PROSA – Process-Oriented Service Level Agreements for Providing a Single Point of Contract[1]

Rico Knapper[1], Tobias Conte[1], Benjamin Blau[1], Hans Richter[2]

[1]Institute of Information Systems and Management (IISM),
KIT Karlsruhe Institute of Technology
{firstname.lastname}@kit.edu
[2]EnBW Energie Baden-Württemberg AG
h.richter@enbw.com

1   Introduction

The fundamental paradigm shift from a product- to a service-oriented economy implies novel technical and organizational challenges. The value generated by a service is mainly represented by intangible elements exposed at execution (Hill 1977). Therefore, a service consumer expects a service to function reliably and to deliver a consistent outcome at a variety of levels, i.e. Quality of Service (QoS). To provide, control and assure QoS it is necessary to focus on functional properties of a service as well as on non-functional aspects. The context of a service also influences its quality, which is experienced by the consumer, e.g. the partner network that comes with a service, its reputation in certain communities or advertisement campaigns promoting the service. From an economic perspective, QoS is the most important characteristic that differentiates service offerings and leverages market advantage, as price competition is tough due to low variable costs of service provisioning. Thus, QoS is the key criterion to keep the business side competitive as it has serious implications on the provider and consumer side (Papazoglou 2008). The provision of services with a defined QoS over electronic networks such as the Web is challenging due to issues like infrastructure problems, unpredictable reliability, low performance of Web protocols and many more. In addition, the distributed nature of Web service environments and their high degree of complexity requires a comprehensive description of Web service quality characteristics, both functional and non-functional. For detailed information about the main aspects of

---

QoS in a Web service context, the interested reader is referred to (Cardoso et al. 2004; Liu et al. 2004; Mani and Nagarajan 2002; Papazoglou 2008; Zeng et al. 2003).

From a business perspective, QoS characteristics defined based on technical services within the infrastructural layer have to be aggregated to more business-relevant Key Performance Indicators (KPIs) (Blake and Cummings 2007; Jaeger et al. 2004; Unger et al. 2008). KPIs represent service quality that is highly related to the business's performance (e.g. maximum downtime of a business service) and are crucial for achieving predefined goals in order to stay competitive in the market. An agreement between service provider and service consumer about the quality to be delivered must be founded on a legal basis, i.e. by specifying a Service Level Agreement (SLA). A SLA is a contract that defines mutual understandings and expectations of a service between service provider and service consumer (Jin et al. 2002). It defines service characteristics and the quality to be delivered by the provider i.e. Service Level Indicators (SLIs), Service Level Objectives (SLOs) and monetary penalties in case of non-performance. Such a contract represents a guarantee for the service consumer, which assures the delivery of the defined quality or an adequate charge-back mechanism. Service-oriented architectures (SOAs) enable the seamless integration of distributed services – even across organizational boundaries – into end-to-end business processes. Companies tend to concentrate on their core competencies while requesting modularized business services from different service providers. The situation today requires that companies agree on multiple SLAs that specify the SLOs for each business service as depicted in Figure 1. This implies a tremendous effort for service providers as well as for service requesters that results from monitoring and managing the contractual relations.

In this paper we provide a comprehensive methodology for implementing Process-oriented Service Level Agreements (PROSA) which enables a single point of contract for service requesters while significantly reducing the managerial overhead generated from multiple contractual relations. We evaluate our concept by providing an industrial case study from the application of smart metering in the EnBW AG. The evaluation results are discussed in detail and managerial implications are derived that lead to strategic recommendations for implementing PROSA in an enterprise context.

The remainder of the paper is structured as follows: Section 2 provides an overview on related literature. The PROSA methodology is outlined in Section 3, describing both the provider and the customer perspective. Subsequently, an industrial case study underpins the PROSA methodology. We conclude with a summary and our future research agenda.
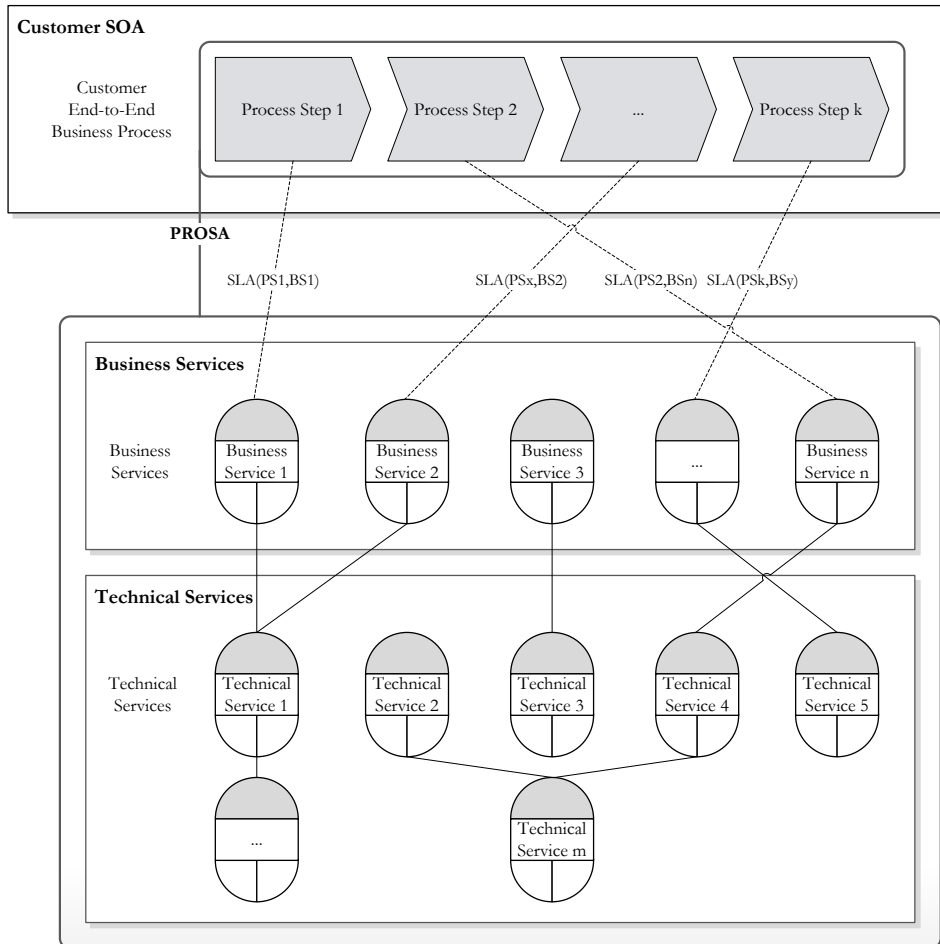
Figure 1: Legal framework for flexible binding of external services into end-to-end business processes. Situation today requires multiple SLAs for fine granu lar services. In contrary, the PROSA methodology enables a single point of contract for customers.

2   Related Literature

There is a wide choice of related literature concerning the aggregation of SLAs. Approaches of close relationship to our work can be roughly categorized in three areas whose boundaries blur to some extent. Models which aggregate the SLOs of single SLIs in a mathematical way are introduced in (Blake and Cummings 2007; Jaeger et al. 2004; Unger et al. 2008). Models which cover the PROSA characteristic of being a document, that is, they provide a framework for building a single

document out of a set of SLA documents (SLAs of the single services invoked by one BP) are discussed in (Ludwig and Franczyk 2008; Muthusamy et al. 2007; Schmidt 2000). Finally, (Daly et al. 2002; Xiao et al. 2008) elaborate models which validate the SLOs on BP level by means of simulations. The related work delivers some valuable insights into two main aspects of the in hand research domain. On the one hand approaches to technically aggregate SLIs and on the other hand approaches which deal with the SLA characteristic of being a document that is aggregating the single SLAs to one document. But some highly interesting and important issues are not covered. Presented models are bottom-up-approaches. Looking at the motivation our approach is customer-oriented. That is a customer who wants to facilitate his business processes by IT-services delivers the objectives concerning the SLOs of the PROSA to the provider(s). Therefore these objectives have to be drilled down to a deep level of technical services – a top-down-approach. Whereas a bottom-up-approach deals with the attributes of technical services and aggregates them bottom-up which is not suitable for our addressed issues. Additionally the mentioned approaches do not cover both aspects customer-orientation and provider-methodology. They are all driven by the providers' perspective. In summary, current approaches deliver first contributions to the domain of SLA aggregation. But they do not cover the customer as well as the provider perspective in an adequate way. Especially the motivated customer orientation is not represented as much as required.

3   The PROSA Methodology

As stated in Section 1 the customer delivers SLOs on business process level to the provider(s). In contrast to current academic contributions mainly focused on the provider perspective, we additionally include the closer examination of the customer perspective in order to present a holistic framework.

Consequently, the big picture of PROSA is a two-sided approach with the involvement of the customer as well as the provider of the services. The main goal of PROSA is bringing both sides together on the level of the customer's BP. Provider-sided we introduce a methodology to aggregate non-functional aspects of Services as exemplified by the SLI "maximum downtime" (Section 3.1). On customer-side, we recommend the adoption of service value networks (SVN) as presented in Section 3.2. The definition for PROSA evolves as follows:

Definition *PROSA is a methodology to implement a single point of contract (SPC) to specify the legal relationship between service providers and a service customer to ensure a single contract concerning all invoked services in respect of the whole customer business process.*

3.1   The Service Provider Perspective

The provider has to break down the customer-delivered SLOs to smaller subordinated units of the business services[2] he provides. These units are the technical services[3], shared technical services[4], underpinning services[5] und the therein used configuration items (CIs)[6]. To break down the customer delivered SLOs and to aggregate the values afterwards, an aggregation metric is needed which observes customer orientation and hence is a metric fulfilling the top-down-paradigm mentioned above.

In the following we present a methodology to drill-down the given SLOs. As above-mentioned, we consult maximum downtime as exemplary SLI. However, PROSA can be used with nearly every non-functional aspect which is used as an SLI and provides a numerical co-domain. As depicted in Figure 2, four elementary steps have to be passed through in the procedure model of the methodology. All steps are explained in detail in the remainder of this section.
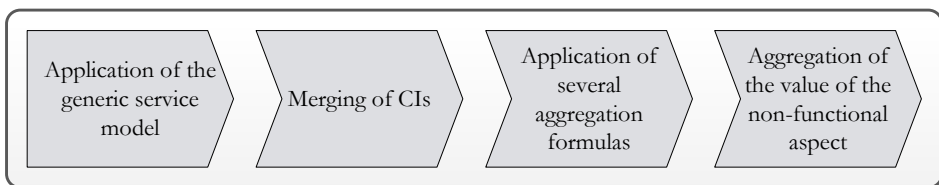

Figure 2: Procedure Model to Establish PROSA from Provider Perspective

The generic service model
The generic service model is an abstract manual for modeling the BP, the business services and the technical services to apply the procedure model. It consists of nine swim lanes representing a type of CI each: Business process, business service, application, database, virtualization, server hardware, storage, infrastructure, and other. In addition there are relation types defined to model the links between the CIs. By our definition only the strongest valid relations are depicted.

---

[2] Service based on the adoption of IT which is delivered to one or more customers by a service provider and supports the BP of the customer.

[3] Technical services are internal subordinated services to support business services. Therefore a technical service is not used by a customer; it is needed by a provider to support a business service he provides.

[4] Services (business services as well as technical services) which are not specific for one single service are called shared service. Therefore services which provide functionalities invoked by different services are shared services.

[5] Technical services which are delivered by an external provider are so called underpinning services. Indeed they have to be treated like regular technical services concerning the aggregation.

[6] Every IT component which delivers a certain needed functionality (Elsässer 2006)

Table 1 shows a list of possible relation types. To differentiate between the types in the model based on the swim lanes, different colors are used. Relations are always directed.

Table 1: List of Possible Relation Types

Relation Type CI1→ CI2	Selection Criterion
Depends on (black)	CI2 not available → CI1 not available
Redundantly depends on (green)	CI2 not available → CI1 available as long as additional CIs with relation type "redundantly depends on" are available
Depends on with time buffer (red)	Although CI2 is not available CI1 is available for a defined time

Application of the generic service model
Instantiating the generic service model to a given process/service by breaking down the BP to CIs, applying every CI to a swim lane and utilizing the relation types to link the CIs, the specific service model of this process is created.

Shared services do not have to be drilled-down. They are used as encapsulated units in the service model with pre-calculated values of the non-functional aspect (calculated using the same methodology).

Merging of CIs to obtain technical services, shared technical services and underpinning services
Having created the service model with the involved CIs the technical services have to be derived from these CIs. Like business services the technical services are available in different categories (e.g. standard, advanced, premium). It has to be decided whether a new technical service has to be registered or a new category for an existing service is sufficient.

Possible reasons to create a new category are equal service descriptions, equal underlying infrastructure, different service qualities, or equal service manager

After identifying the business services, technical services and underpinning services the so called service hierarchy can be build. To distinguish between the different types of services different colors are used.

Application of several aggregation formulas to the links in the service hierarchy and aggregation of the non-functional aspect using the aggregation formulas
Having created the service hierarchy the ties in the directed graph of the service hierarchy have to be classified. The classification depends on the distinction how the subordinated technical services depend on each other concerning the aggregation. In respect to the SLI "maximum downtime", our model includes three aggregation types as listed in Table 2.

Table 2: Aggregation Types

Operation	Explanation
additive (+)	Values from the subordinated technical service and of the superior (technical) service have to be added together regarding the aggregated non-functional aspect.
no significance (X)	Values from the subordinated technical service do not have any importance for the superior (technical or business) service regarding the aggregated non-functional aspect.
maximum (MAX)	The maximum value of the aggregated non-functional aspect either of the subordinated technical service and the superior (technical) service or of all parallel links with the same superior (technical) service is taken.

In Section 4 we provide an industrial case study illustrating the concrete adoption of the methodology.

3.2 An Integrated View incorporating the customer perspective – Service Value Networks

As described earlier, PROSA allows for an agreement with aggregated SLOs on a customer business process level for each SLI. Therefore it is not the single process step but the overall business process that matters for the customer's choice of services which is based upon the utility the process yields for the customer (e.g. through minimizing costs or maximizing the service fit). When composing business services, service attributes or prices of the whole process are considered instead of the characteristics of single business services since only the full process yields a positive utility for the customer.

Such business software faced radical changes recently. Large monolithic software applications implying are being replaced by the on-demand delivery of flexible service components operated in service oriented architectures (SOA). Such modular components can easily be adapted and extended by additional services. This conceptual and technical change offers customers the opportunity to purchase services on-demand, contractual ties to single vendors are being dissolved. Service modules can rather be composed from the offers of different providers tightly focused on required features – pushing the center away from the actual vendor that becomes less important from a customer perspective. Such modularity is one of the most promising answers to the question of how to face rising demands for sophisticated, customized services. Once serving the whole value chain by what has become famous as vertical integration, service providers now tend to

engage in networked value creation in ecosystem-like environments called service value networks (Blau et al. 2009).

Figure 3 depicts the main components of service value networks (SVNs) and their interdependencies in a simplified manner.
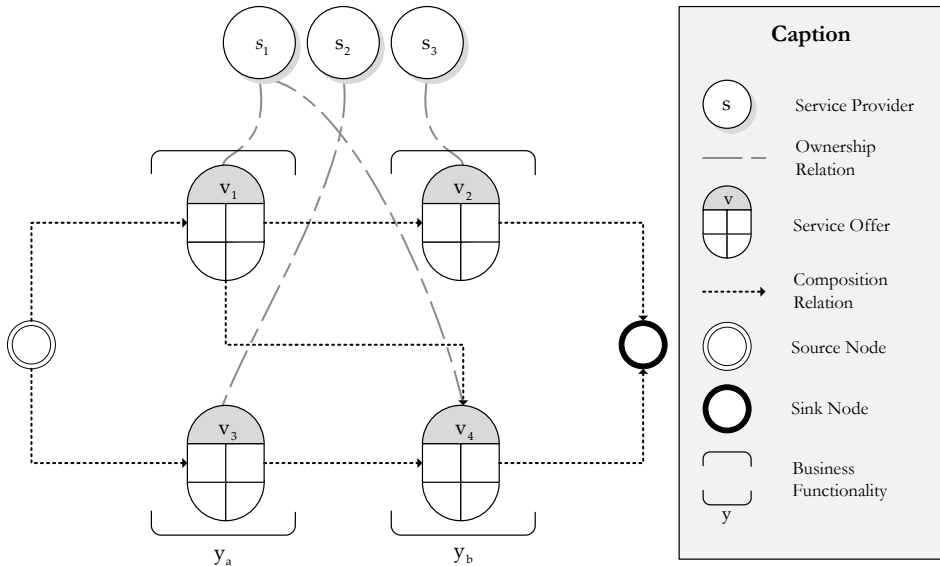

Figure 3: Service Value Network Model According to (Blau et al. 2009)

A service value network consists of a set of *service providers* $s \in S$ that supply a portfolio of *service offers* $v \in V$. Service offers that are compatible, i.e. they are interoperable regarding their interfaces and input and output capabilities, expose a directed *composition relation*. Their connections form a graph-like structure that is directed and acyclic starting from a source node and ending at a sink node. Each feasible connected set of service offers within this graph is called a *path* and represents a possible instantiation of a *customer process* that consists of functionality from each *candidate pool*. That is, the service value network in Figure 3 depicts a specific customer request which consists of the *business functionalities* y_a and y_b. Each of these functionalities can be performed by a multitude of substitutive business services which are potential candidates for the process to be purchased. Each service provider can own one or multiple service offers indicated by an *ownership relation*. However, from a customer perspective the ownership relations are not relevant as the process is most probably provided by several providers anyway whose SLAs are aggregated to an overall PROSA. According to the example in Figure 3 such a process can be reproduced either by a composition of v_1 and v_2 or v_1 and v_4, or v_3 and v_4.

4 Industrial Case Study – Smart Metering

The presented methodology was successfully validated by applying it to the EnBW AG. Part of this company is a subsidiary which is dealing with IT services of all kinds. Other subsidiaries of the company exclusively use these services. Therefore the methodology was validated in a field which delivers all aspects from service customers and providers as both sides are represented in this company. For illustration we took only a small part out of a complex BP which deals with the functionality of a product called "smart electric meter". This product provides a real-time based online version of every customer's electric meter to have a real-time overview concerning the current consumption. The meter is called "smart" because by-and-by it learns which electricity consumers are powered and gives hints how to save energy. In order to have a low level of complexity, our example deals only with a small part of the BP with a single invoked service. The example at hand can easily be transformed to any business case with a finite number of invoked services.
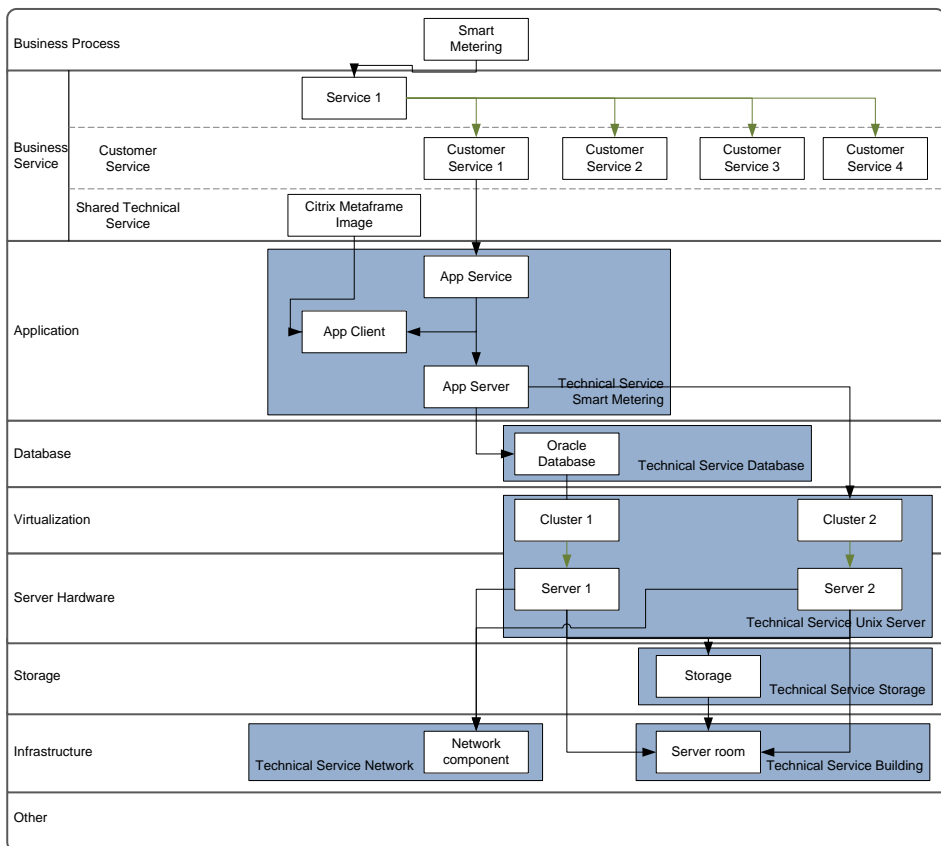

Figure 4: Adapted Service Model

Figure 4 shows the adapted service model with swim lanes, relation types, CIs and as introduction to
Figure 5 the technical services which have to be derived out of the service model. Out of it the service hierarchy can be built.
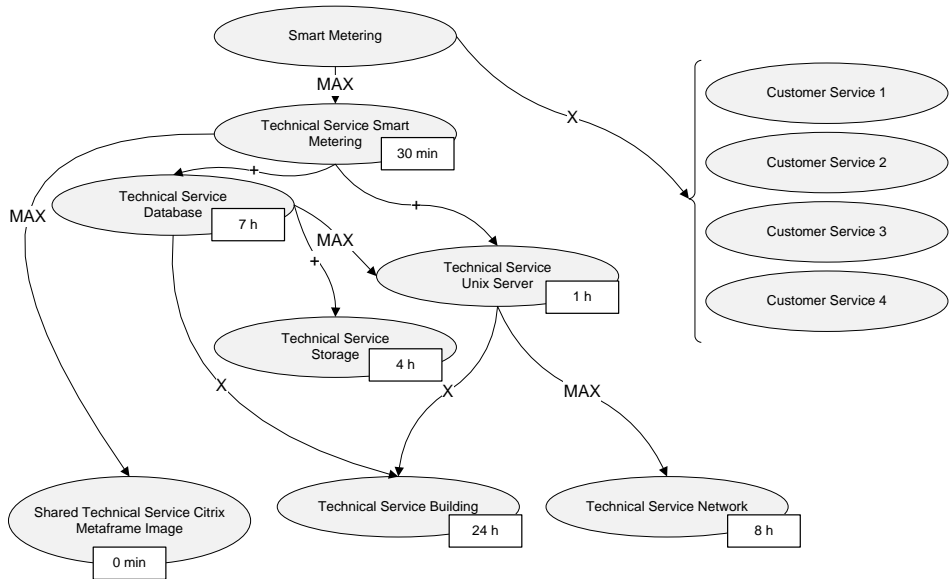

Figure 5: Service Hierarchy

The attached time periods at the technical services represent the stand alone downtime of the technical services. Together with the aggregation rules (derived out of the service model and the therein deposited relation types) the aggregation can be implemented (see Table 3: Aggregation Example).

5   Conclusion

Driven by the above-mentioned fundamental paradigm shift from a product- to a service-oriented economy supporting business processes with IT services has become common. Customer orientation is an important issue to deal with in every business and in particular in service-centric economies.

The two-sided PROSA methodology provides the long overdue customer-oriented and therefore BP-oriented perspective concerning services and SLAs. With this work we cover the main aspects of PROSA and the main steps to a seamless implementation. Our approach enables a single point of contract for service requesters and significantly reduces the managerial overhead generated from multiple SLAs. We evaluated our methodology by providing an industrial

case study from the EnBW AG. The evaluation results demonstrate how the seam-less implementation of PROSA in an enterprise context can be conducted.

In our future work, we intend to delve deeper into the providers' perspective of the PROSA methodology.

Dealing with the technical side is another fundamental area since monitoring of SLIs on BP level is much more difficult than monitoring only the compliance of a SLA from a single invoked service. When monitoring the compliance of PROSA, a lot of complex BP specific constraints have to be regarded. Using Business Activity Monitoring (BAM) could be a promising approach to deal with this issue.

Additionally, as briefly mentioned, we have to consider the issue of how much the provider has to know about the customer's BP. We call this research field "Information Sensitivity", i.e. knowing how much information is required, how it is secured, and what appropriate governance structures are in the context of sensitive business information.

Table 3: Aggregation Example

Superior Technical Service	Subordinated Technical Service	Aggregation Type	Result	Aggregation Result
Technical Service Unix Server	Technical Service Building	X	1 h	8 h
	Technical Service Network	MAX	8 h	
Technical Service Database	Technical Service Building	X	7 h	11 h
	Technical Service Storage	+	11 h	
	Technical Service Unix Server	MAX	11 h	
Technical Service Smart Metering	Technical Service Database	+	11.5 h	11.5 h
	Technical Service Unix Server	+	8.5 h	
	Shared Technical Service Citrix Meta-frame Image	MAX	11.5 h	
Process		MAX	11.5 h	11.5 h

References

Blake, M. B., Cummings, D. J. (2007) Workflow Composition of Service Level Agreements. IEEE International Conference on Services Computing, 138-145.

Blau, B., Krämer, J., Conte, T., van Dinther, C. (2009) Service Value Networks. In: Proceedings of the 11th IEEE Conference on Commerce and Enterprise Computing, 194-201.

Cardoso, J., Sheth, A., Miller, J., Arnold, J., Kochut, K. (2004) Quality of Service for Workflows and Web Service Processes. Web Semantics: Science, Services and Agents on the World Wide Web. 1 (3): 281-308.

Daly, D., Kar, G., Sanders, W. H. (2002) Modeling of service-level agreements for composed services. Lecture Notes in Computer Science. 4-15.

Elsässer, W. (2006) ITIL Einführen und Umsetzen: Leitfaden für effizientes IT-Management durch Prozessorientierung. Hanser.

Hill, T. P. (1977) On Goods and Services. Review of Income and Wealth. 23 (4): 315-338.

Jaeger, M. C., Rojec-Goldmann, G., Muhl, G. (2004) Qos aggregation for web service composition using workflow patterns. In: Proceedings of the 8th IEEE International Enterprise Distributed Object Computing Conference, 49-159.

Jin, L., Machiraju, V., Sahai, A. (2002) Analysis on Service Level Agreement of Web Services. HP.

Liu, Y., Ngu, A. H.,Zeng, L. Z. (2004) QoS Computation and Policing in Dynamic Web Service Selection. In: Proceedings of the 13th international World Wide Web conference on Alternate Track Papers & Posters ACM, New York, NY, USA, 66-73.

Ludwig, A., Franczyk, B. (2008) COSMA - An Approach for Managing SLAs in Composite Services. In: Proceedings of the 6th International Conference on Service-Oriented Computing, 626-632.

Mani, A., Nagarajan, A. (2002) Understanding quality of service for Web services. IBM Developer Works.

Muthusamy, V., Jacobsen, H. A., Coulthard, P., Chan, A., Waterhouse, J.,Litani, E. (2007) SLA-driven business process management in SOA. In: Proceedings of the Conference of the center for advanced studies on Collaborative research, 264-267.

Papazoglou, P. (2008) Web Services: Principles and Technologies. Prentice Hall.

Schmidt, H. (2000) Service Level Agreements based on Business Process Modeling. HP Open View University Association Workshop.

Unger, T., Leymann, F., Mauchart, S., Scheibler, T. (2008) Aggregation of Service Level Agreements in the Context of Business Processes. In: Proceedings of the Enterprise Distributed Object Computing Conference, 43-52.

Xiao, H., Chan, B., Zou, Y., Benayon, J. W., O'Farrell, B., Litani, E., Hawkins, J. (2008) A Framework for Verifying SLA Compliance in Composed Services. In: Proceedings of the IEEE International Conference on Web Services, 457-464.

Zeng, L., Benatallah, B., Dumas, M., Kalagnanam, J., Sheng, Q. Z. (2003) Quality Driven Web Services Composition. In: Proceedings of the 12th international conference on World Wide Web ACM, Budapest, Hungary, 411-421.