, pages 79-84
DOI: 10.5220/0002188600790084
Copyright © SciTePress

we discuss related work (section 5).

2 A COMPOSITE LOGISTICS SERVICE SCENARIO

Logistics services (e.g. transport and storage of goods) are an example of business services. In our scenario (see Fig. 1) the 4PL (fourth party logistics provider) ”Global Transport” acts as organizer of a logistics process offered as composite service ”Transport DD-HH”. The process is created from three services offered by different providers. The service is bought by “Heal Pharma”, which needs to ship pharmaceutical goods from Dresden to Hamburg. Service “Truck DD“ offered by provider ”DD Logistics” is responsible for picking up the goods at the “Heal Pharma” factory in Dresden and delivering them to the train station. There the goods are received by provider ”Railway Services” and shipped to Hamburg via service “Train Shipping DD-HH”. Finally, “HH Logistics” delivers the goods to the customer of ”Heal Pharma” via service“Truck HH”. During the provisioning of this composite service the 4PL manages the underlying process.

Figure 1: Simple Logistics Process.

3 SERVICE DEPENDENCIES IN SERVICE COMPOSITIONS

When services are composed to service compositions, they are implicitly collaborating to achieve a business goal. We consider collaboration of services as implicit because the atomic service providers may not be aware of the full process and the different services involved. Only the 4PL has all the information.

In our use case dependencies occur regarding the time of transport and the amount of goods. These aspects are regulated by SLAs. The 4PL needs to assure that the SLAs negotiated with the different providers enable the smooth execution of the process. Problems during the execution of one atomic service or the renegotiation of its SLA may affect other atomic services of the composition as well as the composition as a whole. We will now analyse different types of dependencies and illustrate them in our logistics use-case.

3.1 Types of Dependencies

We classify dependencies with respect to their occurrence either between atomic services or between atomic services and the composite service. We call dependencies, which occur between atomic services of a service composition, *horizontal dependencies*, because they affect services on the same hierarchical level of composition. In cases where horizontal dependencies are managed properly they might not show to the consumers of the composition. However, this is not always possible. In such cases the effects of the problem will be propagated through the whole composition and thus affect it. Problems would then be visible to the consumers of the composition. Horizontal dependencies can be further classified into direct and indirect dependencies. Direct dependencies occur between two interacting atomic services. Indirect dependencies occur between services which do not directly interact with each other, but where a transitive relationship exists via an intermediate service.

Problems of an atomic service may affect the overall composition. We call these dependencies, which occur across a hierarchical level of composition, *vertical dependencies*. If vertical dependencies exist, SLO violations of atomic services as well as their renegotiation affect the SLA negotiated with the customer. In many cases horizontal and vertical dependencies occur at the same time. Fig. 2 illustrates the presented types of dependencies.

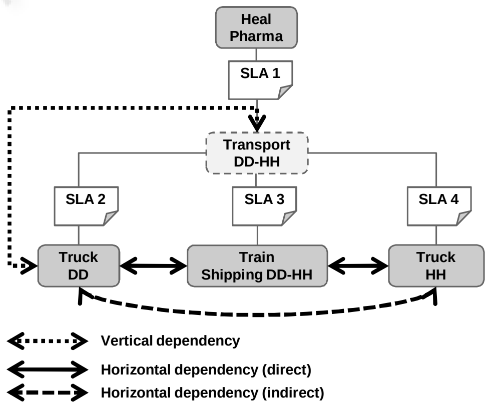

Figure 2: SLAs and Dependencies in a Composite Service.

3.2 Dependencies in a Logistics Process

We will now illustrate the different types of dependencies based on two SLO violation examples of our logistics use case: the delayed delivery of goods and the delivery of a false quantity of goods. Late delivery: The service ”Train Shipping DD-HH” follows a fixed schedule which sets the departure time of the train to 6pm each day. In order for the transshipping of goods from truck to train to be completed by that time, goods need to be delivered to ”Train Shipping DD-HH” by 4pm. When the train has reached Hamburg the next morning, the goods need to be picked up by the ”Truck HH” service between 6am and 8am. Violations of such time constraints may lead to problems for multiple stakeholders of the process. If ”Truck DD” cannot deliver the goods on time, the 4PL needs to inform not only ”Railway Services” as the provider of “Train Shipping DD-HH” but also ”HH Logistics”, the provider of “Truck HH”, since there will not be any goods to be picked up the next day. While the dependency between ”Truck DD” and ”Train Shipping DD-HH” is quite obvious (direct horizontal dependency), the dependency between ”Truck HH” and ”Truck DD” is not. This is an indirect horizontal dependency which can only be derived by analysing the underlying process of the composite service. It may also be necessary for ”Global Transport” to renegotiate the SLA with “Heal Pharma”. Here we find a vertical dependency between ”Truck DD” and ”Transport DD-HH”.

False quantity of goods: ”Heal Pharma” contracted ”Global Transport” to transport 10 pallets of pharmaceutical goods. In case that an error occurs during the loading or unloading of the goods, parts of the cargo might get lost. Let us assume that the error occurred during the loading of the truck at the warehouse of ”Heal Pharma”. The problem will be recognized during the unloading procedure at the train station and is propagated to ”Global Transport” as a violation of the respective SLO by ”Truck DD”. Now all contracts with the other services (”Train Shipping DD-HH” and ”Truck HH”) need to be adapted in such a way that the transport of goods should be carried out for 9 pallets only. These are examples for horizontal dependencies. There is also a vertical dependency to the composite service due to the fact that it also cannot be completed successfully any more. Thus, the contract with ”Heal Pharma” will be violated or needs to be renegotiated.

It might not always be necessary to adapt a process once problems occur. The SLA might state that deliveries need to be on time in 95 percent of all transports. Nevertheless it is important for many logistics providers to avoid unnecessary costs. Such costs may occur, for example, when ”Global Transport” does not inform ”HH Logistics” about the fact that the goods do not need to be transported any more. Providing stakeholders with the right information or renegotiating SLA are possible ways for avoiding unnecessary costs.

4 MANAGING SERVICE DEPENDENCIES

In many cases service dependencies cannot be avoided but need to be actively managed by the composite service provider. In this section we present an approach which will support this work.

Our solution approach is based on an architecture for service modeling and description, SLA (re-)negotiation, and service monitoring. It is extended by functionality for service dependency analysis in service compositions based on SLAs and the business process and a central model for capturing service dependencies. This dependency model is created during dependency analysis at the time of the creation of the service composition and evaluated at runtime upon the occurrence of SLO violations or a request for SLA renegotiation.

In this chapter we present our conceptual architecture, explain our approach to the formal modeling of service descriptions and SLAs, and present first steps towards a dependency model.

4.1 Conceptual Architecture

The proposed architecture, which is shown in Fig. 3 (in FMC notation[1]), consists of the service design time tools (ISE Development Environment), a marketplace for trading services (Service Management Platform - SMP), and a runtime environment (Tradable Services Runtime - TSR). It is based on joint work together with our partners within the TEXO project.

The ISE Development Environment (Cardoso et al., 2008) supports the modeling and description of services based on the Universal Service Description Language (USDL) (Cardoso et al., 2009) and the creation of service compositions. The analysis of service dependencies and the creation of a dependency model takes place as part of the development process. This will be explained in more detail later. Following the successful development of a service, it is deployed to a TSR and registered at the SMP.

The SMP enables service providers to register their services and thus make them available to interested consumers (creators of compositions as well as end users). Besides functionality such as service registration and billing for service usage, SLA negotiation is supported via the SLA Manager component. We implemented SLA negotiation based on the WS-Agreement specification (Andrieux et al., 2007). SLA

[1]Fundamental Modeling Concepts website: http://www.fmc-modeling.org/home

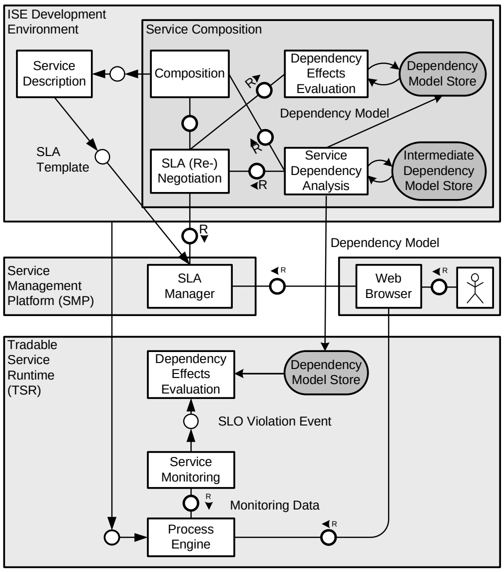

Figure 3: Conceptual Architecture.

templates are offered and can be requested from interested consumers, who will then modify the template according to their specific needs. The resulting agreement proposal document will then be returned to the SLA Manager and forwarded to the service provider, who may decide to either accept or reject the agreement. The final SLA will then be stored by the SLA Manager from where it will be available for the provider as well as the consumer for monitoring purposes.

The TSR provides support for executing and monitoring services. To determine SLO violations, the service monitoring component accesses SLA information and compares negotiated SLO with monitoring data. Upon the detection of SLO violations the dependencies on other services are evaluated by the Dependency Effects Evaluation component.

4.2 Modeling Services and Service Level Agreements

USDL forms the base for the formal description of operational, technical, and business attributes of services. It covers attributes which are common for most services independent of their nature (rather technical or more business oriented), including provider information, terms of use and pricing. At the same time USDL does not cover attributes which are specific to any application domain of a service. Thus, we have extended USDL with a language called *Logistics4USDL* to express special attributes of the logistics domain such as delivery time and transport capacity, to only mention a few. The following listing presents a simplified version of a USDL service description enriched with Logistics4USDL elements.

```
service {
 serviceName Train Shipping DD-HH
 description Goods transport via train
 business {
  providerName Railway Services
  providerAddress Traubestr. 17, Dresden
  log4usdl:logistics{
   log4usdl:quote{
    log4usdl:pickUpAddress Seestr. 1, Dresden
    log4usdl:deliveryAddress Im Tal 3, Hamburg
    log4usdl:pickupTime 2009-02-10T16:00:00Z
    log4usdl:deliveryTime 2009-02-10T16:00:00Z
    log4usdl:transportCapacity 10 pallets
  }}
  reliability 95%
  termsOfUse www.rw-services.com/ToU.html
 }
 operational{...}
 technical{...}
}
```

This formal service description provides the base for formalizing contracts (SLA) (Cardoso et al., 2009). SLA templates are created from the service description as part of the service engineering process and made available at the SLA Manager component for SLA negotiation. These templates are based on the WS-Agreement specification and are augmented with service information using the USDL and Logistics4USDL notations.

Based on the SLA information dependencies between different services are analysed and formalized in a dependency model. The analysis of dependencies requires a formal description of SLAs in order to be able to relate service aspects of different services with each other. This formal base is provided by USDL and Logistics4USDL. A simplified version of an agreement including USDL and Logistics4USDL terms is presented in the following listing.

```
agreement {
 Name Train Shipping DD-HH SLA
 ServiceDescriptionTerm {
  ServiceName Train Shipping DD-HH
  usdl:description Goods transport via train
  usdl:providerName Railway Services
  usdl:providerAddress Traubestr. 17, Dresden
 }
 ServiceProperties {
  VariableSet {
   Variable {
    Name deliveryTime
    Metric xsd:dateDuration
   }
   Variable {
    Name transportCapacity
    Metric log4usdl:pallet
 }}}
 GuaranteeTerm {
```

```
  Name BasicService_GUARANTEE
  monitored true
  ServiceScope Train Shipping DD-HH
  ServiceLevelObjective {
   SLOName deliveryTime
   ServiceLevel 2009-02-10T16:00:00Z
  }
  ServiceLevelObjective {
   SLOName transportCapacity
   ServiceLevel 10
}}}
```

4.3 The Dependency Model

We are currently developing a model for representing the dependencies between services in service compositions. In (Bodenstaff et al., 2008), (Ludwig and Franczyk, 2008), and (Zhou et al., 2008) three approaches to expressing dependencies are presented. While they form a good base for our work, they have some shortcomings. While the first two approaches are only concerned with vertical dependencies, the third one is not concerned with dependencies on the base of SLAs. Please see the related work section for more details. Thus, they need to be extended to fit the needs of business services from the logistics domain.

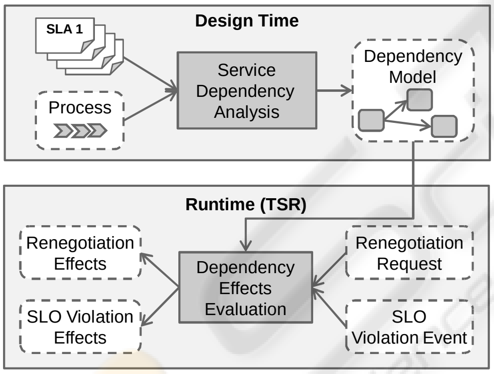

Figure 4: Dependency Analysis and Dependency Effects Evaluation.

The dependency model will be created at design time containing information about the SLAs which were negotiated with the service consumer and the different service providers, as well as the horizontal and vertical dependencies between different SLOs of all involved SLAs.

The creation of the dependency model will take place in two steps. During the process of creating the service composition, SLAs are negotiated with the single service providers. The formal description of SLOs based on USDL and Logistics4USDL will be used as a base for determining the dependencies. The created process determines the relationships among services and thus helps analysing the dependencies. The dependency model will be updated once an SLA has been negotiated with a consumer of the composite service.

We currently envision an automatic approach to dependency analysis and model creation based on the formal process and SLA description. This process is shown in Fig. 4. Further work is necessary to determine whether additional manual modeling effort will be necessary.

Dependency effects evaluation is triggered by SLO violations detected during service monitoring or renegotiation requests. The Service Dependency Monitoring component uses the dependency model for evaluating horizontal and vertical dependencies and provides information on affected services (see Fig. 4). Finally the handling of detected issues is started.

5 RELATED WORK

Most approaches to SLA monitoring (Flehmig et al., 2006; Ameller and Franch, 2008) are concerned with the evaluation of single contracts between a service provider and a consumer. This is not sufficient in the context of composite services since they have horizontal and vertical dependencies between the different services which need to be considered. Thus, we build on the different approaches and extend them with the functionality for managing service dependencies.

The approach presented in (Bodenstaff et al., 2008) monitors vertical service dependencies. A model for formalizing SLA dependencies is described. It covers the aspects of response time and cost. This approach helps to determine to what extend different services are responsible for SLO violations of a composite service. Problems occurring between single services are not covered. Thus, the approach is too limited for the comprehensive management of dependencies between services in compositions.

The COSMA approach (Ludwig and Franczyk, 2008) supports the management of SLAs in service compositions where the SLOs negotiated for the composite service depend on the respective SLOs negotiated for the atomic services of the process. In a central document called COSMAdoc the vertical dependencies for different QoS parameters as well as price information are expressed using aggregation formulas. Horizontal dependencies between the different services are not handled.

In (Zhou et al., 2008) the authors discuss control and data dependencies in business processes in the light of sequencing constraints. They present an approach for deriving a dependency model from semantically annotated business activities by evaluating their pre-conditions, effects, and parameters. It is destined to support the handling of sequencing constraints at runtime. No work has been done with regard to analysing SLAs, representing dependencies between different SLOs or for the determination of SLO violation and renegotiation effects.

Related to our work is also the functionality provided by Business Activity Monitoring (SAP, 2006) approaches. Their goal is to monitor business activities (e.g. sales process) spanning across organizational boundaries. Events from different sources are collected and correlated in order to determine problems such as conflicting quantities of line items between an order and received goods. This provides business people with information about problems within processes only after they have occurred. It enables them to quickly react and potentially solve the problem. It does not enable the prediction of effects of the problem on other parts of the business process.

6 CONCLUSIONS

In this paper we have presented the problem of horizontal and vertical dependencies between services in compositions and argued for a need to manage them. We have illustrated these dependencies based on two examples from the logistics domain.

We also presented our solution approach to dependency management in form of a conceptual architecture. It supports the development of composite services, trading of services on a marketplace, and service provisioning via a service runtime platform. The management of dependencies consists of two main steps: the analysis of dependencies and creation of a dependency model at design time, and the dependency effects evaluation at runtime based on the dependency model. Components for analysing and evaluating service dependencies were integrated into the conceptual architecture as important building blocks.

As first steps towards the analysis of service dependencies we presented how logistics services can be formally described based on the Universal Service Description Language and the Logistics4USDL service description extension. This description provides the base for formalizing SLAs which are then analysed for dependencies between services.

We outlined our envisioned dependency model, as well as steps and artefacts needed to create this model at design time and evaluate it at runtime. While large parts of the conceptual architecture have already been implemented as part of the TEXO project, the development of the dependency model, as well as the dependency analysis and evaluation components is ongoing work.

REFERENCES

Ameller, D. and Franch, X. (2008). Service level agreement monitor (salmon). In *ICCBSS '08: Proceedings of the Seventh International Conference on Composition-Based Software Systems (ICCBSS 2008)*, pages 224–227, Washington, DC, USA. IEEE Computer Society.

Andrieux, A., Czajkowski, K., Dan, A., Keahey, K., Ludwig, H., Nakata, T., Pruyne, J., Rofrano, J., Tuecke, S., and Xu, M. (2007). Web services agreement specification (ws-agreement). Technical report, Open Grid Forum.

Bodenstaff, L., Wombacher, A., Reichert, M., and Jaeger, M. C. (2008). Monitoring dependencies for slas: The mode4sla approach. In *IEEE SCC (1)*. IEEE Computer Society.

Cardoso, J., Voigt, K., and Winkler, M. (2008). Service engineering for the internet of services. To appear in Enterprise Information Systems, Lecture Notes in Business Information Processing (LNBIP).

Cardoso, J., Winkler, M., and Voigt, K. (2009). A service description language for the internet of services. Proceedings of ISSS 2009.

Flehmig, M., Troeger, P., and Saar, A. (2006). Design and integration of sla monitoring and negotiation capabilities. Adaptive Services Grid Deliverable D5.II-7.

Ludwig, A. and Franczyk, B. (2008). Cosma an approach for managing slas in composite services. In Bouguettaya, A., Krueger, I., and Margaria, T., editors, *ICSOC 2008*, number 5364 in LNCS, page 626632. Springer-Verlag Berlin Heidelberg.

Pressebuero, T. (2008). Texo - Business Webs im Internet der Dienste. http://theseus-programm .de/scenarios/de/texo.

SAP (2006). Business activity monitoring. White paper, SAP AG.

Winkler, M., Cardoso, J., and Scheithauer, G. (2008). Challenges of business service monitoring in the internet of services. In *Proceedings of iiWAS2008*. Linz, Austria.

Zhou, Z., Bhiri, S., and Hauswirth, M. (2008). Control and data dependencies in business processes based on semantic business activities. In *Proceedings of iiWAS2008*. Linz, Austria.