UNIVERSIDADE TÉCNICA DE LISBOA

INSTITUTO SUPERIOR TÉCNICO

A Method Based on DEMO for Managing Service Quality

Carlos Manuel Martins Mendes

Supervisor: Doctor Miguel Leitão Bignolas Mira da Silva

Thesis approved in public session to obtain the PhD Degree in

Information Systems and Computer Engineering

Jury final classification: Pass with Merit

Jury

Chairperson: Chairman of the IST Scientific Board

Members of the Committee:

Doctor João Bernardo de Sena Esteves Falcão e Cunha
Doctor Miguel Leitão Bignolas Mira da Silva
Doctor André Ferreira Ferrão Couto e Vasconcelos
Doctor Antonia Albani-Erni

2013

UNIVERSIDADE TÉCNICA DE LISBOA

INSTITUTO SUPERIOR TÉCNICO

A Method Based on DEMO for Managing Service Quality

Carlos Manuel Martins Mendes

Supervisor: Doctor Miguel Leitão Bignolas Mira da Silva

Thesis approved in public session to obtain the PhD Degree in

Information Systems and Computer Engineering

Jury final classification: Pass with Merit

Jury

Chairperson: Chairman of the IST Scientific Board

Members of the Committee:

Doctor João Bernardo de Sena Esteves Falcão e Cunha, Professor Catedrático da Faculdade de Engenharia, da Universidade do Porto

Doctor Miguel Leitão Bignolas Mira da Silva, Professor Auxiliar do Instituto Superior Técnico, da Universidade Técnica de Lisboa

Doctor André Ferreira Ferrão Couto e Vasconcelos, Professor Auxiliar do Instituto Superior Técnico, da Universidade Técnica de Lisboa

Doctor Antonia Albani-Erni, Investigadora, Institute of Information Management, University of St.Gallen, Switzerland

2013

Title:

A Method Based on DEMO for Managing Service Quality

Abstract

In this thesis we focus on reducing the difference between customers' expectations and perceived service, and on the other four gaps that influence this difference. There are several solutions that contributed to closing the gaps, but none solved the problem completely. Most of these solutions are function-oriented solutions and these are not sufficient because, they lack of an appropriate, deep understanding of enterprises and enterprises networks. Functional knowledge is appropriate and sufficient for the use and control of enterprises, but in order to change them, knowledge about their construction and operation is needed.

We propose a service quality method based on the Enterprise Ontology (EO) and respective methodology DEMO (Design & Engineering Methodology for Organizations). Based on EO, a Service Level Agreement (SLA) definition is given and a service quality specification solution is derived. Additionally, we propose that SLAs can be specified at execution time (dynamic service levels) in contrast to the usual SLAs specified at design time.

The research methodology used in this thesis is the Design Science Research Methodology.

The proposal was evaluated in six real world field studies and we used the employees' feedback, the Moody and Shanks framework and questionnaires to evaluate the results. The first two field studies revealed that the identified services were more customer oriented than the existing services and in the following three field studies the proposal showed to be mature enough to model the service provider reality regarding service quality levels. The sixth field study verified that dynamic SLAs have a positive impact in the gaps.

Key-words:

Service Quality, Service Quality Gaps, Service Level Agreement, DEMO, Enterprise Ontology

Título: Um Método Basedo em DEMO para a Gestão da Qualidade dos Serviços

Nome: Carlos Manuel Martins Mendes

Doutoramento em: Engenharia Informática e de Computadores

Orientador: Doutor Miguel Leitão Bignolas Mira da Silva

Resumo

Esta tese foca-se em reduzir a diferença entre a expecativas dos clientes e a sua percepção do serviço recebido, bem como nos restantes quatro *gaps* que influenciam esta diferença. Existem algumas soluções que contribuíram para fechar estes *gaps*, mas nenhuma conseguiu resolvê-los por completo. A maioria dessas soluções são soluções orientadas à função o que não é suficiente uma vez que não têm um adequado e profundo conhecimento das empresas e rede de empresas. O conhecimento funcional é apropriado e suficiente para o uso e controlo de empresas, mas de forma a alterá-las, é necessário conhecimento relacionado com a sua construção e operação.

Propomos uma solução para especificar a qualidade dos serviços baseada em *Enterprise Ontology* (EO) e respectiva metodologia DEMO (*Design & Engineering Methodology for Organizations*). Baseado em EO, apresentamos uma definição de nível de serviço e uma solução para especificação da qualidade dos serviços. Adicionalmente, propomos que os níveis de serviço possam ser especificados em tempo de execução (níveis de serviço dinâmicos) em contraste com os níveis de serviço usuais especificados em *design time*.

A metodologia de investigação escolhida é *Design Science Research Methodology*.

A proposta foi avaliada em seis experiências práticas onde utilizámos o feedback dos funcionários das organizações, a framework *Moody and Shanks* e questionários para avaliar os resultados. As primeiras duas experiências revelaram que os serviços identificados eram mais orientados aos clientes do que os serviços existentes e nas três experiências seguintes a proposta mostrou ser madura o suficiente para modelar os fornecedores no que concerne os níveis de qualidade dos serviços. Na sexta experiência verificámos que os níveis de serviço dinâmicos têm um impacto positivo nos *gaps*.

Palavras-chave:

Qualidade de Serviços, *Gaps* na Qualidade dos Serviços, Níveis de Serviço, DEMO, Enterprise Ontology

Acknowledgements

Since September of 2009 when I started this thesis I had the pleasure of collaborating with personalities from different contexts.

First, I would like to thank my supervisor, Professor Miguel Mira Silva, to all the effort dedicated to this thesis. Every discussion was a true inspiration.

I would also to thank the professors that lectured the courses included in the PhD program, namely Professor Artur Caetano, Professor André Vasconcelos, Professor Diogo Ferreira, Professor José Delgado and Professor José Tribolet. Their feedback was priceless.

Thanks to Carlos Tomaz, a practitioner who is expert in service management and who helped me in the first months of this process.

Thanks to all the field studies participants that made it possible to evaluate my proposal. Special thanks to Rosário Dias, Gabriela Salvador and Gonçalo Silva (Wines4All), Nuno Rebelo (CSP), Nuno Salvador (Pombal City Council) and Rui Costa (private bank).

Thanks to the master students that collaborated with me: João Ferreira, Mário Almeida, David Dias and Pedro Carvalho. Their beliefs in my advices gave me a new sense of responsibility and confidence.

Thanks to my close friends who shared their opinions about this thesis. Their opinions gave me different perspectives about this work.

Thanks to my family, especially to my parents who have supported me in all the possible ways that I ever needed. Besides, their example of hard work gave me strength and inspiration in the darkest times.

Thanks to my fiancée, Telma Augusto, who supported me all the time. Without her loving support, I would have not been able to accomplish this thesis. I want to particularly thank her patience during my most stressed periods.

Table of Contents

List of Figures

List of Tables

1 Introduction

This Chapter is divided in seven sections. First, we present this thesis motivation in Section 1.1. In Section 1.2 we describe the service quality gaps and then in Section 1.3 we describe the research approach, including the research area and methodology. Afterwards, we present the research questions to be addressed (Section 1.4) and we specify the solution requirements (Section 1.5). We then describe the main contributions in Section 1.6, and finally, we describe the remaining structure of the document (Section 1.7).

1.1 Motivation

The motivation for this research began in the researcher master thesis in which the costs of an Information Technology (IT) department of a public organization were associated to their clients via the department services (Mendes C. , 2009). Figure 1 illustrates an example of the proposal. The costs of the department were assigned to the services, and then the costs associated to the services were assigned to the customers.

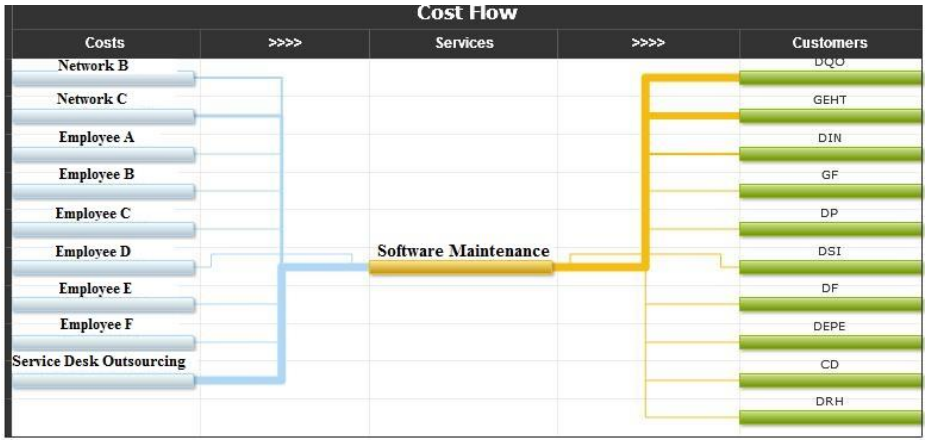


Figure 1 – Costs associated to customers through a service (Software Maintenance)

Figure 1 illustrates the example of the Software Maintenance service. In the left side and in blue there are the costs that are being allocated to this service (networks, employees and an outsourcing contract). In the middle and in orange there is the service Software Maintenance and in the right and in green there are the customers that consume the service. The percentage of costs

associated is proportional to the lines thickness. For example, the cost that most contributes to the service total cost is the Service Desk Outsourcing contract and the customer that most pays for the Software Maintenance service is the DQO.

One of the prerequisites for assigning costs to clients was the identification of the department services, which had not been identified previously in the studied department. Consequently, and in order to accomplish the work, we had to identify ourselves the services of the department.

In the first attempt, we defined the services using the IT department perspective and knowledge, and the result was that the services were IT oriented and did not represent the customers' perspective. When the customers saw the invoices of the IT department, they did not recognize their IT needs in the invoices lines (each corresponding to a service used in the previous month). So, in order to align the services list to the customers' needs, we had to redefine the services using the customers' feedback. This feedback motivated the researcher to study the services concept.

As services are a nuclear concept of most organizations and are needed in some processes (such as this case), it was surprising to find that the services provided by that IT department were not even identified. It was also surprising to discover that service identification was not an easy task to perform. In fact, one conclusion of the master thesis was that the service identification process should always be customer oriented, because customers will only pay for a service that they can understand.

At the more general level, a uniform theoretical framework for service studies is lacking because service studies conducted in different disciplines are more or less separate from the bodies of knowledge in the other disciplines. The frame of reference is often tied to the discipline-specific framework (Chesbrough & Spohrer, 2006).

1.2 Service Quality Gaps

Today services mean jobs and growth, but the organizations who have been leading the charge lack a strong conceptual foundation (Chesbrough & Spohrer, 2006). This lack contributes to the gaps that reduce the services quality because, without a solution to specify the services quality, it is difficult for the service providers and their customers to align their expectations about the services.

Service quality poses a number of challenges and research topics. We decided to focus on the gaps (Parasuraman, et al., 1985) that influence the services quality as they represent the factors

that diminish the services quality. Service marketers often use these gaps to illustrate how differences between perceived service delivery and expected service can come about (Figure 2).

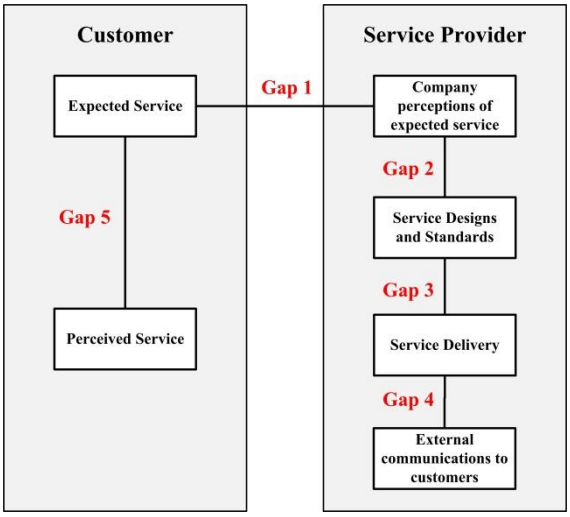


Figure 2 – Gaps Model of Service Quality. Adapted from **(Parasuraman, et al., 1985)**

The fifth gap represents the difference between customers' expectations and perceived service and this gap is caused by the four preceding gaps (Zeithaml & Bitner, 1996):

- **Gap 1** – The expected service as perceived by the service provider differs from the service as expected by the customer. The service provider perception of what the customer expects may differ from the real expected service due to inadequate market research, lack of communication between operational employees and management, or insufficient relationship focus. For example, the service organization aims to satisfy certain availability constraints (e.g. 99% availability), while the actual customer concern is related with maximum downtime (e.g. no longer than one hour per failure);
- **Gap 2** – The service specification as used by the service provider differs from the expected service as perceived by the service provider. The service designs and standards may not match the service requirements as perceived by the provider by a lack of customer-driven standards, absence of process management, lack of a formal process for setting service quality goals, poor service design or inadequate service leadership. For example, the customer expects a quick restart of the system, while the standard procedure of the maintenance organization is focused on analysing the reason for the crash;
- **Gap 3** – The actual service delivery differs from the specified services. Service delivery may not follow the service designs and standards because of deficiencies in human resource policies, failures to match demand and supply, or customers not fulfilling their

role. For example, customers bypass the helpdesk by contacting the maintainer of their system directly, and thus circumventing a well-designed incident management process;

- **Gap 4** – Communication about the service does not match the actual service delivery because of ineffective management of customer expectations, overpromising, or inadequate horizontal communications (i.e. insufficient communication between sales and operations or between advertising and operations, or differences in policies and procedures across the organization). For example, a customer is not informed about the reparation of a bug he or she reported.

In services, customers' perceptions of their experiences are as important as the design and delivery of the service. Experience points are opportunities to help frame their expectations of what they will experience. Customers' satisfaction with a service will be determined by a combination of what is delivered and how that compares to what they expected to receive ((Chesbrough H. , 2011), page 58). Hence, service quality can be increased by closing the first four gaps, thus bringing the perceived service in line with the expected service.

This gap model has been chosen because it describes the five major gaps that influence the services quality. We could think of other gaps that may influence the services quality, but in this thesis we propose a solution to mitigate these gaps that have become a reference in the services marketing research area.

1.3 Research Approach

This Section describes the research area (Service Science) and methodology (Design Science Research Methodology).

1.3.1 Research Area – Service Science

This research is within the Service Science discipline. This discipline is a multi-disciplinary research and academic effort that integrates aspects of established fields like computer science, operations research, engineering, management, marketing, social and cognitive sciences, and legal sciences. Therefore, this research scope is based on both hard and soft sciences. The hard sciences usually rely on experimental, empirical and quantifiable data, while soft science is a term used in groups fields which are supposedly ”scientific”, although not based on reproducible experimental data and/or a mathematical explanation of that data.

Service Science is defined as the discipline “*that focuses on fundamental science, models, theories and applications to drive innovation, competition, and quality of life through co-*

creation of value" ((Ostrom, 2010), p. 5). Originating in the IT sector, service science is a relatively new interdisciplinary approach for research in services. Service Science sets out to embrace and integrate all communities that deal with services. Service science aims to create the basis for systematic service innovation by combining organization and human understanding with business and technological understanding to explain the origins and growth of service systems (Spohrer, et al., 2007).

Service Science focuses not merely on one aspect of service, but rather on service as a system of interacting parts that include people, technology, and business (Chesbrough & Spohrer, 2006). Service Science is the study of service systems and the co-creation of value in complex configurations of resources (Vargo, et al., 2008).

Today, the development of Service Science curricula is a topic of concern for many academic institutions. Some examples can be found in (Hefley & Murphy, 2008), (Dubois, et al., 2011) and (Macaulay, et al., 2012).

1.3.2 Research Methodology - DSRM

Our research was conducted using the Design Science Research Methodology (DSRM) that aims to create and evaluate IT artifacts intended to solve identified organizational problems (Hevner, et al., 2004). These artifacts include constructs (vocabulary and symbols), models (abstractions and representations), methods (algorithms and practices) and instantiations (implemented and prototype systems).

Additionally, DSRM established seven guidelines (Hevner, et al., 2004):

- **Design as an Artifact** – Produce a viable artifact in the form of a construct, a model, a method, or an instantiation. The artifacts developed in this thesis are described in Chapter 4 (Thesis Proposal);
- **Problem Relevance** – Develop technology-based solutions to important and relevant business problems. The relevance of this thesis problem is described in Section 1.1 (Motivation) and 1.2 (Service Quality Gaps);
- **Design Evaluation** – The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well-executed evaluation methods. The artifacts were evaluated in six field studies (Chapters 5, 6, 7, 8, 9 and 10);
- **Research Contributions** – Provide clear and verifiable contributions in the areas of the design artifact, design foundations, and/or design methodologies. This thesis main contributions are listed in Section 1.6 (Main Contributions);

- **Research Rigor** – Application of rigorous methods in both the construction and evaluation of the design artifact. This Section explains the methods that we used to construct (DSRM) and evaluate the artifacts;
- **Design as a Search Process** – The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment. Progress is made iteratively as the scope of the design problem is expanded. As means, ends, and laws are refined and made more realistic, the design artifact becomes more relevant and valuable. The six field studies allowed to iteratively improve our artifact. Additionally, each field study had a specific contribution to the research questions;
- **Communication of Research** – The research must be presented effectively to technology-oriented as well as management-oriented audiences. A total of eight papers were published in this thesis context. The complete list of publications is available in page 149.

DSRM includes the following phases (Peffers, et al., 2008):

- **Problem Identification** – Define the specific research problem and justify the value of a solution (Chapters 1, 2, and 3);
- **Objectives Definition** – Infer the objectives of a solution from the problem definition and knowledge of what is possible and feasible. The objectives can be quantitative, e.g., terms in which a desirable solution would be better than current ones, or qualitative, e.g., a description of how a new artifact is expected to support solutions to problems not hitherto addressed (Chapter 1);
- **Design and Development** – Create the artifact. Such artifacts are potentially constructs, models, methods, or instantiations (each defined broadly) or "new properties of technical, social, and/or informational resources". Conceptually, a design research artifact can be any designed object in which a research contribution is embedded in the design. This activity includes determining the artifact's desired functionality and its architecture, and then creating the actual artifact (Chapter 4);
- **Demonstration** – Demonstrate the use of the artifact to solve one or more instances of the problem. This could involve its use in experimentation, simulation, case study, proof, or other appropriate activity (Chapters 5, 6, 7, 8, 9 and 10);
- **Evaluation** – Observe and measure how well the artifact supports a solution to the problem. This activity involves comparing the objectives of a solution to the actual observed results from using the artifact in the demonstration (Chapters 5, 6, 7, 8, 9 and 10);

- **Communication** – Communicate the problem and its importance, the artifact, its utility and novelty, the rigor of its design, and its effectiveness to researchers and other relevant audiences, such as practicing professionals, when appropriate (Chapter 11).

Figure 3 depicts the methodology employed to address the particular characteristics of the research performed in this thesis.

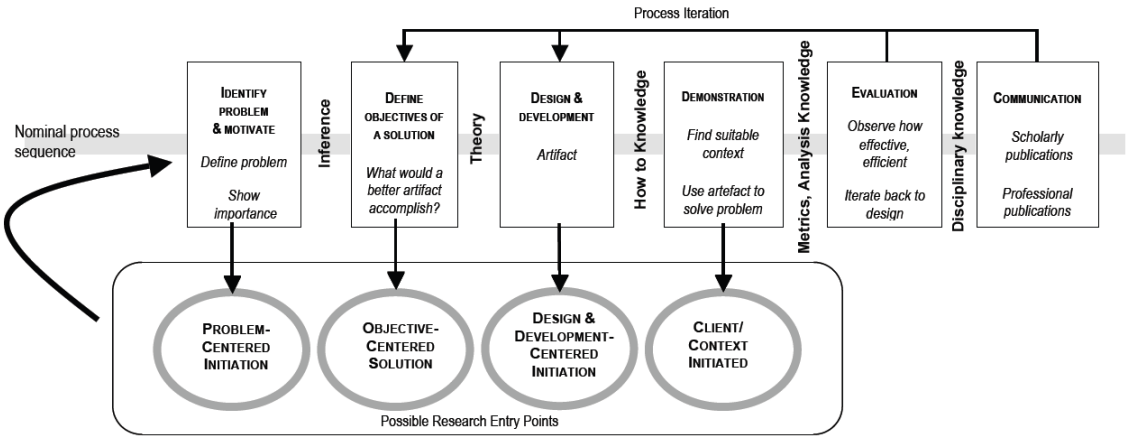


Figure 3 – Design Science Research Methodology Process Model **(Peffers, et al., 2008)**

This research methodology has been chosen because DSRM is appropriate for research that seeks to extend the boundaries of human and organizational capabilities by creating new and innovative artifacts. DSRM is also active with respect to technology, engaging in the creation of technological artifacts that impact people and organizations (Hevner, et al., 2004).

1.4 Research Questions and Hypotheses

In order to specify the context of this thesis, we narrowed the focus to specific questions to be answered and predictions based on hypotheses to be tested. The research questions represent unsolved problems and are based on the gaps model presented in Section 1.2. The hypotheses connect the problem domain with the solution domain, referring and justifying the main concepts that are used throughout the thesis. Additionally, the hypotheses narrow the scope of this thesis by specifying the specific parts of the research questions that will be addressed. In the end, we are able to infer if the hypotheses are valid in the solution for solving the given problem.

We followed the guidelines described in (Creswell, 2009) to structure the research questions and hypotheses. This thesis research questions assume two forms: a central question and four

associated sub-questions. The central question is a broad question that asks for an exploration of the central phenomenon or concept in a study (Creswell, 2009).

The central question that this research seeks to answer is:

How to diminish the gap between expected and perceived service?

This research question has four sub-questions associated that are described in the next four subsections.

1.4.1 Research Question Q1 How to specify services according to customers' expectations?

Hypothesis H1.1 *The way of work defined in DEMO can be used as a basis to identify customer-oriented services.*

Hypothesis H1.2 *The presence of both customer and provider in the service identification process contributes to a more customer-oriented service catalogue.*

Research Question 1 is directly related with Gap 1 and 2, because the customers' expectations should be traceable in the provider's services. Besides, service identification is crucial since services are the basis for several processes. So, if services are poorly identified, these processes will be negatively influenced. For example, 30% of IT Service Management (ITSM) projects do not finish because of problems related with service definition (Cole S. , 2008). Additionally, in a study with around 100 companies that tried to implement a Service Catalogue, only 57% reported that the project was successful while 12% reported that the project was completely unsuccessful. Furthermore, 34% of those companies mentioned service definition as one of the "top risks" for successful catalogue implementation (Cole S. , 2008).

Hypothesis H1.1 is related with the research question Q1 since it tries to validate the possibility of using the way of work defined in DEMO to identify services. This way of work is composed by six steps that allow to find the first ontological model of any organization starting from a textual description of this organization. This first ontological model has a concept (transaction) that, as we will demonstrate later, is connected with services. Towards this thesis we stress the applicability of these six steps and we propose some changes to better specify services.

DEMO (Design & Engineering Methodology for Organizations) is a methodology for modelling, (re)designing and (re)engineering organizations and networks of organizations (Dietz J. , 2006). DEMO allows to model the Enterprise Ontology (EO) of an organization. Enterprise

Ontology is defined as the fully implementation independent understanding of the essence of an organization. We have chosen EO because this theory can help us to expand the expressiveness of the service descriptions and, consequently, to allow a better alignment between expectations and perceptions.

We also propose to find evidences on who should be included in the service identification process (H1.2) because there seems to exist a belief among service providers that they can identify their own services without theirs clients feedback. We do not share this belief, since the providers tend to define their services with an internal focus. Additionally, a study revealed that one of the most relevant challenges from the customers' point of view is the technical focus of providers (Unterharnscheidt & Kieninger, 2010). Therefore, we believe that using both customer and provider feedback in the service identification process contributes to a more customer oriented service catalogue.

1.4.2 Research Question Q2

Research Question Q2 How to specify service level agreements according to customers' expectations?

Hypothesis H2.1 *The Service Level Agreement (SLA) concept is related with the proposition/result of Enterprise Ontology transactions.*

Hypothesis H2.2 *The Enterprise Ontology patterns of transactions represent a structure to define non-functional attributes of SLAs.*

In order to capture customers' expectations, providers should also specify customer oriented Service Level Agreements (SLAs). This involves defining the SLA concept and the attributes that can be used in an SLA to represent the customers' expectations. Research Question Q2 is related with gaps 1 and 2 since the answer to Q2 captures customers' expectations using SLAs.

Regarding the SLA definition, Hypothesis H2.1 will allow to test if there is a relation between the SLA concept and the transaction results/propositions. We believe there is a relation between the two because the propositions are negotiated by customers and service providers in every transaction. The propositions define *something that is or could be the case in the production world* (Dietz J. , 2006). Production world are the effects of the production acts that are performed by the elements of the system.

Concerning the SLA attributes, Hypothesis H2.2 will allow to verify if the Enterprise Ontology universal pattern of transactions represent a structure to define non-functional attributes of SLAs. These patterns, also called transactions, always involve two actor roles and are aimed at

achieving a particular result. Additionally, the universal patterns define the possible sequence of steps that can be executed in order to achieve the expected result. Therefore, we believe that knowing the steps that can be performed, independently of what is the result, give us the possibility to define performance targets that these steps should comply with.

1.4.3 Research Question Q3 How to deliver services according to the service specifications and customers' expectations?

Hypothesis H3.1 *Dynamically defined SLAs diminish the gap between customers' expectations and perceived service.*

Research Question Q3 is concerned with Gap 3. Having the customers' expectations specified in Enterprise Ontology-based SLAs, organizations should be interested in delivering services according to these specifications. We are evaluating if dynamically defined SLAs have a positive impact on the gaps (Hypothesis H3.1). One of the more challenging aspects of service quality is to provide for the uniqueness of each situation and of each customer (Ukens, 2007). Each service interaction between a provider and a customer is an opportunity for the provider to delight, satisfy or disappoint a customer. Additionally, the customers' expectations may change for each interaction. Therefore, defining a number of SLAs at design time, and restricting the customers' options to this number, limits the possibilities for the customers to express their expectations.

We propose to let customers freely express what they are looking for in each interaction. This allows a dynamic SLA negotiation instead of only using the SLAs defined at design time (static SLAs). Therefore, we are evaluating if dynamically defined SLAs have a positive impact in the gap between customers' expectations and perceived service.

1.4.4 Research Question Q4 How to align the two universes of discourse in such a way that the perceived service quality is aligned with the service delivery?

Hypothesis 4.1 *A system that registers all the coordination acts involved in the service exchange diminishes the gap between customers' expectations and perceived service.*

Research Question Q4 is concerned with Gap 4 because, even after the service provision, service providers and customers should understand each other when discussing and evaluating the

service result. Therefore, there must be a solution to align the two universes of discourse in service exchange. **Although we present this research question, we will not address research question Q4 on this thesis context.** This decision is aligned with the recommendation from the CAT to reduce the context of the proposed thesis.

Once more, we believe Enterprise Ontology theory can help us to answer this question. In the future, we intend to evaluate if a system that registers all the coordination acts involved in the service exchange diminishes the gap between service delivery and communication with customers, and consequently contributing to close Gap 5. Coordination acts represent commitments regarding the performance of production acts. Additionally, we believe it would be interesting to research the impact of tacit knowledge (the kind of knowledge that is difficult to transfer to another person by means of writing it down or verbalizing it) in Gap 4, since tacit knowledge interferes with the ability of suppliers and customers to communicate with one another ((Chesbrough H. , 2011), page 54).

These hypotheses satisfy both criteria for experimental hypotheses (Creswell, 2009), since each hypothesis represents a prediction and they are all testable. In other words, the hypotheses predict the anticipated outcomes of the thesis, and once we have collected and evaluated our data, we know whether or not the hypotheses are valid.

Figure 4 illustrates how the gaps and the research questions relate to each other.

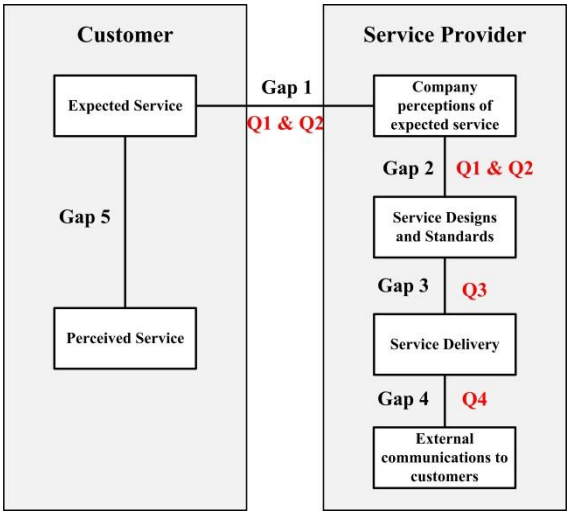


Figure 4 – Gaps Model of service quality and Research Questions

The research questions and hypotheses are the result of the problem identification step of Design Science Research Methodology (DSRM). In Section 1.6 we show how these questions re-

late to the contributions presented throughout the thesis. Furthermore, Chapters 5 to 10 describe how the thesis proposal answers these questions.

1.5 Solution Requirements

This Section describes the objectives definition step of Design Science Research Methodology (DSRM). Based on the research questions and hypotheses, we developed the requirements that this thesis proposal is intended to fulfil.

The solution requirements are:

- **Requirement 1** – Have a clear service definition;
- **Requirement 2** – Define the attributes that should be used to specify a service;
- **Requirement 3** – Define clear steps to identify services;
- **Requirement 4** – Have a clear Service Level Agreement (SLA) definition;
- **Requirement 5** – Define the attributes that should be used to specify an SLA;
- **Requirement 6** – Define the services and SLAs from the customer point of view;
- **Requirement 7** – Provide for the uniqueness of each situation and of each customer.

The relations between requirements, contributions, hypotheses, and research questions are illustrated in Table 1. Additionally, these requirements will be used to analyse the related work solutions and compare these solutions with this thesis proposal (Table 9 – page 50).

1.6 Main Contributions

One type of contributions of design-science research is the design artifact itself. The artifact must enable the solution of relevant problems. To do so, the artifact may extend the knowledge base or apply existing knowledge in innovative ways. Another type of contribution is the development of appropriately evaluated constructs and models that extend and improve the existing foundations in the design-science knowledge base (Hevner, et al., 2004).

The main contributions of this research relate to the above two types and are the following:

Contribution 1 *Application of the DEMO methodology for the identification of services*

Field studies in Chapters 5 and 6 stress the advantages and limitations of using the method for the development of the ontological aspect models of DEMO to identify services.

Contribution 2 *Application and validation of the Generic Service Specification Framework to several field studies*

The GSSF (Terlouw & Albani, 2013) defined the service concept using the Enterprise Ontology (EO) as theoretical foundation. We adopted GSSF in our proposal, and after using GSSF in several field studies, we found some limitations. Figure 24 (page 56) illustrates the new version of the GSSF and the field studies in Chapters 8 and 9 explain why we introduced these changes to the proposal.

Contribution 3 *The mapping between the gaps model and the EO basic transaction pattern*

Section 2.4 (page 23) explains why the Enterprise Ontology theory does not solve the gaps problem and it defines what each gap represents in the basic transaction pattern (Figure 13).

Contribution 4 *A EO-based design artifact that specifies the attributes of an SLA*

Figure 25 (page 57) illustrates the Enterprise Ontology-based SLA. Field studies in Chapters 7, 8, 9 and 10 evaluate the advantages and limitations of the Enterprise Ontology-based artifact that specifies the attributes of an SLA.

Contribution 5 *An ontological specification for the dynamic SLA negotiation*

Section 4.3 (page 59) presents an ontological specification for the dynamic SLA negotiation. This specification contains the actor roles, the transactions, the business rules and the entities involved in the dynamic SLA negotiation. The field study in Chapter 10 (page 133) describes the implementation of dynamic Enterprise Ontology-based SLAs. We enumerate the impacts of our proposal to Gap 5 in terms of five dimensions: reliability, assurance, tangibles, empathy, and responsiveness.

Table 1 identifies how the contributions presented above relate to the research questions, hypotheses and requirements.

		Research Questions	Hypotheses	Requirements
Contributions	C1	Q1	H1.1, H1.2	R1, R3, R6
	C2	Q1	H1.1, H1.2	R1, R2, R6
	C3	Q2	H2.1	R4
	C4	Q2	H2.2	R4, R5, R6
	C5	Q3	H3.1	R6, R7

Table 1 – How the contributions relate to the research questions, hypotheses and requirements

These contributions were communicated to the research community in a number of publications (page 149).

1.7 Thesis Structure

The thesis is structured as illustrated in Table 2.

Chapter	DSRM Phases
1 Introduction	Problem identification and objectives definition
2 Theoretical Foundation	Problem identification
3 Related Work	Problem identification
4 Thesis Proposal	Design and development
5 Comparing Services Using DEMO	Demonstration and evaluation
6 Using DEMO to Identify Services	Demonstration and evaluation
7 Specifying SLAs in a Bank	Demonstration and evaluation
8 Specifying SLAs in a City Council	Demonstration and evaluation
9 Specifying SLAs in a Cloud Services Provider	Demonstration and evaluation
10 Closing the Gaps in Wines4All	Demonstration and evaluation
11 Conclusion	Communication

Table 2 – Thesis Structure

In the current Chapter we describe this thesis motivation, the chosen research approach, the research questions and hypotheses addressed (problem identification of DSRM), the solution requirements (objectives definition of DSRM), and the main contributions. In Chapter 2 we introduce the theoretical background of this research, the Enterprise Ontology theory. Next, we provide an overview of the literature on the research questions area (Chapter 3).

Afterwards in Chapter 4, we present our proposal, namely the SLA definition using EO terms and our Enterprise Ontology-based proposal to specify the services quality (design and development phases of DSRM). Additionally, Chapter 4 presents the ontological model of the dynamic SLA negotiation. From Chapter 5 to 10, we describe the field studies we used to evaluate the proposal (demonstration and evaluation phase of DSRM) and, finally, we present our conclusions in Chapter 11.

2 Theoretical Foundation – Enterprise Ontology (EO)

This Chapter has three main objectives: (1) to explain the reason why we have chosen EO and DEMO (Section 2.1), (2) to briefly describe the Enterprise Ontology theory (Section 2.2) and respective DEMO methodology (Section 2.3) and, finally, (3) to demonstrate that, by itself, this theory is not able to solve the problem (Section 2.4).

2.1 Why Enterprise Ontology?

We have chosen EO because this theory can help us to expand the expressiveness of the service descriptions and, consequently, allow a better alignment between expectations and perceptions. At first glance, EO and the service concept may not look related. However, a recent research paper defined the service concept using EO terms (Terlouw & Albani, 2013). This research work gave us the connection point needed to use the EO theory as the basis for our proposals. The contributions that EO and DEMO may have to this thesis problem were mentioned in the definition of the research questions and hypotheses (Section 1.4 – page 7).

First, we will use the way of work defined in DEMO to identify services (Hypothesis 1.1). This way of work is composed by six steps that allow to find the first ontological model of any organization starting from a text description of this organization. This first ontological model has a concept (transactions) that as we will demonstrate later is connected with the services.

Second, we will validate if there is a relation between the SLA concept and the transactions results/propositions (Hypothesis 2.1). We believe there is a relation between the two because the propositions are negotiated by customers and service providers in every transaction.

Third, we will verify if the Enterprise Ontology universal patterns of transactions represent a structure to define non-functional attributes of SLAs (Hypothesis H2.2). These patterns, also called transactions, always involve two actor roles and are aimed at achieving a particular result. Additionally, the universal patterns define the possible sequence of steps that can be executed in

order to achieve the expected result. Therefore, we believe that knowing the steps that can be performed, independently of what is the result, give us the possibility to define performance targets that the steps should comply with.

Besides, the EO theory produces conceptual models that are (Dietz J. , 2006):

- **Coherent –** the distinguished aspect models constitute a logical and truly integral whole;
- **Comprehensive –** all relevant issues are covered, the whole is complete;
- **Consistent –** the aspect models are free from contradictions or irregularities;
- **Concise –** no superfluous matters are contained in it, the whole is compact and succinct;
- **Essential –** the conceptual model shows only the essence of the enterprise (its deep structure) and abstracts from all realization and implementation issues.

Therefore, EO is a candidate to fulfil the mentioned lack of a strong conceptual foundation (Chesbrough & Spohrer, 2006). Besides that, Enterprise Ontology is a theory widely accepted in the research community. Many papers have been published based on this theory in journals and conferences. Some of these papers are listed at the website of the International CIAO! Network (www.ciao.tudelft.nl) and at the website of the Design and Engineering Methodology for Organizations (DEMO) Knowledge Center (www.demo.nl). Furthermore, Enterprise Ontology relies on fifteen years of practical experience using the DEMO methodology (Dietz J. , 2006).

2.2 Enterprise Ontology

The theory behind Enterprise Ontology (Dietz J. , 2006) is based on four axioms – operation, transaction, composition and distinction – and the organization theorem. The **operation axiom** states that the operation of an enterprise is constituted by the activities of actor roles that are elementary chunks of authority and responsibility, fulfilled by subjects (Figure 5). These subjects perform two kinds of acts: production acts (P-acts) and coordination acts (C-acts). These acts have definite results: production facts (P-facts) and coordination facts (C-facts), respectively.

By performing production acts the subjects contribute to bringing about the goods and/or services that are delivered to the environment of the enterprise. By performing coordination acts subjects enter into, and comply with, commitments towards each other regarding the performance of production acts.

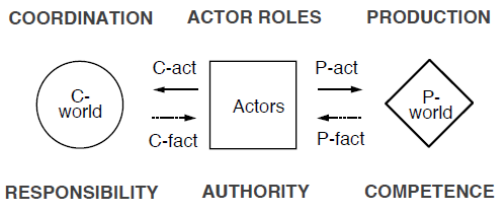


Figure 5 – Graphical representation of the operation axiom and the notions of responsibility, authority, and competence **(Dietz J. , 2006)**

Competence is defined as the collective knowledge, know-how and experience that is necessary and sufficient for a subject to perform production acts of a particular kind. Competence is related to profession (e.g. plumber). Authority is defined as the being authorized of a subject by an institution, e.g., by a company (employee) or by a society (client), to perform particular production acts and/or coordination acts (e.g. plumber of Company X). Responsibility is defined as the socially felt need by a subject to perform the coordination acts for which it is authorized, in an accountable way (e.g. plumber of Company X -> client of Company X)

The **transaction axiom** states that coordination acts are performed as steps in universal patterns. These patterns, also called transactions, always involve two actor roles (initiator and executor) and are aimed at achieving a particular result. A transaction evolves in three phases: the order phase (O-phase for short), the execution phase (E-phase for short), and the result phase (R-phase for short). In the O-phase the two actors agree the expected result of the transaction, in the E-phase the executor executes the production act needed to create the expected result, and in the R-phase the two actors discuss if the transaction result is equal to the expected result. Figure 6 illustrates the basic pattern of a transaction.

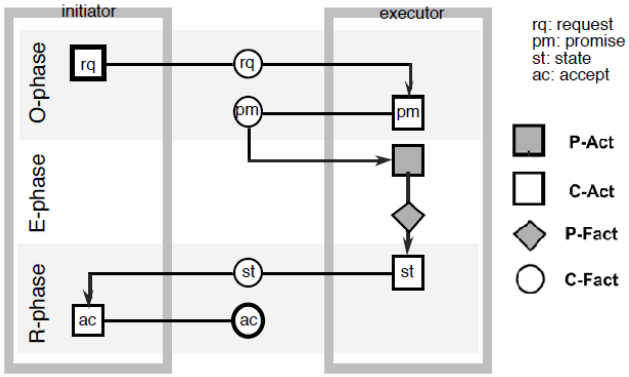


Figure 6 – The standard pattern of a transaction **(Dietz J. , 2006)**

The **composition axiom** establishes the relationships between transactions (Figure 7). This axiom states that every transaction is enclosed in another transaction, or is a customer transac-

tion of another transaction, or is a self-activation transaction. The latter case refers to transactions that give rise to further transactions of the same type.

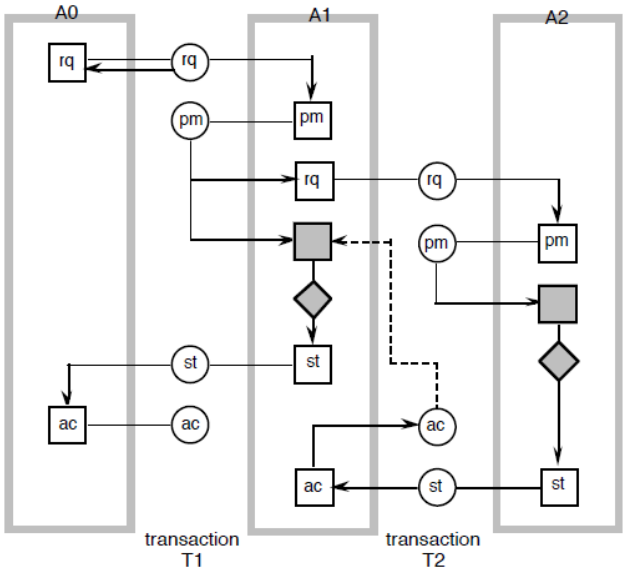


Figure 7 – The structure of enclosing a transaction and of self-activation **(Dietz J. , 2006)**

The **distinction axiom** states that there are three distinct human abilities playing a role in the operation of actors, called performa, informa, and forma (Figure 8). An ontological act (performa) is an act in which new original things are brought about. Deciding and judging are typical ontological production acts. Regarding the coordination between people, typical ontological acts are requesting and promising.

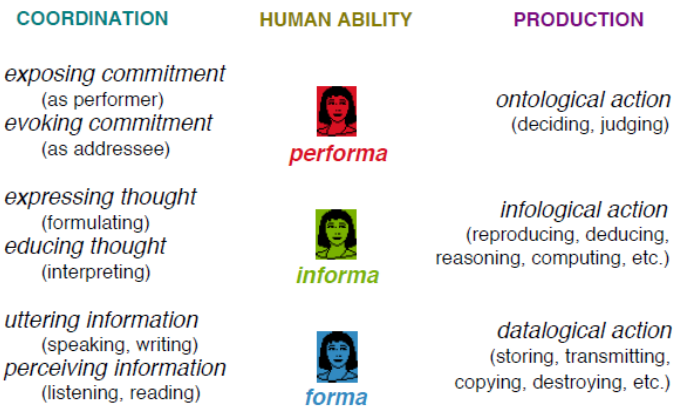


Figure 8 – The distinction axiom **(Dietz J. , 2006)**

An infological production act is an act in which one is not concerned about the form but, instead, about the content of information. Typical infological acts are inquiring, calculating, and reasoning. Regarding the coordination between people, formulating thoughts (in written or spo-

ken sentences) and interpreting (through listening or reading) perceived sentences are typical infological coordination acts. Acts like copying, storing, and transmitting data are typical datalogical acts, while speaking, listening, writing, and reading are typical datalogical coordination acts.

The **organization theorem** states that the organization of an enterprise is a heterogeneous system that is constituted as the layered integration of three homogeneous systems (Figure 9): the **B-organization** (from Business), the **I-organization** (from Intellect), and the **D-organization** (from Document).

The relationships among them are that the D-organization supports the I-organization, and the I-organization supports the B-organization. The integration is established through the cohesive unity of the human being.

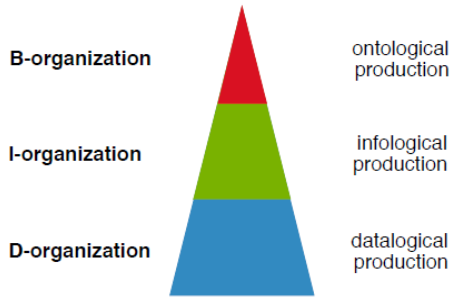


Figure 9 – The organization theorem **(Dietz J. , 2006)**

A conceptual model of this theory is presented in Appendix A – Related Work Concept Models.

2.3 Modelling with DEMO

DEMO (Design & Engineering Methodology for Organizations) is a methodology for modelling, (re)designing and (re)engineering organizations and networks of organizations. DEMO consists of four aspect models, represented by particular diagrams, tables and lists:

- **The Construction Model** (CM) – specifies the identified transaction types and the associated actor roles, as well as the information links between the actor roles and the information bank. The CM specifies the composition, environment, and structure of an organization:
  - **The Interaction Model** (IAM) – shows the active influences between actor roles i.e. the execution of transactions;

- **The Interstriction Model** (ISM) – shows the passive influences between actor roles, i.e., the taking into account of existing facts by an actor role when being active.

- **The Process Model** (PM) – contains, for every transaction type in the CM, the specific transaction pattern of the transaction type. The PM also contains the causal and conditional relationships between transactions. These relationships determine, in addition to the transaction patterns, the possible trajectories in the Coordination-world. In other words, the PM specifies the state space as well as the transition space of the Coordination -world;
- **The Action Model** (AM) – specifies the action rules that serve as guidelines for the actors in dealing with their agenda. For every coordination step, there is an action rule that guides how the performing actor role should respond to the reached status. At the ontological level of abstraction there is nothing below the AM;
- **The State Model** (SM) – specifies the state space of the P-world: the object classes and fact types, the result types, and the ontological coexistence rules. SM is the ideal starting point for developing and maintaining the data dictionary of an enterprise and it simplifies the identification of business components (software components), based on the chunks of fact types around categories.

These models (depicted in Figure 10) constitute the complete ontological model of Business-organization and subsequently represent the ontological model of the corresponding enterprise.

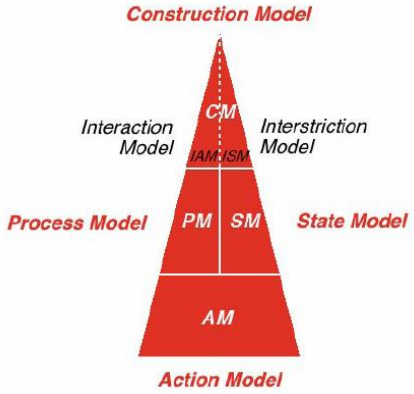


Figure 10 – The ontological aspect models **(Dietz J. , 2006)**

The logical sequence of producing the aspect models is anticlockwise, starting with the interaction model (IAM).

On the left of Figure 11, the diagram types in which the aspect models are expressed are shown, while on the right, the cross-model table types are shown. The Interaction Model (IAM) is expressed by the Actor Transaction Diagram (ATD) and the Transaction Result Table (TRT).

The Process Model (PM) is expressed by the Process Structure Diagram (PSD) and the Information Use Table (IUT). The Action Model (AM) is expressed by action rules specifications. The State Model (SM) is expressed by the Object Fact Diagram (OFD), TRT, and IUT. Finally, the Interstriction Model (IM) is expressed by the Actor Bank Diagram (ABD) and the Bank Contents Table (BCT).

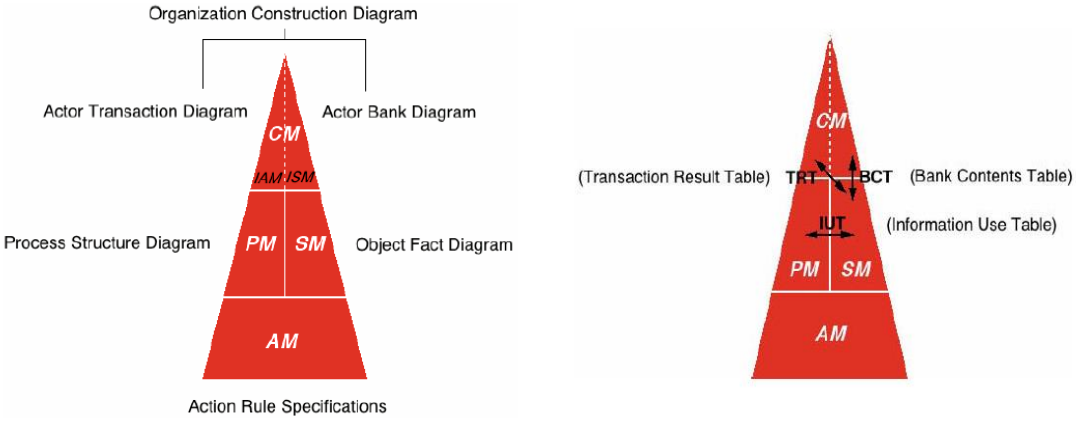


Figure 11 – The diagrams (left) and the cross-model tables (right) **(Dietz J. , 2006)**

Although our research uses all these models, the most relevant model for our proposal is the Construction model because it contains the transactions/services of the organization.

2.4 Enterprise Ontology and the Gaps Problem

Although we recognize the qualities of the models derived from this theory, these models by themselves cannot close the service quality gaps (Mendes & Mira da Silva, 2012). This happens because the gaps existence depends on how the ontological models are implemented. For instance, the occurrence of these gaps can be potentiated by a concept that almost all organizations use, the delegation. In EO, by delegation is understood the allowance by the authorized subject to another subject to perform one or more steps in one or more transactions of the corresponding transaction kind. If the expectations and perceptions of two actors can create the mentioned gaps, then we can imagine that the expectations and perceptions of three, four or more actors can be even more difficult to align.

To understand what each gap represents in EO, one has to keep in mind that a coordination act consists of two concurrent acts: the intention act and the proposition act. In the intention act, the performer proclaims its ‘social attitude’ with respect to the proposition. In the proposition act, the performer proclaims the fact and the associated time the intention is about (Figure 12).

In this example the performer of the coordination act (Mr Bean) is requesting the Florist (addressee) to deliver a bouquet of 15 roses (fact) and this request was done in 05/06/2013.

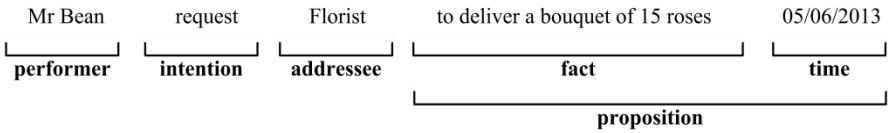


Figure 12 – Standard notation of a coordination act. Adapted from **(Dietz J. , 2006)**

However, without a clear specification of the proposition, the two actors may have different understandings of the proposition. In this example, the time in which Mr. Bean needs the bouquet is not explicitly mentioned although it is assumed to be as soon as possible. Nevertheless, as it is not specified, the performer (Mr. Bean) may assume that the bouquet will be given as soon as possible, while the addressee (Florist) may think that the request is not urgent. This would represent a clear gap between Mr. Bean's expectations and the Florist's perception of those expectations that corresponds to Gap 1. Therefore, in Enterprise Ontology models, the gaps problem is caused by the possible misalignment among notions (one from the initiator/customer and another from the executor/provider) of the agreed proposition.

Figure 13 shows the five gaps in the Basic Transaction Pattern.

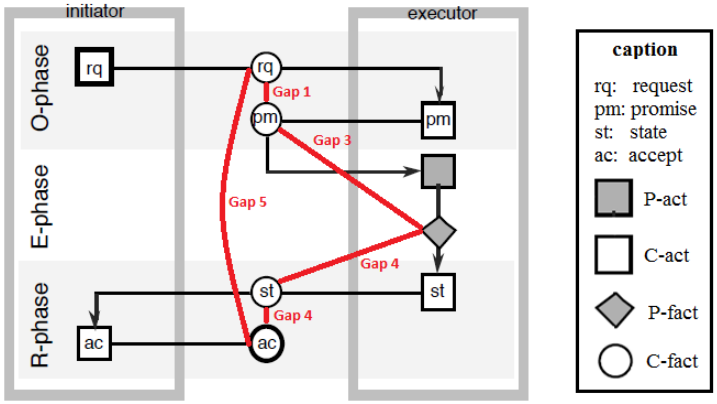


Figure 13 – Basic Transaction Pattern and the gaps model **(Mendes & Mira da Silva, 2012)**

The Gap 1 represents the misalignment between the customer's expectations and the provider's perception of those expectations, i.e. the difference between the proposition in the C-fact request and the proposition in the C-fact promise. This gap occurs when the customer expectations are not specified in clear service level agreements.

Gap 2 represents the misalignment between the provider's perception of customer's expectations and what is designed, so this gap is created at design time and therefore has no representa-

tion in Figure 13. However, this gap may be visible in the O-phase, more precisely in the difference between the proposition in the C-fact request and the proposition in the C-fact promise.

Gap 3 is the difference between what is promised (C-fact promise) and what is executed (P-fact). Consequently, this gap is influenced by the E-phase. As only the provider knows about the P-fact's existence, the initiator only becomes aware of this gap when the C-fact state occurs.

Gap 4 occurs when the communication about the service does not match the actual service delivery, so this gap represents the misalignment between the propositions of the coordination acts involved in the R-phase. This gap may also result of the misalignment between the P-fact and the C-fact state.

Gap 5 is the difference between what the customer requested and what really gets, i.e. this gap is the difference between the proposition of the C-fact request and the C-fact accept, and therefore is influenced by all the transaction phases and by the other four gaps.

EO already has a mechanism to deal with these gaps. In the R-phase, if the customer is not satisfied with the transaction result, he has the possibility to reject it. However, in these situations, both actors lose. The customer does not get the expected result and the service provider will probably lose this customer, not mentioning the time and resources both used in the transaction. This solution is what Information Technology Infrastructure Library (ITIL) calls a "workaround": a temporary fix that implies that a genuine solution to the problem is needed (Office of Government Commerce, 2007a). We also know that dealing with problems only after they occur is normally much more expensive than preventing these problems from occurring in the first place.

We propose to diminish the misalignments at their root, so the quality of the service exchange must be specified at the O-phase, stored and available to any actor that participates in the service delivery. This way customers have more control. They get all the information they want, whenever they want it.

2.5 Summary

The mapping between the gaps model and the EO basic transaction pattern (Figure 13) corresponds to the contribution 4 of this thesis (Section 1.6 – page 12). This mapping explains why the Enterprise Ontology theory does not solve the gaps problem by defining what each gap represents in the basic transaction pattern. Therefore, this mapping is the starting point to understand the contributions that EO and DEMO may have to this thesis problem.

3 Related Work

The conceptualization and measurement of service quality have generated a great deal of on-going discussion in the literature. Research papers from services marketing, services management and organizational psychology emphasize the importance of service quality in attracting, satisfying and retaining customers (Storbacka, et al., 1994) (Heskett, et al., 1997) (Schneider, et al., 1998).

This literature provides evidence that service quality has an integrating role between organizations and their customers. This integrating role is due to service quality being the outcome of internal organizational policies and practices, and fundamental in the service sequence that leads to customer value, satisfaction and loyalty (Storbacka, et al., 1994) (Zeithaml, et al., 1996) (Heskett, et al., 1997) (Cronin, et al., 2000).

This Chapter is divided in five sections. In Section 3.1 we introduce some service definitions (including the one used in this research) and in Section 3.2 we present the current solutions to identify services. Afterwards, we describe the solutions to specify the services quality (Section 3.3) and in Section 3.4 we provide an overview on the methods to measure services quality. Finally, we present our conclusions regarding the related work in Section 3.5.

3.1 Service Definition

The origin of the term services in English derives from the Latin *Servitium*, meaning "slavery". The related word servant has a similar derivation (Oxford Dictionaries). The modern usage of the term "service" arose out of the 1930s U.S. Department of Commerce's Standard Industrial Classification (SIC) codes. In these codes, the major economic sectors were agriculture, manufacturing, and services. At that time, services were a residual category for other activities that did not fit into agriculture or manufacturing (Foster & Metcalfe, 2001). Today, stretched beyond the point of being meaningful, that residual is the bulk of economic activity, and by far the fastest growing part of economic activity.

Today, "*a service is a change in the condition of a person, or a good belonging to some economic entity, brought about as the result of the activity of some other economic entity, with the approval of the first person or economic entity*" (Hill, 1977).

The U.S. government has accepted this definition as the basis for defining service products in the new North American Product Classification System (NAPCS) (Mohr & Russel, 2002). Other definitions of services can be found that emphasize an exchange between two or more parties and/or a transformation (potentially intangible) received by a customer (Giarini, 1987) (Quinn, 1988) (Foster & Metcalfe, 2001).

According to the Information Technology Infrastructure Library (ITIL), a service is "*a means of delivering value to customers by facilitating outcomes customers want to achieve without the ownership of specific costs or risks*" (Office of Government Commerce, 2007a). Although this definition represents a good starting point to understand what a service is, it is an abstract definition and does not identify all the service features, components and relations with the organization assets.

In the majority of the dictionaries, the service concept is associated with intangibility, being defined as an action, not a thing, that cannot be stored or reused. A service responds to user needs by adding benefits to their operations.

According to (International Organization for Standardization, 2000), a service is the result of at least one activity necessarily performed at the interface between supplier and customer, and is generally intangible. Provision of a service can involve, for example, the following:

- an activity performed on a customer-supplied tangible product (e.g. automobile to be repaired);
- an activity performed on a customer-supplied intangible product (e.g. the income statement needed to prepare a tax return);
- the delivery of an intangible product (e.g. the delivery of information in the context of knowledge transmission);
- the creation of ambience for the customer (e.g. in hotels and restaurants).

Service-dominant (S-D) logic (Vargo & Lusch, 2004) is an alternative to the traditional, goods-dominant (G-D) paradigm for understanding economic exchange and value creation. This service-centered view is based on the idea that a service – the application of competences for the benefit of another – is the basis of all exchanges.

The S-D logic can be summarized as follows (Vargo & Akaka, 2009):

- There are no "services." There is service, the act of doing something for another party, directly or through a good. We can serve but we cannot make services;
- There is no new service economy. Service has always been the basis of exchange. Manufacturing (and thus the Industrial Revolution) is just a special case of service provision;
- Value is always cocreated. If goods are used as vehicles of service they might be coproduced but the cocreation of value is not optional.

S-D logic has been identified as an appropriate philosophical foundation for the development of service science (Spohrer, et al., 2008). Figure 56 in Appendix A – Related Work Concept Models summarizes the concepts used in S-D logic.

A different research proposes an Ontological Foundation for Services Science and state that services are complex temporal entities (events) based on the central notion of commitment (Ferrario, et al., 2011a). This research uses the following service definition: *a service is present at a time t and location l if, at time t, an agent is explicitly committed to guarantee the execution of some type of action at location l, on the occurrence of a certain triggering event, in the interest of another agent and upon prior agreement, in a certain way* (Figure 57 and Figure 58 in Appendix A summarize the concepts used in this research).

3.1.1 Service Definition Chosen

We adopted the service definition that is based on the standard transaction pattern of Enterprise Ontology (Terlouw & Albani, 2013). Although a service has many similarities with a transaction, they are not equal. While a transaction includes all acts of the initiator and the executor, the service concept emphasizes more the executor side than the initiator side. Therefore, in DEMO a service is defined as part of a transaction rather than a whole transaction.

A service is a universal pattern of coordination and production acts, performed by the executor of a transaction for the benefit of its initiator, in the order stated in the standard pattern of a transaction. When the service is implemented the executor has the ability:

- *to get to know the coordination facts produced by the initiator;*
- *to make available to the initiator the coordination facts produced by itself.*

When looking at the basic transaction pattern (Figure 14), everything except the coordination acts of the initiator (request, quit, reject and accept) are part of the service. But in order to communicate with the executor of the service, the initiator needs to be aware of the standard transaction pattern.

This definition of service is very generic, since it holds for two kinds of providers: human actors and IT systems. Services executed by human actors or IT systems only differ in the way they are implemented; Human/Business services are implemented by human beings, whereas IT/web services are implemented by IT systems (Terlouw & Albani, 2013). These systems assist human actors in their activities; therefore parts of a human service may also be executed by IT systems.

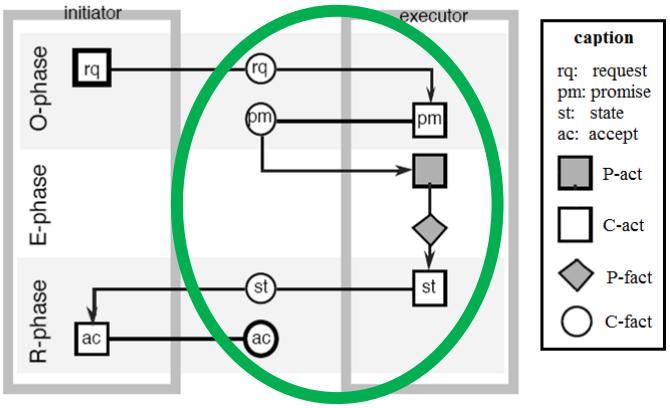


Figure 14 – Basic Transaction Pattern (Service elements are inside the green line)

We use this definition in this thesis, because this definition is the only one that, as our research, uses Enterprise Ontology as a conceptual foundation.

3.2 Service Identification

In this Section we analyse both Information Technology Infrastructure Library (ITIL) and Capability Model Integration for Services (CMMI-SVC), each containing processes that deal with Service Catalogues and thus service identification. We briefly review other two of the most studied service identification approaches: Business Components Identification and Service Oriented Architectures.

Service Portfolio Management (SPM) is one of the ITIL processes and it defines a Service Portfolio that includes the Service Catalogue (Office of Government Commerce, 2007b). This portfolio describes the services of a provider in terms of business value, articulating business needs and the service provider's response to those needs and, finally, requires the identification and definition of the provided services. However, ITIL does not explicitly present a way for identifying services.

CMMI-SVC is a process improvement framework for service provider organizations that helps to set process improvement goals and priorities, provides guidance for quality processes and a point of reference for appraising current ones, directed at organizations concerned with the delivery of services in general (Software Engineering Institute, 2010).

CMMI-SVC provides 24 process areas and, in the context of this work, the most relevant area is the Strategic Service Management. This process area involves activities such as analysing capabilities and needs for services that span multiple customers and agreements, and establishes and maintains standard services, its levels and descriptions. On the one hand, these standard services are typically described in a Service Catalogue that is oriented to the information needs of customers. On the other hand, like ITIL, CMMI-SVC does not provide a specific process to identify services.

In order to promote software reuse, Component-based Software Engineering (CBSE) techniques have been adopted to ease the development of large-scale complex information systems (Fan-Chao, et al., 2005). These techniques differ from traditional software engineering methodologies since their main focus is on acquiring reusable components which provide accessible services.

As described in (Wang, et al., 2005), there are three kinds of service identification techniques, based on components: Domain Engineering based methods, CRUD Matrix based methods and Cohesion-Coupling based Clustering Analysis methods. Instead of identifying services directly, these methods identify the components and, since each component provides services to the exterior, the authors claim we can also identify its services. Still, a specific process to perform this task is not provided.

There is also a preliminary work of using DEMO to identify business components and their provided services (Albani & Dietz, 2011). This work uses a three dimensional method for Business Components Identification (BCI-3D). BCI-3D aims at grouping business tasks and their corresponding information objects into business components. BCI-3D also allows to identify the services provided and required by each component. This solution allows for reduction of complexity of domain models and for identification of stable business components, which is suitable for the development of an information system. However, this solution was not designed to identify business services and the service definition is different from the one used in this thesis.

Another type of solutions to identify services are the ones based on Service Oriented Architecture (SOA). SOA is an architectural approach for designing, architecting and delivering enterprise applications that support business operations as a set of meaningful services. Many researchers suggested various methodologies to guide the migration to SOA, each with its own

approach to service identification. For example, in (Dietz & Terlouw, 2010) there is a review of some SOA methodologies such as SOMA (Arsanjani, et al., 2008) and SOAF (Erradi, et al., 2006). These methodologies provide a theoretical basis to achieve SOA, but they do not describe all phases in a practical way. Moreover, they are technology-based, focusing on Web services.

3.3 Service Quality

Service quality is closely related to customer satisfaction. Effectively managing customer satisfaction is a topic of vast interest because of their positive impact on customer behaviour and firm performance (Zeithaml, et al., 1996) (Rust, et al., 2002) (Morgan, et al., 2005) (Homburg, et al., 2005). Customer satisfaction is one of the primary factors leading to customer loyalty and continuation of relationships (Rust & Chung, 2006). For example, various studies have found that higher level of customer satisfaction lead to greater customer loyalty and word of mouth recommendations (Oliver, 1980) (Bolton & Drew, 1991) (Anderson & Sullivan, 1993) (Guo, et al., 2009) (Lai, et al., 2009).

We analysed several solutions to specify the services quality, including the Service Level Engineering solution, Service Level Management best practices, web services based solutions, Dynamic SLA solutions, Customer Service Experience solutions, and the Generic Service Specification Framework (GSSF). In spite of the different backgrounds, all contributed to the service quality specification.

In the end of each subsection we analyse the solutions regarding the fulfilment of the solution requirements defined in Section 1.5 (page 12). The symbol ● is used when the requirement is fulfilled, the symbol ◐ is used when the requirement is partially fulfilled, and the symbol ○ is used when the requirement is not fulfilled.

3.3.1 Service Quality Dimensions

In this Section, we present some classification systems that categorize the attributes used to specify the services quality.

According to the ISO 9000 (International Organization for Standardization, 2000), a quality of an artifact, a product or a service is the degree to which a set of inherent characteristics of the artifact, process, or service fulfils requirements.

Probably the most common classification of characteristics is the separation between functional and non-functional. A functional characteristic defines a function of a system or one of its components. A function is described as a set of inputs, the behaviour, and outputs. Functional characteristics may be calculations, technical details, data manipulation and processing, or a specific functionality that define what a system is supposed to accomplish.

Functional requirements are supported by non-functional requirements (also known as quality requirements), which impose constraints on the design or implementation (such as performance requirements, security, or reliability). Generally, functional requirements are expressed in the form "system must do <requirement>", while non-functional requirements are "system shall be <requirement>". The plan for implementing functional requirements is detailed in the system design. The plan for implementing non-functional requirements is detailed in the system architecture. According to (O'Sullivan, 2006), the categories of non-functional properties of services include temporal and locative availability, payment, price, obligations, rights, quality, security, trust, penalties and discounts. **In this thesis we will focus on the non-functional properties of services.**

The research in (Parasuraman, et al., 1985) reveals that, regardless of the type of service, consumers basically used similar criteria in evaluating service quality. These criteria were divided in 10 key categories: tangibles, reliability, responsiveness, communication, credibility, security, competence, courtesy, understanding/knowing the customer, and access. Later, they restricted the list to only five categories (Parasuraman, et al., 1988):

- **Tangibles** – physical facilities, equipment and appearance of personnel;
- **Reliability –** ability to perform the promised service dependably and accurately;
- **Responsiveness –** willingness to help customers and provide prompt service;
- **Assurance** – knowledge and courtesy of employees and their ability to inspire trust and confidence;
- **Empathy –** caring, individualized attention the organization provides to customers.

We will evaluate the impact of this thesis proposal in these five categories.

According to ISO 9000 (International Organization for Standardization, 2000), there are various classes of characteristics, such as physical (e.g. mechanical, electrical, chemical or biological characteristics), sensory (e.g. related to smell, touch, taste, sight, hearing), behavioural (e.g. courtesy, honesty, veracity), temporal (e.g. punctuality, reliability, availability), ergonomic (e.g. physiological characteristic, or related to human safety) and functional (e.g. maximum speed of an aircraft).

A classification system for goods that can also be applied to services was proposed in (Darby & Karni, 1973) and (Nelson, 1974). This system classifies the goods characteristics according to three categories: search properties, experience properties and credence properties. The first one, search properties, represents attributes that can be determined by the customer before the product/service purchase. Search attributes include attributes such as colour, style or price. Experience properties are attributes that can only be discerned after purchase or during consumption (taste, usability or dependability).

The last category, credence properties, represents characteristics which the customer may find impossible to evaluate even after purchase and consumption. Examples of offerings high in credence properties include appendectomies and engine repairs. Few consumers possess medical or mechanical skills sufficient to evaluate whether these services are necessary or are performed properly, even after they have been prescribed and produced by the seller.

3.3.2 Service Level Engineering

Service Level Engineering (SLE) is a systematic engineering approach to determine business-relevant service level indicators (SLIs) and efficient service level objectives (SLOs) for Service Level Agreements (SLAs) (Kieninger, et al., 2011a). SLE is related to the research fields of Service Engineering, Service Level Management, Business Driven IT Management and Quality Management.

This approach defines two costs (Kieninger, et al., 2011a):

- **Variable service cost** – The cost that the provider incurs for providing the service, including all direct and indirect costs that are variable with respect to the SLOs, i.e. that depend on the quality of service;
- **Business opportunity cost** – The cost for the customer that is incurred by an imperfect service compared to the cost of a perfect service.

An example of cost curves for variable service cost and business opportunity cost is shown in Figure 15.

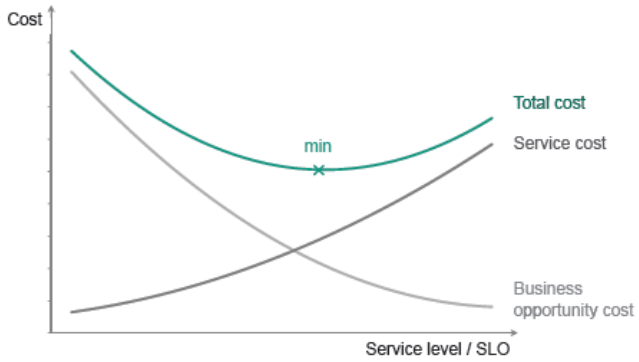


Figure 15 – Exemplary cost curves depending on the SLO **(Kieninger, et al., 2011a)**

With an increasing SLO, variable service cost rises, while business opportunity cost declines. Summing up both cost functions leads to a total cost function, which can be minimized. An SLO that minimizes total cost will be denoted as system-efficient SLO.

In (Kieninger, et al., 2011a) the authors argue that both parties (providers and customers) can always be better off by choosing system-efficient SLOs instead of SLOs that are not system-efficient.

Table 3 illustrates how the solution requirements defined in Section 1.5 (page 12) are addressed by the Service Level Engineering solution. As mentioned before, the symbol ● is used when the requirement is fulfilled, the symbol ◐ is used when the requirement is partially fulfilled, and the symbol ○ is used when the requirement is not fulfilled.

Requirements	**Analysis**
1 – Have a clear service definition	◐
2 – Define the attributes that should be used to specify a service	○
3 – Define clear steps to identify services	◐
4 – Have a clear service level agreement (SLA) definition	●
5 – Define the attributes that should be used to specify an SLA	○
6 – Define the services and SLAs from the customer point of view	●
7 – Provide for the uniqueness of each situation and of each customer	◐

Table 3 – Solution Requirements and Service Level Engineering

The Service Level Engineering solution describes an IT service *as a service that includes a crucial involvement of an IT system* and presents a framework that can be used in order to identify services (Kieninger, et al., 2011b). Additionally, an example of a service catalogue that contains IT services and service level indicators from a customer perspective is presented. Howev-

er, the service concept is not clearly defined, the services are only defined by their name, the SLA attributes are not specified, and although the uniqueness of each customer is taken into count the uniqueness of each situation is not. Therefore, requirements 1, 3 and 7 are partially fulfilled, requirements 2 and 5 are not fulfilled, and requirements 4 and 6 are fulfilled.

3.3.3 Service Level Management

Service Level Management is proposed by best practice frameworks, such as ITIL (Office of Government Commerce, 2007b) or CMMI-SVC (Software Engineering Institute, 2010).

According to ITIL, Service Level Management (SLM) is one of the key processes by which organizations manage their services, being an interface between the customer and the provider. At the most basic level, Service Level Management is involved in the following activities: define, agree, record and manage levels of service.

There are a number of key elements required to ensure that services are fit for purpose and fit for use, and remain so throughout their lifetime: service level requirements, targets and agreements (Office of Government Commerce, 2007b). Basically, to understand the Service Level Requirements (SLR) means that the customers' needs and wants are understood, i.e. an SLR is a customer requirement for an aspect of a service. SLRs are based on business objectives and are used to negotiate Service Level Targets (SLT) that are commitments documented in SLAs. SLTs are based on SLRs, and are needed to ensure that the service is fit for purpose. SLTs should be SMART: specific, measurable, attainable, realistic and timely. Finally, a Service Level Agreement (SLA) is an agreement between a provider and a customer that describes the service, documents the SLTs, and specifies the responsibilities of the provider and the customer. Over the years SLAs have been the chosen concept to specify services quality (Office of Government Commerce, 2007b).

There are some solutions based on ITIL that are focused on specific areas. Two examples are the Microsoft Operations Framework (MOF) and the Enhanced Telecom Operations Map (eTOM). MOF aims to be a practical guidance for everyday IT practices and activities, helping users establish and implement reliable, cost-effective IT services (Microsoft, 2008). It includes operational guidance to IT Service Management (ITSM) and provides operational "job aids" in support of service delivery. Instead of aiming to be generic, the MOF is tailored for environments using Microsoft solutions. MOF includes templates and demonstrates how generic concepts can be refined to serve infrastructure and operations specific needs.

A sophisticated multi-level process model, supported by an object-oriented information model, can be found in the Enhanced Telecom Operations Map (eTOM) (TeleManagement Forum,

2007). As it is designed to serve the needs of telecommunications providers the processes and data model are very specific but offer a good insight of how to align processes across customers and providers to use/offer services transparently.

Current approaches for Service Level Management have two main flaws. First, they lack a strong conceptual foundation because they were derived from best practices - not from a well-founded theory. The inexistence of a theory may cause incoherencies among those solutions (second flaw). Service Level Management solutions are process driven and not service driven. The solutions based on best practice frameworks are designed to work individually as processes and the interactions between these processes (such as Request Fulfilment, Service Level Management and Incident Management) are usually unclear (Figure 16). For instance, the connection between an incident and an SLA is not clearly explained in ITIL, or in CMMI.

Finally, these solutions are focused on the activities that must be executed to support the organization, instead of being designed to be customer-driven and focused on the interaction with the customer. This is clearly a disadvantage because, as we have mentioned before in Section 1.1, the solution should always be customer-driven.

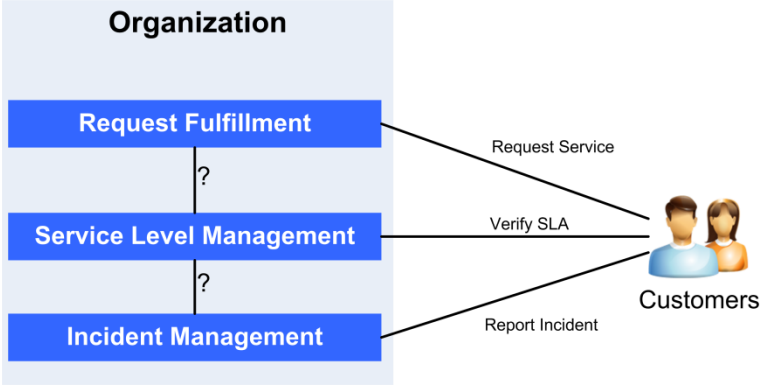


Figure 16 – Processes integration problem in best practices frameworks

Table 4 illustrates how the solution requirements defined in Section 1.5 are addressed by the Service Level Management solutions. The service and SLA definitions provided by the SLM solutions revealed to be too generic and, therefore, these solutions partially fulfil the requirements 1 and 4. SLM solutions do not define the attributes that should be included in the service (requirement 2) and SLAs (requirement 5) specification. SLM solutions define guidelines and not clear steps to identify services, so the requirement 3 is only partially fulfilled. Some include the customer in the service and SLAs specification, but still with some problems as mentioned before. Thus, the requirement 6 is also partially fulfilled.

Requirements	Analysis
1 – Have a clear service definition	◐
2 – Define the attributes that should be used to specify a service	○
3 – Define clear steps to identify services	◐
4 – Have a clear service level agreement (SLA) definition	◐
5 – Define the attributes that should be used to specify an SLA	○
6 – Define the services and SLAs from the customer point of view	◐
7 – Provide for the uniqueness of each situation and of each customer	○

Table 4 – Solution Requirements and Service Level Management Solutions

Finally, the last requirement is not fulfilled since these solutions do not provide for the uniqueness of each situation and of each customer.

3.3.4 Web Service based Solutions

There are also some solutions to specify the services quality that had origin in the web services community.

In recent years, the WSDL and UDDI standards arose as ad-hoc standards for the definition of service interfaces and service registries. The Web Service Definition Language (WSDL) standard enables provider and consumer to have a common view on the interface of the service by defining services into WSDL documents. A WSDL document uses the following elements in the definition of network services (W3C, 2001):

- **Types** – a container for data type definitions using some type system (such as XML Schema Definition - XSD);
- **Message** – an abstract, typed definition of the data being communicated;
- **Operation** – an abstract description of an action supported by the service;
- **Port Type** – an abstract set of operations supported by one or more endpoints;
- **Binding** – a concrete protocol and data format specification for a particular port type;
- **Port** – a single endpoint defined as a combination of a binding and a network address;
- **Service** – a collection of related endpoints.

The Universal Description Discovery Integration (UDDI) can be used as a means for publishing some service information (Bellwood, et al., 2004). UDDI enables businesses to publish service listings (WSDL documents) and discover each other, and defines how the services or software applications interact over the Internet.

WSDL and UDDI together do not provide enough basis to deal with questions regarding, for instance, the availability of the service, and the costs of calling the service. These standards do not offer enough basis for a service consumer to get a full understanding of the behaviour of a service. In practice this often leads to a serious mismatch between the provider's intent and the consumer's expectations concerning the functionality of the corresponding service ((Terlouw L. , 2011), page 113).

The Web Service Description Language (WSDL) and Web Service Flow Language (WSFL) were used to specify SLAs (Sahai, et al., 2002). However, this research work does not try to specify business services. For instance, the specifications do not include penalties or prices. The research presented in (Tosic, et al., 2002), (Dobson, 2004) and (Frolund & Koistinen, 1998) has the same limitation. Nevertheless, there are some recent papers that try to overcome this limitation. For example, the Web Service Level Agreement (WSLA) framework is a novel framework for specifying and monitoring SLAs for Web Services (Keller & Ludwig, 2003). This framework is applicable to any inter-domain management scenario, such as business process and service management or the management of networks, systems and applications in general.

There are other solutions from the web services community that included business criteria in SLAs (Andrieux, et al., 2007) (Liu, et al., 2004). In the first paper, the specification proposed consists of three parts that may be used in a composable manner: a schema for specifying an agreement, a schema for specifying an agreement template, and a set of port types and operations for managing agreement life-cycle, including creation, expiration, and monitoring of agreement states (Andrieux, et al., 2007).

The second paper proposes an extensible Quality of Service (QoS) model that is open, fair and dynamic for both service requesters and service providers. The paper considers three generic quality criteria that can be measured objectively for elementary services: execution price, execution duration, and reputation (Liu, et al., 2004). These three solutions represent a new movement in the web services community. However, none is based on a strong conceptual foundation.

Another solution is an SLM approach that is based on the Model Driven Architecture (MDA) (Debusmann, et al., 2004). The first step of this approach is to specify abstract reusable SLA patterns which are bound later to a concrete management platform and are finally physically deployed into the managed environment. This approach benefits from the advantages of the MDA and provides a mechanism for transforming SLA patterns for different SLA management platforms. This approach was prototypically implemented for the Web Services Level Agreement (WSLA) environment. Furthermore, this solution provided a relevant contribution to the

SLM area since it allows deploying SLAs across different platforms. However, this solution does not specify the attributes that each SLA should have.

An additional solution is the Unified Contract Management (Kabilan, et al., 2004) that is an ontology-based methodology intended to cover all phases of the lifecycle of a contract: conceptualization, negotiation and storage. This methodology assumes that a contract is the key to a successful relationship between partners who do not need to know each other (e-commerce) and includes a three layered approach to the creation of that contract.

The first layer consists of key concepts for any contract; the second one comprises the concept and terminology pertaining to a specific type of contract; and the third layer is a library of templates modelled on the basis of a shared domain ontology. This solution neither specifies the attributes belonging to each layer nor does it provide an explanation about how this could be implemented for services outside of the web services world.

Table 5 illustrates how the solution requirements defined in Section 1.5 are addressed by the Web Service based Solutions.

Requirements	**Analysis**
1 – Have a clear service definition	●
2 – Define the attributes that should be used to specify a service	●
3 – Define clear steps to identify services	◐
4 – Have a clear service level agreement (SLA) definition	●
5 – Define the attributes that should be used to specify an SLA	●
6 – Define the services and SLAs from the customer point of view	◐
7 – Provide for the uniqueness of each situation and of each customer	○

Table 5 – Solution Requirements and Web Service based Solutions

The web service based solutions fulfil requirements 1, 2, 4, and 5 since these are solutions with a clear service and SLA definitions and some define the attributes to specify services and SLAs. However, these solutions partially fulfil requirement 3 and 6, and do not fulfil requirement 7. The web service based solutions do not propose clear steps to identify services and some only allow to identify the components of the software that provide the services. These solutions suffer from the web tunnel vision since do not include business criteria in their specifications, thus the requirement 6 is partially fulfilled. Finally, these solutions do not provide for the uniqueness of each situation and of each customer and, consequently, do not fulfil requirement 7.

3.3.5 Dynamic Service Level Agreements

The concept of dynamic SLAs appeared in some research works. For example, in (Hedwig, et al., 2011) a novel integrated management model for information systems is proposed. This model induces economic influence factors into the operation strategy to adapt the performance goals of an enterprise information system dynamically. Other example is presented in (Chieng, et al., 2001) in which an agent-enhanced system that facilitates dynamic Service Level Agreement (SLA) activities is proposed. In (Taleb, et al., 2007) a dynamic service level negotiation scheme specifically tailored to satellite networks is portrayed. Another example is the extensions to the WS-Agreement protocol addressing dynamic SLAs (Pichot, et al., 2009). Other research presents the design principles of Dynamic Service Negotiation Protocol (DSNP). DSNP is a protocol to negotiate the SLS (Service Level Specification) in IP layer (Chen, et al., 2002). Besides these two protocols there are others that address the dynamic SLAs negotiation. Some examples are compared in the research (Sarangan & Chen, 2006).

The research project BREIN (Business objective driven Reliable and Intelligent grids for real busiNess) also contributed to the dynamic SLAs area (BREIN Project). BREIN was partially funded by the European Union and had as its main objective the creation of an infrastructure that supports collaboration among companies in a dynamic and changing environment.

BREIN developed an SLA Management Framework that is able to intelligently process SLAs by taking into account the business entities capabilities, policies and goals. The embedded use of semantic and multiagent concepts makes it more flexible in terms of problems discovery, the reaction to them, and through that provides the needed reliable solution for eBusiness (Frutos & Kotsiopoulos, 2009) (Karaenke & Kirn, 2010). Table 6 illustrates how the solution requirements defined in Section 1.5 are addressed by the dynamic SLAs solutions.

The dynamic SLA negotiation solutions fulfil the requirements 1, 2, 4, and 5. Nevertheless, all these research works are only applicable to web services, do not present clear steps to identify services, and do not stress the possibility of being applied to business services.

Requirements	Analysis
1 – Have a clear service definition	●
2 – Define the attributes that should be used to specify a service	●
3 – Define clear steps to identify services	◐
4 – Have a clear service level agreement (SLA) definition	●
5 – Define the attributes that should be used to specify an SLA	●
6 – Define the services and SLAs from the customer point of view	◐
7 – Provide for the uniqueness of each situation and of each customer	◐

Table 6 – Solution Requirements and current dynamic SLAs solutions

Therefore, requirement 3, 6, and 7 are only partially fulfilled.

3.3.6 Customer Service Experience

In this Section we included the solutions that try to capture the providers' services and processes from a customer point of view. Namely, we analysed, the total quality management, the Multilevel Service Design (MSD), the service blueprint and Process Description Capture Method (IDEF3) technique.

The total quality management techniques arose to solve the following problem: the indirect exchange masks the fundamental unit of exchange. As organizations continued to increase in size, they began to realize that virtually all their workers had lost sense of both the customer (Hauser & Clausing, 1988) and the purpose of their own service provision.

The workers, who performed micro specialized functions deep within the organization, had internal customers, i.e. other workers. One worker would perform a micro specialized task and then pass the work product on to another worker, who would perform another activity; this process continued throughout a service chain. Because the workers along the chain did not pay one another (reciprocally exchange with one another) and did not typically deal directly with external customers, they could ignore quality and both internal and external customers.

To correct this problem, various management techniques were developed under the rubric of total quality management (Cole & Mogab, 1995). The techniques intended to re-establish the focus of workers and the organization on both internal and external customers and quality.

Multilevel Service Design (MSD) is a new interdisciplinary method for designing complex service systems. MSD combines contributions from new service development, interaction design, and the emerging field of service design (Patrício L. , et al., 2011).

MSD enables integrated development of service offerings at three hierarchical levels: (1) designing the organization's service concept with the customer value constellation of service offerings for the value constellation experience; (2) designing the organization's service system, comprising its architecture and navigation, for the service experience; and (3) designing each service encounter with the Service Experience Blueprint (Patrício, et al., 2008) for the service encounter experience.

MSD offers a holistic view, from the service concept level to the multi-interface service system level and to each service encounter. The different levels of MSD provide different views of the service offering that can be used by different members of the design team and different decision makers (Patrício L. , et al., 2011).

MSD offers a great insight on how the customers interact with the providers and on how together they can co-create value. However, this method does not focus on service quality and on how service quality should be captured.

Another solution is the service blueprint that is a technique used for service innovation (Shostack, 1984). The blueprint shows processes within the company, divided into different components which are separated by lines ((Wilson, et al., 2012), page 180):

- **Customer Actions** – This element appears on top of the service blueprint and it contains all of the steps that customers take as part of the service delivery process;
- **Onstage / Visible Contact Employee Actions** – This element is separated from the customer actions by a 'line of interaction'. These actions are face-to-face actions between employees and customers;
- **Backstage / Invisible Contact Employee Actions** – The 'line of visibility' separates the Onstage from the Backstage actions. The elements that appear above the line of visibility can be seen by the customers, while the elements under the line of visibility are invisible for the customers. An example of an action in this element is a telephone call; this is an action between an employee and a customer, but they don't see each other;
- **Support Processes** - The 'internal line of interaction' separates the contact employees from the support processes. The support processes contain the activities carried out by individuals and units within the company who are not contact employees. These activities need to happen in a certain order for the service to be delivered;
- **Physical Evidence** – For each customer action the physical evidence that customers come in contact with is described at the very top of the service blueprint. These are all the tangibles that customers are exposed to that can influence their quality perceptions.

The process of structuring a service blueprint involves six steps:

1. The identification of the service process, that is supposed to be blueprinted;
2. The identification of the customer segment or the customers that are supposed to experience the service;
3. Picturing the service from the customer's perspective;
4. Picturing the actions of the contact employee (onstage and backstage), and/or technology actions;
5. Linking the contact activities to the needed support functions;
6. Adding the evidence of service for every customer action step.

Another solution of Customer Service Experience is presented in (Tseng, et al., 1999). A portraying scheme of service experience of customers based on the Integrated DEFinition for Process Description Capture Method (IDEF3) technique is proposed. The authors argue that they can improve service operations through uncovering the service experience of customers by following the customer's experience mapping. In other words, they get useful insight into service operations improvement, such as disclosing where the service operations should be improved so that the customer value perception can be enhanced, by examining systematically how customers behave, what they come in contact with, and what their journey is regarding the existing service operations system.

Table 7 illustrates how the solution requirements defined in Section 1.5 are addressed by the Customer Service Experience solutions.

Requirements	**Analysis**
1 – Have a clear service definition	●
2 – Define the attributes that should be used to specify a service	○
3 – Define clear steps to identify services	●
4 – Have a clear service level agreement (SLA) definition	○
5 – Define the attributes that should be used to specify an SLA	○
6 – Define the services and SLAs from the customer point of view	◐
7 – Provide for the uniqueness of each situation and of each customer	○

Table 7 – Solution Requirements and Customer Service Experience solutions

The Customer Service Experience solutions allow to identify services by specifying the processes that are triggered by the customers' actions. Therefore, a service is delivered by a service process that is initiated by the customers. This fulfils requirements 1 and 3. However, these so-

lutions do not specify the attributes of the services or identify the SLAs associated with the services. Consequently, these solutions do not fulfil requirements 2, 4, and 5. Customer Service Experience solutions partially fulfil requirement 6 since these solutions picture the service from the customer's perspective, but dot not do the same for the service quality. Finally, Customer Service Experience solutions do not approach the dynamics of the customers' expectations and, therefore, do not fulfil requirement 7.

3.3.7 Generic Service Specification Framework (GSSF)

GSSF is an Enterprise Ontology-based approach and even though the main goal of the GSSF was to specify the services and not the service quality itself, this framework also contributed to the problem area. The Generic Service Specification Framework (Terlouw & Albani, 2013) is based on the generic service definition proposed by Albani et al (Albani, et al., 2009).

This framework defines four main areas of concern for each service:

- **Service executor** – defines who is the provider of the service;
- **Service production** – focuses on the production act to be performed by the executor;
- **Service coordination** – gives the consumer all the information required for realizing a successful communication with the provider;
- **Service contract option** – specifies one or several contract options from which service consumers can choose.

The service executor area contains two aspects: the actor role and the contact information. The actor role aspect specifies the role of the actor that takes final responsibility for the service. In case of a human service, this is the actor role of the human executing the production act, while, in case of an IT service, this is the actor role of the human responsible for executing the production act that has delegated the responsibility to the IT system. The actor role aspect can be obtained from two types of diagrams provided by DEMO, namely the Actor Transaction Diagram and the Process Model (Figure 17).

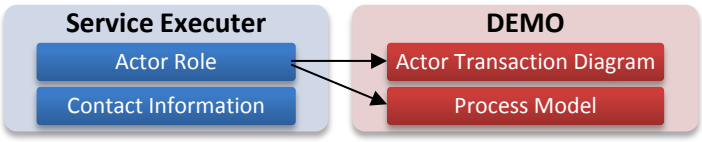


Figure 17 – The aspects of the service executor area and DEMO diagrams

The contact information aspect defines the service executor contacts that the initiator can use to contact the service executor and ask some questions about the service.

Figure 18 illustrates the aspects of the service production area.

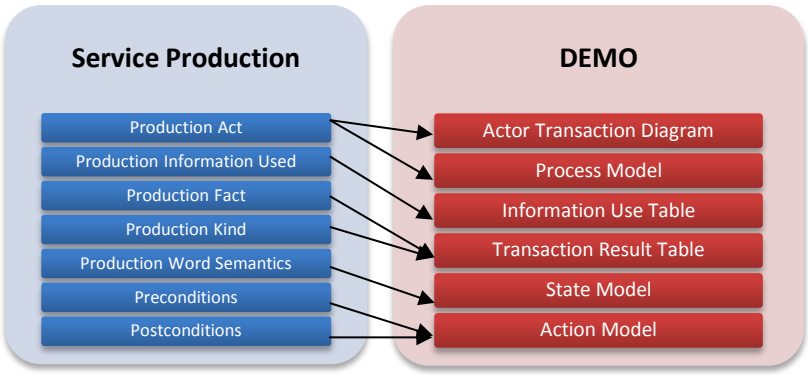


Figure 18 – The aspects of the service production and DEMO diagrams

The service production area contains seven aspects:

- **Production act** – is obtained from the Actor Transaction Diagram or the Process Model of DEMO;
- **Production information used** – is described in the Information Use Table (IUT) that defines for every coordination act and production act which information is required and therefore specifies the required input parameters;
- **Production fact** – is the result of the execution of a production act (the actual value requested by the initiator) obtained from the Transaction Result Table (TRT) of the DEMO;
- **Production kind** – defines the type of service that is being specified (ontological, infological and datalogical). The Transaction Result Table (TRT) defines the result type of each transaction and therefore defines the resulting production fact type for the service;
- **Production world semantics** – is modelled in the State Model of DEMO that allows a common knowledge and understanding about the semantics of the service to be provided;
- **Pre and postconditions** – defines production acts that should be executed before or after the execution of the service. Information about the pre- and postconditions are gained from the Action Model of DEMO. This model defines the operational business rules of an enterprise.

The service coordination area defines four aspects (Figure 19): coordination acts, coordination kind, protocol, and location.

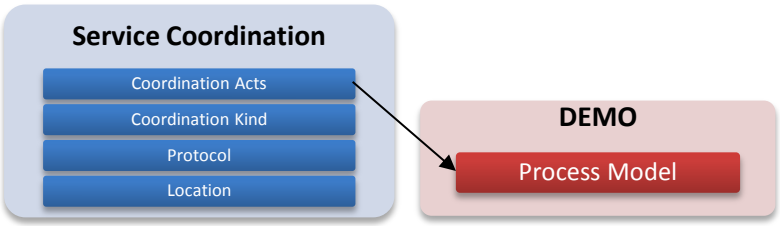


Figure 19 – The aspects of the service coordination area and DEMO diagrams

The first aspect describes the required coordination acts for communicating with the executor. These C-acts can be described using the Process Model of DEMO. The second aspect defines if the service is an IT service or a human service. This is helpful because IT systems and humans communicate in a different way. The protocol aspect enables a successful communication between the customer and the provider. Protocols define the rules governing the syntax, semantics and synchronization of communication. Finally, to request a service the initiator must know the location of the service, either physical or logical.

The service contract option area specifies one or several contract options from which service consumers can choose (Figure 20). The contract option aspect consists of a particular quality level and the price for using the service with this particular quality level. The service executor might define different quality levels in order to anticipate the various needs and financial positions of different consumers. The contract option aspect is not related with DEMO diagrams since this aspect is implementation dependent.

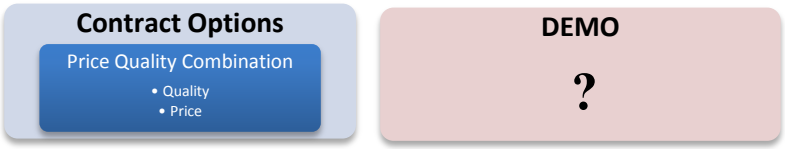


Figure 20 – The aspects of the contract options area and DEMO diagrams

The Generic Service Specification Framework represents a significant contribution in the service specification research area, even though the quality aspects are very basic. In particular, the level of service quality specification is not always sufficient because sometimes customers and providers get different expectations due to lack of specification (Terlouw & Albani, 2013). Table 8 illustrates how the solution requirements defined in Section 1.5 are addressed by the GSSF.

Requirements	Analysis
1 – Have a clear service definition	●
2 – Define the attributes that should be used to specify a service	●
3 – Define clear steps to identify services	●
4 – Have a clear service level agreement (SLA) definition	○
5 – Define the attributes that should be used to specify an SLA	○
6 – Define the services and SLAs from the customer point of view	◐
7 – Provide for the uniqueness of each situation and of each customer	○

Table 8 – Solution Requirements and GSSF

The GSSF fulfils the first three requirements since GSSF has a clear service definition, defines the attributes that should be included in a service specification, and GSSF as clear steps that should be used in order to identify services. However, GSSF does not have an SLA definition or a list of attributes for specifying SLAs. Consequently, GSSF can only identify services from the customer point of view and not SLAs. Finally, GSSF does not allow to provide for the uniqueness of each situation and of each customer.

3.4 Assessing Service Quality

In general, offerings high in search properties (attributes that a consumer can determine prior to purchasing a product) are easiest to evaluate, those high in experience properties (attributes that can only be discerned after purchase or during consumption) more difficult to evaluate, and those high in credence properties (attributes that the consumer may find impossible to evaluate even after purchase and consumption) hardest to evaluate (Parasuraman, et al., 1985). Most services contain few search properties and are high in experience and credence properties, making their quality more difficult to evaluate than quality of goods (Zeithaml V. , 1981).

The most popular measure of service quality is SERVQUAL, an instrument of 22 questions for assessing customer perceptions of service quality (Parasuraman, et al., 1988). SERVQUAL measures the gap between customer expectations and perceptions and was originally composed by 10 aspects of service quality (reliability, responsiveness, competence, access, courtesy, communication, credibility, security, understanding the customer and tangibles).

SERVQUAL has been mostly associated with person-to-person interaction, and, as different tools are needed to understand quality evaluation of technology-enabled services (Meuter, et al.,

2000), new measures of service quality that are suited to the Web environment have emerged. For example, SiteQual (Yoo & Donthu, 2001) and e-Servqual (Parasuraman, et al., 2005).

Over time, a few variants of SERVQUAL have also been proposed. For example, 'SERVPERF' has been proposed by (Cronin & Taylor, 1992). They questioned the conceptual basis of the SERVQUAL scale and found it confusing with service satisfaction. They, therefore, opined that expectation component of SERVQUAL be discarded and instead performance component alone be used.

In (Lages & Fernandes, 2005) the authors suggest that consumer final decisions are taken at a higher-level of abstraction. Similarly to SERVQUAL, the Service Personal Values (SERPVAL) is also multi-dimensional and presents three dimensions of service value: peaceful life, social recognition, and social integration. All three SERPVAL dimensions are associated with consumer satisfaction.

Despite being widely criticized, "SERVQUAL has enjoyed and continues to enjoy widespread acceptance as a measure of service quality" (Carrillat, et al., 2007) p. 473). SERVQUAL has also been extensively used (as is, or in adapted versions) for many different service settings including hospitals (Dean, 1999), restaurants and hotels (Bojanic & Rosen, 1994) (Saleh & Ryan, 1991), information systems companies (Kettinger & Lee, 1994), business schools (Rigotti & Pitt, 1992), and local governments (Donnelly, et al., 1995).

3.5 Summary

Table 9 summarizes the analysis that was done regarding the related work solutions fulfilment of the solution requirements. As mentioned before, the symbol ● is used when the requirement is fulfilled, the symbol ◐ is used when the requirement is partially fulfilled, and the symbol ○ is used when the requirement is not fulfilled.

The business world changed from products to services. The focus shifted away from tangibles to intangibles, such as skills, information, and knowledge, and towards interactivity, connectivity and on-going relationships. The orientation shifted from the producer to the consumer. The academic focus shifted from the thing exchanged to one on the process of exchange (Vargo & Lusch, 2004) (Vargo & Lusch, 2006). This shift highlights the importance of dealing with the gaps that influence the service quality.

Related Work Solutions	Requirements (see Section 1.5 – page 12)						
	Req. 1	**Req. 2**	**Req. 3**	**Req. 4**	**Req. 5**	**Req. 6**	**Req. 7**
Service Level Engineering	◐	○	◐	●	○	●	◐
Service Level Management	◐	○	◐	◐	○	◐	○
Web Service based Solutions	●	●	◐	●	●	◐	○
Dynamic SLAs	●	●	◐	●	●	◐	◐
Customer Service Experience	●	○	●	○	○	◐	○
Generic Service Specification Framework	●	●	●	○	○	◐	○

Table 9 – Solution Requirements and related work solutions

There are several solutions to closing the gaps, but none yet solved the problem completely. Some lack detail in specifying the services quality (like the Generic Service Specification Framework), others were not based on a strong conceptual foundation (such as ITIL or CMMI) and the majority of web services based solutions suffer from the web service tunnel vision.

From the researches mentioned in this Chapter we based our thesis proposal on the service definition from (Albani, et al., 2009) and on Generic Service Specification Framework (Terlouw & Albani, 2013). Additionally, we based our SLA proposal in some of the solutions presented in the Web Service based solutions subsection (subsection 3.3.4). Namely, we included in our SLA proposal attributes that are presented in (Keller & Ludwig, 2003), (Andrieux, et al., 2007) and (Liu, et al., 2004).

4 Thesis Proposal

This Chapter corresponds to the design and development step of Design Science Research Methodology (DSRM).

We start by describing our reasoning behind the proposal to specify the service quality. This reasoning is based on an example of a service exchange that allows to identify the basic attributes that compose an SLA (Section 4.1). Next, we describe the steps that are needed to apply our proposal in practice, we present a new version of the Generic Service Specification Framework (GSSF), and we describe the final SLA attributes proposal (Section 4.2). Then, we detail the dynamic SLA negotiation using DEMO models, i.e. we present a full white-box model for the dynamic SLA negotiation (Section 4.3). Finally, we conclude this Chapter with a summary (Section 4.4).

4.1 Motivation Example

In order to show what an SLA looks like using Enterprise Ontology, let us start with a simple example of a transaction conducted during a face-to-face communication between two subjects (Table 10). The example describes a customer (C) buying a bouquet of flowers from a florist (F). The right column of the table has the analysis of this example in terms of the three kinds of communicative acts (see Section 2.2 – page 18).

The proposition of the performative act in line 2 (request) is fully known in line 10, so the request is considered to be performed successfully. We denote the proposition '*Deliver a bouquet of 15 roses in 5 minutes at 30€, otherwise the customer won't wait'* by P.

Interaction	Enterprise Ontology representation
(1) F: Hello! How can I help you, Mister Bean?	Expressivum; will be disregarded.
(2) C: I want to buy a bouquet of flowers.	Performative act; intention: rq; proposition: <unknown>.
(3) F: Do you want something specific?	Informative act, serving to clarify the proposition.
(4) C: Yes, I want roses.	Informative act, serving to clarify the proposition.
(5) F: How many?	Informative act, serving to clarify the proposition.
(6) C: 15 please.	Informative act, serving to clarify the proposition.
(7) F: When do you need the bouquet?	Informative act, serving to clarify the proposition.
(8) C: In the next 5 minutes.	Informative act, serving to clarify the proposition.
(9) F: Ok, that will cost you 30€.	Informative act, serving to clarify the proposition.
(10) C: No problem, but I only have 5 minutes.	Informative act, serving to clarify the proposition.
(11) F: Ok, one moment, please.	Performative act; intention: promise; proposition: P.
(12) < F makes the bouquet >	Material act related to the production act
(13) < F puts the bouquet in front of C >	Material act related to the production act
(14) F: Here you are, Mister Bean.	Performative act; intention: statement; proposition: P.
(15) C: Thanks.	Performative act; intention: acceptance; proposition: P.

Table 10 – Florist example of service exchange

According to this example, we propose that:

A service level agreement is the proposition that two actors (initiator and executor) build together in the O-phase of any ontological transaction. This proposition is clarified by informative acts.

A first analysis of the agreed proposition shows that an SLA may have at least four elements, as illustrated in Figure 21.

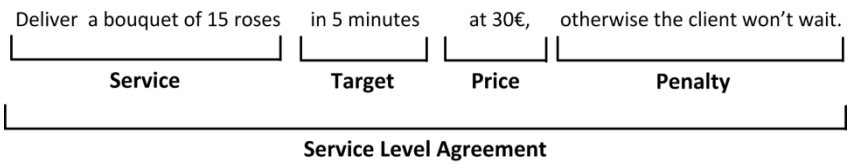


Figure 21 – Service level agreement elements (first analysis)

One may think that EO can capture this type of information but, as we mentioned, EO specifies the organization essence and ignores how the organization is implemented. Consequently, the DEMO models capture the ontology part of the proposition, i.e. the part that corresponds to the service. In this example: 'Deliver a bouquet of 15 roses'. The remaining would be disre-

garded (or it would be included in informal expressions inside the action rules specifications), because it is not part of the organization essence.

The first element that must be defined in an SLA is the service. This is a mandatory element of any SLA, because there is no SLA without a service. In this example, the service is to deliver a bouquet of 15 roses.

The second element is the price that, in fact, represents the price of choosing that specific SLA. In this case the price is 30€. The price should always be quantitative (ex. 30) and the currency should be explicit (ex. € or $), so there is no room for misunderstandings about this element. In the example, the total price of the service is equal to the price of the SLA, but they are conceptually different. The SLA price is just a component of the total price of the service. However, the relations with other components that may also affect the services price will be dealt with later.

The third element is the target that defines a clear goal to be achieved when the service is delivered. As stated before (Section 3.3.3 – page 36), the target should be SMART: specific, measurable, attainable, realistic and timely. In this example, the target is to deliver the bouquet in 5 minutes, i.e. the florist state act should be done up to 5 minutes after the promise act; otherwise the agreed penalty applies.

The last element is the penalty that describes what happens if the target is not achieved. In this example, the penalty is that the customer will leave without completing the transaction, i.e. the customer will cancel his request.

Therefore, each SLA is composed by four elements: service, target, price and penalty. Figure 22 illustrates the relations between these SLA elements and the DEMO models and diagrams.

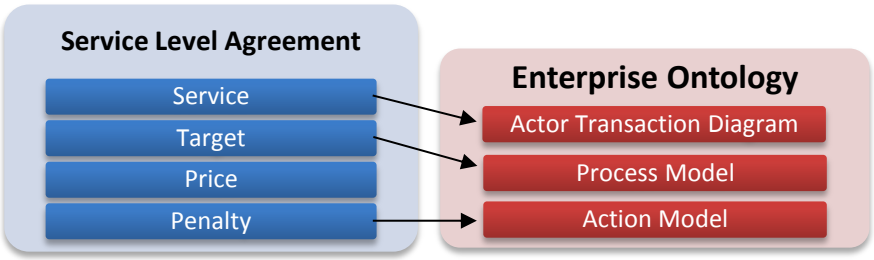


Figure 22 – Service level agreement proposal and relation to Enterprise Ontology

The service can be specified by using the Actor Transaction Diagram and the Generic Service Specification Framework (Terlouw & Albani, 2013). The target may be partially obtained from the Process Model, because this model specifies possible sequence of steps (coordination and production acts) that can be executed in order to achieve the expected result.

The price has no direct representation in the DEMO models and diagrams, as it is implementation dependent. Information about the penalties is gained from the Action Model because this model defines the operational business rules of an organization.

In this case, the service is provided with only one SLA attached. However, a real-world example would be a lot more complex. So we propose that a service may have several SLAs attached. Therefore, the services quality may be specified through a number of combinations of targets, prices and penalties.

Looking at the example, the SLA is defined at runtime, i.e. the provider and the customer decide the service quality when the service is being delivered. However, the service quality may also be defined at design time, i.e. the provider and the customer define several levels of services quality (each involving a different price) from which the customer can choose when requesting the service.

This SLA definition **contributes** to answer the research question Q2 (how to specify service level agreements according to customers' expectations?). Moreover, this definition contributes to the validation of the hypothesis H2.1 **(**the Service Level Agreement (SLA) concept is related with the proposition/result of Enterprise Ontology transactions).

4.2 Proposed Method

Our proposed method is composed by four steps and is illustrated in Figure 23 using the Flow Chart technique.

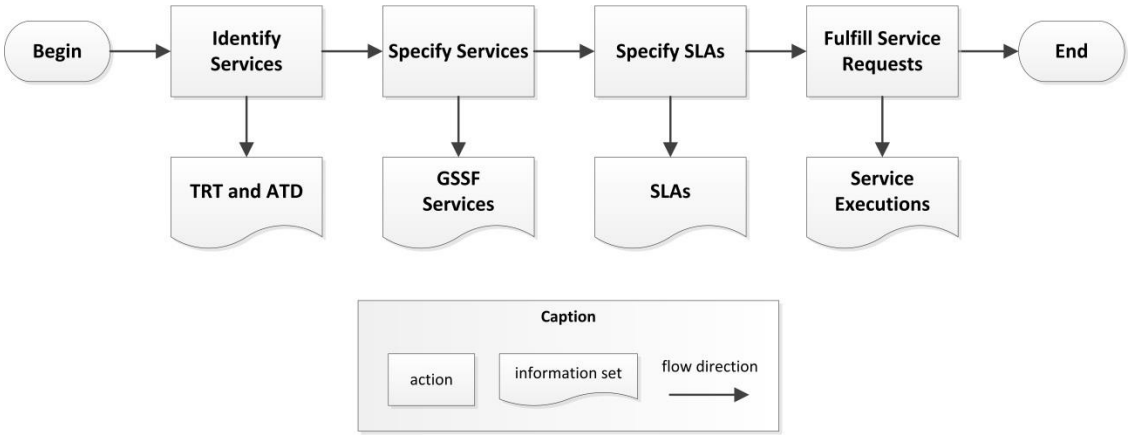


Figure 23 – Proposed method

The **first step** is to identify the services of the provider, using for that purpose a process based on the methodology proposed in (Dietz J. , 2006). This process is composed by six steps:

1. **Enterprise Description:** Write a textual description that summarizes the actions performed by the service provider to fulfil the customer requests. If not provided the enterprise description should be achieved through interviews with both provider and customers. The interviewers should prepare the interviews by collecting documentation about the provider services and processes. The questions asked in the interviews should be focused on the interaction between provider and customers (for example 'Which are the interactions that you have with other employees/customers in order to perform your functions? What do they request from you? What do you request from them?'). It is also important to validate the enterprise description with the provider and customers in order to validate if the enterprise description describes the actions performed by the provider with a vocabulary that both understand;
2. **Performa-Informa-Forma Analysis:** Identify the three kinds of human abilities (Performa, Informa, Forma) performed by the services provider, according to the Distinction axiom, using the text of the previous step;
3. **Coordination-Actors-Production Analysis:** Split the identified Performa items into C-acts/facts, P-acts/facts and Actor roles who perform these acts, corresponding to the Operation axiom;
4. **Transaction Pattern Synthesis:** Identify each Transaction type and result, based on the acts/facts identified in the previous step, according to the Transaction axiom;
5. **Result Structure Analysis:** Check if there are any dependencies between the transaction types identified in the Transaction Pattern Synthesis. Generally, these dependencies occur when the executor of a transaction is the initiator of another (inner) transaction;
6. **Actor Transaction Diagram/Service Identification:** Identify the initiator and executor actor roles of each transaction type. When this mapping between transactions and actor roles is complete, we can identify the services provided by the organization being studied.

The **second step** of the method is to apply a new version of the Generic Service Specification Framework (GSSF). Figure 24 illustrates this new version that is a result of using the original GSSF in some field studies. In each field study, we found some possible improvements to the GSSF that we present bellow (in Chapters 8 and 9 we justify the changes to the original GSSF).

The new GSSF version is composed by five areas of concern:

- **Service Basic Information** – defines the name and the base price of the service;
- **Service Executor** – defines who the provider of the service is (same attributes as the previous GSSF version);

- **Service Production** – focuses on the production act to be performed by the executor (same attributes as the previous GSSF version);
- **Service Coordination** – focuses on the possible coordination acts that can be performed by both provider and customer;
- **Service Configuration** – defines the different configuration options available and respective prices. These configurations and prices have no direct representation in the DEMO models and diagrams, as these are implementation dependent. This area is also composed by the location and protocol attributes that in the previous version of the GSSF were included in the Service Coordination area. We have made this change because these two attributes are directly related with the possible configurations available for each service.

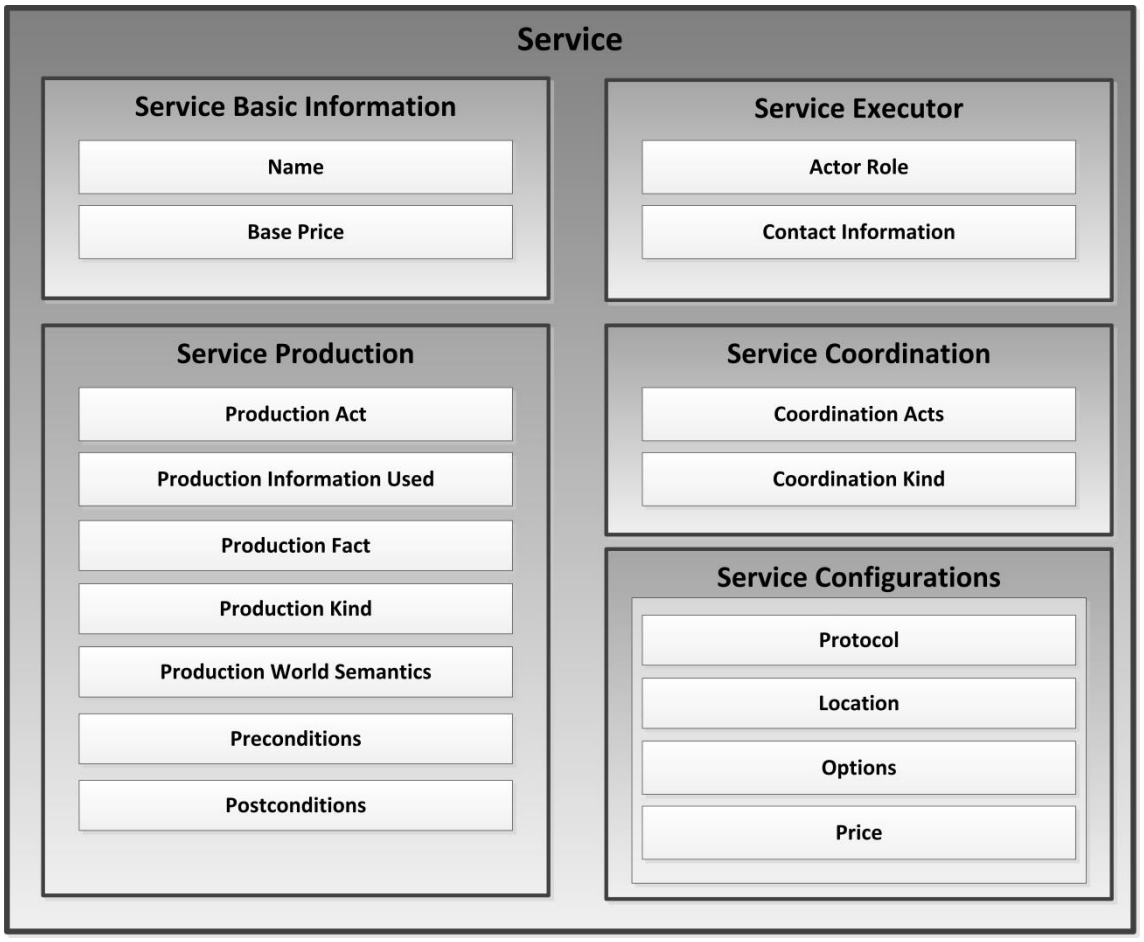


Figure 24 – Generic Service Specification Framework (GSSF) new version

The contract options area of concern of the GSSF is replaced by our definition of SLA presented in Figure 25. That means, for each identified service, one should specify the list of associated SLAs using our SLA definition (**third step**).

Our SLA proposal considers three main areas of concern and 13 attributes. This proposal began with the four attributes gathered from the motivation example presented in page 51. Afterwards, we did a deep literature review in the service level management area. We used the following criteria to choose the attributes:

- First, if the attribute was present in at least two solutions it was taken into consideration;
- Second, the attribute had to be a non-functional attribute;
- Finally, the attribute had to related with the customers' expectations.

Next, we validated the proposal with seven practitioners involved in SLAs specification that suggested the revision of some of the attributes labels. Finally, we evaluated this proposal in several organizations (Mendes & Mira da Silva, 2012) (Mendes, et al., 2012).

The first area is called SLA Basic Information and this area defines the name of the SLA (Name) and the name of the service that the SLA is applied to (Service).

Next, we define an area called SLA Responsibility Information that considers the information about each actor's responsibilities in the execution of this SLA. In this area, two attributes are specified concerning the customer's obligations and duties (Customer Responsibilities) and those of the service provider (Provider Responsibilities). These attributes have no direct representation in the DEMO models.

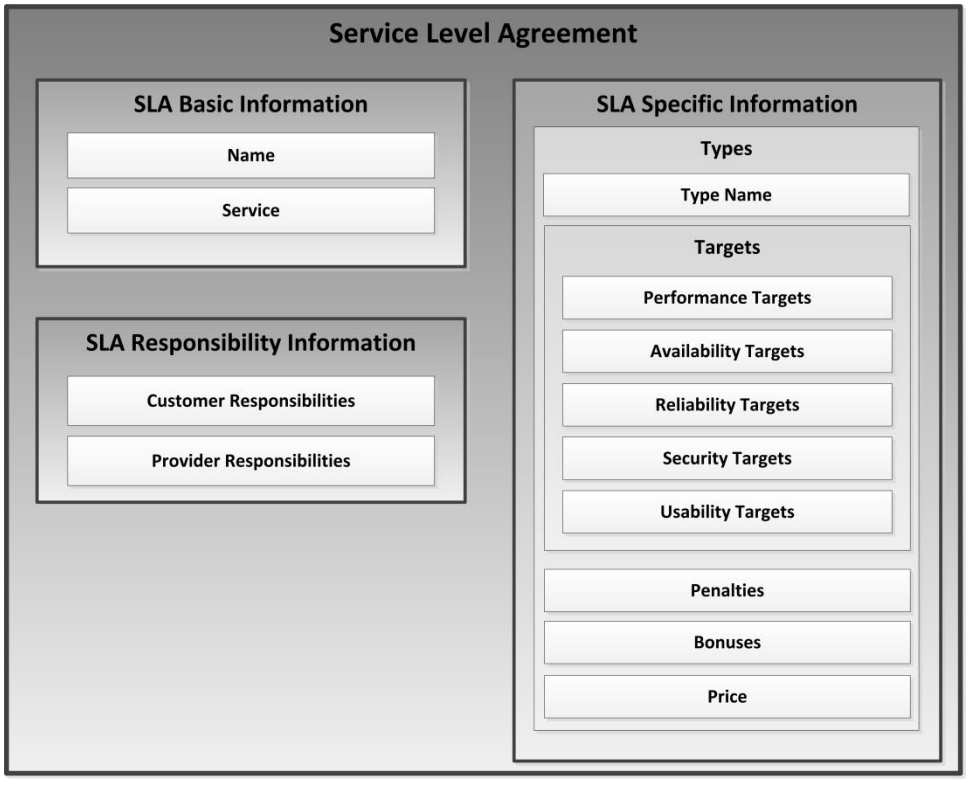


Figure 25 – SLA attributes proposal

Finally, the last area is called Specific SLA Information and this area specifies the types of the SLA (type name), five different types of targets (targets) that can give rise to actions if they are not fulfilled (penalties), or rewards (bonuses) if they are fulfilled. Each type of SLA is also associated to a price (Price). Each SLA can have several types and each type different targets. This way different combinations of service quality can be defined. In one hand, this proposal allows to define specific targets to be achieved in each service request, as in the field studies of Chapters 7, 9 and 10. In other hand, it allows to define ranges of targets as in the field study of Chapter 8. These relations are formalized in the Object Fact Diagram (OFD) (Figure 32 – page 71) of the DEMO white-box model for the dynamic SLA negotiation.

The connections between these SLA attributes and Enterprise Ontology are the following:

- The service attribute can be gained from the GSSF specification;
- The performance targets can be based on the acts present in the universal transaction patterns of Enterprise Ontology as we will demonstrate on field studies 7, 8 and 10;
- The penalties can be described by transactions that must be executed if the targets are not fulfilled;
- The bonuses can be described by transactions that must be executed if the targets are fulfilled.

The other attributes are implementation dependent and therefore have no representation in DEMO models.

At execution time, the initiator of the transaction has two options: choose SLAs from the list specified at design time or, if these SLAs do not express the initiator expectations at that moment, initiate a negotiation for a new SLA, in other words dynamically defining an SLA (**fourth step**). Step 4 is described in detail in Section 4.3.

In summary**, we propose to reduce the gaps by formally specifying the SLAs using the EO theory as foundation.**

Table 11 illustrates how the thesis proposal steps relate to the research questions, hypotheses, contributions, and field studies. The service identification process (step 1) concerns the research question Q1, the hypotheses H1.1 and H1.2, the contribution C1, and was evaluated in the six field studies. The service specification step (step 2) concerns the research question Q1, the hypotheses H1.1 and H1.2, the contribution C2, and was evaluated in the field studies described in Chapters 7, 8, 9, and 10. The SLAs specification step (step 3) concerns the research question Q2, the hypotheses H2.1 and H2.2, and the contributions C3 and C4. The step 3 was evaluated in the field studies described in Chapters 7, 8, 9, and 10. Finally, the service requests fulfilment

step (step 4) concerns the research question Q3, the hypothesis H3.1, and the contribution C5. The step 4 was evaluated in the field study described in Chapter 10.

		Step Result	Research Questions	Hypotheses	Contributions	Field Studies
Method Steps	1	Services Identification	Q1	H1.1, H1.2	C1	Ch. 5 to 10
	2	Services Specification	Q1	H1.1, H1.2	C2	Ch. 7 to 10
	3	SLAs Specification	Q2	H2.1, H2.2	C3,C4	Ch. 7 to 10
	4	Service Requests Fulfilment	Q3	H3.1	C5	Ch. 10

Table 11 – How the method steps relate to the research questions, hypotheses, contributions and field studies

This thesis does not intend to propose changes to the DEMO models. The level of abstraction of the models produced by applying this thesis proposal and the ones from DEMO are **different**. DEMO produces ontology models completely independent of the implementation; the application of GSSF produces models more implementation dependent; and the application our proposal produces models even more implementation dependent. To do so, we propose to extend the Generic Service Specification Framework (GSSF) with a Service Level Agreement concept in order to achieve a white-box model of lower level (i.e. more implementation dependent) that can provide a more detailed description of the services quality.

4.3 Proposal Ontology

In order to formalize the proposal we present a DEMO white-box model for the dynamic SLA negotiation (step 4 of the proposed method). A white-box model is a direct conceptualization of the ontological system definition and captures the construction and the operation of a system, while abstracting from implementation details (Dietz J. , 2006). As mentioned in Chapter 2, this white-box model is composed by four models (see Figure 10 – page 22): Construction Model

(CM), Process Model (PM), Action Model (AM), and State Model (SM). These models can be described using the diagrams and cross-model tables described in Figure 11 (page 23).

4.3.1 Construction Model (CM)

The first model is the Construction Model and is described by the Transaction Result Table (TRT), the Result Structure Chart and the Actor Transaction Diagram (ATD). In order to achieve this model we applied the way of work defined in (Dietz J. , 2006) to a text that describes the dynamic SLA negotiation. First, we distinguished between the Ontological, Infological and Datalogical actions described, as referred in the Enterprise Ontology Distinction axiom (see Section 2.2 – page 18). This step is called Performa-Informa-Forma Analysis. To do that, we defined a notation to differentiate those actions: in this example, we have highlighted the text, using red, green and blue colours to identify, respectively, the Ontological, Infological and Datalogical actions.

The next step concerns the identification of C-acts/facts, P-acts/facts and actor roles, using the Performa (Ontological) items identified in the previous step. We also have considered a notation to differentiate between them, similar to the one used in the Operation axiom: square brackets "[" and "]" to identify actor roles, brackets "(" and ")" to identify C-acts/facts and angled brackets "<" and ">" to identify P-acts/facts. This step is called Coordination-Actors-Production Analysis and here there is a reduction of the complexity, relatively to other methodologies, because from now on we will only consider the Ontological actions identified in this step. The result of these two analyses is presented below:

At execution time, the initiator of the transaction has two options: choose an SLA from the list specified at design time or initiate a negotiation for a new SLA, in other words defining dynamically an SLA (step 4).

*When the [customers] (**request**) the [provider] to <**fulfil**> a certain service, the customers can choose the service configurations and SLAs defined before, or they can (**request**) the service fulfilment with new a service configuration using the structure of Figure 24 and/or with a new SLA using the structure of Figure 25. In both cases, the provider should make available an average of the service requests performance regarding that specific customer. This way the customer has a customized reference about the provider capabilities that he can follow. This allows the customer to align his expectations in the beginning of the service fulfilment.*

*In the second case, the [provider] should <**create**> a new service configuration and/or an SLA according service expectations. Both should be <**approved**> by [someone] that has the*

*knowledge to understand if the organization has the capabilities to comply with the new service configurations and SLAs. The new service configurations and SLAs may involve the <**end**> of available configurations and SLAs due to incompatibilities.*

*The [provider] should also periodically <**monitor**> the SLAs fulfilment in order to measure the SLAs performance. This monitoring allows to notify the service executors accountable for services requests that are close to break SLAs. Furthermore, if a specific SLA is continuously being broken then the [provider] may conclude that this SLA need to be <**ended**> and new one <**created**>. Additionally, the information created by the SLAs fulfilment monitoring should be available to both customers and service executors. With this information in hand the [customers] can (**request**) the application of the agreed penalties if the SLAs were not fulfilled. In contrast, if the SLAs were fulfilled then the [provider] can (**request**) the application of the agreed bonus.*

*If the service executions require a payment by the [customer] then this payment can be (**requested**) directly by the [service executor] in each service interaction, and/or the [provider] may (**request**) these payments periodically.*

After these analyses, we identified the transactions by clustering the identified C-acts/facts and P-acts/facts in what is denominated the Transaction Pattern Synthesis. The Transaction axiom can be helpful in this step, because it guarantees that each P-act/fact or C-act/fact previously found corresponds to a complete transaction. Then, for each identified transaction type, the result type (i.e., the Production fact created) was formulated. The result is presented in the Table 12 below, called Transaction Result Table.

Transaction Types	**Result Types**
T01 – Service fulfilment	R01 – Service Execution E has been completed
I-T01 – Custom-made targets calculation	I-R01 – Custom-made targets have been calculated
T02 – Service Configuration start	R02 – Service Configuration C has been started
T03 – Service Configuration approval	R03 – Service Configuration C has been approved
T04 – Service Configuration end	R04 – Service Configuration C has been ended
T05 – SLA start	R05 – SLA A has been started
T06 – SLA approval	R06 – SLA A has been approved
T07 – SLA end	R07 – SLA A has been ended
T08 – SLA monitoring	R08 – SLA monitoring has been done for Period P
I-T08 – SLA warnings notification	I-R08 – SLA Warnings for Period P has been sent
T09 – Penalty application	R09 – Penalty P has been applied
T10 – Bonus application	R10 – Bonus B has been applied
T11 – Payments management	R11 – Service Execution payments have been managed for period P

Transaction Types	Result Types
T12 – Service payment	R12 – Service Execution E has been paid

Table 12 – Transaction Result Table (TRT)

We propose the existence of 12 ontological transactions and two infological transactions. Transaction T01 represents the executions of the services that the provider has available to customers. I-T01 calculates the customized SLAs targets according to the customer. T02 allows to create new Service Configurations according to the structure defined in Figure 24. The execution of T03 approves the new Service Configurations and the execution of T04 disables available Service Configurations. T05, T06 and T07 follow the same pattern but for SLAs, i.e. T05 creates new SLAs, T06 approves new SLAs and T07 ends SLAs. These SLAs have the structure proposed in Figure 25.

T08 monitors the fulfilment of the SLAs in determined period and T08 allows to notify the accountable persons for service executions that are close to break SLAs (I-T08). T08 also produces information that can be used in two other transactions, T09 and T10. T09 may be requested by the customer when the SLAs are not fulfilled and it applies the agreed penalties. T10 may be requested by the provider when the SLAs are fulfilled and it applies the agreed bonuses. Finally, the last two transactions manage the service execution payments. T11 allows to periodically manage who are the customers with payments to be done and T12 can be used by the provider to request the payment of individual service executions.

Figure 26 illustrate the dependencies between the transaction results. If the dependency is such that the bringing about of a result B is initiated during the process of bringing about a result A, and that the completion of A has to wait for B to be completed, then B is a component of A (Dietz J. , 2006).

The result of T1 (R1) depends directly on two transactions (T2 and T5) and indirectly on four transactions (T3, T4, T6 and T7). The service configuration start (R2) always depends on the service configuration approval (R3) and may need a service configuration end (R4). The SLA start (R5) has a similar pattern since R5 always needs an SLA approval (R6) and it may need an SLA end (R7). The payments management (R11) may initiate several service payments (R12) and the SLA monitoring (R8) may initiate several bonus applications (R9). Finally, the penalty application (R10) has no dependencies.

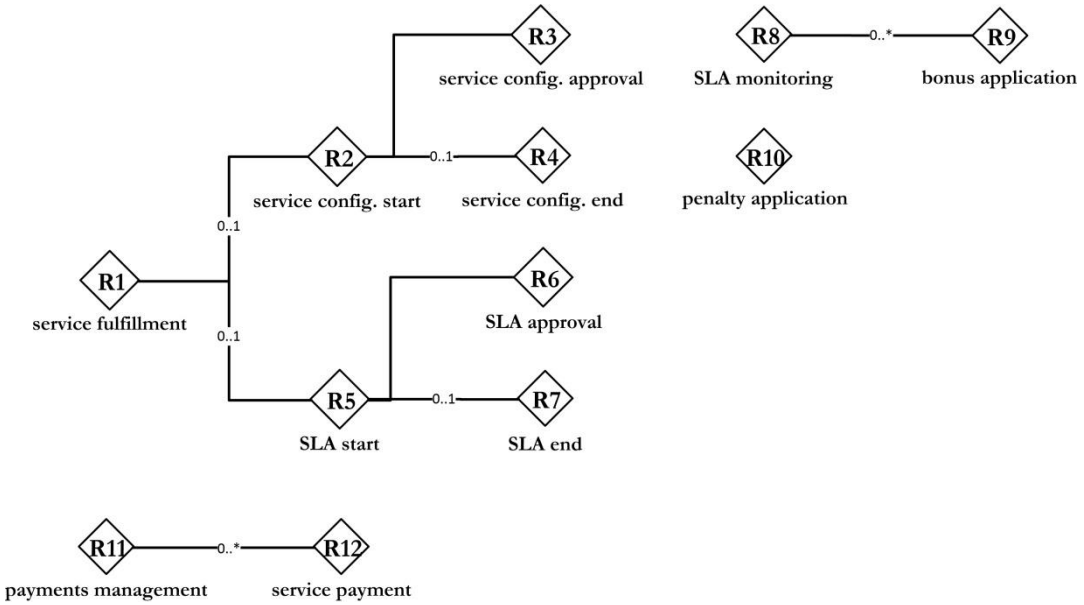


Figure 26 – Result Structure Chart

After identifying the Ontological transaction types, its results and dependencies, we modelled the Actor Transaction Diagram (ATD), represented in Figure 28. In this type of diagrams, a transaction is represented using a diamond in a disk that contains the respective combination of C-acts and a P-act. Each transaction is connected to two boxes, representing the initiator and executor actor roles. The initiator is connected to the transaction symbol using a solid line, while the executor is connected to the transaction using a solid line ending in a black square. The gray boxes refer to composite actor roles, i.e. elements whose exact structure is not known. All the environmental elements (i.e. elements outside the provider) are represented in gray boxes for that reason. The white boxes refer to elementary actor roles (Figure 27).

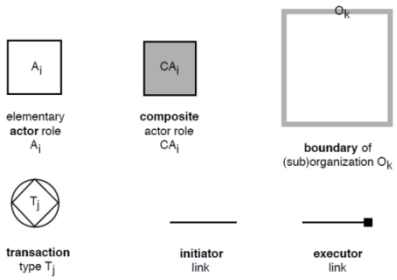


Figure 27 – Actor Transaction Diagram (ATD) caption

We suggest the existence of 11 ontological actor roles: customer, service executer, service configuration starter, service configuration approver, service configuration ender, SLA starter, SLA approver, SLA ender, SLA monitor, penalties applicator and payments manager.

The first one is a composite actor role (CA01) and it should be fulfilled by the service's client. CA01 has four responsibilities: initiator of transactions T01 (service fulfilment) and T10 (penalty application), and executor of transactions T09 (bonus application) and T11 (service payment). According to the DEMO restriction that defines a limit of a single executor by transaction, we can conclude that this composite actor role can be divided at least into two elementary actor roles, i.e. the number of transactions CA01 executes.

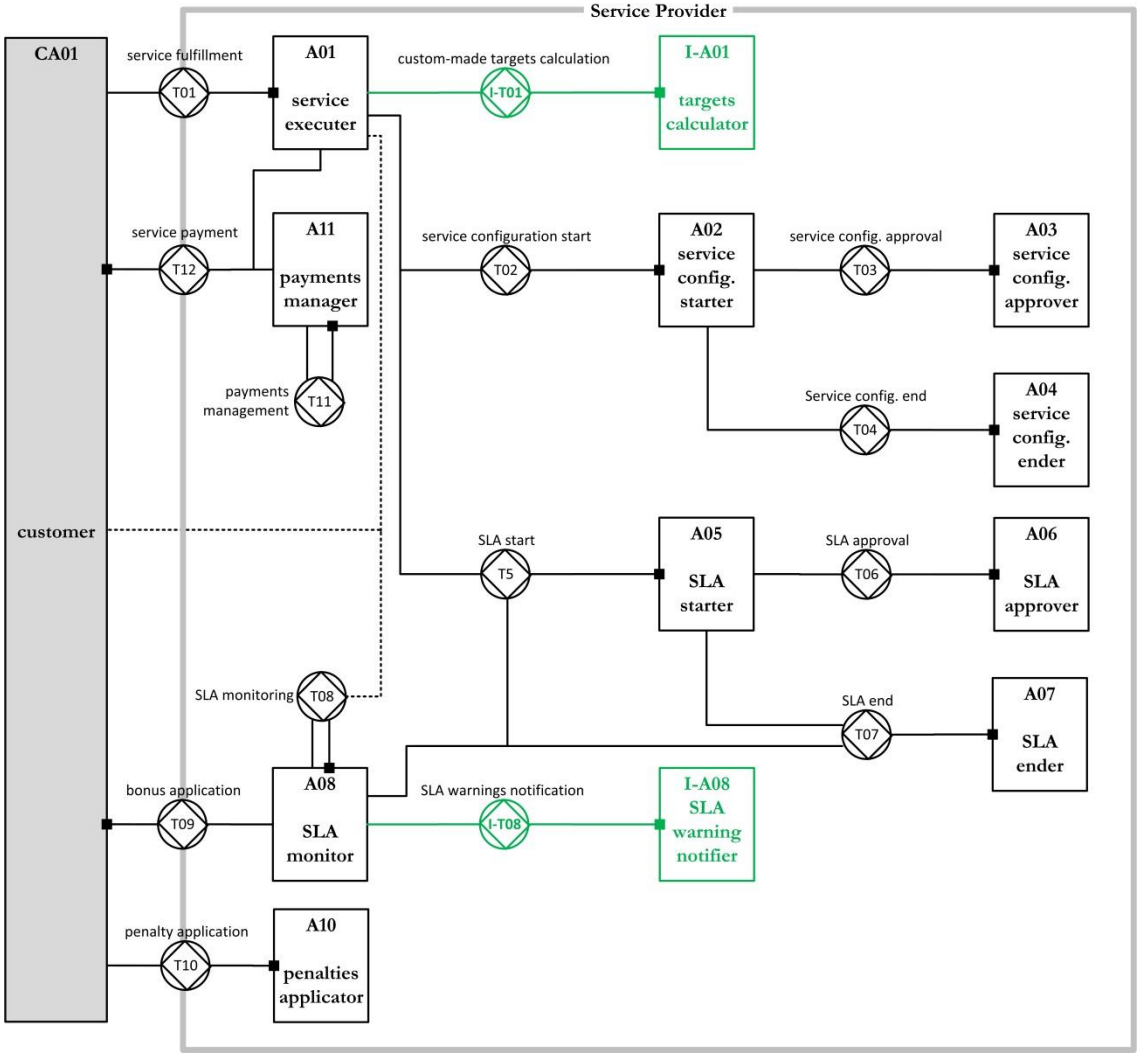


Figure 28 – Proposal Actor Transaction Diagram (ATD)

On the service provider side, we propose 10 ontological actor roles and two infological actor roles. The actor role A01, service executer, is accountable for executing transaction T1, service fulfilment, and for requesting four transactions: I-T01, T02, T05 and T12. The actor role I-A01 calculates the customized SLAs targets according to the customer (CA01), i.e. I-A01 executes

the transaction I-T01. The actor role A02 creates new service configurations (T02), requests the actor role A03 to approve new service configurations (T03) and, finally, may request the actor role A04 to end service configurations (T04). Regarding the SLA concept, the actor role A05, SLA starter, is liable for creating new SLAs, i.e. for executing transaction T5, and for requesting the transactions T06 and T07. The SLA approver actor role (A06) approves the creation of new SLAs and the actor A07, SLA ender, is responsible for ending the existing SLAs (transaction T7).

The actor role A08 is accountable for managing the SLA monitoring (T08). Periodically, A08 activates the transaction T08 and executes it. If applicable, A08 may request the execution of transactions I-T08 and T09. I-T08 is executed by I-A08 that sends warnings to the service executors (A01) accountable for service executions close to break SLAs. A08 may also initiate the creation of a new SLA (T05) and the end of an SLA (T07). T09 (bonus application) is executed by the customer (CA01) when the SLAs are fulfilled. As stated before, the customer (CA01) is also accountable for requesting the application of the penalties (T10 execution) in case of an SLA break. The execution of T10 is responsibility of the actor role penalties applicator (A10). Finally, the actor role A11 is accountable for managing the service executions payments. Periodically, A11 executes T11 and request the customers to pay for the services (T12).

Regarding the motivation example presented in Section 4.1, we have that the customer (Mr. Bean) fulfilled the actor role CA01 – Customer, since he was accountable for requesting the execution of the service fulfilment (in this case the delivery of a bouquet of roses). Then the florist fulfilled the actor roles A01 – Service Executor, A05 – SLA starter and A06 – SLA approver. In other words, the florist executed the requested service, created and approved a new SLA.

The ATD completes the Interaction Model of the Construction Model (CM). The other part of CM is the Interstriction Model that can only be modelled after the specification of the Process, Action and State models.

4.3.2 Process Model (PM)

The next model, the Process Model, relies on the Construction Model (CM) and is described by the Process Structure Diagram (PSD). Figure 29 illustrates the caption of elements used in the PSD.

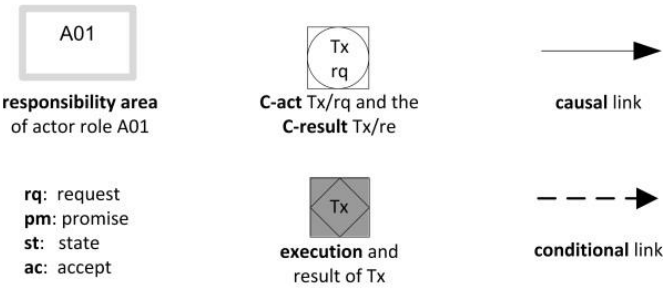


Figure 29 – Proposal Process Structure Diagram (PSD) caption

Figure 30 presents the PSD for the dynamic SLA negotiation proposal. From the left to the right, and top to the bottom, each one the following paragraphs details the PSD.

The process starts by the customer's (CA01) request (T01 rq) for a service execution. Then, the service executor (A01) may promise to execute that service (T01 pm). Therefore, A01 is responsible for deciding if the provider has capabilities to fulfil the service request according the customer's expectations. Afterwards, the service executor (A01) may need to request the creation of a new service configuration or a new SLA if needed. If not, the service executor may execute the service (T01) and inform the customer about the result (T01 st). Then, the customer (CA01) may accept the service execution (T01 ac).

In a more complex scenario, the service executor may need to request the creation of a new service configuration (T02 rq) and/or a new SLA (T05 rq). In order to create a new service configuration, the service configuration starter (A02) must promise (T02 pm), execute (T02) and state (T02 st) T2. Additionally, to execute T2 the service configuration starter (A02) must wait for the promise of the service configuration approval (T03 pm) and, if applicable, for the acceptance of the service configuration end (T04 ac).

Therefore, before A02 can start a new service configuration, A02 must request the approval of this service configuration (T03 rq). Then, the service configuration approver (A03) may promise to approve the creation of the new service configuration (T03 pm). Thus, A03 has the knowledge to understand if the provider can have available a specific service configuration. Afterwards, A03 must approve the new service configuration (T03) and state (T03 st). Then, the service configuration starter (A02) may accept the approval (T03 ac) and, if needed, he may request the end of a conflicting service configuration (T04 rq).

If T04 is requested, then the service configuration ender (A04) may promise (T04 pm), execute (T04) and state (T04 st) the service configuration end. Next, A02 may create the new service configuration (T02) and state (T02 st). The service executor (A01) may now accept the service configura-

tion start () and if the chosen SLAs by the customer exist, then A01 can execute the service () and state it ().

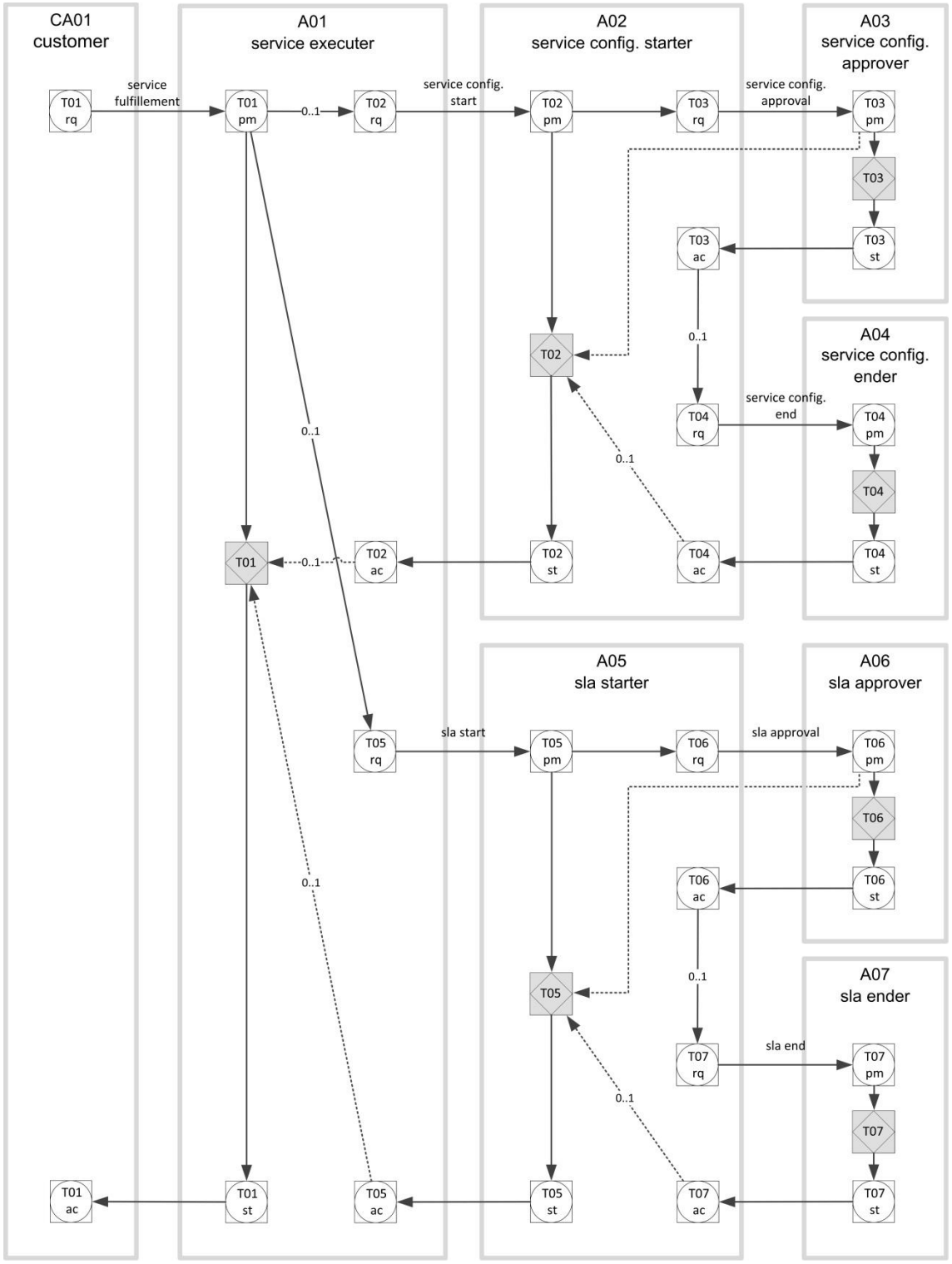


Figure 30 – Proposal Process Structure Diagram (PSD) of transactions T01 to T07

In parallel, if there are new SLAs in the service request, then these SLAs must be created. The process of creating new SLAs is similar to the creation of service configurations, since there is a transaction to create SLAs (T05) that depends always on another (SLA approval – T06) and may also involve the end of another SLA (T07).

The PSDs involving T08 to T12 are simpler and are illustrated in Appendix C – Proposal Process Structure Diagrams.

4.3.3 Action Model (AM)

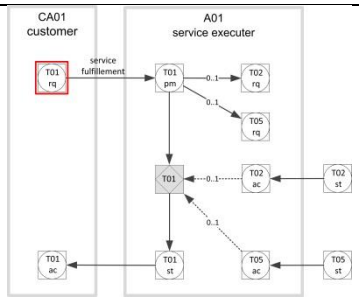
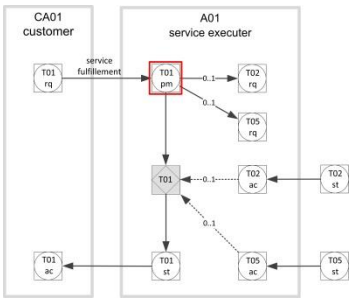
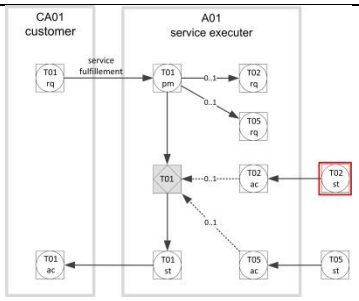
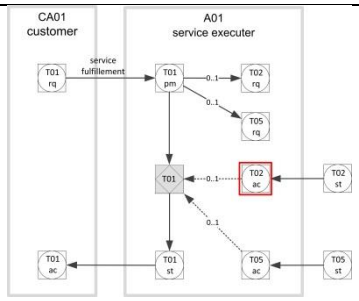
The Action Model (AM) is designed after the Process Model and delivers a fully ontological specification for each process step of the previous defined ontological transactions. The AM is expressed through action rules that we will specify using a pseudo-algorithmic language. An action rule is enclosed by a *when-then* clause. The *when* part specifies the agendum that is being dealt with and the *then* part specifies the actions that have to be taken for dealing with the agendum. Conditional responses (choices) are represented by an *if-then-else* clause. Repeated actions are specified by means of a *for-each* clause. Sometimes it is not possible to specify a rule (completely) formally. In such cases, informal expressions are allowed; they are always put between the angled brackets < and > (Dietz J. , 2006).

We present the action rules for the actor role service executor (A01) for illustration purposes. The remaining action rules are fully described in Appendix D – Proposal action rules. Table 13 illustrates the six action rules that the actor role A01 must follow when fulfilling his responsibilities. These six action rules are responses to: customer (CA01) request for a service fulfilment (T01 request), service executor (A01) promise to execute a service (T01 promise), state of a service configuration start (T02 state), accept of a service configuration start (T02 accept), state of SLA start (T05 state) and accept of an SLA start (T05 accept). Aside with each action rule, we present a sample of the Process Structure Diagram containing the coordination act that the action rule corresponds (highlighted with a red square).

The first action rule concerns the response to the customer (CA01) request of a service execution. Three inputs are needed: the service, the service configuration and the SLA. The service executor (A01) must decide if he can fulfil the service request. If he decides to deliver the service, then A01 must promise the fulfilment of the service, otherwise A01 must decline.

The second action rule concerns the response to the T01 (service fulfilment) promise. First, if the chosen service configuration does not exist then service executor (A01) must request its

creation. Next, the same must be done for the chosen SLA. Finally, if the both service configuration and SLA already exist then A01 may execute the service fulfilment and state it.

#	Process Structure Diagram	Action Rule
1		**when** fulfilment of [ServiceExecution] is requested **with** service of [ServiceExecution] is [Service] **with** sla of [ServiceExecution] is [SLA] **with** serviceConfig of [ServiceExecution] is [ServiceConfig] **if** <canFulfillRequest> **then** fulfilment of [ServiceExecution] must be promised **else** fulfilment of [ServiceExecution] must be declined
2		**when** fulfilment of [ServiceExecution] is promised **if** [Service Configuration] not exists **then** start of [Service Configuration] must be requested **if** [SLA] not exists **then** start of [SLA] must be requested **if** [Service Configuration] exists and [SLA] exists **then** fulfilment of [ServiceExecution] must be executed **and** fulfilment of [ServiceExecution] must be stated
3		**when** start of [Service Configuration] is stated **if** <[Service Configuration] is acceptable> **then** start of [Service Configuration] must be accepted **else** start of [Service Configuration] must be rejected
4		**when** start of [Service Configuration] is accepted **if** <[SLA] is not new> or start of [SLA] is accepted **then** fulfilment of [ServiceExecution] must be executed **and** fulfilment of [ServiceExecution] must be stated

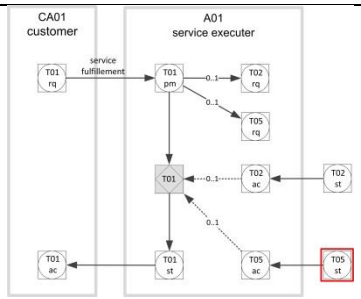
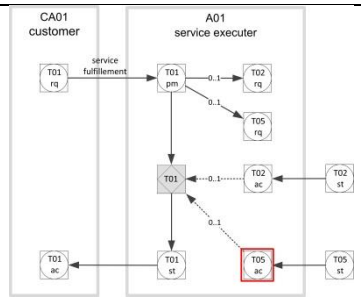
#	Process Structure Diagram	Action Rule
5		**when** start of [SLA] is <u>stated</u> **if** <start of [SLA] is acceptable> **then** start of [SLA] must be <u>accepted</u> **else** start of [SLA] must be <u>rejected</u>
6		**when** start of [SLA] is <u>accepted</u> **if** <[Service Configuration] is not new> or start of [Service Configuration] is <u>accepted</u> **then** fulfilment of [ServiceExecution] must be <u>executed</u> **and** fulfilment of [ServiceExecution] must be <u>stated</u>

Table 13 – Action rules for elementary actor role service executor (A01)

The third action rule concerns the response to the new service configuration state C-act. The service executor (A01) must decide if the new service configuration is acceptable. If A01 decides the service configuration is acceptable then we must accept it, otherwise he should reject the new service configuration.

The fourth action rule concerns the response to the new service configuration accept C-act. If the chosen SLA already exists or a new one was created, then A01 must execute the service and state it.

The fifth action rule concerns the response to the new SLA state C-act. The service executor (A01) must decide if the new SLA is acceptable. If A01 decides the SLA is acceptable then we must accept it, otherwise he should reject it.

The sixth action rule concerns the response to the new SLA accept C-act. If the chosen service configuration already exists or a new one was created, then A01 must execute the service and state it.

The remaining action rules are fully described in Appendix D – Proposal action rules.

4.3.4 State Model (SM)

The State Model is expressed in an Object Fact Diagram (OFD) and an Object Property List (OPL). Figure 32 illustrates the OFD of our proposal. The OFD is fully based on the language

WOSL (Dietz J. , 2005) and shows the categories (external at grey) and the relations between categories and the transactions' results. Figure 31 describes the caption of the OFD.

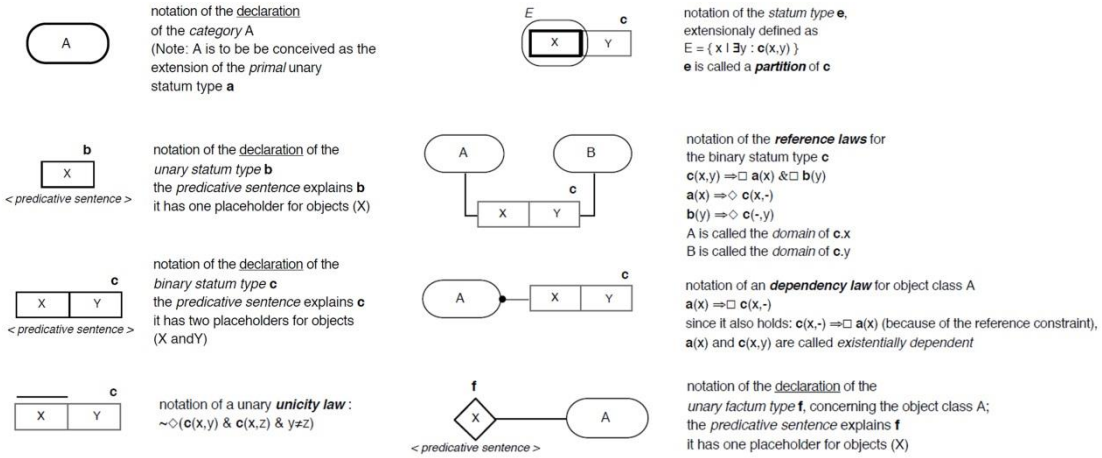


Figure 31 – Object Fact Diagram (OFD) caption (based on **(Dietz J. , 2005)**)

The Transaction Kind, Elementary Actor Role and P-Event Kind are external categories and the relations between those categories are based on the DEMO meta model defined in (Dietz J. , 2009). In this meta model each Transaction kind always has one executor, at least one initiator and just one Production Event (P-Event) Kind that represents the result of the Transaction. Each P-Event Kind can be associated with only one Transaction and an Elementary Actor Role can be the executor of only one Transaction Kind and the initiator of several.

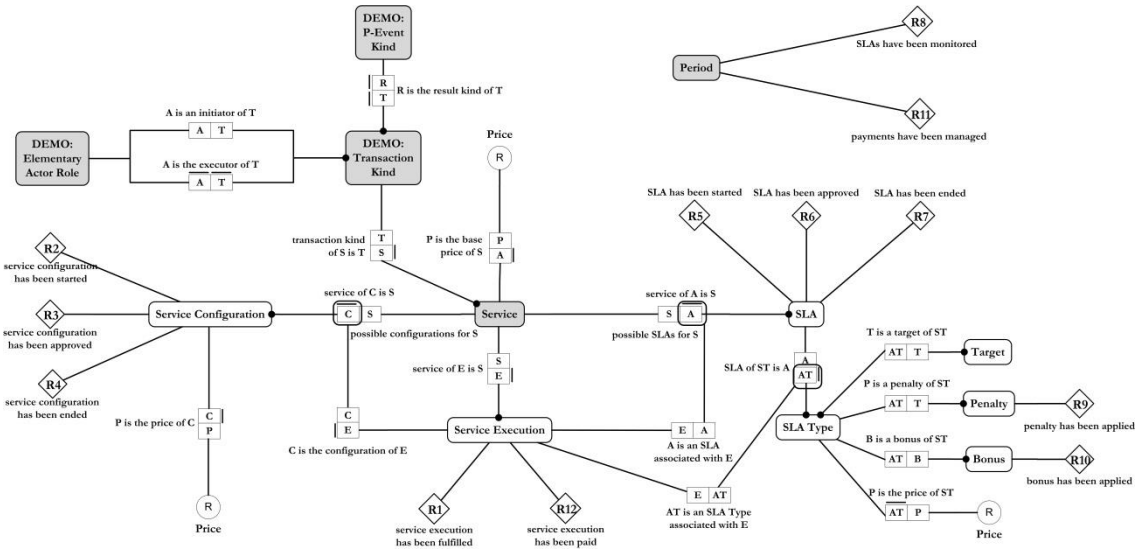


Figure 32 – Proposal Object Fact Diagram (OFD)

The Service category is also external since it is not created in the dynamic SLA negotiation. The Service category represents the services that are available to the market. A service can be associated with only one Transaction Kind and a Transaction Kind can have several services.

In Appendix E – Proposal Object Fact Diagram the same Figure is presented in a larger scale.

Regarding the internal categories, we propose the Service Configuration, the Service Execution, the SLA, the SLA type, the Target, the Penalty and the Bonus. The Service configuration category represents the different configurations that a Service can have, i.e. a Service can have several Service Configurations, but a Service Configuration can be associated with only one Service. The coexistence rules between the categories Service and SLA are similar to the previous since a Service may have several SLAs associated, but an SLA must be associated with only one Service. The Service Execution represents the fulfilment of services by the provider, so this category is associated with the Service category. A Service Execution must be associated with only one Service and a Service may have several Service Executions.

The Service Configurations that have a determined Service are extensively defined as a new category (possible configurations for S). This category represents the configuration options that a client has when requesting a certain Service, so the Service Execution category is associated to it with the following coexistence rules: a Service Execution may have one Service Configuration associated and a Service Configuration may be chosen in different Service Executions.

The relation between Service Executions and SLAs is similar since, the SLAs that have a determined Service are extensively defined as a new category (possible SLAs for S). This category represents the quality options that a client has when requesting a certain Service, so the Service Execution category is associated to it with the following coexistence rules: a Service Execution may have several SLAs associated and an SLA may be chosen in different Service Executions. Additionally, an SLA may have several SLA Types and an SLA Type must be associated with only one SLA. The Service Execution may also have several SLA Types associated through the connection ‘AT is an SLA type associated with E’. Therefore, the connections labelled ‘A is an SLA associated with E’ and ‘AT is an SLA type associated with E’ represents the service quality and, ultimately, the customer expectations regarding a particular Service Execution.

Finally, an SLA Type can be composed by combinations of Targets, Penalties and Bonuses, and just one Price.

The transaction events are associated to the categories as follows: results of transactions T1 (R1) and T12 (R12) are linked to Service Execution category, results of T2, T3 and T4 are related with the Service Configuration category, results of T5, T6 and T7 are related with the SLA category, results of T8 and T11 are related with the external category Period, result of T9 is

connected with the Penalty category, and result of T10 is linked to the Bonus category. The dependencies between results are not specified since these relations were already described in the Result Structure Chart (Figure 26).

Besides the OFD, the state model can also be modelled using the Object Property List (OPL). The OPL is just a convenient way of specifying fact types that are proper (mathematical) functions, and of which the range is a set of values. The OPL is illustrated bellow.

Property Type	**Object Class**	**Scale**
pre_conditions_price (*)	Service Execution	MONEY
service_execution_price (*)	Service Execution	MONEY

Table 14 – Proposal Object Property List

We propose two derived properties for the Service Execution category. These properties can be calculated by the following derivation rules:

pre_conditions_price(E) = *<sum of service execution prices that are preconditions of S >*

service_execution_price(E) = *pre_conditions_price(S)+*
service_base_price(S)+
service_configuration_price(C) +
<sum of SLAs prices associated with E>

The first derivation rule establishes that the *pre_conditions_price* is given by the sum of service executions prices that are preconditions to the given service execution. The second derivation rule establishes that the price of a service execution is the sum of four components: the *pre_conditions_price,* the base price of the respective Service, the service configuration price, and the sum of the prices of the chosen SLAs.

Regarding the florist example presented in Section 4.1 the delivering of the bouquet of roses to Mr. Bean represents an instance of the Service Execution category. In this example, an SLA Type was created and associated with the Service Execution. This SLA Type had a performance target, since the bouquet needed to be ready in 5 minutes. In other words, after the promise to deliver the bouquet (line 11), the state act (line 14) should occur in a 5 minute period. If this target was not fulfilled then a penalty would be applied (the customer could cancel his request and abort the transaction). Additionally, the price of this service execution (service_execution_price property defined in the OPL) was 30€. In this example there were no in-

stances of Service Configurations and Bonuses. A more detailed application of this proposal ontology is presented in the field study of Chapter 10.

With the information provided by the State Model it is possible to specify the Information Usage Table (IUT) and the Interstriction Model (ISM). IUT specifies, for every object class, fact type, and result type from the State Model, in which steps of the Process Model its instances are used. IUT is illustrated in Appendix F – Proposal Information Use Table (IUT).

The ISM constitutes the 'right hand side' of the Construction Model (see Figure 10) and specifies the passive influence bonds between actor roles. ISM is illustrated in Appendix G – Proposal Interstriction Model (ISM).

4.4 Summary

In this Chapter we presented our thesis proposal to specify the service quality. This proposal is based on a strong conceptual foundation (EO) that, by itself, is based on the speech act theory. We have chosen EO because this theory can help us to expand the expressiveness of the service descriptions and, consequently, allow a better alignment between expectations and perceptions.

Our proposal is composed by four steps and covers from the service identification phase to the service delivery. A major innovation of this thesis is the proposal of a dynamic SLA negotiation, so far only applicable to web services (see Section 3.3.5 – page 41). We believe that defining a number of SLAs at design time and restricting the customers' options to this number limits the clients' possibilities to convey their expectations. We suggest letting customers freely express what they are looking for in every interaction. This allows a dynamic SLA negotiation instead of only using the SLAs defined at design time (static SLAs).

5 Comparing Services Using DEMO in Wines4All

This and the next five Chapters explain how we proceeded in the demonstration and evaluation phase of DSRM. These chapters describe the six field studies that we used to validate and mature the proposal. We explain the proposal on the basis of real examples that increase the practical relevance of our study and obtain an in-depth insight into how our proposal can assist in the service quality specification process.

We describe each field study in five parts:

- **Problem** – The specific research question and hypothesis (Section 1.4 – page 7) that the field study addresses;
- **Proposal** – The steps of the proposal (Figure 23 – page 54) that were implemented;
- **Demonstration** – How the steps were implemented;
- **Evaluation** – How the results of the field study were evaluated. In all the field studies, we use the framework proposed in (Pries-Heje, et al., 2004) to describe our evaluation strategy. This framework allows to describe the evaluation in three dimensions: what was evaluated, how it was evaluated and when it was evaluated;
- **Lessons Learned** – The lessons learned from the field study.

This Chapter describes the first field study (Mendes, et al., 2011).

5.1 Problem

In this field study we focused on finding evidences on who should be included in the service identification process because there seems to exist a belief among the service providers that they can identify their own services without theirs clients' feedback. So, we are trying to validate **Hypothesis H1.2 (the presence of both customer and provider in the service identification process contributes to a more customer-oriented service catalogue).**

5.2 Proposal

In order to solve the problem of service identification, we propose to use the DEMO methodology to identify the services provided by organizations to their customers. Since ITSM best practices cannot help (on the one hand, they are too general and do not specify how to identify services, and, on the other hand, the already existing methods that allow this task are focused on technology) we decided to use EO and the definition of service based on this methodology. Moreover, EO and DEMO models focuses on what really matters, i.e. the business layer, not considering the implementation details that are secondary to the clients.

We propose to use the following steps to find the services of an organization: Enterprise Description, Performa-Informa-Forma Analysis, Coordination-Actors-Production Analysis, Transaction Pattern Synthesis, Result Structure Analysis, and Actor Transaction Diagram/Service Identification. These steps are based on the EO theory and will be explained in further detail in the next Section.

5.3 Demonstration

The demonstration was performed in a private company, leader in the wines and spiritual beverages distribution, which we will call from now on Wines4All.

Wines4All has about 80 employees, three of which from the Human Resources (HR) department. This field study aimed to provide evidence against the importance of well-identified services. For that, we decided to study this HR department and to identify the services it provided to the rest of the organization. First, we identified the services provided by the HR department using the provider perspective. The head of the HR department (Gabriela) was interviewed and described the actions she had to take to perform her job. With this description we produced the enterprise description of the HR department (first step of the proposal) and then we applied the following proposal steps in order to find the services from the provider perspective.

In addition, we also applied the service identification proposal using an enterprise description with a customer perspective. For that purpose, we interviewed an employee of the Marketing department (Rosario) in order to identify what interactions she had with the HR department, i.e. what services she perceives the HR department provided to her. So, the final objective was to verify if the services a provider offers differ from the perceived services received by a customer.

5.3.1 Provider Perspective

Firstly, we describe the actions undertaken by the head of the HR department. The starting point to fulfil this task is called Enterprise Description and is characterized by producing a text that summarizes the actions performed by the service provider, such as the presented below (**first step** of the proposal). This text was written by the researchers using Gabriela (head of HR department) feedback. For the sake of readability, we will just present the text after applying the first two analyses.

When the text is written, one should read it carefully and try to recognize and distinguish between the Ontological, Infological and Datalogical actions, as referred in the Enterprise Ontology Distinction axiom (see Section 2.2 – page 18). This step is called Performa-Informa-Forma Analysis and is the **second step** of the proposal. To do that, we should define a notation to differentiate those actions: in this example, we have highlighted the text, using red, green and blue colours to identify, respectively, the Ontological, Infological and Datalogical actions.

The next step (**third one**) concerns the identification of C-acts/facts, P-acts/facts and actor roles, using the Performa (Ontological) items identified in the previous step. We also have considered a notation to differentiate between them, similar to the one used in the Operation axiom: square brackets "[" and "]" to identify actor roles, brackets "(" and ")" to identify C-acts/facts and angled brackets "<" and ">" to identify P-acts/facts. This step is called Coordination-Actors-Production Analysis and here there is a reduction of the complexity, relatively to other methodologies, because from now on we will only consider the Ontological actions identified in this step.

The result of applying these two analyses to the text of HR point of view is presented below:

The Human Resources (HR) department of Wines4All is responsible for the management of the vehicles distribution, infrastructure management, training of the various employees, recruitment of new employees, and insurance, among others. It is constituted by three other employees: Vitor, Patricia and Luisa. Vitor deals essentially with fleet management, post office and banks, Patricia assists the Finance department and General Manager (she is a personal assistant of these two areas), and Luisa is a receptionist who deals with phone calls and Proof of Deliveries (PODs). They both report to Gabriela, head of HR.

*The recruitment process of Wines4All starts when both the **[**HR**]** and the General Manager agree that there is a (need) to <hire> [new employees]. This need can be obtained from previous feedback given by the responsible for each department. It is possible to use the support giv-*

en by the [universities] to <hire> a [new trainee] and give him/her the necessary <training>, or, through advertisements or [temporary employment companies], <hire> [someone already established on the market] and with experience. When there is a need to hire a new trainee, the HR department of Wines4All contacts some specific universities, according to the function and pre-requisites to perform that function. After this first contact, some universities proceed to the selection of resumes, while others send them all to Wines4All, which will select the most promising candidates. Next, these candidates are directly contacted by Wines4All to schedule the first interview with HR. If the candidate is accepted into the next stage of the recruitment process, a second interview meeting is scheduled with the head of the department where the function will take place. Finally, in case of satisfactory performance, the candidate is accepted. This recruitment process based on interviews is similar to all candidates, whether they are graduates or people with work experience.

*About the fleet management, the **[**HR**]** is responsible for the <rental> of certain types of vehicles, which will be used by employees with determined functions. This rental is made to an **[**external renting company**]**, and HR has to deal with distance control, accidents, and further expenses related to these vehicles (highway, fuel). They negotiate a contract that has a certain time limit, and includes various options such as maximum distance that can be travelled. It is also necessary to <make an insurance> for the vehicles. After receiving the vehicles, one must regularly check if they have mechanical problems and if they occur, the car must be <taken> to a workshop for repair. Exchanging vehicles between drivers to guarantee a balanced use of each must also be considered. When the contract is about to end, the vehicle is <taken> to inspection to check if everything is fine.*

The <insurance> is also related to HR. For instance, there are several types of insurance: life insurance, health insurance, vehicle insurance and others. In case of a trainee, the work accidents insurance is the only one to be triggered, if it is a temporary contract, it includes health and work accidents insurance, and if it is a permanent contract, it includes life, health and work accidents insurance. In both cases, the HR communicates with the insurer and Social Security to deliver tax and documents.

The payroll is determined using the budget that was established for that year. The monthly salary for each employee is calculated and pre-determined from the company politics, and it is affected by absences, product discounts, among other factors which will be uploaded to the software that calculates each month's salary. To determine every factor, the head of each department must inform the HR.

*The **[**HR**]** <establishes> telecommunications contracts with specified **[**operators**]** to ease the communication of the employees and checks if these contracts are fulfilled by the operating company which affects the monthly payment that has to be taken. HR is also responsible for the development of the employees' vacation map, control of the documented internal politics of the company and check if they are being carried out by employees, and occupational medicine to ensure that some employees meet certain requirements to perform some actions.*

*At the end of every year, the performance appraisal process is executed, during which the job performance of each [employee] is <evaluated>. The **[**HR**]** develops specific forms for each department, which must be filled in by the employees until a certain date (Self-evaluation). After this phase, the head of each department gathers the feedback given by their employees and then schedules meetings with all members to discuss the performance during the last year.*

*When the performance appraisal process has finished, it is time to check if **[**someone**]** (needs) a particular <training session> in a specific subject. When this need is identified, one tries to identify a group of employees that also needs the same training. After that, it is necessary to plan the training session. This way, the HR starts by checking the availability of the employees, possible dates to execute it, text books, proofs of participation and other logistics steps. In case of internal training, the HR verifies who, inside the company, has the know-how to <develop> that training session, and then, the chosen employee will be held responsible for developing the module. In case of external sessions, the HR also needs to contact an accredited external company to execute the training session.*

*The infrastructure management (office equipment, chairs, tables) and overall function of the headquarters is conducted by the **[**HR**]**. When employees (need) office equipment, they ask Vitor for it. Then, he will <order> this equipment in the office store. When the equipment arrives, the store sends it to Wines4All and it is delivered to the employees. The **[**HR**]** also <deals> with the logistics of the company's events (Christmas dinner, company day and answers the employees' general doubts.*

After these analyses, we identified the transactions by clustering the identified C-acts/facts and P-acts/facts in what is denominated the Transaction Pattern Synthesis (**fourth step**). The Transaction axiom can be helpful in this step, because it guarantees that each P-act/fact or C-act/fact previously found corresponds to a complete transaction. Then, for each identified transaction type, the result type (i.e., the Production fact created) should be correctly and precisely formulated. The result is presented in the Table 15 below, called Transaction Result Table.

Transaction Types	Result Types
T01 – Hire a new employee	R01 – Employee E has been hired
T02 – Rent a vehicle	R02 – Vehicle V has been rented
T03 – Repair vehicle	R03 – Vehicle V has been repaired
T04 – Inspect vehicle	R04 – Vehicle V has been inspected
T05 – Insure a vehicle	R05 – Vehicle V has been insured
T06 – Insure an employee	R06 – Employee E has been insured
T07 – Establish communication contracts	R07 – Communication contract T has been established
T08 – Evaluate job performance	R08 – Job performance J has been evaluated
T09 – Give training session	R09 – Training session S has been given
T10 – Fulfil equipment requests	R10 – Request R has been fulfilled
T11 – Order and receive office equipment	R11 – Office equipment O has been received
T12 – Organize company events	R12 – Event E has been organized

Table 15 – Transaction Result Table (TRT) – HR point-of-view

After defining the transaction types and the respective result types, we checked if there are any dependencies between the transactions results, as the Composition axiom describes. This step is called Result Structure Analysis and can be executed by carefully reading the text one more time (**fifth step**). The following dependencies were found:

- There is a dependency between T01 and T06. After hiring a new employee, the HR department has to insure him/her. T06 is mandatory;
- There is a dependency between T02 and T05. When a vehicle is rented to an external renting company, the HR department is responsible for insuring that same vehicle. T05 is mandatory;
- T10 depends on T11. In order to fulfil the employee's general requests, the HR department contacts the Office Store to order the necessary equipment. T11 is optional.

After identifying the transaction types, its results and dependencies, we determined the environment surrounding the HR department, considering the organization context. The first model to be developed is called the Actor Transaction Diagram (ATD), represented in Figure 33 (**sixth step**).

In this type of diagrams, a transaction is represented using a diamond in a disk that contains the respective combination of C-acts and a P-act. Each transaction is connected to two boxes, representing the initiator and executor actor roles. The initiator is connected to the transaction symbol using a solid line, while the executor is connected to the transaction using a solid line ending in a black square.

The gray boxes refer to composite actor roles, i.e. elements whose exact structure is not known. All the environmental elements, i.e. elements outside the organization that we are studying, are represented with gray boxes for that reason. This also means that we can represent the studied organization with a gray box when referring to the kernel of the organization, which can be further specified using elementary actor roles represented by white boxes. In this example, for simplicity reasons, the HR department has been considered a composite actor role. The gray-lined rectangle "Service Provider" represents the boundary under consideration.

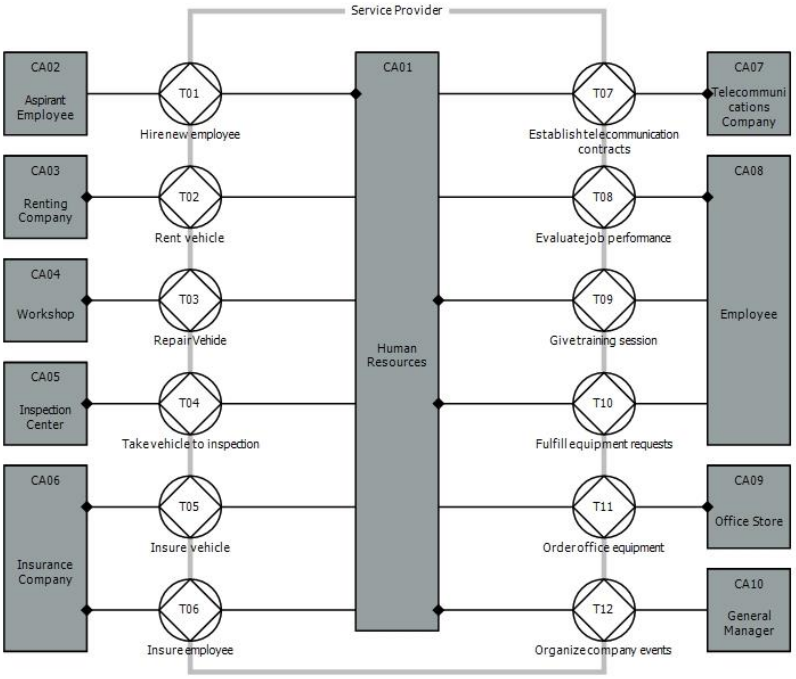


Figure 33 – Actor Transaction Diagram – HR point-of-view

As depicted in Figure 33, and following the service definition based on in the Enterprise Ontology (Terlouw & Albani, 2013), we can identify four services provided by the Human Resources department according to Gabriela: hire a new employee (T01), give a training session (T09) to Wines4All's employees, fulfil requests of the employees related to general office equipment (T10) and, finally, organize Wines4All's events for employees (T12).

5.3.2 Customer Perspective

So far, we have described all the steps to find the services from the HR's perspective (the provider perspective). We will now describe these steps more briefly in order to find the services provided by the HR department, but from the Marketing department's (customer) point-of-view. After collecting the textual description from interviews and applying the first two analyses (Per-

forma-Informa-Forma and Coordination-Actors-Production) we have obtained the following text:

According to Rosario, when she was hired to work at Wines4All, there wasn't any Human Resources (HR) department yet. She was <hired> with the assistance of an external company. She had a first interview with [Vera Ribeiro] (Marketing dept.) and an agent of the external company, followed by a second meeting to discuss the working conditions and sign documents. The first week on the job was called Integration Week, when the new employees can understand what is the done by the various Wines4All departments.

Nowadays, regarding the <evaluation process>, [Rosario] receives emails from the HR containing the forms that must be used in the self-evaluation phase, as well as the internal procedures and politics of the company. About the training sessions, all the logistics <is arranged> by the [HR], as well as the management of the number of training hours. The head of the department is responsible for informing the HR that a specific training might be needed.

Every month, she receives the phone bill. Each employee has a limit that he/she can spend monthly, and if the limit is overreached, then the [employee] has to <pay> the difference.

To <schedule> her vacation, [Rosario] needs to send her proposal to Vera Ribeiro, who analyses and approves it or not. Then, Vera Ribeiro sends the proposal to Vera Martins (Marketing dept.) to guarantee a consensus between the whole departments. From here on, the HR is informed.

When she wanted to change her NIB code, she had to call HR to let them know about the new one and complete the necessary documents. The same thing happened when [she] (wanted) to <include> another beneficiary in her insurance.

In order to correctly calculate her salary, in case of absences, she has to deliver some documents to the HR, and then this department proceeds to the correct discounts. In case of expenses the same procedure must be followed. When these aspects are known, the software automatically calculates the output.

Problems with office equipment <are dealt> by [Vitor]. When she wants to send a letter, she provides it to Vitor, who is in charge of delivering it at the postal offices. Letters received at Wines4All are delivered to the respective recipients by Patricia.

After identifying the Ontological activities and the involved actor roles, we identified each transaction and respective result on the Transaction Pattern Synthesis (see Table 16).

Transaction Types	Result Types
T01 – Hire a new employee	R01 – Employee E has been hired
T02 – Evaluate job performance	R02 – Job performance J has been evaluated
T03 – Give training session	R03 – Training session S has been given
T04 – Pay telecommunications invoice	R04 – Invoice I has been paid
T05 – Schedule vacations	R05 – Vacation schedule S has been developed
T06 – Change insurance status	R06 – Employee E has changed its insurance status I
T07 – Fulfill equipment requests	R07 – Request R has been fulfilled

Table 16 – Transaction Result Table (TRT) – Marketing point-of-view

From the Marketing's point-of-view, there are no dependencies among the identified transactions. The explanation for this difference is due to the fact that Rosario works at the Marketing department, so she does not know how the HR department actually executes those transactions. This step corresponds to the Result Structure Analysis.

The ATD from the point-of-view of the Marketing department is represented in Figure 34.

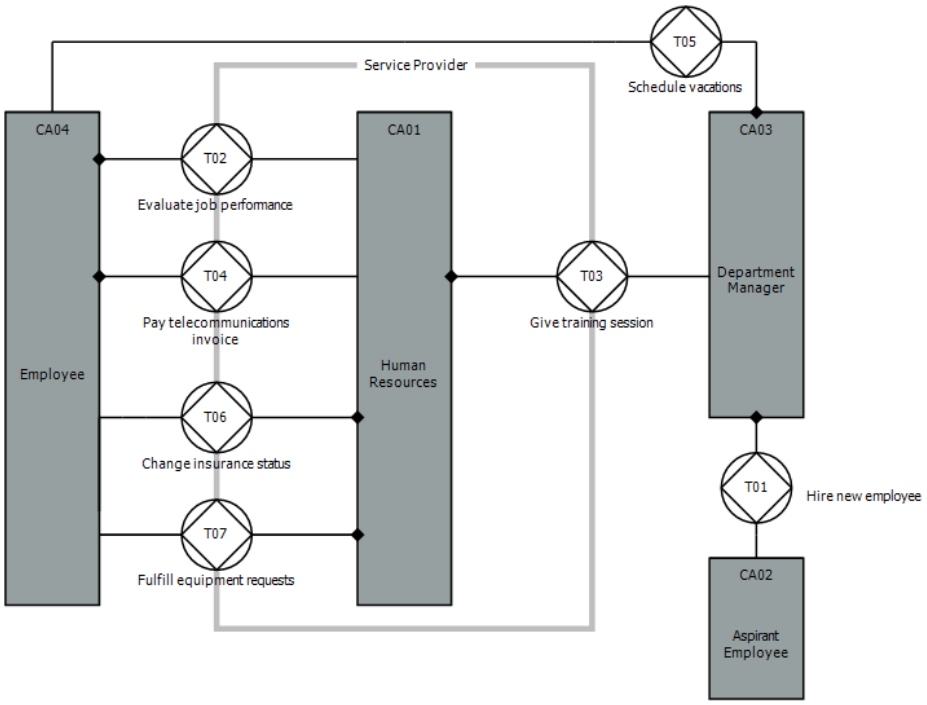


Figure 34 – Actor Transaction Diagram – Marketing point-of-view

We can identify three services provided by the HR department according to Rosario: give a training session (T03), change the status of the insurance (T06) and, finally, fulfil requests of the employees regarding office equipment (T07).

After analysing both perspectives, we conclude that there are similarities and differences between them. On the one hand, we found two services which are similar, “Give training session” and “Fulfil equipment requests”. On the other hand, there were also some mismatches: while in the HR’s point-of-view we identified “Hire new employee” and “Organize company events”, from the Marketing’s point-of-view we found “Change insurance status”. These differences will be discussed in the following sections.

5.4 Evaluation

Our evaluation strategy can be described using the framework defined in (Pries-Heje, et al., 2004):

- **What is actually evaluated?** The artifact evaluated is the proposed service identification process (a design process) and the resulting service list (a design product);
- **How it is evaluated?** We used the Moody & Shanks framework (Moody & Shanks, 2003) and the Wines4All employees’ feedback in a naturalistic evaluation. Thus the evaluation was conducted in a real organization facing real problems. The Moody & Shanks framework was used to evaluate the quality of the produced models;
- **When was it evaluated?** It was evaluated ex post (after the design artifact was developed).

After applying our proposal to identify the services provided by the HR department, both from a provider’s point-of-view and from a customer point-of-view, we evaluated the obtained results using the Moody & Shanks Framework (Moody & Shanks, 2003) to assess the quality of the Actor Transaction Diagrams and collect feedbacks about the results.

In order to evaluate the quality of the produced models, we used the Moody & Shanks framework that is composed by a set of eight quality factors:

- **Completeness**: refers to whether the model contains all user requirements. This factor is influenced by the person who describes the organization or department that is being studied. In the HR perspective, if the person who described the department has forgotten to mention some details it would not be possible to identify all the transactions / services. From the Marketing perspective, Rosario cannot identify all the services provided by the HR department, because the services that Rosario perceives depend on what are the main interactions between the HR and Marketing. For instance, Rosario did not have any idea about the organization of events (which was identified as a service by the HR department), because she had never been involved in this kind of transaction;

- **Integrity**: indicates the extent to which all business rules have been included in the model. The ATD does not express any business rules, but it was possible to gather the dependencies between transactions;
- **Flexibility**: describes the ease with which the model can cope with business and/or regulatory change. ATDs are stable against changes in the composition of members of the environment that interact with the HR department, since these structural changes do not reflect themselves on the ontological level. For example, actor roles such as Insurance company and Workshop, among others, are general enough. From the HR perspective, there are three dependencies between transactions that can restrict this flexibility. However, from the Marketing perspective there are no dependencies at all;
- **Understandability**: refers to the ease with which the concepts and structures can be understood. The produced models were shown to Gabriela and Rosario and the initial reaction revealed some confusion. In fact, one may find it difficult to understand the ATD, if not familiar with the notation; however, the reduced number of element types that composes each model contributes to a fast learning;
- **Correctness**: describes the extent to which the model conforms to the rules and conventions of the modelling technique. Since the model uses a diminished number of elements and the basic transaction pattern, it is possible to verify the accuracy of the ATD model;
- **Simplicity**: refers to the extent to which the model has a minimum number of constructs. Since DEMO models are focused on the Ontological level, there is a significant reduction on the number of elements when we consider the original source, in this case the textual description, which has several Datalogical and Infological aspects. For instance, the ATD of the HR perspective has 12 transactions and 10 actor roles, while the ATD of the Marketing perspective has seven transactions and four actor roles;
- **Integration**: regarding this particular field study, integration is concerned with the extent to which the two different models can be integrated or compared to each other. This quality factor depends on the textual description provided by both parties. Despite the differences which were found and explained, there were also some common points (two services / transactions and several actor roles);
- **Implementability**: denotes the ease with which the model can be implemented within the time, budget and technology constraints. In DEMO models, we are dealing with the Performa (Ontological) elements performed by actor roles, which are implementation independent. DEMO models are not ideal to represent the implementation details.

The evaluation of the obtained models was also performed by interviewing the HR and Marketing employees. In general, the feedback was positive because they understood the identified services. Nevertheless, a general complaint was that our list of services did not include all the functions the employees perform. This is due to DEMO models being implementation-independent, not considering the Infological and Datalogical activities. If one revises the textual descriptions of the HR, from both points-of-view, it is possible to conclude that the majority of the actions are included on the Forma and Informa categories (Datalogical or Infological, respectively).

5.5 Lessons Learned

After applying the proposal, we understood how DEMO models can reduce the complexity of enterprise models: by layering the enterprise in three parts, and focusing only on the part that refers directly to the creation of new original facts (Ontological layer). Despite these advantages, there are also some downsides. In order to apply the DEMO methodology, there is a need for a textual description about the organization that is being studied and the surrounding environment, as input for the identification process. This text is written in natural language, by someone who has some insight about the tasks performed by the organization, something that can lead to misunderstandings due to the lack of expressivity.

We also have to consider that the models produced by the DEMO methodology just contemplate the Ontological aspects performed by employees. This way, actions that are categorized as Infological or Datalogical are not directly included in these models. So, in order to have a complete overview of the actions performed by an actor, we have to consider those three kinds of abilities. This field study revealed that, in some situations, it would be useful to model the most relevant infological and datalogical transactions.

An important lesson is that different people have different notions of what is performed by one another, even when they have a customer-provider relation. This was apparent from the different textual descriptions and, consequently, the different services we obtained from the HR and Marketing's perspectives. This difference represents the gap between the customers' expectations and the service provider's perception of those expectations. This gap was first identified two decades ago (Parasuraman, et al., 1985) and the fact that, nowadays, it still occurs in companies shows the importance of the service identification process.

We also conclude that **both customer and service provider should be included in the service identification process and not only the service provider**, **validating Hypothesis H1.2**

(the presence of both customer and provider in the service identification process contributes to a more customer-oriented service catalogue). In fact, there seems to exist a belief among the service providers that they can alone identify their own services and that their customers should have a minor role in this process. With the results of this field study we demonstrated that customers and providers have different perspectives. The services are the connection point between providers and customers and having services poorly defined is the first step to not satisfy customers' expectations regarding services.

6 Using DEMO to Identify Services in an IT Service Provider

This field study focused on the service exchange between the European private company from the previous field study (Wines4All) and an external IT service provider. These services were already identified by the IT service provider and the results were: Development of New Features, Change Existing Features, and Management (Mendes & Mira da Silva, 2010) (Mendes & Mira da Silva, 2011).

6.1 Problem

Having the Hypothesis H1.2 (the presence of both customer and provider in the service identification process contributes to a more customer oriented service catalogue) validated, in this field study we focused on validating **Hypothesis H1.1 (the way of work defined in DEMO can be used as a basis to identify customer-oriented services).**

The objective of the field study was to identify which services the IT service provider does supply the Wines4All with and to verify if the services that our proposal identifies are preferred and understood by customers, when comparing these to the already identified by the IT provider.

6.2 Proposal

The proposal is similar to the one from Chapter 5 but the enterprise description (result from step 1) should be gathered from clients and service provider feedback. This is motivated by our

main conclusion of the previous field study: both customer and service provider should be included in the service identification process.

The proposal steps will be used to compare if the services identified by the way of work defined in DEMO are more significant than the ones already identified by the IT provider, i.e. if the customers prefer the new services list to the old one. According to DSRM, the set of steps in the proposal is the artifact that we will use to identify services.

6.3 Demonstration

Once more, the starting point of the proposal is the Enterprise Description that summarizes the actions performed by the service provider. In order to produce the enterprise description, we interviewed the IT service provider developer and the three users from Wines4All with more service requests at the time. We merged the interviews into a single text and then validated the result with the developer and the three interviewed users. For the sake of readability, we just present the text after applying the first two analyses.

Afterwards, we read the text carefully and tried to recognize and distinguish between the Ontological, Infological and Datalogical actions described (Performa-Informa-Forma Analysis). To do that, we defined a notation to differentiate those actions: in this example, we have highlighted the text using red, green and blue colours to identify the Ontological, Infological and Datalogical actions, respectively.

The next step concerned the identification of C-acts/facts, P-acts/facts and actor roles, using the Performa (Ontological) items identified in the previous step. We also considered a notation to differentiate between them: square brackets "[" and "]" to identify actor roles; brackets "(" and ")" to identify C-acts/facts; and angled brackets "<" and ">" to identify P-acts/facts (Coordination-Actors-Production Analysis). The result of applying these two analyses to the original text is presented below:

The services provided by the service provider to Wines4All are fulfilled by a developer that goes one day per week to the Wines4All's facilities, but who can also provide services remotely. There are no restrictions on who makes requests, because [everybody] in the organization can (request) a service. The requester can do it directly with the developer or he can submit the request in the Service Manager tool (used to record and manage all the requests). In both cases the requester explains the intent, specifies the application, functionalities involved, the name

and the priority of the request. If the requester approaches the developer directly, then the last one must register the request in the Service Manager.

Afterwards, the [developer] analyses the request and <implements it>. If necessary he requests more information about the request. The request may involve the integration with other applications out of the service provider's scope. If so, the [developer] must <prepare> an integration plan with the representative of the company accountable for the application to integrate. Normally the integration is done by means of web services.

When the implementation is over, the application involved is republished in the development server (if the request is about a bug with critical impact in the applications then the application is directly republished in the production server). From then on a [beta tester] takes the lead and <tests> the new functionality to see if it is ok. When the functionality is mature enough the application is published by the developer in the production server. If the request can be fulfilled by just changing a configuration at run time, the publishing step is not taken.

The Wines4All's [employees] may (request) training about new or existing functionalities, which is given by the same developer. The Wines4All's [employees] may also <schedule> meetings with the developer to debate several questions, such as solved bugs, current problems or future work.

After these analyses, we identified the transactions in the text by clustering the C-acts/facts and P-acts/facts in what is referred to as Transaction Pattern Synthesis. The Transaction axiom can be helpful in this step because it guarantees that each previously found P-act/fact or C-act/fact corresponds to a complete transaction. Then, for each transaction type, the result type (i.e., the P-fact created) should be formulated. The result is represented in Table 17.

Transaction Types	**Result Types**
T01 - Development	R01 – Feature F has been developed
T02 - Outside Application Integration	R02 – Integration I implemented
T03 – Testing	R03 – Feature F has been tested
T04 – Training	R04 – Training T has been given
T05 – Meeting	R05 – Meeting M occurred

Table 17 – Transaction Result Table (TRT)

After defining the transaction types and the respective result types, we checked if there are any dependencies between the transactions as described in the Composition axiom (Result Structure Analysis step):

- There is a dependency between R01 and R02: before the development of the feature, one checks if integrations with outside applications, such as web applications, are necessary. T02 is optional;
- There is also a dependency between R01 and R03: after the development of the feature and before the customer acceptance, the feature must be tested by a beta tester. T03 is mandatory;
- There are no dependencies involving T04 and T05.

After identifying the Ontological transaction types, its results and dependencies, we determined the environment surrounding the IT service provider. The first model is called the Actor Transaction Diagram (ATD), represented in Figure 35.

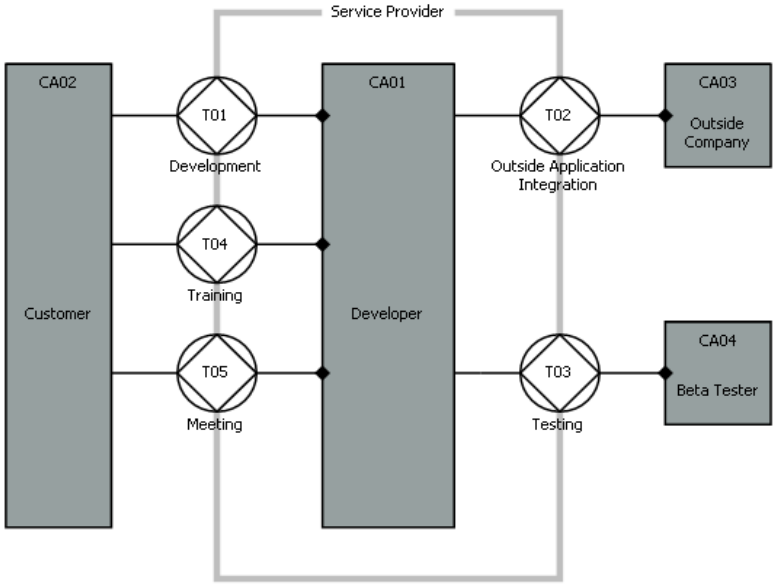


Figure 35 – Actor Transaction Diagram (ATD)

We can now identify the services by using the already stated definition of service, based on the Transaction Pattern of Enterprise Ontology (Terlouw & Albani, 2013). We conclude that the services the provider delivers correspond to the transactions in which the developer actor role is the executor. This only happens with three transactions: T01 – Development, T04 – Training and T05 – Meeting. Consequently, we can identify three services.

6.4 Evaluation

The evaluation strategy used in this field study can be described using the framework defined in (Pries-Heje, et al., 2004):

- **What is actually evaluated?** The artifact evaluated is the proposed service identification process (a design process) and the resulting service list (a design product);
- **How it is evaluated?** We used a questionnaire and the Moody & Shanks framework (Moody & Shanks, 2003) in a naturalistic evaluation. Thus, the evaluation was conducted in a real organization facing real problems. The questionnaire helped collecting the customers' feedback about the new services and the Moody & Shanks framework was used to evaluate the quality of the produced models;
- **When was it evaluated?** It was evaluated ex post (after the design artifact was developed).

The questionnaire was answered by 17 arbitrarily chosen employees of Wines4All, representing about 22% of the employees. The questionnaire was composed by a single question: "From 1 to 5, classify the two lists of services. With 5 the list represents in perfection the services provided by the provider and with 1 the services list is completely misaligned with reality."

The new list of services was composed by the services that were found using the DEMO methodology (Development, Training and Meeting), while the old services list was composed by Development of New Features, Change Existing Features and Management.

Table 18 illustrates the results of the questionnaire.

	1	2	3	4	5	Avg.	S.D.
Old Services List	0	2	7	5	3	3.53	0.9
New Services List	0	0	1	8	8	4.41	0.6

Table 18 – Questionnaire answers

From the 17 employees, 12 classified the new services list as more appropriate, 4 employees classified the new and old services equally and only one evaluated the old services as being a better representative of the reality. The standard deviation (column S.D.) is higher for the old list (0.9 against 0.6) showing that employees used a wider range of values in the evaluation of the old services list.

In order to evaluate the quality of the produced models, we used again the Moody & Shanks framework (Moody & Shanks, 2003). The results were similar to the previous field study, the

Actor Transaction Diagram presented high quality regarding Flexibility, Simplicity, Completeness, Integrity, and Correctness, and minor quality concerning Implementability and Understandability.

6.5 Lessons Learned

The new services list was different from the old services list. This is interesting because the old services list was identified by the IT service provider itself, so how could they not be the best possible representation of the service provider's offer? Indeed, the questionnaire results showed that the new services list was more accepted by the users than the old services list.

The old services list had two services (Development of New Features and Change Existing Features) that in the new list were represented by only one (Development). In fact, these two services represented two activities that the service provider executes in order to provide the Development service. They are technical concepts that only concern the service provider and that the customer does not (need to) know. Moreover, they were merged into a single service because they have the same result: Feature F has been developed.

The evaluation with the Moody & Shanks framework showed that the produced models were of sufficient high quality. In two of the eight quality factors (Flexibility and Simplicity) the DEMO models presented high quality and in three others (Completeness, Integrity and Correctness) the models showed high level despite the dependency on the quality of the enterprise description.

Despite the strong points of DEMO models, the evaluation also identified two limitations: Implementability and Understandability of DEMO models. Regarding the Implementability limitation, this is implicit to DEMO models because the theory in which DEMO is based on was designed to be implementation independent. However, this limitation did not compromise the quality of the new services list. Due to the specific notation, DEMO models may also be hard to understand by those who are not familiar with the notation. Nevertheless, the fact that these models are composed by few element types diminishes the learning period.

This field study contributed for the research question Q1 (how to specify services according to customers' expectations?). We used the way of work defined in DEMO to identify services and the new list of services was more accepted than the older. This fact **contributes** to the **validation of Hypothesis H1.1** (the way of work defined in DEMO can be used as a basis to identify customer-oriented services).

7 Specifying SLAs in a Bank

This field study took place in the Information Technology Division (ITD) of a private bank. In Portugal, this bank has about 2 400 000 clients and 872 branches. ITD has 482 employees and provides services to about 10 500 users (Mendes & Mira da Silva, 2012).

7.1 Problem

This field study focus on the research question Q2 (how to specify service level agreements according to customers' expectations?), namely we concentrated on validating **Hypothesis H2.2 (the Enterprise Ontology patterns of transactions represent a structure to define non-functional attributes of SLAs)**.

7.2 Proposal

In this field study we applied the first three steps of our proposal, i.e.:

1. Identify the services:
   1.1. Enterprise Description;
   1.2. Performa-Informa-Forma Analysis;
   1.3. Coordination-Actors-Production Analysis;
   1.4. Transaction Pattern Synthesis;
   1.5. Result Structure Analysis;
   1.6. Actor Transaction Diagram/Service Identification.
2. Specify the executor, production and coordination of the services using Generic Service Specification Framework (GSSF);
3. Specify the SLAs for each identified transaction/service. We applied the first version of the SLAs described in Figure 21.

We did not applied the fourth step of the proposal (fulfil the service requests according to the agreed SLAs) since the first three are sufficient to validate Hypothesis 2.2.

7.3 Demonstration

The **first step** was to identify the services ITD provides their customers with. Therefore, we applied the service identification process composed by the six steps: Enterprise Description, Performa-Informa-Forma Analysis, Coordination-Actors-Production Analysis, Transaction Pattern Synthesis, Result Structure Analysis, and Actor Transaction Diagram/Service Identification.

In order to write the enterprise description, we interviewed six employees from the ITD and three key users. During the interviews, the participants were asked to describe the activities performed by the ITD. The interviews were recorded and transcribed, as well as checked and discussed by two interviewers, ensuring unbiased findings and avoiding misinterpretation (Kvale, 2007).

The next step was to perform the Perfoma-Informa-Forma Analysis and the Coordination-Actors-Production Analysis, both described in the EO book (Dietz J. , 2006) and used in the two previous field studies. After these analyses, we identified the transactions by clustering the identified C-acts/facts and P-acts/facts (Transaction Pattern Synthesis). Then, for each identified transaction type, the result type was identified. The result is presented in Table 19.

Transaction Types	**Result types**
T01 – Internal Policies Production	R01 – Internal Policies P have been produced
T02 – Employee Training	R02 – Employee E has been trained for Internal Policies P
T03 – Improvement Implementation	R03 – Improvement I has been implemented
T04 – Feature Development	R04 – Feature F has been developed
T05 – Implementation Plan Change	R05 – Implementation Plan IP has been changed
T06 – Production Environment Change	R06 – Production Environment PE has been changed
T07 – Audit	R07 – Audit A has been done
T08 – Implementation Plan Production	R08 – Implementation Plan IP has been produced
T09 – Equipment Access	R09 – Equipment Access EA has been provided
T10 – Voice & Data Communication Installation	R10 – Communication Network N has been installed
T11 – Security Access	R11 – Secure Access SA has been granted
T12 – File Storage	R12 – File Storage FS has been provided
T13 – Specialized Software Access	R13 – Specialized Software Access SSA has been provided
T14 – Failure Support	R14 – Failure FA has been solved
T15 – General Employee Satisfaction Evaluation	R15 – Satisfaction Of Employee E and Semester S relating ITD function has been evaluated
T16 – Employee Satisfaction Evaluation	R16 – Employee Satisfaction Evaluation of Failure FA has been done
T17 – Mandatory Feature Certification	R17 – Feature F has been certified as mandatory
T18 – Risk Feature Judgment	R18 – Feature F has been judged as risky
T19 – Business Case Benefits Decision	R19 – Decision about the benefits of the business case B has been taken

Table 19 – Transaction Result Table (TRT) of the ITD

After identifying the transactions and the actor roles involved, we checked if there are any dependencies between the transactions as described in the Composition axiom (Result Structure Analysis step) and we developed the Actor Transaction Diagram (ATD) presented in Figure 36.

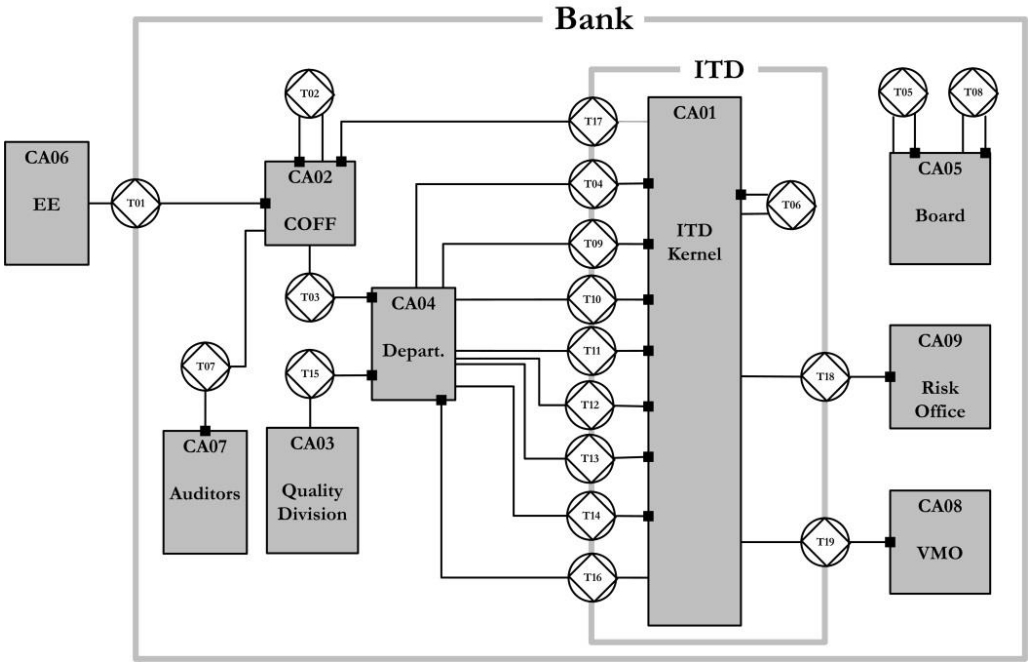


Figure 36 – ATD of the bank

We can now identify the services provided by ITD by using the already stated definition of service (page 29). We conclude that the services the ITD delivers correspond to the transactions in which the ITD Kernel actor role (CA01) is the executor. This happens with eight transactions: T04 - feature development, T06 - production environment change, T09 - equipment access, T10 - voice & data communication installation, T11 - security access, T12 - file storage, T13 - specialized software access, and T14 - failure support.

In order to proceed to the **second step** of the proposal (specify the executor, production and coordination of the services), first we had to model the Process Model, the Action Model, and the State Model of ITD, since some aspects of the GSSF (used in the second step of the proposal) depend on these models.

The Process Model (PM) is expressed in a Process Structure Diagram (PSD) and an Information Use Table (IUT). The PSD specifies, for each identified transaction, the process steps that are allowed to be taken. From now on, we will focus on the transaction T14 (failure support) because T14 represents the service with more levels of quality. Figure 37 represents the PSD in which the transaction T14 is involved.

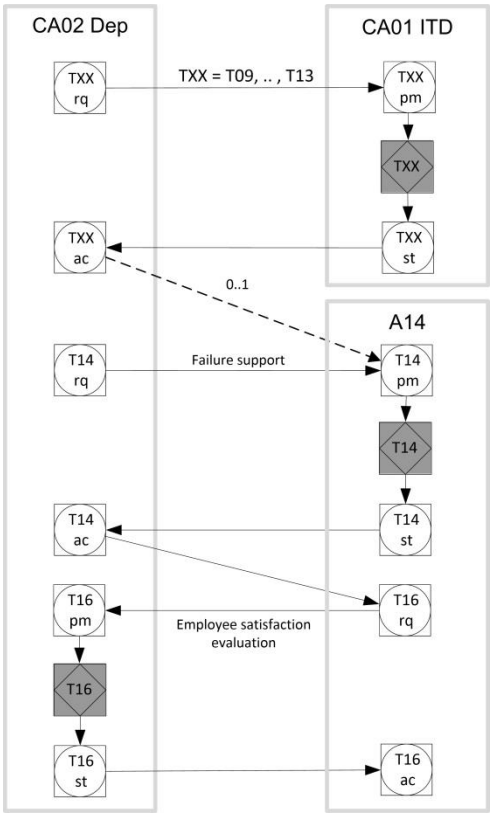


Figure 37 – The Process Structure Diagram of the business process involving T14

A business department employee (actor role CA02) starts by asking ITD (actor role CA01) for a particular service (TXX rq). After its promise (TXX pm), execution (TXX ex), and statement (TXX st), the employee (CA02) will accept the service result (TXX ac). After this acceptance, there may be something wrong with the service, so the employee (CA02) can use the ITD's Help Desk to request support (T14 rq). Then, after proceeding with the necessary help (T14 pm, ex, and st) and respective acceptance of the employee (T14 ac), the actor role A14 requests the employee to evaluate the quality of the support (T16 rq). Finally, the employee (CA02) evaluates the quality of the support (T16 pm, ex, and st).

The Action Model of an organization consists of a set of action rules and provides a full account of the essential operational decisions in the enterprise, leaving out all non-essential matters. Once more time, we focus on the action rules involved in T14.

when support of [Failure] is requested **with** employee of [Failure] is [Employee]
  if #services_in_use([Employee]) > 0 and <can fulfil request>
    then support of [Failure] must be promised
    else support of [Failure] must be declined

when support of [Failure] is promised
then support of [Failure] must be executed
then support of [Failure] must be stated

when support of [Failure] is stated
if <failure solution is acceptable >
then support of [Failure] must be accepted
else support of [Failure] must be rejected

when support of [Failure] is accepted
then employee satisfaction evaluation of [Failure] must be requested

These action rules specify what should be done after each C-act of the transaction T14 (failure support). For example, when a support to solve a failure (FA) is requested by an employee (E), the executor of T14 should verify if this employee E uses some service from ITD (#service_in_use(E)). If employee E uses services from ITD and it is possible to fulfil the request, then the executor of T14 (actor role CA01) should promise to solve the failure FA (promise T14(FA)) otherwise, CA01 should decline the employee E request (decline T14(FA)).

The State Model (SM) of an organization is the specification of the state space of the P-world. SM specifies the object classes, the fact types, and the result types, as well as the existential laws between them. SM is useful to define the Production World Semantics aspect of the production area in the GSSF and the Production World Semantics aspect allows a common knowledge and understanding about the semantics of the service to be provided (Terlouw & Albani, 2013).

The SM is expressed in an Object Fact Diagram (OFD) and an Object Property List (OPL). The OFD of the ITD is illustrated in Figure 38. There are six external object classes (in gray): Auditors, Department, Employee, Specialized Software, Semester and Equipment. There are ten internal object classes: Improvement, Internal Policy, Communication Network, File Storage, Failure, Audit, Feature, Business Case, Production Environment and Implementation Plan. The instances of the last ones are created by the identified transactions.

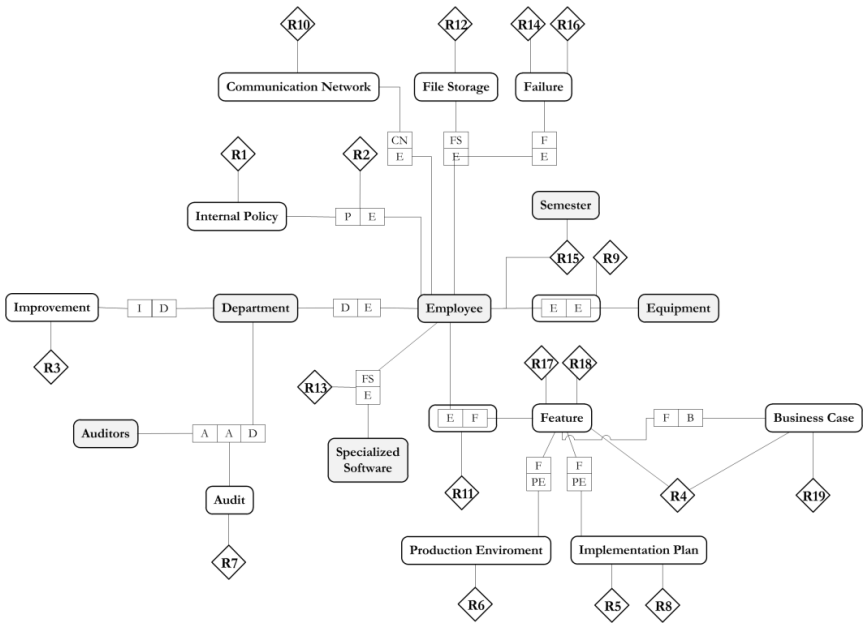


Figure 38 – Object Fact Diagram (OFD) of the ITD

The OPL is illustrated in Table 20.

Property Type	**Object Class**	**Scale**
#services_in_use(*)	EMPLOYEE	NUMBER
mandatory	FEATURE	BOOLEAN
risk	FEATURE	BOOLEAN

Table 20 – Object Property List (OPL) of the ITD

One of these properties is a derived fact type, indicated by an asterisk between brackets. The derivation rule is as follows: #services_in_use(E) = <sum of total services accepted by E>.

Having defined the State Model, it is possible to define the Information Use Table (IUT) from the Process Model (PM). The IUT specifies the object class, fact type, and result type in the steps of the Process Structure Diagram (PSD). Table 21 illustrates the IUT of the ITD.

For example, the INTERNAL POLICY object class is used in the request act of transactions T1 and T2. The IUT is useful to define for every coordination act and production act which information is required. Hence, IUT is used to define the Production Information Used aspect of the Service Production area in the GSSF (Terlouw & Albani, 2013).

Object class, fact type or result type	Process steps
INTERNAL POLICY	T01/rq, T02/rq
employee E has been trained for internal policies P	T02/rq
IMPROVEMENT	T03/rq
PRODUCTION ENVIROMENT	T06/rq
EMPLOYEE	T02/rq, T09/rq, T10/rq, T12/rq, T13/rq, T14/rq, T15/rq
FEATURE	T03/pm, T04/rq, T05/rq, T06/rq, T08/rq, T11/rq
COMMUNICATION NETWORK	T10/rq
FILE STORAGE	T12/rq
FAILURE	T14/rq
IMPLEMENTATION PLAN	T05/rq, T08/rq,
BUSINESS CASE	T04/rq,
DEPARTMENT	T03/rq
SEMESTER	T15/rq
EQUIPMENT	T09/rq
SPECIALIZED SOFTWARE	T13/rq
AUDIT	T07/rq

Table 21 – IUT of the ITD

We applied the Generic Service Specification Framework (GSSF) to specify the service 'failure support' that implements the T14 transaction (**second step** of the proposal).

The service executor area specifies the role of the actor that takes final responsibility for the service. The specification of this area is illustrated in the Table 22 below.

Service Specification - *Failure Support (T14)*	
Service Executor	
Actor Role	Help Desk Employee
Contact Information	General Email: Help Desk Phone Number Email: xxxxx Phone: xxxxx
Service Production	
Production Act	Failure support is the act of receiving, following and solving an unexpected service interruption or dysfunction. Failure support involves gathering information and solving the failure.
Production Information Used	The information used aspect is derived from the IUT (Table 21). The object classes used in this service are Failure and Employee.
Production Fact	The production fact can be derived from the Transaction Result Table (Table 19): 'Failure FA has been solved'.
Production Kind	The production kind of this transaction is 'ontological', since T14 forms part of the ontological model of the bank.

Production World Semantics	The production world semantics can be gained from the State Model that is illustrated in Figure 38.
Preconditions	Pre and post-conditions are gained from the Action Model. As pre condition we have: **when** support of [Failure] is requested **with** employee of [Failure] is [Employee] **if** #services_in_use([Employee]) > 0 and <can fulfil request> **then** support of [Failure] must be promised **else** support of [Failure] must be declined This service can only be provided if the employee has access to at least one service provided by ITD.
Postconditions	Regarding the post condition, we have the following action rule: **when** support of [Failure] is accepted **then** employee satisfaction evaluation of [Failure] must be requested Consequently, when the service failure support is accepted by the employee, ITD requests the same employee to evaluate the quality of the support.
Service Coordination	
Coordination Acts	The coordination acts aspect can be derived from the PSD that is illustrated in Figure 37.
Coordination Kind	This service is a Human Service
Protocol	1. Contact the Help Desk 2. Provide information about the failure 3. Wait for the statement 4. Test the failure solution 5. Accept the failure solution
Location	Web portal address, phone number and email.

Table 22 – Service "Failure Support" Specification

The service production area specifies the actual result that the Help Desk Employee actor role offers to the service initiator and the service coordination specifies the coordination acts, the coordination kind, the protocol and the service location.

As we mentioned before, each service may have several SLAs associated. We focused on the SLAs associated with the service 'failure support' (transaction T14) and we identified five SLAs for this service (**third step** of the proposal). We used the first version of our SLA proposal illustrated in Figure 21. The first SLA is related with the time to answer the phone calls. ITD employees accountable for the failure support service have orders to answer the phone calls in one minute. The SLA attributes are described in Table 23.

Service	Failure support (T14)
Target	Answer the phone in less than 60 seconds (the first coordination act must occur in a 60 seconds period after the request)
Price	0 €
Penalty	Not defined

Table 23 – SLA attributes for the service 'failure support'

The following four SLAs are related to the time to solve the failures. Each SLA represents a different priority. Table 24 describes the SLA attributes for the service 'failure support' with critical (top) priority.

Service	Failure support (T14)
Target	Solve failure with critical priority in 6 hours (the state act must occur in a 6 hours period after the promise)
Price	0 €
Penalty	Not defined

Table 24 – SLA attributes for the service 'failure support' with critical priority

The failure support service also had an SLA with high priority (the state act must occur in less than a 12 hour period after the promise), an SLA with medium priority (the state act must occur in less than a 60 hour period after the promise) and SLA with low priority (the state act must occur in less than a 120 hour period after the promise). The ITD clients can choose one of these four SLAs on requesting time.

7.4 Evaluation

Our evaluation strategy can be described using the framework defined in (Pries-Heje, et al., 2004):

- **What is actually evaluated?** The artifacts evaluated were the first three steps of the proposed method (Figure 23) and the first version of the SLA attributes proposal (Figure 21);
- **How it is evaluated?** We modelled the ITD services and SLAs using our proposal and we analysed the relation between the SLAs and the Enterprise Ontology transaction patterns. Thus the evaluation was conducted in a real organization facing real problems;
- **When was it evaluated?** It was evaluated ex post (after the design artifact was developed).

The identified SLAs revealed to be related with the acts defined in the Enterprise Ontology transaction patterns. Figure 39 illustrates these relations.

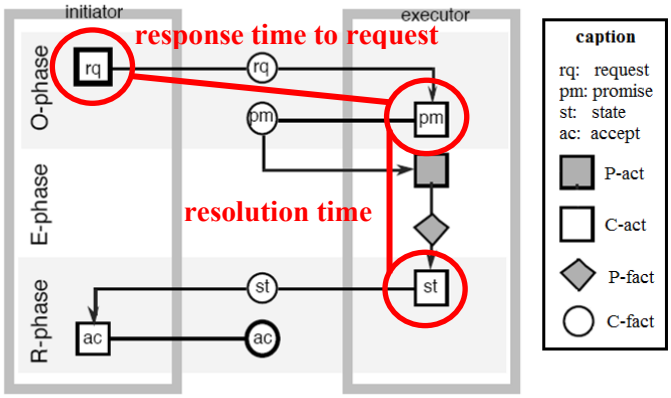


Figure 39 – The basic pattern of a transaction and the failure support SLAs

The SLA defined in Table 23 has a target 'Answer the phone in less than 60 seconds', which means that the first coordination act must occur in a 60 seconds period after the request. Therefore, this target is concerned with the response time to a request (first coordination act of a transaction). This response can be an ontological act (promise or decline) or an infological act (acknowledge of the request).

The SLA described in Table 24 has a target 'Solve failure with critical priority in 6 hours' which means that the state act must occur in a 6 hours period after the promise. Therefore, this target is concerned with the resolution time.

7.5 Lessons Learned

First, we conclude that the current attributes were sufficient to specify the ITD SLAs. Indeed, the proposal includes two attributes that were not used in this case, the price (always 0€) and the penalty. Despite this fact, these two attributes are important in our proposal. The price is always 0 € in this case, because the services are provided to others departments in the same organization and there is no chargeback. The penalty is also important in the definition of the SLAs, because not having penalties may diminish the efficiency of the service exchange. If there are no consequences for the deal breakers, then they will tend to do consequently worst.

On the other hand, this field study **indicates** that the Enterprise Ontology patterns of transactions can be used as a basis to define the targets of the SLAs, thus **contributing to validate Hypothesis 2.2** (the Enterprise Ontology patterns of transactions represent a structure to define

non-functional attributes of SLAs). As the EO theory describes the interaction between the customer and the provider in a very formal way, and since the Service Level Management acts as the interface between customer and provider, the EO provides a solid basis for formalizing the notion of SLA.

8 Specifying SLAs in a City Council

This field study occurred in a Portuguese city council named Pombal (CMP). Pombal is located in Leiria District and is composed of 17 parishes. It has a total area of 626.1 km² and a total population of 58 617 inhabitants. The population of the city of Pombal is about 18 000 inhabitants. CMP employs a total of 389 people with 203 men and 189 women and has five major departments divided into divisions, units and sections. In 2010 CMP spent a total of 20 553 200 € from which 7 542 250 € in human resources (Pombal City Council, 2011).

8.1 Problem

The main objective of this field study was to verify the maturity of the proposal, namely to validate if the SLA attributes were detailed enough to model the services provided by the Information Technology division of CMP. Therefore, this field study contributes to answer the research question **Q2 (how to specify service level agreements according to customers' expectations?)** and to validate **Hypothesis H2.2 (the Enterprise Ontology patterns of transactions represent a structure to define non-functional attributes of SLAs)**.

8.2 Proposal

In this field study we applied the first three steps of our proposal, i.e.:

1. Identify the services:
   1.1. Enterprise Description;
   1.2. Performa-Informa-Forma Analysis;
   1.3. Coordination-Actors-Production Analysis;
   1.4. Transaction Pattern Synthesis;
   1.5. Result Structure Analysis;

1.6. Actor Transaction Diagram/Service Identification.

2. Specify the executor, production and coordination of the services;
3. Specify the SLAs for each identified transaction/service using the structure of Figure 40.

We did not applied the fourth step of the proposal (fulfil the service requests according to the agreed SLAs) since the first three are sufficient to answer the research question Q2.

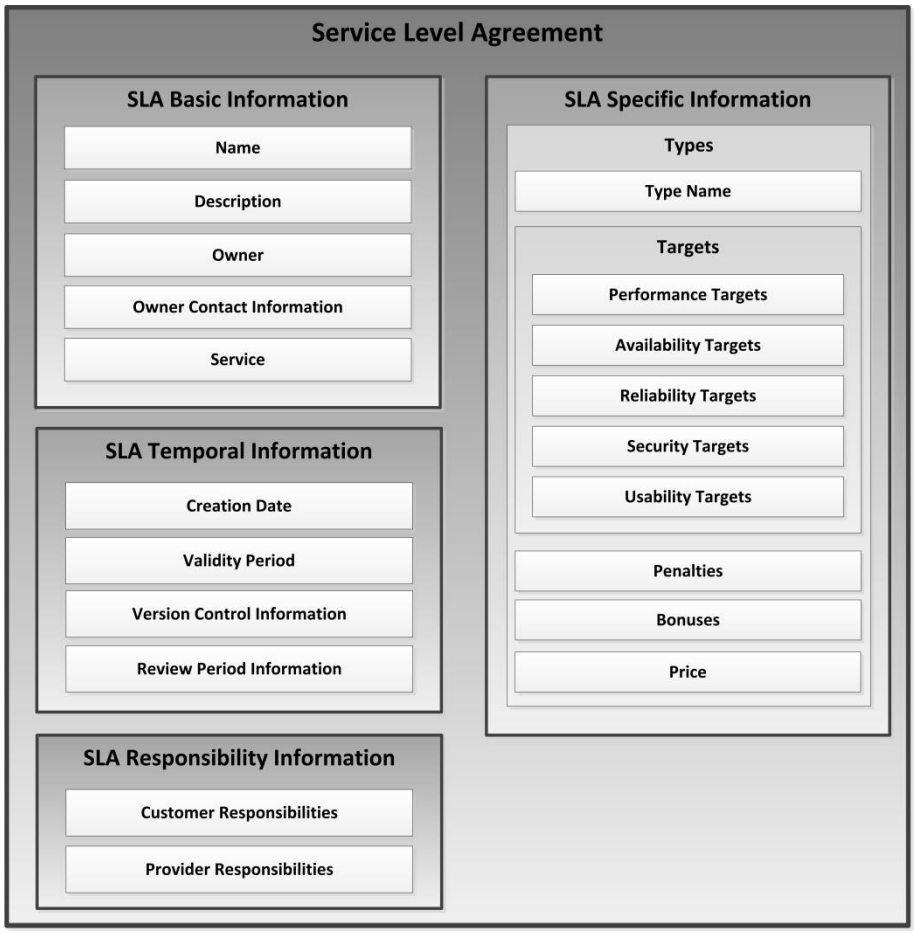


Figure 40 – Structure and attributes of the Enterprise Ontology-based SLA

Our SLA proposal considers four main areas of concern with their respective attributes that we will now explain. The first area is called **SLA Basic Information** and it defines the name of the SLA (*Name*) and the SLA purpose (*Description*). Additionally, the SLA Basic Information describes who owns this SLA (*Owner*), it also provides a contact of this person (*Owner Contact Information*) and, finally, this area defines the name of the service that the SLA applies to (*Service*).

The second considered area contains information concerning the dates of the SLA and is called **SLA Temporal Information**. In this area the date on which the SLA was established

(*Creation Date*) is defined as well as the time interval on which the SLA is valid (*Validity Period*), the information related to the SLA modification dates by the customer (*Version Control Information*) and the information concerning the SLA review dates performed by an entity related to the service provider (*Review Period Information*).

Next, we define an area called **SLA Responsibility Information** that regards the information about the responsibilities of each actor in the execution of this SLA. In this area two attributes are specified concerning the obligations and duties of the customer (*Customer Responsibilities*) and the service provider (*Provider Responsibilities*).

Finally, the last area is called **SLA Specific Information** and for each type of SLA (*Type Name*) it specifies five different types of targets (*Targets*), which can give rise to actions if they are not fulfilled (*Penalties*), but if they are fulfilled, this should be rewarded (*Bonuses*). Each type of SLA is also associated to a price (*Price*).

We reached this SLA version by doing a deep literature review in the service level management area and then by validating it with seven recognized practitioners involved in SLAs specification.

8.3 Demonstration

In order to identify the services (**step one** of the proposal) we interviewed individually 17 employees from CMP. With the purpose of having an overall perspective of the entire CMP we have selected employees from all the departments. During the interviews participants were asked to describe the activities performed by CMP. The interviews were recorded and transcribed as well as checked and discussed by two interviewers each ensuring unbiased findings and avoiding misinterpretation as specified in (Kvale, 2007).

The interviews allowed us to develop an enterprise description of CMP that was used as input for the service identification step (proposal first step). We do not fully describe the six sub steps of the service identification step, nevertheless these sub steps are based on the DEMO methodology (Dietz J. , 2006) and are described in the first two field studies. The result of this first step is the Actor Transaction Diagram (ATD) (Figure 41).

We identified 173 services of which 145 are ontological, 17 are infological and 11 are datalogical. These services correspond to all services provided by the five major departments that constitute the City Council. Figure 41 illustrates nine major Composite Actor Roles (Citizen, Customer Kernel, President, IT Kernel, Employee, HR Kernel, Special Mobility, Candidate and Doctor).

We focused on the services provided by the IT division. The IT division operates and maintains the computer equipment, develops new tools, supports their applications, and conducts courses to enhance learning of the new features.

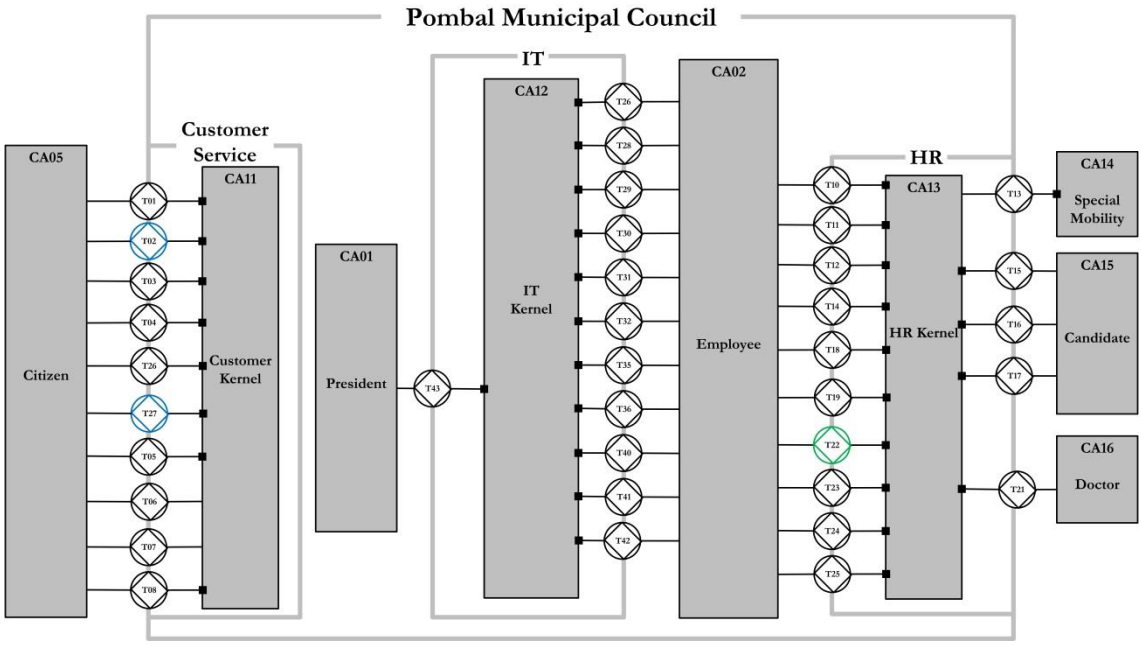


Figure 41 – ATD with HR and IT

Table 25 details the results of the transactions performed the city council IT division.

Transaction types	**Result types**
T26 – Network Configuration	R26 – Network N has been configured
T28 – Hardware Installation	R28 – Hardware H has been installed
T29 – Hardware Uninstallation	R29 – Hardware H has been uninstalled
T30 – Application Development	R30 – Application APP has been developed
T31 – Incident Resolution	R31 – Incident I has been resolved
T32 – Database Management	R06 – Database Management has been done for Period P
T35 – Software Installation	R35 – Software S has been installed
T36 – Software Uninstallation	R36 – Software S has been uninstalled
T40 – Backup Realization	R40 – Backup B has been done
T41 – Handbook Definition	R41 – Handbook HB has been defined
T42 – Training	R42 – Training T has been given
T43 – Business Intelligence Study Realization	R43 – Business Intelligence Study BIS has been done

Table 25 – Transaction Result Table (TRT) of the city council IT Division

We identified 12 services provided by the IT division: *Network Configuration* (T26), *Hardware Installation* (T28), *Hardware Uninstallation* (T29), *Application Development* (T30), *Incident Resolution* (T31), *Database Management* (T32), *Software Installation* (T35), *Software*

Uninstallation (T36), *Backup Realization* (T40), *Handbook Definition* (T41), *Training* (T42) and *Business Intelligence Study Realization* (T43).

In order to proceed to the **second step** of the proposal (specify the executor, production and coordination of the services), first we had to model the Process Model, the Action Model, and the State Model of Pombal City Council, since some aspects of the GSSF (used in the second step of the proposal) depend on these models. For the sake of readability, we only present the models used in the service specified below.

We applied the Generic Service Specification Framework (GSSF) to specify the services of Figure 41 (second step of the proposal). An example of this specification is illustrated in Table 26. This table describes the attributes of the service *Application Development* that implements the T30 transaction and is provided by the IT Division.

Service Specification - *Application Development (T30)*	
Service Executor	
Actor Role	*Developer (A02)*
Contact Information	General Email: suporte@cm-pombal.pt Email: xxxxx Phone: xxxxx
Service Production	
Production Act	Application Development is the act of designing and developing new applications required by other units of the Municipal Council or by the IT Division itself.
Production Information Used	The object classes used in this services are: Application, Handbook, Employee
Production Fact	The production fact can be derived from the Transaction Result Table (Table 25): Application APP has been developed
Production Kind	The production kind of this transaction is: Ontological
Production World Semantics	See Object Fact Diagram (Figure 42)
Preconditions	Pre and post-conditions are gained from the Action Model. There are no preconditions for this service.
Postconditions	Regarding the post condition, we have the following action rule: **when** development of [Application] is accepted **then** training of [Application] must be requested **and** definition of [Handbook] must be requested
Service Coordination	
Coordination Acts	See Figure 43
Coordination Kind	This service is a Human Service

Protocol	1. Contact the IT Division; 2. Specifies the Application requirements; 3. Wait for the end of development; 4. Accept/Decline Application;
Location	Email, Phone

Table 26 – Service "Application Development" Specification

The service specified above is called *Application Development* and is carried out by a Developer whose contact is available in the Service Executor area. In the Service Production area, it is specified that a new application is produced, which makes this an ontological service and, based on the State Model (Figure 42), we found the information classes used in this service: Handbook, Application and Employee.

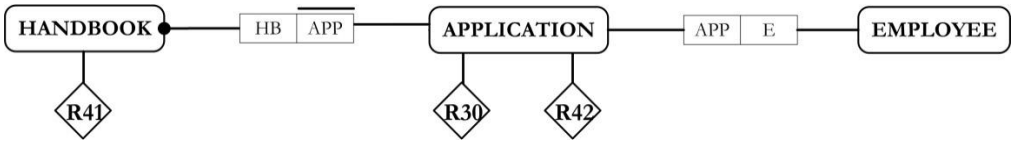


Figure 42 – State Model for Service 'Application Development'

The ontological coexistence rules between these classes are the following: an Application may have a Handbook and Handbook may describe several applications. A Handbook always requires the existence of an Application. An Application may be used by several Employees and an employee may have access to several applications.

Regarding the preconditions there are none associated with the *Application Development* service. As post conditions, the execution of T42 and T41 must be requested. In other words, the training and the handbook writing are the postconditions of the *Application Development* service.

Concerning the Coordination area, the coordination acts involved in this service are illustrated by the Process Structure Diagram (PSD) (Figure 43). In this diagram we see that the employee (CA02) makes a request for a new application (T30 rq) and this request is handled by the developer (A02). Which means that A02 may promise to develop the new application (T30 pm), develop the application (T30 ex) and state it (T30 st). Then, the employee (CA02) may accept the result of the development (T30 ac). When this happens, the developer (A02) has to start two new transactions: write the user manual (T41 rq) and schedule a training on this new application (T42 rq). At that moment, the standard pattern may be repeated for these two transactions (T41 and T42). In other words, T41 and T42 may be promised, executed, stated and accepted.

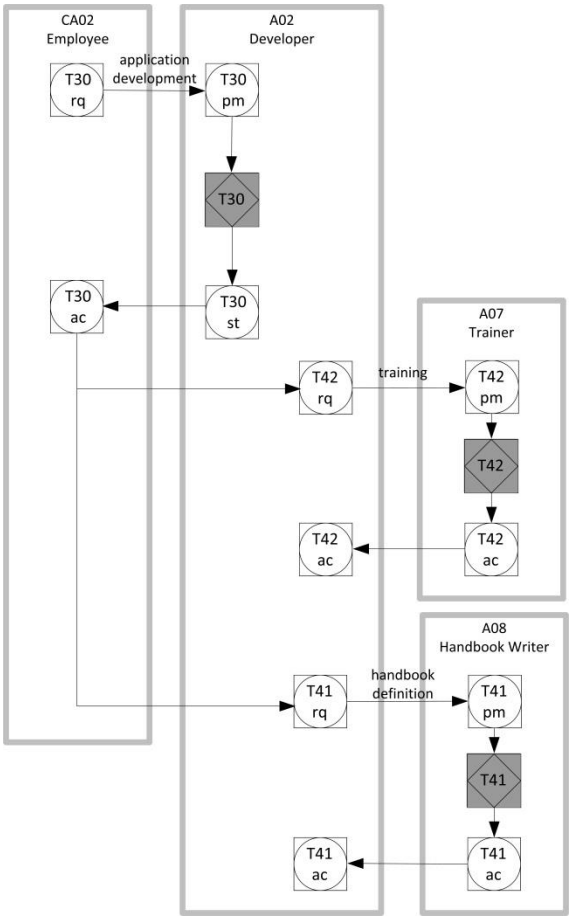


Figure 43 – PSD for Service *'Application Development'*

In addition, in the Coordination area the protocol to successfully contact the service provider is specified, as well as the location of the service that, in this case, is by email or phone.

In the **third step** of the proposal we specified the SLAs associated to the CMP services using our SLA proposal (Figure 40). Table 27 illustrates an example of this specification for the service *Application Development*.

This SLA concerns the development of an application named WebDoc2.0 and the SLA Owner is the IT Division Chief. Multiple contacts of the IT Division Chief are specified in order to be contacted by the service customer at any time. This SLA was made on the first day of January of 2011, was valid until 31 of November of 2011 and was changed in July 15. To fulfil this SLA, the service provider needs to complete all the points specified in the SLA Provider Responsibilities (completion of ERP integration and putting into production all features developed).

Service Level Agreement Specification	
SLA Basic Information	
Name	**WebDoc 2.0 Development**
Description	SLA concerning the development of a tool for document management.
Owner	Nome: Nuno Salvador Category: IT Division Chief Organic Unit: IT Division
Owner Contact Information	Email: xxx Phone: xxx
Service	*Application Development* (T30)
SLA Temporal Information	
Creation Date	January 1, 2011
Validity Period	Until November 31, 2011
Version Control Information	July 15, 2011
Review Period Information	NA
SLA Responsibility Information	
Customer Responsibilities	NA
Provider Responsibilities	1 - Finish ERP Integration; 2 - Place in operation all the features developed.
SLA Specific Information	
Type	**Overcome Goal**
Targets	
Performance	Until November 15, 2011 (the state act occurs until November 15, 2011)
Penalties	NA
Bonuses	Evaluation Score: 5 Career: Allows career evolution
Price	0 €
Type	**Fulfilment Goal**
Targets	
Performance	Until November 31, 2011 (the state act occurs until November 31, 2011)
Penalties	NA
Bonuses	Evaluation Score: 3
Price	0 €
Type	**Non Fulfilment Goal**
Targets	
Performance	After November 31, 2011 (the state act occurs after November 31, 2011)
Penalties	Evaluation Score: 1 Career: Can be fired with probable cause
Bonuses	NA
Price	0 €

Table 27 – SLA "WebDoc 2.0 Development" Specification

This SLA has three types that depend on the date of completion of the development. Penalties and bonuses are translated into career points that influence the career development.

In case of SLA Type "Overcome goal" (the state act occurs until November 15 of 2011), the SLA Owner wins five career points and can evolve in his career. In case of SLA Type "Fulfilment goal" (the state act occurs until November 31 of 2011), the SLA Owner wins three career points. Finally, if the service provider does not meet the deadline (the state act occurs after November 31 of 2011) the SLA Type "Non Fulfilment Goal" applies and the SLA Owner only wins one career point and can be fired with probable cause.

Note that this SLA has no price defined because this is an internal service to the CMP and no chargeback is made among the CMP departments.

8.4 Evaluation

To explain the evaluation we use the framework proposed in (Pries-Heje, et al., 2004). This framework identifies what is actually evaluated, how it is evaluated and when the evaluation takes place.

Table 28 illustrates the answers to the three main questions that this framework proposes to answer:

- **What is actually evaluated?** The artifact evaluated is the proposed set of steps of Section 8.2 (a design process) and the results of applying these steps to the CMP (Services and SLAs; a design product);
- **How is it evaluated?** We used CMP employees' and customers' feedback to evaluate the Enterprise Ontology-based SLA structure and the CMP services and SLAs. This represents a naturalistic evaluation since it was conducted in a real organization facing real problems;
- **When was it evaluated?** It was evaluated ex post (after the design artifact was developed).

P summarizes the essential characteristics of the evaluation Process, while C indicates the evaluation Criteria.

The evaluation was naturalistic since we applied our proposal in a real organization with real data. The evaluation was ex post since it occurred after the demonstration in the CMP. We evaluated both the proposed set of steps (including the Enterprise Ontology-based SLA structure) and the results of applying these steps to the CMP (Services and SLAs specification).

	Ex Ante	Ex Post
Naturalistic	Design Process / Design Product	P: CMP employees' feedback & customers' feedback C: SLA Attributes Quality
Artificial	Design Process / Design Product	Design Process / Design Product

Table 28 – Evaluation strategy

In order to evaluate the Enterprise Ontology-based SLA structure (see Figure 40) and the Services and SLAs specification from CMP we collected feedback from 23 CMP employees and 7 CMP customers. They were arbitrarily chosen and were asked to classify the attributes of the SLA proposal from 1 to 10 according to the importance (being 1 irrelevant and 10 essential).

Figure 44 illustrates the average and standard deviation per attribute. As can be seen there was little variation in the answers of the interviewees in most attributes.

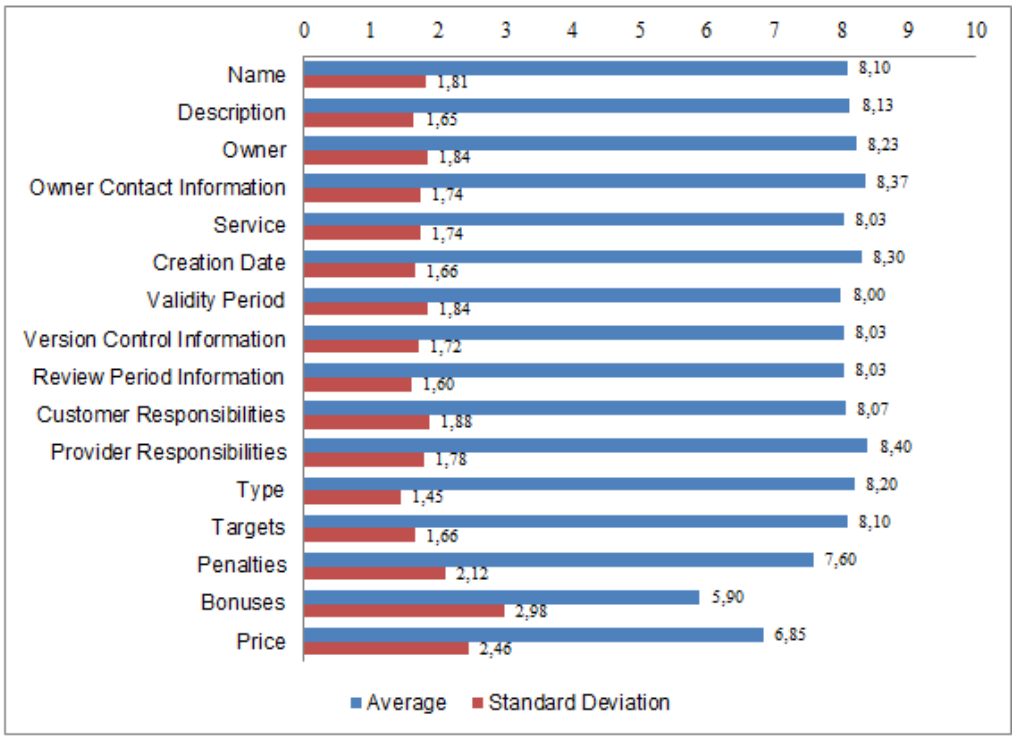


Figure 44 – Rating of proposal attributes

The first 14 attributes had high average scores (from 7.60 to 8.40) and the remaining two (Bonuses and Price) had lower classifications (5.90 and 6.85).

These results indicate that the majority of the proposed attributes (14 in 16) were classified as important since they scored a minimum of 7.60 in 10 possible points. The remaining two attributes (Bonuses and Price) scored 5.90 and 6.85 revealing that they were classified as less important when comparing to the first 14. These results can be explained by the fact that there are no chargeback among the departments of the 30 inquired persons. Therefore, they value more the attributes that describe the service quality than the ones that capture the costs.

Besides this validation, we also validated the service catalogue from the IT Division with the IT chief that agreed with all 12 identified services.

The definition of a city council service catalogue has great potential, because the services that this type of organizations provides to the citizens are similar. For instance, in Portugal there are 308 city councils and in theory they all have the same purpose. Having identified the service catalogue of one city council we can validate if it is applicable to other city councils and eventually find some services that could be provided in cooperation.

Knowing the services and the type of services provided allows one to understand how the service provider can improve his performance. For instance, the service *Collect Water Usage* – T155 is a datalogical transaction since it neither involves the creation of new original facts (ontological) nor information processing (infological). Hence, it has large potential to be optimized, since technology can be used to reduce the effort needed to execute datalogical acts (Dietz J. , 2006). CMP has four employees dedicated to collect water usage. Assuming that CMP spends 77 555 € a year with these four employees ((7 542 250 € / 389 employees) * 4 employees) and the other 307 city councils in Portugal do not have this service automated and use similar resources on it, then this would represent an expense of 23 809 468 € a year in a service that has great potential to be automated. This value is estimated using only the costs of the employees' wages so if we add the supporting costs (IT support, HR support, etc.) then the estimated value would certainly be higher.

This kind of analysis has special value because of the current situation Portugal is in since it points some solutions to a number of current challenges imposed by the Troika memorandum.

8.5 Lessons Learned

The employees' and customers' feedback revealed that in the CMP context 14 in 16 of the proposal attributes were considered important. This indicates that this new version of Enterprise

Ontology-based SLAs has more potential to capture the customers' expectations than the older version of the proposal that had only 4 attributes. By specifying these attributes, customers can structurally define their expectations which may help the alignment between customers and service providers. The specification of the customers' expectations into SLAs helps the service providers to understand those expectations and consequently reduce the gaps among the two.

Although the questionnaire results do not show evidences about it, during the process of collecting the employees' and customers' feedback, we also found that they had problems understanding the attributes from the SLA Temporal information area. They had difficulties in differentiating the Version Control Information attribute and the Review Period Information attribute. In addition, the Validity Period was also confused with the SLA targets. Therefore, we decided to remove the SLA Temporal area from the SLAs proposal.

Additionally, we found a particular service that was not possible of being completely modelled by our proposal. This service had a different price according to the way it was requested. By the internet the price of the service was 50% less than going physically to CMP. Therefore we decided to add a new attribute to the SLAs (service configuration). Therefore, this field study **contributed** to answer the research question **Q2 (how to specify service level agreements according to customers' expectations?).** On the other hand, this field study also **indicates** that the Enterprise Ontology patterns of transactions can be used as a basis to define the targets of the SLAs, thus **contributing to validate Hypothesis 2.2** (the Enterprise Ontology patterns of transactions represent a structure to define non-functional attributes of SLAs).

Even though it was not an objective of this proposal we found some possible improvements for the CMP services. Since our proposal is based on the Enterprise Ontology theory and identifies services that do not create new original results, then our proposal can identify services with great potential to be automated. In this particular city council we found services that if automated could save the Portuguese state millions of euros.

9 Specifying SLAs in a Cloud Services Provider

This field study took place in a cloud services provider that, for confidentiality purposes, we call CSP. CSP is one of the main Portuguese operators of fixed telecommunications providing a combined offer of Voice, Data, Internet and Service components aimed at large companies, SMEs, the public sector and other telecom operators. CSP employs a total of 247 people and in 2010 had revenues of 136 600 000 €. Over two thirds of these revenues were derived from complex data and communications services.

9.1 Problem

This field study had the same objective of the previous field study: to verify the maturity of the proposal, namely to validate if the SLA attributes were detailed enough to model the service provided by the CSP. Therefore, this field study contributed to answer the research question **Q2 (how to specify service level agreements according to customers' expectations?)**.

9.2 Proposal

Similarly to the previous two field studies, in this field study we applied the first three steps of our proposal, i.e.:

1. Identify the services:
    1.1. Enterprise Description;
    1.2. Performa-Informa-Forma Analysis;
    1.3. Coordination-Actors-Production Analysis;
    1.4. Transaction Pattern Synthesis;
    1.5. Result Structure Analysis;
    1.6. Actor Transaction Diagram/Service Identification.

2. Specify the executor, production and coordination of the services;
3. Specify the SLAs for each identified transaction/service using the structure of Figure 45.

We did not applied the fourth step of the proposal (fulfil the service requests according to the agreed SLAs) since the first three are sufficient to answer the research question Q2.

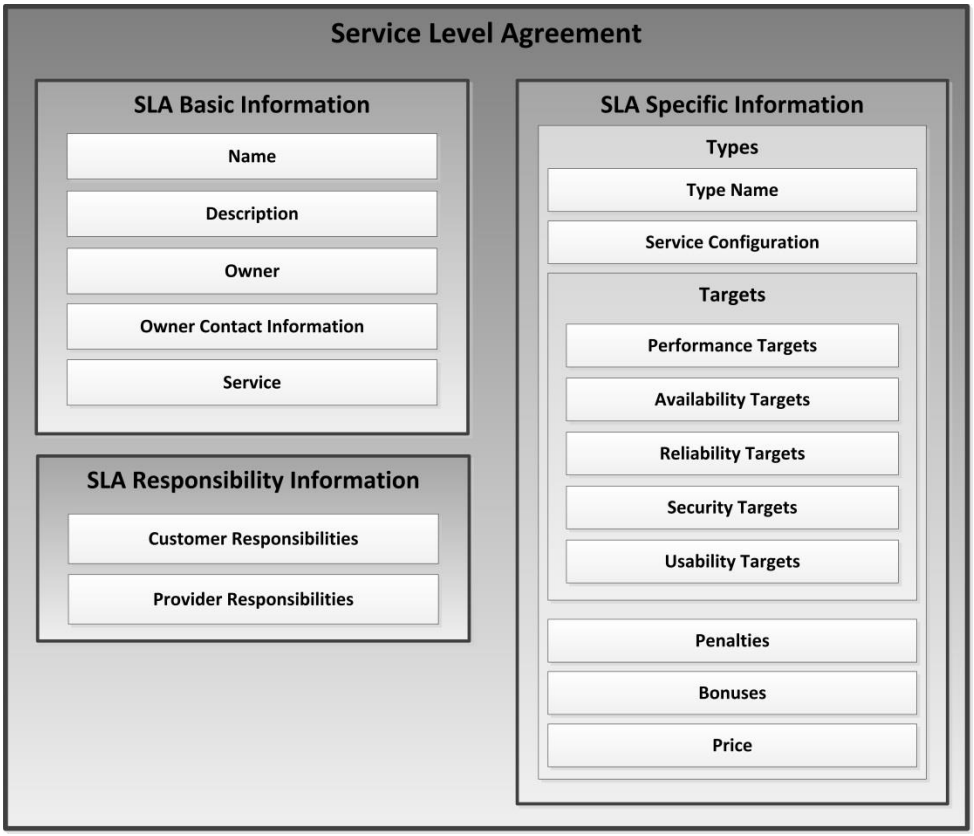


Figure 45 – Structure and attributes of the Enterprise Ontology-based SLA

Our SLA proposal considers three main areas of concern and 17 attributes. The first area is called SLA Basic Information and it defines the name of the SLA (Name) and the SLA's purpose (Description). Additionally, the SLA Basic Information describes who owns this SLA (Owner) and also provides this person's contact (Owner Contact Information). Finally, this area defines the name of the service that the SLA is applied to (Service). This service must be specified using the first three areas of the GSSF (executor, production and coordination).

Next, we define an area called SLA Responsibility Information that considers the information about each actor's responsibilities in the execution of this SLA. In this area two attributes are specified concerning the customer's obligations and duties (Customer Responsibilities) and those of the service provider (Provider Responsibilities).

Finally, the last area is called SLA Specific Information and for each type of SLA (Type Name) it specifies the service configuration and five different types of targets (targets) that can give rise to actions if they are not fulfilled (penalties), or rewards (bonuses) if they are fulfilled. Each type of SLA is also associated to a price (Price).

9.3 Demonstration

Recently, CSP invited the authors to help redesign their cloud services in order to be more customer oriented and aligned with customers' expectations. Therefore, this demonstration is focused on a specific area of the CSP services, the cloud services offering.

In order to identify the services (**step one** of our proposal) we interviewed four CSP employees individually. The participants were asked to describe the activities performed by CSP. The interviews were recorded and transcribed as well as checked and discussed by two interviewers, each ensuring unbiased findings and avoiding misinterpretation as specified in (Kvale, 2007).

The interviews allowed us to develop an enterprise description of CSP (first sub-step of service identification step). We do not fully describe the six sub-steps of the service identification step, nevertheless these sub-steps are based on the DEMO methodology (Dietz J. , 2006) and are described in the first two field studies.

An important result of this step is the list of transactions/services. This list identifies, for each transaction type, the result type (i.e., the Production fact created). The list is presented in the Table 29 below.

Transaction Types	**Result Types**
T01 – Data Centre Access Start	R01 – Customer C access to Data Centre DC has been started
T02 – Server Access Start	R02 – Customer C access to Server S has been started
T03 – Server Configuration	R03 – Server S has been configured
T04 – File System Configuration	R04 – File System FS has been configured in Server S
T05 – Processor Configuration	R05 – Processor P has been configured in Server S
T06 – RAM Configuration	R06 – RAM R has been configures in Server S
T07 – Internet Configuration	R07 – Internet I has been configured in Server S
T08 – Operating System Configuration	R08 – Operating System OS has been configured in Server S
T09 – Server Access End	R09 – Customer C access to Server S has been ended
T10 – Payment Management	R10 – Payments Management for Period P has been done
T11 – Server Access Payment	R11 – Customer C access to Server S for Period P has been paid
T12 – Failure Support	R12 – Failure F has been solved
T13 – Problem Diagnose	R13 – Problem B has been diagnosed

Transaction Types	Result Types
T14 – Problem Solve Internally	R14 – Problem B has been solved internally
T15 – Problem Solve Externally	R15 – Problem B has been solved externally

Table 29 – Transaction Result Table (TRT) of CSP

After identifying the transaction types and results, we determined the environment surrounding CSP. The first model developed is called the Actor Transaction Diagram (ATD) and is depicted in Figure 46.

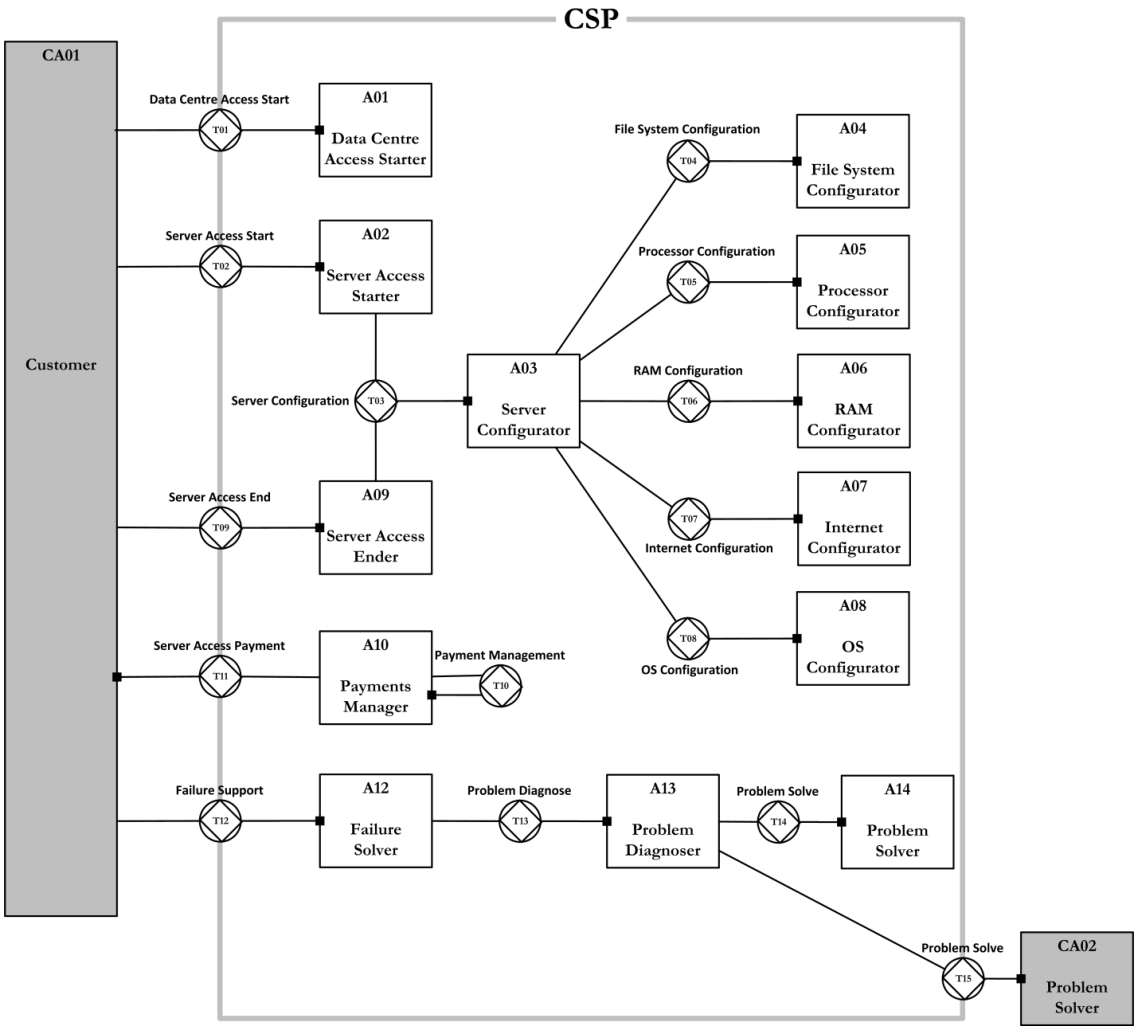


Figure 46 – CSP Actor Transaction Diagram (ATD)

In the ATD, a transaction/service is represented using a diamond in a disk that contains the respective combination of C-acts and a P-act. Each transaction is connected to two boxes, representing the initiator and executor actor roles. The initiator is connected to the transaction symbol using a solid line while the executor is connected to the transaction by a solid line ending in a

black square. The grey boxes refer to composite actor roles, i.e. elements whose exact structure is not known. All the environmental elements, i.e. elements outside the organization, are represented with grey boxes. This also means that we can represent the organization with a grey box when referring to the kernel of the organization, which can be further specified by using elementary actor roles represented by white boxes.

We identified 15 ontological services (T01 to T15), two composite actor role (CA01 – Customer and CA02 – Problem Solver) and 13 elementary actor roles (A01 to A14). Transaction T01 gives access to a Virtual Data Centre, a new service provided by CSP. T01 is completely automated and CSP customers can customize their own Data Centre by requesting the following transactions. Transaction T02 gives access to a Virtual Private Server. In order to provide this service, CSP must execute transactions T03 to T08, i.e. a server must be configured (T03) and the chosen options installed: File System (T04), Processor (T05), RAM (T06), internet connection (T07), and Operating System (T08). Transaction T09 ends the access to a server.

Transactions T10 and T11 represent the payment process. Every month, T10 is self-activated by the Actor Role A10 that requests the customers to pay for the services they use from CSP (T11).

Transaction T12 supports failures and it may involve three other transactions: T13, T14 and T15. T13 is executed by CSP's first level of support and their major goal is to diagnose the root cause of the failure and, if possible, to immediately solve the problem (T14). If it is not possible, they can request internal (T14) or external (T15) experts to solve the problem.

In order to proceed to the proposal's **second step** (specify the executor, production and coordination of the services), we had to model the Process Model, the Action Model, and the State Model of CSP, since some aspects of the GSSF (used in the second step of the proposal) depend on these models. For the sake of readability, we only present the models used in the service specified below.

In the second step of the proposal we applied the Generic Service Specification Framework (GSSF) to specify the services of Figure 46. An example of this specification is illustrated in Table 30 that describes the attributes of the service Server Access Start that implements the T02 transaction.

Service Specification – *Server Access Start (T02)*	
Service Executor	
Actor Role	*Server Access Starter (A02)*
Contact Information	General Email: -----

	Email: ----- Phone: -----
Service Production	
Production Act	Give customer C access to a Server with the chosen configurations
Production Information Used	The object classes used in this services are: Customer, Server, Operating System, Processor, RAM, Internet and File System
Production Fact	The production fact can be derived from the Transaction Result Table (Table 29): Customer C access to Server S has been started
Production Kind	The production kind of this transaction is: Ontological
Production World Semantics	See Object Fact Diagram (Figure 47)
Preconditions	Pre and post-conditions are gained from the Action Model. As precondition we have: **when** start of access to [Server] is promised **then** configuration of [Server] must be requested
Postconditions	There are none postconditions for this service
Service Coordination	
Coordination Acts	See Figure 48
Coordination Kind	This service is a Human Service
Protocol	Contact the Service Executor Negotiate the service options
Location	Service Executor Contacts

Table 30 – Service "Server Access Start" Specification

This service is executed by the actor role Server Access Starter (A02) that is fulfilled by a CSP employee. An email and a phone number are provided to the customers.

In the service production area, a description of the action performed by the service provider is given (production act attribute), the categories used in this act are listed (production information used) and the resulting production fact is described (Customer C access to Server S has been started). As this production fact is an original one (this fact cannot be derived from previous facts), then the service can be defined as ontological (see Theoretical Background Chapter – page 17). The production world semantics attribute can be obtained from the Object Fact Diagram (OFD) illustrated in Figure 47. The OFD is based on the language WOSL (Dietz J. , 2005) and shows the categories (external in grey), the relations between categories, and the transactions' results (diamonds).

The external categories (not created/modified in this context) are Customer, Operating System (OS), Internet, RAM, Processor and File System. The Customer category represents all the customers of CSP, the Operating System, the Internet, the RAM, the Processor and the File System categories represent the options the customers have to install in their virtual private servers.

The internal categories are: Data Centre, Server, Failure and Problem. The Data Centre category represents the Virtual Data Centres that CSP's customers can access. The coexistence rules of this category are: one customer can have several Data Centres; a Data Centre belongs to only one customer; a Data Centre can have several servers; and, a server can only exist in one Data Centre. The Server category identifies the virtual private servers. The server is connected to the other categories by the following coexistence rules: a customer may have several servers, but a server may only be associated with one customer and an operating system may be installed in several servers, but a server can only have one operating system. The same applies for File System, Processor, RAM and Internet.

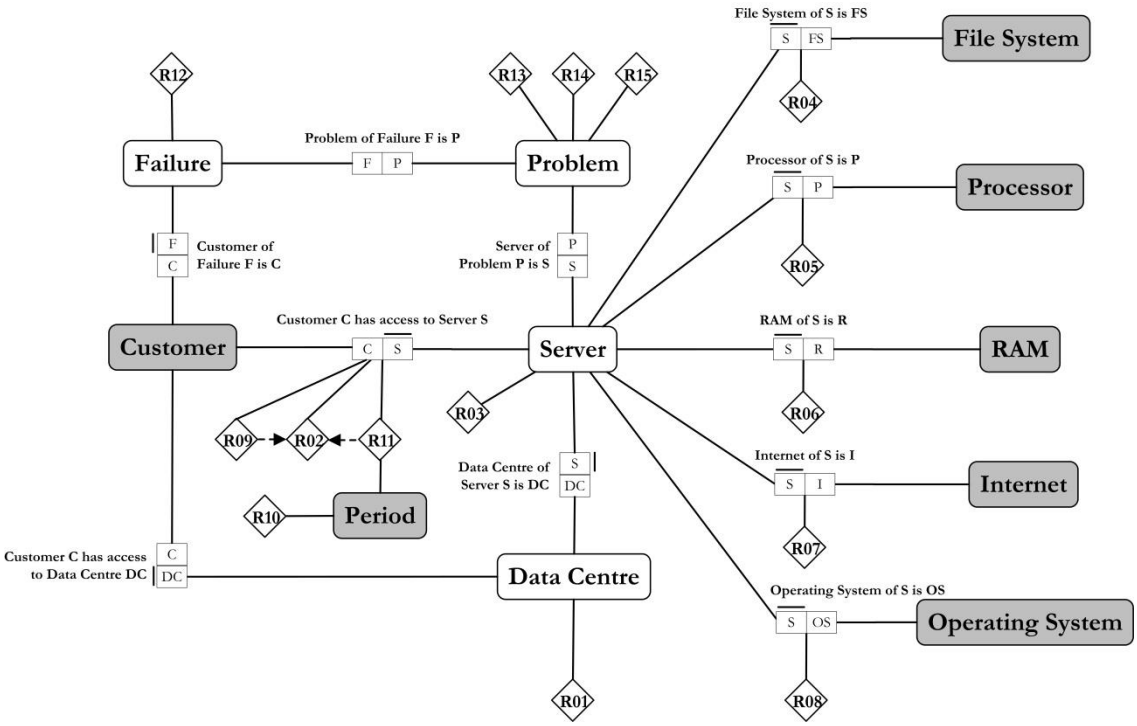


Figure 47 – CSP Object Fact Diagram (OFD)

A failure is associated with only one customer, but a customer can have several associated failures. A failure may be connected to many problems and a problem can cause several failures. A problem might also be associated with several servers and a server can have several problems.

The transaction events are associated to the categories as follows: the result R01 is associated to the Data Centre category, the results of transactions T02, T09, and T11 (R02, R09 and R11) are linked to the connection between Customer and Server. The results R09 and R11 are dependent of the result R02 since a server access can only be ended (R09) if it has been started (R02) and a customer will only pay for a server (R11) that he has access to (R02).

The result R03 (Server S has been configured) is related to the Server category, the result R08 (Operating System OS has been configured in Server S) is associated with the connection between Server and Operating System; the result R07 (Internet I has been configured in Server S) is associated to the connection between Server and the Internet; the result R06 (RAM R has been configured in Server S) is linked to the connection between Server and RAM; the result R05 (Processor P has been configured in Server S) is associated to the connection between Server and Processor; and, the result R04 (File System FS has been configured in Server S) is linked to the connection between Server and File System. The result R12 (Failure F has been solved) is associated to Failure category and the results R13, R14 and R15 to the Problem Category.

Regarding the preconditions attribute, before the service Server Access Start is executed, the provider must first configure the server, i.e. execute T03. After the service execution, there are none conditions (postconditions attribute).

The service coordination area is composed by four attributes: coordination acts, coordination kind, protocol and location. These attributes can be obtained from the Process Structure Diagram (PSD) that is part of the Process Model (Dietz J. , 2006). The PSD represents the process that must be carried out to fulfil the service and Figure 48 represents an example of such a process.

The process starts by the customer's (CA01) request (T02 rq) of transaction T02 (Server Access Start). A CSP employee (fulfilling actor role A02) promises to deliver this service (T02 pm) and requests (T03 rq) the execution of the service T03 (Service Configuration). Then the actor role A03 promises (T03 pm) to configure the server, executes the configuration and informs (T03 st) the requester (A02).

Afterwards, the actor role A02 accepts (T03 ac) the result of the service T03, creates the server access for the customer (T02 execution) and sends the access information to the customer (T02 st). Finally, the customer (CA01) accepts the T02 result. Note that, in order to execute the service T03, the actor role A03 must request the execution of T04, T05, T06, T07 and T08 (server options configuration). We do not show the entire pattern of these transactions in order to simplify this diagram.

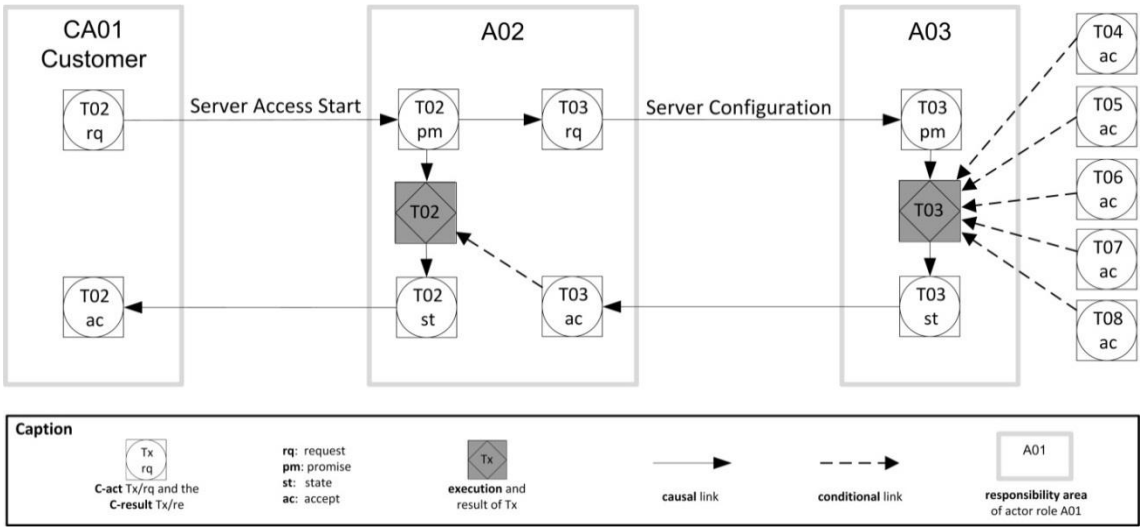


Figure 48 – CSP Process Structure Diagram (PSD) and caption

In the **third step** of the proposal we specified the SLAs associated to the CSP services using our SLA proposal (Figure 45). Table 31 illustrates an example of this specification for the service Server Access Start (T02).

Service Level Agreement Specification		
SLA Basic Information		
Name	Virtual Private Server Configuration	
Description	SLA concerning the configuration of a Virtual Private Server	
Owner	*Server Access Starter (A02)*	
Owner Contact Information	Email: ----- Phone: -----	
Service	*Server Access Start* (T02)	
SLA Responsibility Information		
Customer Responsibilities	1 – Choose a server configuration;	
Provider Responsibilities	1 – Configure the server with the requested options; 2 – Inform the customer about the price; 3 – Measure SLAs targets fulfilment.	
SLA Specific Information		
Type	**Configuration 1 – Standard**	**Configuration 1 - Enterprise**
Service Configuration	Windows Standard, (n) core, (x) GB RAM, (z) GB storage	Windows Enterprise, (n) core, (x) GB RAM, (z) GB storage
Targets		
Availability	99,95%	99,95%

Penalties	2% of the price	2% of the price
Bonuses	Not Defined	Not Defined
Price	62,50 € monthly	112,50 € monthly
Type	**Configuration 2 – Standard**	**Configuration 2 - Enterprise**
Service Configuration	Windows Standard, (n+1) core, (x+1) GB RAM, (z+100) GB storage	Windows Enterprise, (n+1) core, (x+1) GB RAM, (z+100) GB storage
Targets		
Availability	99,95%	99,95%
Penalties	2% of the price	2% of the price
Bonuses	Not Defined	Not Defined
Price	102,50 € monthly	152,50 € monthly
Type	**Configuration 3 – Standard**	**Configuration 3 - Enterprise**
Service Configuration	Windows Standard, (n+3) core, (x+3) GB RAM, (z+250) GB storage	Windows Enterprise, (n+3) core, (x+3) GB RAM, (z+250) GB storage
Targets		
Availability	99,95%	99,95%
Penalties	2% of the price	2% of the price
Bonuses	Not Defined	Not Defined
Price	192,50 € monthly	242,50 € monthly

Table 31 – SLA "Virtual Private Server Configuration" Specification

As stated before, our SLA proposal is composed by three areas: SLA Basic Information, SLA Responsibility Information and SLA Specific Information. In the first one, we defined the name of the SLA (Virtual Private Server Configuration), the SLA description (SLA concerning the configuration of a Virtual Private Server), the SLA owner (Server Access Starter - A02), the SLA owner contacts, and the service (Server Access Start - T02). In the SLA Responsibility Information area we defined both the customers' and the provider's responsibilities.

Finally, in the third area, we specified the quality options of the service that correspond to the six SLAs types: Configuration 1 – Standard, Configuration 1 – Enterprise, Configuration 2 – Standard, Configuration 2 – Enterprise, Configuration 3 – Standard, and Configuration 3 – Enterprise. When requesting access to a server (T02), the customers can only choose one of these SLAs types. These six SLAs types represent different combinations of Operating Systems, number of processor cores, RAM capacity, and storage capacity that CSP has available for the

service Server Access Start (T02) (the last three attributes were specified using variables n, x and z due to confidentiality purposes).

Every combination is described in the Service Configuration attribute and each one is associated to a different monthly price (from 62,50€ to 242,50€). All the SLA types have a common target regarding the service availability, since the server made accessible in T02 should be available 99,95% of the time. If not, the penalty applies and the customer receives a discount of 2% in the price (penalty attribute). In this service there are no bonuses.

9.4 Evaluation

This section corresponds to the evaluation phase of DSRM in which we use the framework proposed in (Pries-Heje, et al., 2004). This framework identifies what is actually evaluated, how it is evaluated and when the evaluation takes place.

In Table 32 we illustrate the answers to the three main questions proposed by this framework:

- **What is actually evaluated?** The artifact evaluated is the proposed set of steps (a design process) and the Services and SLAs that result from applying these steps to the CSP (a design product);
- **How is it evaluated?** We used CSP employees' feedback, the Four Principles from (Österle, et al., 2011) and the Moody and Shanks framework (Moody & Shanks, 2003) to evaluate the Enterprise Ontology-based SLA structure as well as the CSP services and SLAs. This represents a naturalistic evaluation since it was conducted using a real artifact in a real organization facing real problems;
- **When was it evaluated?** It was evaluated ex post.

In the Table 32 bellow, P summarizes the essential characteristics of the evaluation Process, while C indicates the evaluation Criteria.

	Ex Ante	**Ex Post**
Naturalistic	Design Process / Design Product	P: CSP employees' feedback C: SLA Attributes Quality
Artificial	Design Process / Design Product	Design Process / Design Product

Table 32 – Evaluation strategy

We collected feedback from four CSP employees. They were arbitrarily chosen and asked to freely comment on the proposal. The feedback was positive and the identified benefits are described in the last principle of (Österle, et al., 2011). Overall, the Four Principles from (Österle, et al., 2011) were accomplished:

- **Abstraction**: the artifact can be applied to any cloud service from any given enterprise description. As the proposal provides abstract models that focus on the business layer, not considering the implementation details, it can be applied to organizations in different contexts. In fact, previously we evaluated this proposal in several organizations, from public city councils (Mendes, et al., 2012) to private banks (Mendes & Mira da Silva, 2012);
- **Originality**: the proposed artifact is not present in the body of knowledge of the domain since it was designed by relating independent subjects, such as service level management and DEMO;
- **Justification**: the artifact is supported by the related work, described by textual and graphical representations, and justified and validated in different ways;
- **Benefit**: the artifact provides a structured working approach for specifying customers' expectations. By specifying the proposal attributes, customers can structurally define their expectations which may help the alignment between customers and service providers. The specification of the customers' expectations into SLAs helps the service providers understand those expectations and consequently reduce the gaps between the two. Additionally, the interviewees recognized the value of our proposal in the specification of their services, since it allowed the service provider to rethink some attributes of the services that had not been initially designed (for example, the Bonus attribute). Finally, the advantages of modelling the services interaction with DEMO were also mentioned, namely the simplicity of the ATD (Figure 46).

In order to evaluate the quality of the produced models, we used the Moody & Shanks framework (Moody & Shanks, 2003). The results were similar to the previous field studies since the models present high quality regarding Flexibility, Simplicity, Completeness, Integrity, and Correctness, and minor quality concerning Implementability and Understandability. Regarding the Implementability limitation, this is implicit to DEMO models because the theory in which DEMO is based on was designed to be implementation independent. However, this limitation did not compromise the quality of this field study's results. Due to the specific notation, DEMO models may also be hard to understand by those who are not familiar with the notation. Nevertheless, the fact that these models are composed by few element types diminishes the learning period.

9.5 Lessons Learned

While a lot of research is currently taking place in the cloud technology itself, there is an equally urgent need for understanding the business-related issues surrounding cloud services (Marston, et al., 2011). Even though it might be impossible to conjecture all the technological changes in the near future, the economic forces shaping this phenomenon, in contrast, are very logical and almost inexorable in nature (Marston, et al., 2011).

On the other hand, the difference between customers' expectations and the perceptions of those expectations by the service providers is an existing problem in the cloud services area despite the technical changes that this type of services are going through.

As part of our proposal focus on the ontological acts (creation of new original facts) and ignores the implementation details, it was not surprising to find that applying Enterprise Ontology-based SLAs to model cloud services is not so different from modelling business services as we have done in previous field studies. The feedback from CSP employees was positive and they recognized value in the proposal and in the role that it fulfilled in maturing their services specification. For example, our proposal allowed the service provider to find some service attributes that had not been defined. Additionally, the CSP employees also contributed to the redesign of the SLAs proposal. Their feedback was that the service configuration was more related with the protocol and location attributes than with the SLA targets. Therefore, despite of being related the service configuration and the SLAs should be modelled as different concepts. So, we updated our proposal and designed a new version for the Generic Service Specification Framework (GSSF) and redesigned the SLAs structure. We removed the service configuration attribute from the SLA proposal and we added a new service configuration area to the GSSF with the following attributes: protocol, location, options and price.

Another change to our SLA structure came from the fact that the SLA owner was always the same as the service owner (also in line with the previous two field studies). Consequently, we decided to remove the SLA owner and the respective contacts information from the SLA structure proposal and we assume that the accountable for a service (defined in the GSSF as the Service Owner) is also accountable for the SLAs of that particular service.

Therefore, this field study **contributed** to answer the research question **Q2 (how to specify service level agreements according to customers' expectations?).**

10 Closing the Gaps in Wines4All

This field study focused on the service exchange between the European private company from the second field study (Wines4All) and the same external IT service provider. As mentioned before, Wines4All is a European private company that is leader in the wines and spiritual beverages distribution and hired an IT service provider to develop computer applications. Wines4All employs about 80 people that can request the IT provider services using a web application. These requests are handled by two developers.

10.1 Problem

This field study focused on the research question Q3 (how to deliver services according to the service specifications and customers' expectations?), namely we concentrated on validating **Hypothesis H3.1 (dynamically defined SLAs diminish the gap between customers' expectations and perceived service).**

10.2 Proposal

In this field study we applied all the steps of our proposal, i.e.:

1. Identify the services:
   1.1. Enterprise Description;
   1.2. Performa-Informa-Forma Analysis;
   1.3. Coordination-Actors-Production Analysis;
   1.4. Transaction Pattern Synthesis;
   1.5. Result Structure Analysis;
   1.6. Actor Transaction Diagram/Service Identification.
2. Specify the services using the new version of the GSSF (Figure 24 – page 56);

3. Specify the SLAs for each identified service using the structure of Figure 25 (page 57);
4. Fulfil service requests by implementing the proposal ontology of Section 4.3 (page 59).

This proposal consolidates the findings from the previous field studies and includes all the steps of the proposed method.

10.3 Demonstration

In order to identify the services (**step one** of our proposal), we interviewed an IT service provider developer and three employees from Wines4All individually (we chose the employees with more service requests at the time). The participants were asked to describe the activities performed by the IT service provider. The interviews were recorded and transcribed as well as checked and discussed by two interviewers, each ensuring unbiased findings and avoiding misinterpretation as specified in (Kvale, 2007).

We merged the interviews results into a single text and we validated it with the developer and the three interviewed users (**sub-step 1**). This text was used as input for the remaining sub-steps of the service identification step.

Then, we distinguished the Ontological, Infological and Datalogical actions described, as was referred in the Enterprise Ontology Distinction axiom. This step is called Performa-Informa-Forma Analysis (**sub-step 2**). To do that, we defined a notation to differentiate those actions: in this example, we have highlighted the text, using red, green and blue colours to identify the Ontological, Infological and Datalogical actions, respectively.

The **third sub-step** concerns the identification of C-acts/facts, P-acts/facts and actor roles, using the Performa (Ontological) items identified in the previous step. We have also considered a notation to differentiate between them: square brackets "[" and "]" to identify actor roles; brackets "(" and ")" to identify C-acts/facts; and angled brackets "<" and ">" to identify P-acts/facts. This step is called Coordination-Actors-Production Analysis and here, in comparison to other methodologies, there is a reduction of the complexity because, from now on, we will only consider the Ontological actions identified in this step.

After these analyses, we defined the existing transactions in the text by clustering the identified C-acts/facts and P-acts/facts in what is referred to as Transaction Pattern Synthesis (**sub-step 4**). The Transaction axiom can be helpful in this step because it guarantees that each previously found P-act/fact or C-act/fact corresponds to a complete transaction. Then, for each identified transaction type, the result type (i.e., the P-fact created) was formulated. The result is called Transaction Result Table (TRT) and is represented in Table 33.

Transaction Types	Result Types
T01 – Feature Development	R01 – Feature F has been developed
T02 – Application Integration	R02 – Integration I implemented
T03 – Testing	R03 – Feature F has been tested
T04 – Training	R04 – Training T has been given
T05 – Meeting	R05 – Meeting M occurred

Table 33 – Transaction Result Table (TRT) of the IT service provider

After having defined the transaction types and the respective result types, we must check if there are any dependencies between the transactions/P-facts (results), as described in the Composition axiom. This step is called Result Structure Analysis (**sub-step 5**) and the practical way to perform this step is to read the description of a case once more and to look for phrases in which dependencies between production acts or results are expressed. In this case, two dependencies were identified. The result 'Feature F has been developed' depends on the results 'Integration I implemented' and 'Feature F has been tested'. The remaining results do not have dependencies. The Result Structure Chart illustrates these dependencies (Figure 49).

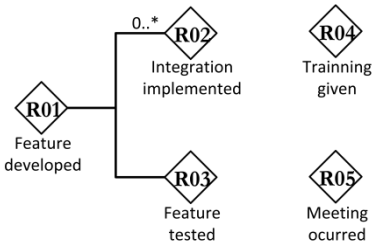


Figure 49 – Result Structure Chart

After identifying the Ontological transaction types, its results and dependencies, we must determine the environment surrounding the software developer. The first model to be developed is called the Actor Transaction Diagram (ATD), represented in Figure 50. In this type of diagrams, a transaction is represented using a diamond in a disk that contains the respective combination of C-acts and a P-act. Each transaction is connected to two boxes, representing the initiator and executor actor roles. The initiator is connected to the transaction symbol using a solid line, while the executor is connected to the transaction using a solid line ending in a black square. These gray boxes refer to composite actor roles, i.e. elements whose exact structure is not known. All the environmental elements (i.e. elements outside the provider we are studying) are represented in gray boxes for that reason.

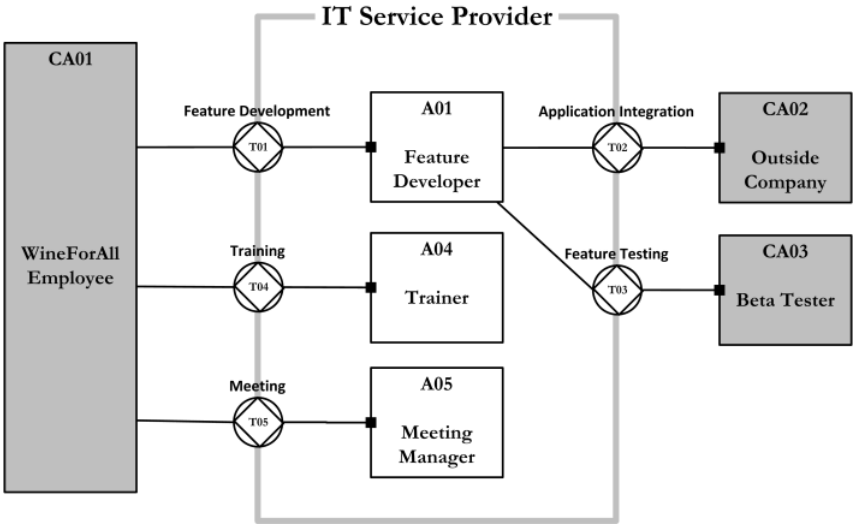


Figure 50 – Actor Transaction Diagram (ATD) of the IT service provider

After defining the transactions between the service provider and its environment, we can identify the services that the developer provides. By using the already stated definition of service, based on the Transaction Pattern proposed by Dietz (Dietz J. , 2006), we conclude that the services the service provider delivers partially correspond to the transactions in which the service provider is the executor. This only happens with three transactions: T01 – Feature Development, T04 – Training and T05 – Meeting.

In order to proceed to the proposal's **second step** (specify services), we had to model the Process Model, the Action Model, and the State Model of CSP, since some aspects of the GSSF (used in the second step of the proposal) depend on these models. For the sake of readability, we only present the models used in the service specified below.

In the **second step** of the proposal we applied the new version of the Generic Service Specification Framework (GSSF) to specify the services of Figure 50. An example of this specification is illustrated in Table 34 that describes the attributes of the service Feature Development that implements the T01 transaction.

Concerning the service basic information area, this service name is Feature Development and has not defined a base price. Regarding the service executor area, this service is executed by the actor role Feature Developer (A01), which is fulfilled by an IT service provider employee. An email and a phone number are provided to the customers.

Service Specification – *Feature Development (T01)*	
Service Basic Information	
Name	Feature Development
Base Price	Not Defined
Service Executor	
Actor Role	*Feature Developer (A01)*
Contact Information	General Email: ----- Email: ----- Phone: -----
Service Production	
Production Act	A feature has been developed by the IT Service provider according to the needs of the Wines4All employee
Production Information Used	The object classes used in this services are: Feature and Application
Production Fact	The production fact can be derived from the Transaction Result Table (Table 33): Feature F has been developed
Production Kind	The production kind of this transaction is: Ontological
Production World Semantics	See Object Fact Diagram (Figure 51)
Preconditions	Pre and post-conditions are gained from the Action Model. As precondition we have: **when** development of [Feature] is promised **if** <needs application integration> **then** implementation of [Integration] must be requested **else** development of [Feature] must be executed **and** test of [Feature] must be requested
Postconditions	The postcondition is expressed in the same action rule of the precondition
Service Coordination	
Coordination Acts	See Figure 52
Coordination Kind	This service is a Human Service
Service Configuration	
Protocol	1. Access a web-based application 2. Choose the service options 3. Submit the request
Location	Web-based application
Options	Not Defined
Price	Not Defined

Table 34 - Service "Feature Development" Specification

In the service production area, a description of the action performed by the service provider is given (production act attribute), the categories used in this act are listed (production information used) and the resulting production fact is described (Feature F has been developed). As this pro-

duction fact is an original one (this fact cannot be derived from previous facts), then the service can be defined as ontological (see Section 2.2 – page 18). The production world semantics attribute can be obtained from the Object Fact Diagram (OFD) illustrated in Figure 51. The OFD is based on the language WOSL (Dietz J. , 2005) and shows the categories (external in gray), the relations between categories, and the transactions' results (diamonds).

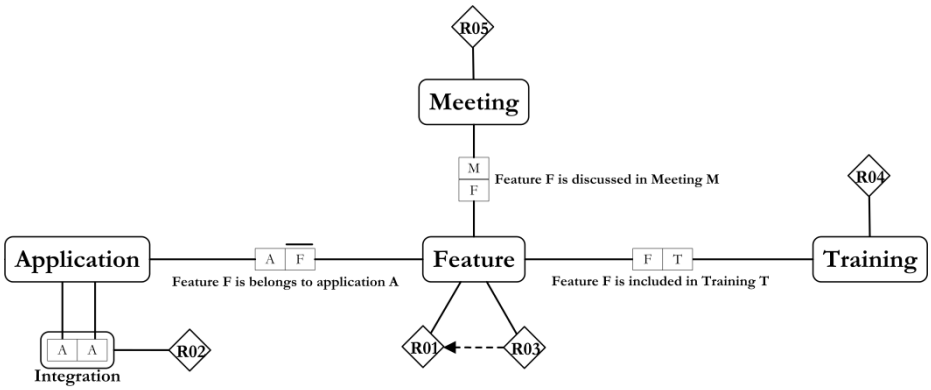


Figure 51 – Object Fact Diagram (OFD) of the service provider

The internal categories are Feature, Application, Integration, Meeting and Training. The Feature category represents the features that the Wines4All employees request to the developer of the IT Service provider. An Application can have several features, but a feature can only be associated with one application. The Integration category is extensionally defined by the relation between two Applications. In a Meeting several Features can be discussed. Likewise a Feature can be discussed in several Meetings. The same coexistence rule applies for Features and Trainings.

The transaction events are associated to the categories as follows: results R01 and R03 are associated to the Feature category, result R02 with the Integration category, result R04 with the Training category and result R05 is associated with the Meeting category. Result R03 is dependent of result R01 since a Feature can only be tested if it has been developed.

Regarding the preconditions, this service may need the execution of the service T02 (Application Integration). Concerning the postcondition, the feature developed must be always tested, i.e. T03 must be executed.

The service coordination area is composed by the coordination acts and the coordination kind. These attributes can be obtained from the Process Structure Diagram (PSD) that is part of the Process Model (Dietz J. , 2006). The PSD represents the process that must be carried out in order to fulfil the service. Figure 52 represents an example of such a process.

The process starts by the Wines4All employee's (CA01) request (T01 rq) of transaction T01 (Feature Development). An IT service provider employee (fulfilling actor role A01) promises to deliver this service (T01 pm) and, if needed, requests (T02 rq) the execution of the service T02 (Application Integration). Then, the actor role CA02 promises (T02 pm) to integrate the application, executes the promise and informs (T02 st) the requester (A01).

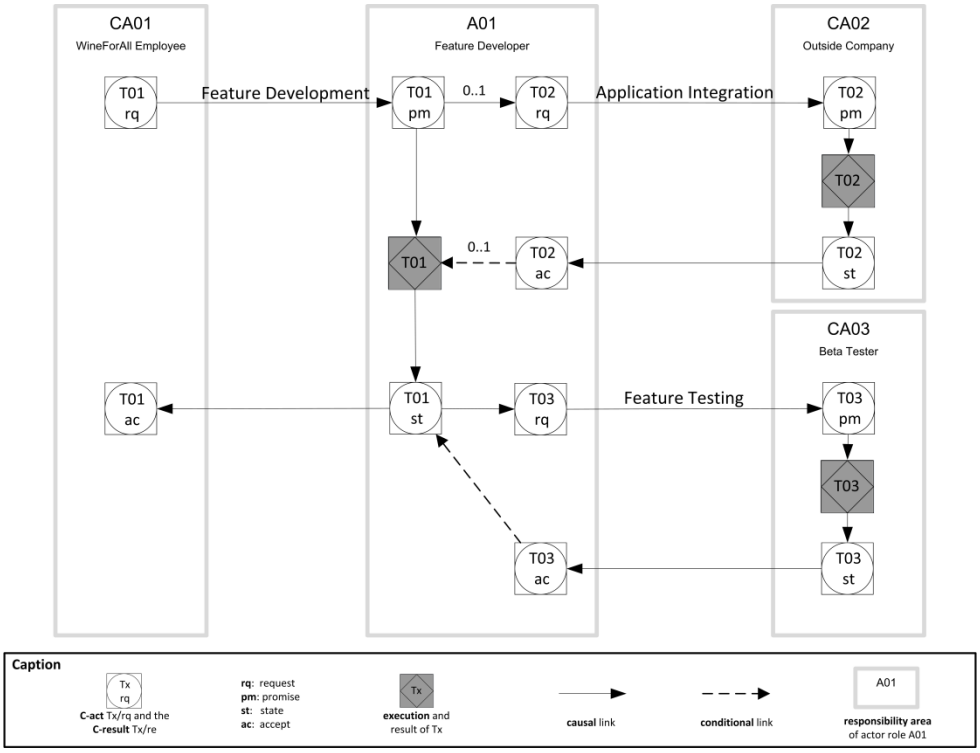


Figure 52 –Process Structure Diagram (PSD) and caption

Afterwards, the actor role A01 accepts the result of T02 (T02 ac), and develops the feature (T01 execution), but before he can inform the customer (T01 st) about the execution, he must first request the beta tester (CA03) to assess the feature (T03 rq). Afterwards, the beta tester (CA03) promises to test the feature (T03 pm) and, once that has been done (T03 ex), CA03 informs the developer (T03 st). Then, the developer (A01) can accept the feature test (T03 ac) and inform the customer about the developed feature (T01 st). At last, the customer (CA01) can accept the feature development.

Finally, the service configuration area specifies the protocol, location, options and price. The protocol that the Wines4All employees must follow is to contact the IT service provider through a web-based application (location of the service). In this application the employees can choose the service options and submit their requests. The service configuration options and the respective prices are not defined.

In the **third step** of the proposal we specified the SLAs associated to the IT provider services using our SLA proposal (Figure 25). Table 35 illustrates an example of this specification for the service 'Feature Development'.

Service Level Agreement Specification	
SLA Basic Information	
Name	Feature Development SLA
Service	*Feature Development (T01)*
SLA Responsibility Information	
Customer Responsibilities	1 – Inform the provider about his needs;
Provider Responsibilities	1 – Develop the feature with the requested options; 2 – Measure SLAs targets fulfilment.
SLA Specific Information	
Type 1	**Critical**
Targets	
Performance	Response Time – 2 hours
Performance	Resolution Time – 4 hours
Penalties	Not Defined
Bonuses	Not Defined
Price	Not Defined
Type 2	**Urgent**
Targets	
Performance	Response Time – 4 hours
Performance	Resolution Time – 12 hours
Penalties	Not Defined
Bonuses	Not Defined
Price	Not Defined
Type 3	**Normal**
Targets	
Performance	Response Time – 8 hours
Performance	Resolution Time – 24 hours
Penalties	Not Defined
Bonuses	Not Defined
Price	Not Defined

Type 4	Low
Targets	
Performance	Response Time – 24 hours
Performance	Resolution Time – 48 hours
Penalties	Not Defined
Bonuses	Not Defined
Price	Not Defined

Table 35 – SLA attributes for the service 'Feature Development'

As stated before, our SLA proposal is composed by three areas: SLA Basic Information, SLA Responsibility Information and SLA Specific Information. In the first one, we defined the name of the SLA (Feature Development SLA) and the service (Feature Development - T01). In the SLA Responsibility Information area we defined both the customers' and the provider's responsibilities. Finally, in the third area, we specified the quality options of the service that correspond to the four SLAs types: Critical, Urgent, Normal and Low. When requesting a feature development (T01), the customers can choose one of these SLAs types. Each type has two performance targets: response and resolution time. The first one concerns the time from the request coordination act to the promise/decline coordination act and the second one the time from the promise coordination act to the state coordination act. This service does not have defined bonuses, penalties or price.

In order to fulfil the **fourth step** of the proposal we implemented part of the actor roles and transactions included in the Actor Transaction Diagram proposal (Figure 28). In other words, we integrated the proposed actor roles and transactions for the dynamic SLAs negotiation (Figure 28) with the actor roles and transactions that we identified in the first step of the proposal (Figure 50).

Figure 53 highlights the actor roles and transactions from the proposed ATD for the dynamic SLA negotiation that were implemented in this field study. The Wines4All employees implemented the actor role CA01 – customer. They had the responsibility of requesting the service fulfilment executions that in this case correspond to the three transactions that the employees from the IT provider execute: Feature Development, Training and Meeting. To do so, the Wines4All employees used a web application (developed the IT service provider) that simplified the SLA negotiation. Basically, besides the SLAs defined in Table 35, when requesting the services the employees could freely choose the date on which they needed the service provider to respond (response time target) and also the date they needed the service to be provided (reso-

lution time target). This way the Wines4All employees could request services according to their expectations at that moment.

One IT provider dev\eloper implemented the following actor roles: A01 – service executer, A05 – SLA starter, A06 – SLA approval, and A07 – SLA ender. As A01 the developer was accountable for executing the Feature Development, Training and Meeting services. As A05 the developer had to create new SLAs when the combination of response time and resolution time was not already created. As A06 the developer had to approve new SLAs and as A07 the developer had to end SLAs.

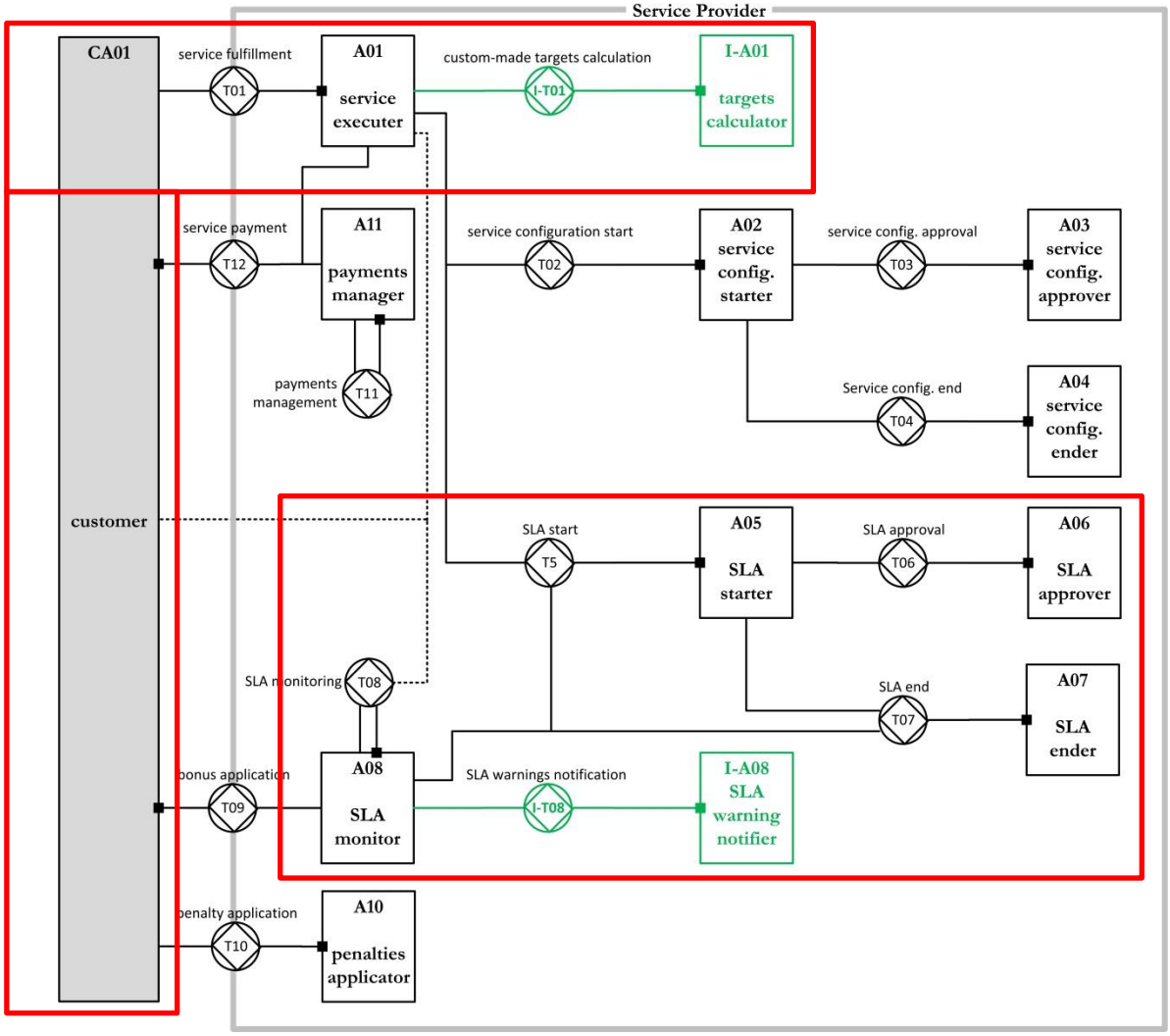


Figure 53 – Evaluated transactions and actors from the proposed Actor Transaction Diagram (Figure 28)

The following actor roles were implemented by the web application that was used to manage the services fulfilment: A08 – SLA monitor, I-A08 SLA warning notifier, and I-A01 SLA tar-

gets calculator. The web application implemented the A08 actor role by doing a daily monitoring of the SLAs fulfilment. When an SLA was close to be broken, then the application also had the responsibility of warning the developer about the fact, thus implementing the actor role I-A08. Additionally, the application suggested customized SLAs according to the user (I-A01). When the users were requesting services, the application suggested response and resolution time targets using the average of the previous requests of that specific user and chosen SLA Type (critical, urgent, normal and low). This feature contributed to a more customized negotiation of SLAs. The remaining actor roles and transactions were not implemented since they were not required for the Wines4All context.

We have monitored 360 service requests from 25 employees in which the SLAs were dynamically negotiated. From these, 24 were not fulfilled: in 18 the response time target was not achieved, in 11 the resolution time target was not achieved and in 5 both the targets were not achieved.

10.4 Evaluation

This Section corresponds to the evaluation phase of DSRM and in order to explain the evaluation, we use the framework proposed in (Pries-Heje, et al., 2004). This framework identifies what is actually evaluated, how it is evaluated and when the evaluation takes place:

- **What is actually evaluated?** The artifact evaluated is the proposed set of steps (a design process);
- **How is it evaluated?** We used a four step process based on SERVQUAL to evaluate the artifact. This represents a naturalistic evaluation since it was conducted in a real organization facing real problems;
- **When was it evaluated?** It was evaluated ex post (after the design artifact was developed).

Our evaluation process was composed by the following steps:

1. Measure Gap 5 using SERVQUAL (Parasuraman, et al., 1988);
2. Apply the proposal;
3. Measure Gap 5 using SERVQUAL;
4. Compare results.

We measured Gap 5, i.e. the difference between the customers' expectations and perceptions about the IT provider's services, before we have applied the proposal. For this purpose, we used SERVQUAL (Parasuraman, et al., 1988), a 22-item instrument for assessing customer percep-

tions of service quality. We decided to use SERVQUAL because it is the reference on measuring customer perceptions of service quality (Carrillat, et al., 2007). SERVQUAL is a questionnaire with 22 questions and allows us to measure five dimensions of service quality:

- **Reliability** – ability to perform the promised service dependably and accurately;
- **Assurance** – knowledge and courtesy of employees and their ability to inspire trust and confidence;
- **Tangibles** – appearance of physical facilities, equipment, personnel, and communication materials;
- **Empathy** – caring, individualized attention the firm provides its customers;
- **Responsiveness** – willingness to help customers and provide prompt service.

In each question, the Wines4All employees were asked to rank from 1 to 7 their minimum expectation (ME), desired expectation (DE) and perceived service (P) regarding the IT provider services. This way, expectations were measured as a range as opposed to a single, scaled point. The range between employees' minimum expectations and desired expectations constitutes what is known as the "zone of tolerance". This range and the perceived quality allowed us to calculate other indicators:

- **Service Adequacy Gap Score (P-ME) –** This score indicates the degree to which end users basic, minimum service levels are being met. This score is computed by subtracting the minimum expectation (ME) from the perceived service (P). A positive number indicates the extent that perceived service levels exceeds end users minimum expectations whereas a negative number indicates a gap between the perceived performance and minimum expectations.
- **Service Superiority Gap Score (P-DE)** – This score indicates the degree to which end users desired service levels are being met. This score is computed by subtracting the desired expectation (DE) from the perceived service (P). A positive number indicates the extent that perceived service exceeds end users desired expectations while a negative number indicates a gap between perceived service performance and end users desired expectations.

We have collected feedback from 16 employees that were arbitrarily chosen. Figure 54 illustrates the results of the first SERVQUAL questionnaires. In the columns we have the five dimensions of SERVQUAL (tangibles, reliability, responsiveness, assurance, and empathy) and respective average. The grey rectangles represent the zone of tolerance between the Wines4All employees' minimum and desired expectations regarding the IT provider services. The little

triangles represent the Wines4All employees' perceptions of the service delivery of the same services.

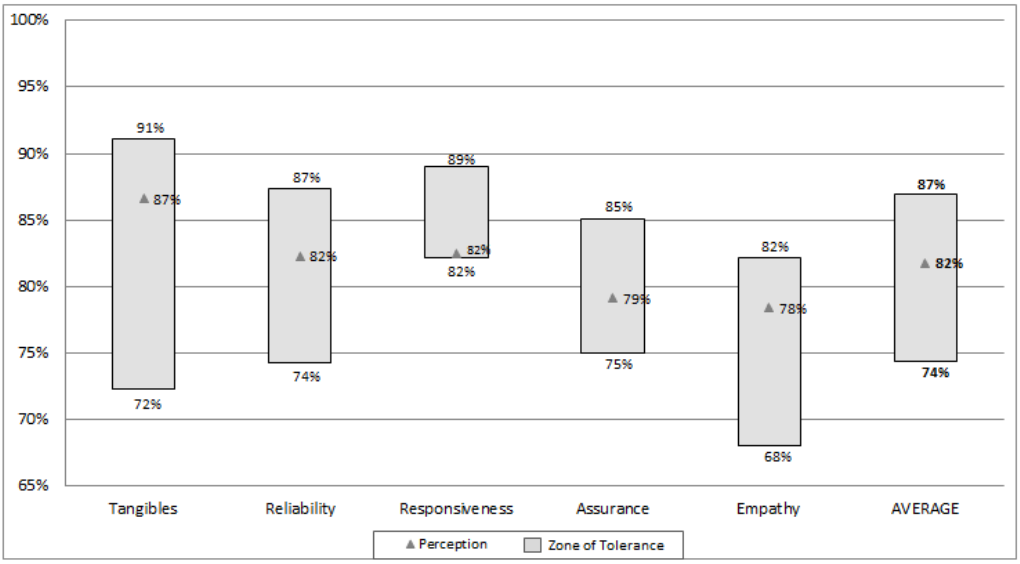


Figure 54 – Initial SERVQUAL results (before the proposal implementation)

In all dimensions the perceived service quality was among the minimum and desired expected service, i.e. the minimum expectations were fulfilled, but the desired service was not satisfied in any dimension. Therefore, there was a gap between customers' expectations and perceptions with an average of -5,14% and the dimension with worst result was the responsiveness with a gap of -6,55% (Table 36). The remaining differences were: -5,95% in the assurance dimension, -5,00% in the reliability dimension, -4,46% in the tangibles dimension, and -3,75% in the empathy dimension.

Then, we implemented the proposal steps, i.e. we found the services of the IT service provider, applied the new version of the Generic Service Specification Framework and specified the SLAs for each identified service. Afterwards, the employees from Wines4All requested services using the SLAs defined at design time or, if needed, requested services with SLAs defined dynamically.

After 6 months of service exchange using the SLA proposal, we measured Gap 5 again using SERVQUAL. This way, we found new information about the difference between the customers' expectations and perceptions (Figure 55).

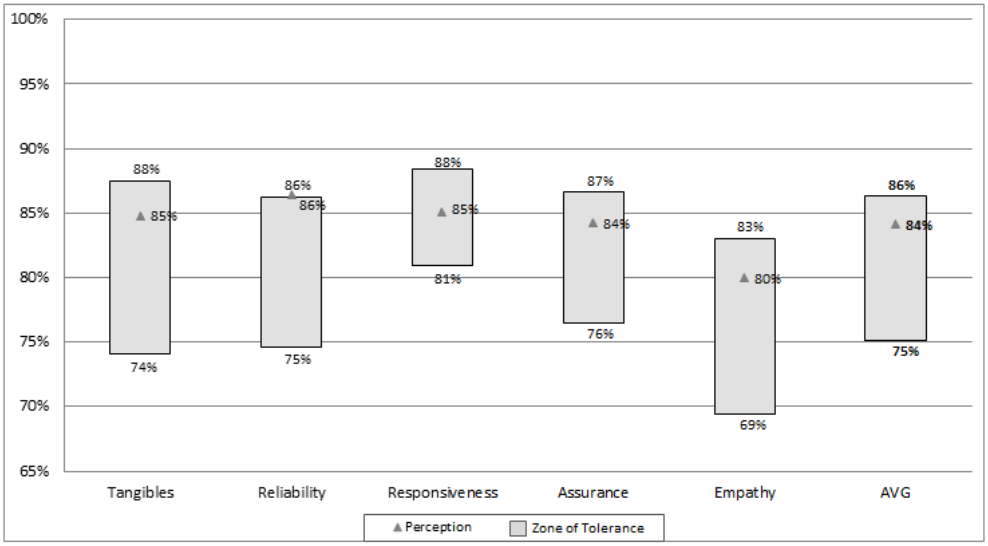


Figure 55 – Final SERVQUAL results (after the proposal implementation)

This time, the minimum service was fulfilled in all dimensions and the desired service was fulfilled in one dimension: reliability. Still, in the other four dimensions the desired service was not fulfilled. Table 36 compares the results of the two SERVQUAL questionnaires.

	Minimum Expectation (ME)		**Desired Expectation (DE)**		**Perception (P)**		**GAP 5 (P-ME)**		**GAP 5 (P-DE)**	
	1st	*2nd*	*1st*	*2nd*	*1st*	*2nd*	*1st*	*2nd*	*1st*	*2nd*
Tangibles	72,32%	74,11%	91,07%	87,50%	86,61%	84,82%	**14,29%**	**10,71%**	**-4,46%**	**-2,68%**
Reliability	74,29%	74,64%	87,32%	86,25%	82,32%	86,43%	**8,04%**	**11,79%**	**-5,00%**	**0,18%**
Responsiveness	82,14%	80,95%	88,99%	88,39%	82,44%	85,12%	**0,30%**	**4,17%**	**-6,55%**	**-3,27%**
Assurance	75,00%	76,49%	85,12%	86,61%	79,17%	84,23%	**4,17%**	**7,74%**	**-5,95%**	**-2,38%**
Empathy	68,04%	69,46%	82,14%	83,04%	78,39%	80,00%	**10,36%**	**10,54%**	**-3,75%**	**-3,04%**
Average	***74,36%***	***75,13%***	***86,93%***	***86,36%***	***81,79%***	***84,12%***	***7,43%***	***8,99%***	***-5,14%***	***-2,24%***

Table 36 – SERVQUAL results (before and after the proposal implementation)

With both results, we can now compare and conclude if the evolution of Gap 5 was for the better or not. The minimum expectation average increased 0,77% (from 74,36% to 75,13%) due to the increase of 1,79% in the tangibles dimension, 0,36% in the reliability dimension, 1,49% in the assurance dimension, 1,43% in the empathy dimension, and the decrease of 1,19% in the responsiveness dimension. However, the desired expectation decreased in 0,57% as a result of the decrease of 3,57% in the tangibles dimension, 1,07% in the reliability dimension, 0,60% in the responsiveness dimension, and the increase of 1,49% in the assurance dimension and 0,89% in the empathy dimension. The perception evaluated positively in 2,33% due to the increases of

5,06% in the assurance dimension, 4,11% in the reliability dimension, 2,68% in the responsiveness dimension and 1,61% in the empathy dimension, and the decrease of 1,79% in the tangibles dimension.

We conclude that there was a reduction of Gap 5 (difference between customers' expectations and perceived service) in all the dimensions. The reduction was more significant in the reliability, assurance and responsiveness dimensions with an improvement of respectively 5,18%, 3,57%, and 3,27%. The tangibles dimension had an improvement of 1,79% and the empathy dimension had an improvement of 0,71%.

These improvements resulted in an overall decrease of 2,90% in Gap 5, meaning that Gap 5 was still present with a value of -2,24%.

10.5 Lessons Learned

Overall, the Four Principles from (Österle, et al., 2011) were accomplished:

- **Abstraction**: the artifact can be applied to any service provider from any given enterprise description. As the proposal provides abstract models that focus on the business layer, not considering the implementation details, it can be applied to organizations in different contexts. In fact, previously we evaluated part of this proposal in several organizations, from public city councils (Mendes, et al., 2012) to private banks (Mendes & Mira da Silva, 2012);
- **Originality**: the proposed artifact is not present in the body of knowledge of the domain since it was designed by relating independent subjects, such as service level management and DEMO;
- **Justification**: the artifact is supported by the related work, described by textual and graphical representations, and justified and validated in different ways;
- **Benefit**: the artifact provides a structured working approach for specifying customers' expectations. By specifying the proposal's attributes, customers can structurally define their expectations which may help the alignment between customers and service providers. The specification of the customers' expectations into SLAs helps the service providers understand those expectations and consequently reduce the gaps between the two. Besides, the dynamic SLAs helped the IT service provider to improve the reliability, responsiveness and assurance of their services. In other words, the ability to dependably and accurately perform the promised service, the willingness to help customers and provide a prompt service, and the knowledge and courtesy of employees and their ability to inspire

trust and confidence were improved. The tangibles and empathy did not suffer significant differences, since the performance SLAs do not relate to these dimensions.

The proposal showed to be fruitful in three of the five dimensions for service quality proposed in SERVQUAL. The reliability dimension had an improvement of 5,18% that allowed to close completely the Gap 5 in this dimension. Therefore, the customers' expectations were fulfilled in the reliability dimension. The assurance dimension had an improvement of 3,57%, and the responsiveness dimension had an improvement of 3,27%.

These improvements translated into a reduction of 2,90% in the Gap 5, i.e. the difference between customers' expectations and perceived service was reduced in 2,90% and ended up being -2,24%. Therefore, this field study **indicates** that **dynamic SLAs have a positive impact in the gaps,** thus **contributing to validate Hypothesis H3.1** (dynamically defined SLAs diminish the gap between customers' expectations and perceived service)**.**

Although these results we also found some limitations. In the first place, the resistance of some users in specifying their expectations, since they preferred not to do it and risk being surprised by the provider in the service delivery. In order to fight this limitation, we tried to enhance the interface of the web application that allows employees to request services and specify their expectations. One example of this effort is the customized suggestion of SLAs by the application described before.

Another limitation is the fact that we assume that the service provider must know if he can provide the service with the expected service quality or not. How this is done is out of this research context. We do not focus on how the service provider should be implemented in order to satisfy the SLAs. We just provide a way of dynamically specifying the customers' expectations in SLAs and we assume that the service provider has the necessary knowledge to know if he can comply with the negotiated services quality.

11 Conclusion

In this research we propose a method to close the gaps that influence the services quality. There are several solutions that contribute to closing the gaps, but none solved the problem completely. Some lack detail in specifying the services quality (such as the Generic Service Specification Framework), others are not based on a strong conceptual foundation (such as ITIL or CMMI) and some are only applicable to a specific type of services such as the majority of the web services based solutions.

In this thesis we present a proposal to solve the gaps problem based on a methodology (DEMO) with a strong theoretical background, thus using the existing knowledge in an innovative way (one of the design-science research objectives). This proposal was evaluated in six field studies conducted in four different organizations. The compatibility of these field studies with the thesis proposal does not indicate that the proposal is fully complete; it only indicates that the proposal is compatible with the realities of the given cases. However, the incremental results that were obtained in each field study give a qualitative sense of confidence towards the usefulness of the solution.

The expectations regarding the performance of a provider are dynamic since they change according to each customer and even each situation. Even the same customer may have different expectations depending on previous experience. The uniqueness of each situation and of each customer is one of the more challenging aspects of quality service (Ukens, 2007). During this research we realized that the thesis problem (gap between customers' expectations and perceived service) requires a continuous follow-up in order to be minimized. Even if a provider has solved the gap in a certain period, this is not a warranty that the gap will stay in that way. The close up between customers' expectations and perceived service requires an endless monitoring and acting.

11.1 Main Contributions

First, we mapped the gaps model to the EO basic transaction pattern (Figure 13), showing that, by itself, EO cannot answer the research problem. Then, we proposed a service identification

process based on the DEMO methodology that was evaluated in two field studies. The first field study revealed that both client and provider should be included in the service identification process (validating Hypothesis H1.2) and in the second first field the new services list identified by the proposal was more accepted by the customers than the older services list (contribution to answer the research question Q1 and validating Hypothesis H1.1).

Nevertheless, the customers' expectations cannot be specified only using information about the services, so we also proposed an Enterprise Ontology-based SLA definition (validating Hypothesis H2.1) and showed how to use Enterprise Ontology for specifying SLAs in three field studies (contribution to answer the research question Q2 and Hypothesis H2.2). These field studies validated the EO transactions patterns potential to define performance targets of the SLAs.

Additionally, we evaluated the dynamic SLAs negotiation in a field study (contribution to answer the research question Q3 and Hypothesis H3.1). In this field study, the overall gap between customers' expectations and perceptions decreased by 2,90%, result of a significant improvement in three dimensions. The reliability dimension had an improvement of 5,18%, the assurance dimension had an improvement of 3,57%, and the responsiveness had an improvement of 3,27%.

Another contribution is the proposal ontology described in Section 4.3 (page 59). This proposal represents a practical guide for anyone who desires to implement the dynamic SLA negotiation, since it describes the actors, the transactions, the business process and rules, and the entities that compose this part of the proposal.

These contributions are summarized in nine papers already published or accepted for publication:

- Mendes, C., Mira da Silva, M., Implementing the Service Catalogue Management, 7th International Conference on the Quality of Information and Communications Technology (QUATIC), pp.159-164, IEEE Computer Society, 2010
- Mendes, C., Mira da Silva, M., Implementing a Request Fulfilment Process, 2nd International Conference on Exploring Services Sciences (IESS), Lecture Notes in Business Information Processing, Vol. 82, pp. 113-126, Springer, 2011
- Mendes, C., Ferreira, J., Mira da Silva, M., Comparing Service Using DEMO, 3rd International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K) - Special Session on Enterprise Ontology, Paris, 2011 (Best Student Paper Award)

- Mendes, C., Ferreira, J., Mira da Silva, M., Identifying Services from a Service Provider and Customer Perspectives, Lecture Notes in Communications in Computer and Information Science, Springer, 2012
- Mendes, C., Mira da Silva, M., DEMO-based Service Level Agreements, 3rd International Conference on Exploring Services Sciences (IESS), Lecture Notes in Business Information Processing, Springer, 2012
- Mendes, C., Ferreira, J., Mira da Silva, M., Using DEMO to Identify Services, 8th International Conference on the Quality of Information and Communications Technology (QUATIC), IEEE Computer Society, 2012
- Mendes, C., Almeida, M., Salvador, N., Mira da Silva, M., Using Enterprise Ontology-based SLAs for Improving City Council Services, 4rd International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K) - Special Session on Enterprise Ontology, Barcelona, 2012 (Best Paper Award)
- Almeida, M., Mendes, C., Mira da Silva, M., Extended Enterprise Ontology-based SLAs to specify customers' expectations, 4th International Conference on Exploring Service Science (IESS), Porto, 2013
- Mendes, C., Almeida, M., Salvador, N., Mira da Silva, M., Modelling Services Using DEMO, Lecture Notes in Communications in Computer and Information Science, Springer, 2013

Additionally, two more papers were submitted and are waiting for a decision:

- Mendes, C., Almeida, M., Mira da Silva, M., Applying DEMO-based SLAs to Cloud Services, Journal of Service Science Research
- Mendes, C., Mira da Silva, M., Closing service quality gaps using DEMO-based SLAs, Journal of Information Systems Research

These papers contribute for the last phase of DSRM (communication phase).

11.2 Limitations

The fact that we did apply our proposal in real organizations brings some disadvantages. As the environment is not completely controlled, then some unpredictable factors may influence the field study results. For instance, in the last field study (Chapter 10) we were unable to verify what other variables may have influenced the gaps. In order to mitigate those variables, we tried to maintain the same context during the proposal demonstration, so the proposal would be the

only relevant difference. To accomplish that, the service provider employees, the customers and the nature of the service requests were maintained during the proposal demonstration.

Additionally, in a natural context the researchers cannot interview as they please. For example, in the field study carried out in the Cloud Service Provider (Chapter 9) we did not have access to the company customers.

Another limitation is the fact that not all the service providers are willing to open their SLA negotiation. Some argue that by letting their customers freely express their expectations they will in one hand lose control of what they do, and, on other hand, the customers will also request services with the maximum quality and the minimum price. In the field study that we described in Chapter 10, we found that these arguments did not apply. First, the capacity of the organization to fulfil requests is the same, what changes is the way that capacity is used. With the dynamic SLA negotiation, the provider capacity is deployed considering the customers' expectations, and therefore, more customer-oriented.

In another perspective, dynamic SLA negotiation is a negotiation. It is not only the customer that decides what is done and when it is done. The provider must have the knowledge to decide if can comply with the requested service quality or not. As mentioned before, the way this is done is out of this research context.

We do not focus on how the service provider should be implemented in order to satisfy the SLAs. We just provide a way to dynamically specify the customers' expectations in SLAs and we assume that the provider has the necessary knowledge to know if the provider can comply with the negotiated services quality. Besides, if the providers let the customer choose always the maximum quality and the minimum price, then eventually the provider will start to break their promises which will impact negatively the customers' expectations, and ultimately will result in a reduction of the requested service quality.

Still regarding the dynamic SLA negotiation, another limitation is the time the provider needs to negotiate the SLAs. In the field study described in Chapter 10, we evaluated the dynamic SLA negotiation in a small scale: during six months a developer provided services to 25 employees. In that context, the effort required by the developer to dynamically negotiate SLAs was similar to the effort that was needed to manage the requests that used static SLAs. This is explained by the fact that the majority of the actor roles of the dynamic SLA negotiation proposal (see Figure 28 – page 64) were implemented by a single person (the developer). Therefore, there was no waiting time between the transaction executions. In a more complex context, this would not be the case and, consequently, it would need to be evaluated what would be the impact of dynamic SLA negotiation is such cases.

11.3 Future Research

As future work, Research Question Q4 (how to align the two universes of discourse in such a way that the perceived service quality is aligned with the service delivery?) could be addressed. Currently, a master student is dealing with this research question and he is designing a system that registers all the coordination acts involved in the service exchange. This system is based on this thesis' artifacts.

Also related with Research Question Q4 is the role of semiotics in the Gap 4 (communication about the service does not match the actual service delivery). Semiotics is the study of signs and sign processes, indication, designation, likeness, analogy, metaphor, symbolism, signification, and communication. Semiotics is closely related to the field of linguistics, which, for its part, studies the structure and meaning of language more specifically. Semiotics could bring contributions in three perspectives: semantically, syntactically and pragmatically. The first perspective concerns the relation between signs and the things to which they refer (meaning). The second perspective describes the relations among signs in formal structures, and the third perspective concerns the relation between signs and the effects they have on the people who use them.

Another possible future research is the integration of the SLAs concepts directly into the DEMO models, for instance to enrich the ATD with an SLA symbol. Another model that would benefit from the SLAs concepts would be the Action Model because, in the action rules specification, it is not always possible to specify a rule (completely) formally. In such cases, informal expressions are allowed and they are usually put between the angled brackets < and >. It would be interesting to replace those informal expressions by conditions that mentioned the performance of the SLAs.

Another possible line of research is the integration of this thesis proposal with the GOD-theory research (Aveiro, 2009). Our dynamic SLAs negotiation solution could provide input that would imply the change of the construction of the organization and not only the change of its Service Configurations and SLAs.

Finally, another possible future research is the development of different viewpoints for the same service catalogue design. In this thesis, we focused on the attributes that relate to the customers' expectations due to the problem that we have addressed. However, one could also focus on the attributes that may concern only the provider and therefore develop a different viewpoint of the service catalogue. In this line of research, a possible contributor could be the ISO/IEC/IEEE 42010 that is an international standard for architecture descriptions of systems and software (International Organization for Standardization, 2007). ISO 42010 de-

fines requirements on the description of system, software and enterprise architectures and includes the possibility of defining different viewpoints for the same system, software, or architecture.

Bibliography

Albani, A., & Dietz, J. (2011). *Enterprise ontology based development of information systems.* International Journal of Internet and Enterprise Management, Vol.7, No.1, pp.41-63.

Albani, A., Terlouw, L., Hardjosumarto, G., & Dietz, J. (2009). *Enterprise Ontology Based Service Definition.* Amsterdam, The Netherlands: 4th International Workshop on Value Modeling and Business Ontologies.

Anderson, E., & Sullivan, M. (1993). *The Antecedents and Consequences of Customer Satisfaction for Firms.* Marketing Science, Vol.12, No. 2, pp. 125-143.

Andrieux, A., Czajkowski, K., Dan, A., Keahey, K., Ludwig, H., Nakata, T., et al. (2007). *Web Services Agreement Specification (WS-Agreement).* Open Grid Forum, http://www.ogf.org/documents/GFD.107.pdf.

Arsanjani, A., Ghosh, S., Allam, A., Abdollah, T., Ganapathy, S., & Holley, K. (2008). *SOMA: a Method for Developing Service-oriented Solutions.* IBM Systems Journal, Vol. 47, No. 3.

Aveiro, D. (2009). *Towards a GOD-theory for organizational engineering: continuously modeling the (re)Generation, Operationalization and Discontinuation of the enterprise.* PhD thesis, Instituto Superior Técnico.

Bellwood, T., Capell, S., Clement, L., Colgrave, J., Dovey, M., Feygin, D., et al. (2004). *UDDI Spec Technical Committee Draft 3.0.2.* https://www.oasis-open.org/committees/uddi-spec/doc/spec/v3/uddi-v3.0.2-20041019.htm.

Bojanic, D., & Rosen, D. (1994). *Measuring service quality in restaurants: An application of the SERVQUAL instrument.* Journal of Hospitality and Tourism Research, Vol.18, No.1, pp.3-14.

Bolton, R., & Drew, J. (1991). *A Multistage Model of Customers' Assessments of Service Quality and Value.* Journal of Consumer Research, Vol.17, No. 4, pp.375-384.

BREIN Project, Accessed on October 2012, http://www.eu-brein.com

Carrillat, F., Jaramillo, F., & Mulki, J. (2007). *The validity of the SERVQUAL and SERVPERF scales: A meta-analytic view of 17 years of research across five continents.*

International Journal of Service Industry Management, Vol.18, No.5, pp.472-490, Emerald Group Publishing Limited.

Chen, J.-C., McAuley, A., Sarangan, V., Baba, S., & Ohba, Y. (2002). *Dynamic service negotiation protocol (DSNP) and wireless DiffServ.* International Conference on Communications, IEEE Computer Society, Vol.2, pp. 1033-1038.

Chesbrough, H. (2011). *Open Services Innovation: Rethinking Your Business to Grow and Compete in a New Era.* Jossey-Bass.

Chesbrough, H., & Spohrer, J. (2006). *A Research Manifesto for Services Science.* Communications of the ACM, Vol.49, No.7, pp.35-40.

Chieng, D., Marshall, A., Ho, I., & Parr, G. (2001). *Agent-Enhanced Dynamic Service Level Agreement in Future Network Environment.* Management of Multimedia on the Internet, Lecture Notes in Computer Science, Vol.2216, pp.299-312, Springer.

Cole, S. (2008). *Service Catalog Trends and Best Practices Survey Highligths.* Enterprise Management Associates.

Cole, W., & Mogab, J. (1995). *The Economics of Total Quality Management: Clashing Paradigms in the Global Market.* Cambridge, MA: Blackwell.

Creswell, J. (2009). *Research Design - Qualitative, Quantitative, and Mixed Methods Approaches* (3rd ed.). SAGE.

Cronin, J., & Taylor, S. (1992). *Measuring Service Quality: A Reexamination and Extension.* Journal of Marketing, Vol.56, No.3, pp.55-68, American Marketing Association.

Cronin, J., Brady, M., & Hult, G. (2000). *Assessing the effects of quality, value, and customer satisfaction on consumer behavioral intentions in service environments.* Journal of Retailing, Vol.76, No.2, pp. 193-218.

Darby, M., & Karni, E. (1973). *Free Competition and the Optimal Amount of Fraud.* Journal of Law and Economics, Vol.16, No.1, pp.67-88.

Dean, A. (1999). *The applicability of SERVQUAL in different health care settings.* Health Marketing Quarterly, Vol.16, No.3, pp.1-15.

Debusmann, M., Kroger, R., & Geihs, K. (2004). *Unifying service level management using an MDA-based approach.* Network Operations and Management Symposium (NOMS) IEEE/IFIP , Vol.1, pp.801-814.

Dietz, J. (2005). *A world ontology specification language.* OTM Workshops.

Dietz, J. (2006). *Enterprise Ontology: Theory and Methodology.* Springer.

Dietz, J. (2009). *Demo meta model specification.* forthcoming in www.demo.nl.

Dietz, J., & Terlouw, L. (2010). *A Framework for Clarifying Service-oriented Notions – How to Position Different Approaches.* Journal Enterprise Modeling and Information Systems Architecture, Vol.5, No. 1, pp.26-43, German Informatics Society.

Dobson, G. (2004). *Quality of Service in Service-Oriented Architectures.* Accessed on June 2011, de http://digs.sourceforge.net/papers/qos.html

Donnelly, M., Wisniewski, M., Dalrymple, J., & Curry, A. (1995). *Measuring service quality in local government: The SERVQUAL approach.* International Journal of Public Sector Management, Vol.8, No.7, pp.15-20.

Dubois, E., Falcão e Cunha, J., & Léonard, M. (2011). *Towards an Executive Master Degree for the New Job Profile of a Service Systems Innovation Architect.* 2011 Annual SRII Global Conference, pp.572-578, IEEE Computer Society.

Erradi, A., Anand, S., & Kulkarni, N. (2006). *SOAF: An Architectural Framework for Service Definition and Realization.* International Conference on Services Computing (SCC), pp.151-158, IEEE Computer Society.

Fan-Chao, M., Den-Chen, Z., & Xiao-Fei, X. (2005). *Business Component Identification of Enterprise Information System: A hierarchical clustering method.* International Conference on e-Business Engineering (ICEBE), pp.473-480, IEEE Computer Society.

Ferrario, R., Guarino, N., & Fernández-Barrera, M. (2011a). *Towards an Ontological Foundation for Services Science: The Legal Perspective.* Approaches to Legal Ontologies - Law, Governance and Technology Series, Vol. 1, pp.235-258, Springer.

Ferrario, R., Guarino, N., Janiesch, C., Kiemes, T., Oberle, D., & Probst, F. (2011b). *Towards an Ontological Foundation for Services Science: The General Service Model.* Wirtschaftinformatik Proceedings 2011. Paper 47, AIS.

Foster, J., & Metcalfe, J. (2001). *Frontiers of Evolutionary Economics.* Edward Elgar.

Frolund, S., & Koistinen, J. (1998). *QML: A Language for Quality of Service Specification.* HP Software Technology Laboratory.

Frutos, H., & Kotsiopoulos, I. (2009). *BREIN: Business Objective Driven Reliable and Intelligent Grids for Real Business.* International Journal of Interoperability in Business Information Systems, Vol.3, No.1, pp. 39-42.

Giarini, O. (1987). *The Emerging Service Economy.* Pergamon Press.

Guo, L., Xiao, J., & Tang, C. (2009). *Understanding the psychological process underlying customer satisfaction and retention in a relational service.* Journal of Business Research, Vol.62, No.11, pp.1152-1159, Elsevier.

Hauser, J., & Clausing, D. (1988). *The House of Quality.* Harvard Business Review, Vol.66 (May–June).

Hedwig, M., Malkowski, S., & Neumann, D. (2011). *Dynamic Service Level Agreement Management for Efficient Operation of Elastic Information Systems.* International Conference on Information Systems (ICIS), Association for Information Systems.

Hefley, B., & Murphy, W. (2008). *Service science, management and engineering: education for the 21st century.* Springer.

Heskett, J., Sasser, W., & Schlesinger, L. (1997). *The Service Profit Chain: How Leading Companies Link Profit and Growth to Loyalty, Satisfaction and Value.* New York: The Free Press.

Hevner, A., March, S., Park, J., & Ram, S. (2004). *Design Science in Information Systems Research.* MIS Quarterly, Vol.28, No.1, pp.75-105, Society for Information Management and The Management Information Systems Research Center.

Hill, T. (1977). *On goods and services.* The Review of Income and Wealth, Vol.23, No.3, pp.315-338.

Homburg, C., Koschate, N., & Hoyer, W. (2005). *Do satisfied customers really pay more? A study of the relationship between customer satisfaction and willingness to pay.* Journal of Marketing, Vol.69, No.2, pp.84-96.

International Organization for Standardization. (2000). *ISO 9000.* Quality management systems – Fundamentals and vocabulary.

International Organization for Standardization. (2007). *ISO/IEC/IEEE 42010.* Systems and software engineering.

Kabilan, V., Johannesson, P., & Rugaimukamu, D. (2004). *An Ontological Approach to Unified Contract Management.* Kitakyushu, Japan: Information Modelling and Knowledge Bases XV, 13th European-Japanese Conference on Information Modelling and Knowledge Bases (EJC 2003), Frontiers in Artificial Intelligence and Applications, Vol.105, pp.106-110, IOS Press.

Karaenke, P., & Kirn, S. (2010). *A Multi-tier Negotiation Protocol for Logistics Supply Chains.* Pretoria, South Africa: 18th European Conference on Information Systems (ECIS 2010).

Keller, A., & Ludwig, H. (2003). *The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services.* Journal of Network and Systems Management, Vol.11, No. 1, pp.57-81.

Kettinger, W., & Lee, C. (1994). *Perceived Service Quality and User Satisfaction with the Information Services Function.* Decision Sciences, Vol.25, No.5-6, pp.737-766.

Kieninger, A., Baltadzhiev, D., Schmitz, B., & Satzger, G. (2011b). *Towards Service Level Engineering for IT Services: Defining IT Services from a Line of Business Perspective.* SRII Global Conference (SRII), pp.759-766.

Kieninger, A., Westernhagen, J., & Satzger, G. (2011a). *The Economics of Service Level Engineering.* 44th Hawaii International Conference on System Sciences (HICSS), Vol.11, pp.1-10, IEEE Computer Society.

Kvale, S. (2007). *Doing interviews.* London: Sage Publications.

Lages, L., & Fernandes, J. (2005). *The SERPVAL scale: a multi-item instrument for measuring service personal values.* Journal of Business Research, Vol.58, No.11, pp.1562-1572, Elsevier.

Lai, F., Griffin, M., & Babin, B. (2009). *How quality, value, image, and satisfaction create loyalty at a Chinese telecom.* Journal of Business Research, Vol.62, No.10, pp.980-986, Elsevier.

Liu, Y., Ngu, A., & Zeng, L. (2004). *QoS Computation and Policing in Dynamic Web Service Selection.* New York, USA: 13th International World Wide Web Conference, pp.66-73, ACM.

Macaulay, L., Moxham, C., Jones, B., & Miles, I. (2012). *Innovation and Skills: Future Service Science Education.* Service Science: Research and Innovations in the Service Economy, pp.717-736, Springer.

Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., & Ghalsasi, A. (2011). *Cloud computing — The business perspective.* Decision Support Systems, Vol.51, pp. 176–189, Elsevier.

Mendes, C. (2009). *Financial Management Implementation.* Master Thesis, Instituto Superior Técnico.

Mendes, C., & Mira da Silva, M. (2010). *Implementing the Service Catalogue Management.* Porto, Portugal: 7th International Conference on the Quality of Information and Communications Technology (QUATIC), pp.159-164, IEEE Computer Society.

Mendes, C., & Mira da Silva, M. (2011). Implementing a Request Fulfillment Process. In *Exploring Services Science, Lecture Notes in Business Information Processing.* Geneva, Switzerland: 2nd International Conference on Exploring Service Science, Vol.82, pp.113-126, Springer.

Mendes, C., & Mira da Silva, M. (2012). DEMO-based Service Level Agreements. In *Exploring Services Science, Lecture Notes in Business Information Processing.* Geneva,

Switzerland: 3rd International Conference on Exploring Service Science, pp.227-242, Springer.

Mendes, C., Almeida, M., Salvador, N., & Mira da Silva, M. (2012). *Using DEMO-based SLAs for Improving City Council Services.* Barcelona, Spain: 4th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management - Special Session on Enterprise Ontology (Best Paper Award).

Mendes, C., Ferreira, J., & Mira da Silva, M. (2011). *Comparing Service Using DEMO.* Paris, France: 3rd International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management - Special Session on Enterprise Ontology (Best Student Paper Award).

Meuter, M., Ostrom, A., Roundtree, R., & Bitner, M. (2000). *Self-Service Technologies: Understanding Customer Satisfaction with Technology-Based Service Encounters.* Journal of Marketing, Vol.64, No.3, pp.50-64.

Microsoft. (2008). *Microsoft Operations Framework (MOF) 4.0.*

Mohr, M., & Russel, S. (2002). North American Product Classification System: Concepts and Process of Identifying Service Products. Nantes: 17th Annual Meeting of the Voorburg Group on Service Statistics.

Moody, D., & Shanks, G. (2003). *Improving the Quality of Data Models: Empirical Validation of a Quality Management Framework.* Journal Information Systems Vol.28, No.6, pp.619-650, Elsevier.

Morgan, N., Anderson, E., & Mittal, V. (2005). *Understanding Firms' Customer Satisfaction Information Usage.* Journal of Marketing, Vol.69, No.3, pp.131-151.

Nelson, P. (1974). *Advertising as Information.* Journal of Political Economy, Vol.82, No.4, pp.729-754.

O'Sullivan, J. (2006). *Towards a precise understanding of service properties.* PhD thesis, Queensland University of Technology .

Office of Government Commerce. (2007a). *ITIL v3 – Service Strategy.* The Stationery Office.

Office of Government Commerce. (2007b). *ITIL v3 – Service Design.* The Stationery Office.

Oliver, R. (1980). *A Cognitive Model of the Antecedents and Consequences of Satisfaction Decisions.* Journal of Marketing Research, Vol.17, No.4, pp.460-469.

Österle, H., Becker, J., Frank, U., Hess, T., Karagiannis, D., Krcmar, H., et al. (2011). Memorandum on Design-Oriented Information Systems Research. *European Journal on Information Systems, Vol.10, pp.7-10.*

Ostrom, A. B.-D. (2010). *Moving Forward and Making a Difference: Research Priorities for the Science of Service.* Journal of Service Research, Vol.13, No.1, pp.4-36.

Oxford Dictionaries. Accessed on July 2012 http://oxforddictionaries.com/definition/service?q=service

Parasuraman, A., Zeithaml, V., & Berry, L. (1985). *A Conceptual Model of Service Quality and its Implication for Future Research.* Journal of Marketing, Vol.49, No.4, pp.41-50.

Parasuraman, A., Zeithaml, V., & Berry, L. (1988). *SERVQUAL: A Multiple-Item Scale for Measuring Consumer Perceptions of Service Quality.* Journal of Retailing, Vol.64, No.1, pp.12-40.

Parasuraman, A., Zeithaml, V., & Malhotra, A. (2005). *E-S-QUAL: A Multiple-Item Scale for Assessing Electronic Service Quality.* Journal of Service Research, Vol.7, No.3, pp.213-233.

Patrício, L., Fisk, R. P., & Falcão e Cunha, J. (2008). *Designing Multi-Interface Service Experiences: The Service Experience Blueprint.* Journal of Service Research, Vol.10, No.4, pp.318-334.

Patrício, L., Fisk, R. P., Falcão e Cunha, J., & Constantine, L. (2011). *Multilevel Service Design: From Customer Value Constellation to Service Experience Blueprinting.* Journal of Service Research, Vol.14, No.2, pp.180-200.

Peffers, K., Tuunanen, T., Rothenberger, M., & Chatterjee, S. (2008). *A Design Science Research Methodology for Information Systems Research.* Journal of Management Information Systems, Vol.24, No.3, pp.45-77, M. E. Sharpe.

Pichot, A., Wäldrich, O., Ziegler, W., & Wieder, P. (2009). *Towards Dynamic Service Level Agreement Negotiation:An Approach Based on WS-Agreement.* Web Information Systems and Technologies, Lecture Notes in Business Information Processing, Springer, Vol. 18, pp. 107-119.

Pombal City Council. (2011). *2011 Pombal City Council Budget.* Pombal City Council, Pombal.

Pries-Heje, J., Baskerville, R., & Venable, J. (2004). *Strategies for Design Science Research Evaluation.* 16th European Conference on Information Systems (ECIS), pp. 255-266.

Quinn, J. (1988). *Technology in Services: Policies for Growth, Trade, and Employment.* National Academy of Engineering.

Rigotti, S., & Pitt, L. (1992). *SERVQUAL as a Measuring Instrument for Service Provider Gaps in Business Schools.* Management Research News, Vol.15, No.3, pp.9-17.

Rust, R., & Chung, T. (2006). *Marketing Models of Service and Relationships.* Marketing Science, Vol.25, No.6, pp.560-580.

Rust, R., Moorman, C., & Dickson, P. (2002). *Getting Return on Quality : Revenue Expansion, Cost Reduction, or Both.* Journal of Marketing, Vol.66, No.4, pp.7-24.

Sahai, A., Durante, A., & Machiraju, V. (2002). *Towards Automated SLA Management for Web Services.* Technical Report, Hewlett-Packard Company.

Saleh, F., & Ryan, C. (1991). *Analysing service quality in the hospitality industry using the SERVQUAL model.* The Service Industries Journal, Vol.11, No.3, pp.324-345.

Sarangan, V., & Chen, J.-C. (2006). *Comparative study of protocols for dynamic service negotiation in the next-generation Internet.* Communications Magazine, IEEE Computer Society, Vol. 33, No. 3, pp. 151-156.

Schneider, B., White, S., & Paul, M. (1998). *Linking service climate and customer perceptions of service quality: test of a causal model.* Journal of Applied Psychology, Vol.83, No.2, pp.150-63.

Shostack, G. L. (1984). *Designing Services That Deliver.* Harvard Business Review.

Software Engineering Institute. (2010). *CMMI for Services, Version 1.3.* Carnegie Mellon University.

Spohrer, J., Maglio, P., Bailey, J., & Gruhl, D. (2007). *Steps Toward a Science of Service Systems.* Computer, Vol.40, No.1, pp.71–77, IEEE Computer Society.

Spohrer, J., Vargo, S., Caswell, N., & Maglio, P. (2008). *The Service System Is the Basic Abstraction of Service Science.* 41st Annual Hawaii International Conference on System Sciences (HICCS), Vol.7, No.4, pp.104-116, IEEE Computer Society.

Storbacka, K., Strandvik, T., & Grönroos, C. (1994). *Managing Customer Relationships for Profit: The Dynamics of Relationship Quality.* International Journal of Service Industry Management, Vol.5, No.5, pp.21-38.

Taleb, T., Hashimoto, K., Kato, N., & Nemoto, Y. (2007). *A Dynamic Service Level Negotiation Mechanism for QoS Provisioning in NGEO Satellite Networks.* International Conference on Communications, IEEE Computer Society, pp.1-6.

TeleManagement Forum. (2007). *Enhanced Telecom Operations Map (eTOM) - The Business Process Framework.*

Terlouw, L. (2011). *Modularization and Specification of Service-Oriented Systems.* PhD thesis, Delft University of Technical.

Terlouw, L., & Albani, A. (2013). *An Enterprise Ontology-Based Approach to Service Specification.* Transactions on Services Computing, IEEE Computer Society, Vol.36, No.1, pp.89-101.

Tosic, V., Patel, K., & Pagurek, B. (2002). *WSOL – Web Service Offerings Language.* London, UK: International Workshop on Web Services, E-Business, and the Semantic Web, pp.57-67, Springer.

Tseng, M., Qinhai, M., & Su, C.-J. (1999). *Mapping customers' service experience for operations improvement.* Business Process Management Journal, Vol.5 No.1, pp.50-64.

Ukens, L. (2007). *101 Ways to Improve Customer Service: Training, Tools, Tips, and Techniques* . Pfeiffer.

Unterharnscheidt, P., & Kieninger, A. (2010). *Service Level Management – Challenges and their Relevance from the Customers' Point of View.* Americas Conference on Information Systems, AIS.

Vargo, S., & Akaka, M. (2009). *Service-Dominant Logic as a Foundation for Service Science: Clarifications.* Journal Service Science, Vol.1, No.1, pp. 32-41.

Vargo, S., & Lusch, R. (2004). *Evolving to a New Dominant Logic for Marketing.* Journal of Marketing, Vol.68, pp.1-17.

Vargo, S., & Lusch, R. (2006). *The Service-Dominant Logic of Marketing: Dialog, Debate, and Directions.* Armonk, New York: pp.43-56, M. E. Sharpe.

Vargo, S., Maglio, P., & Akaka, M. (2008). *On value and value co-creation: A service systems and service logic perspective.* European Management Journal, Vol.26, pp.145–152, Elsevier.

W3C. (2001). *Web services description language.* http://www.w3.org/TR/wsdl.

Wang, Z., Xu, X., & Zhan, D. (2005). *A Survey of Business Component Identification Methods and Related Techniques.* International Journal of Information Technology, Vol.2, No.4.

Wilson, A., Zeithaml, V., Bitner, M., & Gremler, D. (2012). *Services Marketing: Integrating Customer Focus Across the Firm.* McGraw-Hill Higher Education, 2nd Edition.

Yoo, B., & Donthu, N. (2001). *Developing a Scale to Measure the Perceived Quality of An Internet Shopping Site (SITEQUAL).* Quarterly Journal of Electronic Commerce, Vol.2, No.1, pp.31-47.

Zeithaml, V. (1981). *How consumer evaluation processes differ between goods and services.* Chicago: Marketing of Services, Vol.9, No.1, pp.186-190, American Marketing Association.

Zeithaml, V., & Bitner, M. (1996). *Service Marketing.* New York: McGraw-Hill.

Zeithaml, V., Berry, L., & Parasuraman, A. (1996). *The Behavioral Consequences of Service Quality.* Journal of Marketing, Vol.60, No.2, pp.31-46, American Marketing Association.

Appendix A – Related Work Concept Models

In this chapter we present the concept models that we produced from the related work. These models helped to understand and structure the studied solutions.

Figure 56 summarizes the main concepts that appear in the Service Dominant Logic (S-D Logic).

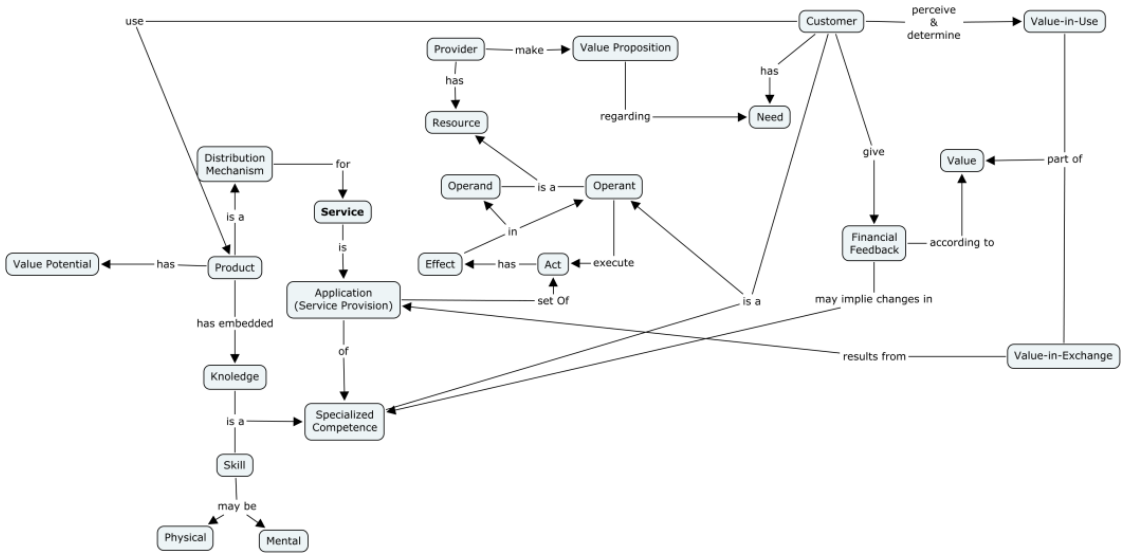


Figure 56 – Service Dominant Logic Conceptual Map

In S-D Logic there are no "services." There is service, the act of doing something for another party, directly or through a good. Therefore, a product is seen as a distribution mechanism for a service. S-D logic states that value is perceived and determined by the consumer on the basis of "value in use." Value results from the beneficial application of operant resources sometimes transmitted through operand resources. Providers can only make value propositions regarding customers' needs. In S-D Logic customers are primarily operant resources and are active participants in relational exchanges and coproduction (Vargo & Lusch, 2004).

Figure 57 summarizes the main concepts that appear in Ontological Foundation for Service Science.

A service system process consists of different interconnected processes and events, resulting from complex interactions involving intentional agents and technological artifacts. A service system process is composed by two main elements: the service (itself composed by the service commitment and the service process) and the service value exchange. A service commitment is an agent's explicit commitment to guarantee the execution of some type of actions, on the occurrence of a certain triggering event, in the interest of another agent and upon prior agreement, according to a certain specification (service description) which constraints the way service actions will be performed (service process) (Ferrario R. , et al., 2011b).

A service implies a concrete commitment (from the side of a provider) to guarantee the production of a certain content, consisting in actions of a certain kind executed in a certain way. Altogether, the various actions (performed by the service producer on behalf of the provider) constitute the service process. Additionally, the service process realizes the service commitment i.e. the service process is the execution of the actions described in the service description. (Ferrario, et al., 2011a).

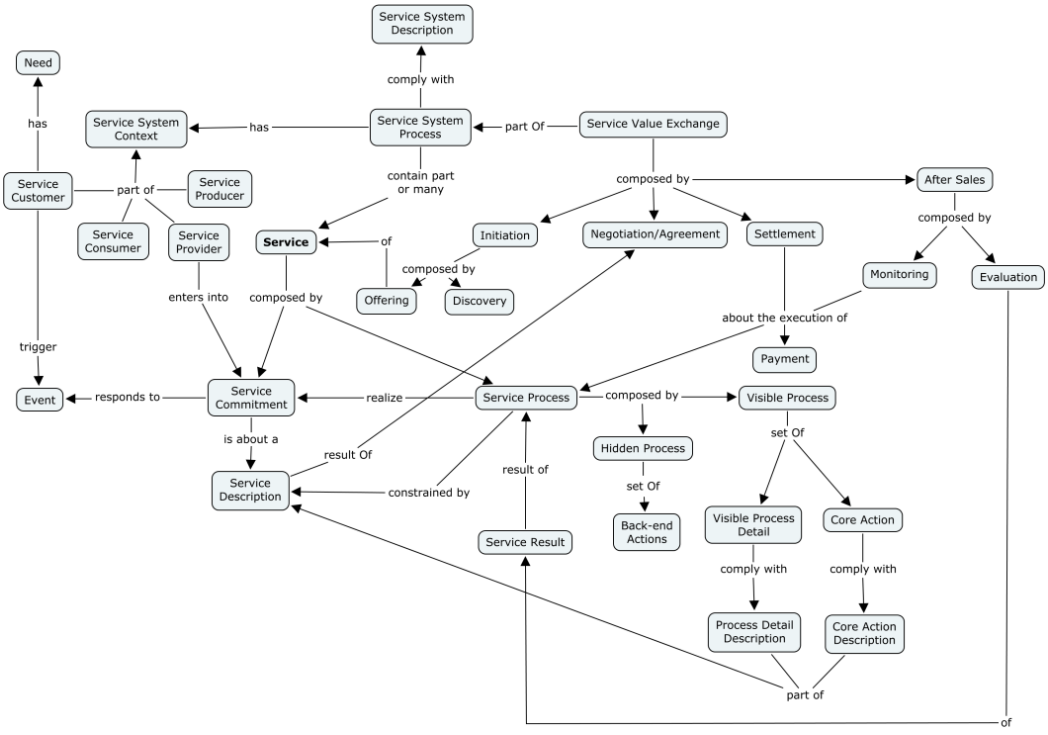


Figure 57 – General Conceptual Model of Ontological Foundation for Service Science

The service value exchange is composed by four phases: initiation, negotiation/agreement, settlement, and after sales. The actors involved in these four phases are the service provider, service customer, service producer, and service consumer.

Figure 58 focus on the actor roles that appear in Ontological Foundation for Service Science.

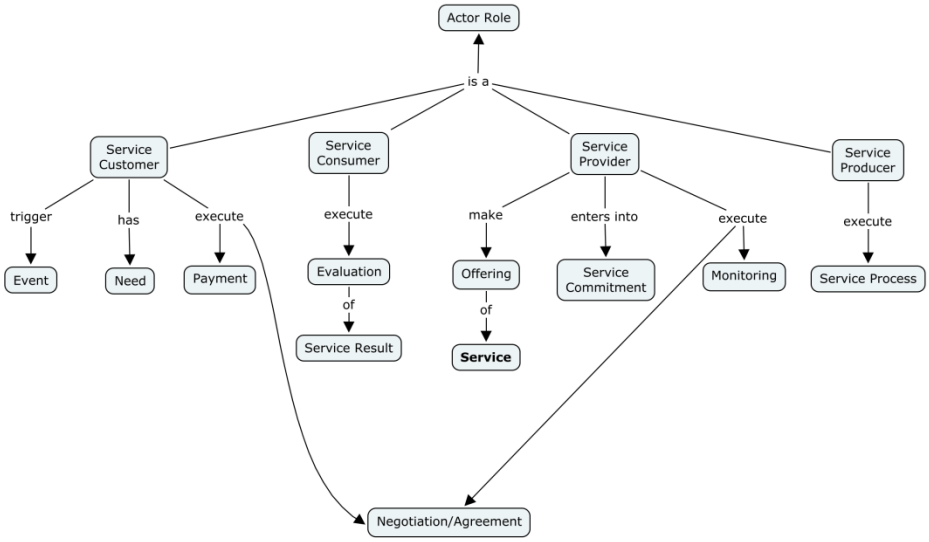
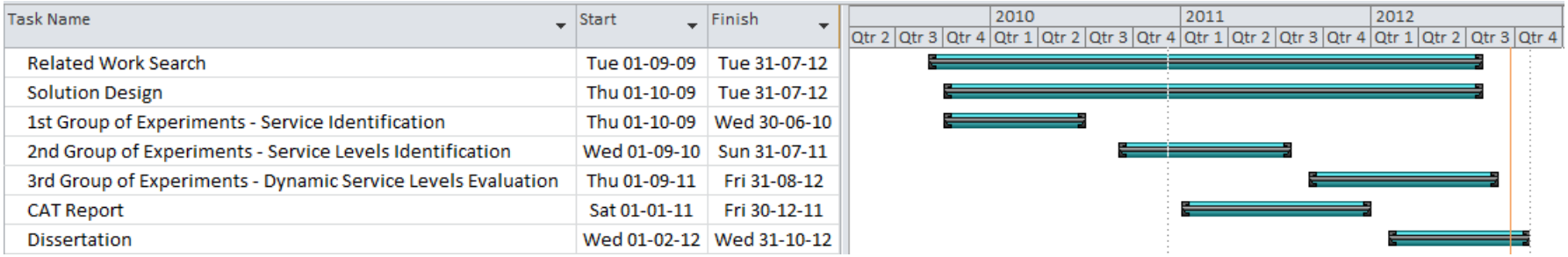


Figure 58 – Conceptual Model of Ontological Foundation for Service Science (Actor Roles)

The service customer is accountable for triggering the event that origins the service delivery, negotiates with the service provider the service result and pays for the service. The service consumer evaluates the service result in the after sales phase of the service value exchange. The service provider makes services offerings, enters into service commitments, negotiates the service result with the service customer, and monitors the execution of the service process. Finally, the service producer is accountable for executing the service process (Ferrario R. , et al., 2011b)

Appendix B – Thesis work plan

Figure 59 illustrates what was this thesis work plan.



Task Name	Start	Finish
Related Work Search	Tue 01-09-09	Tue 31-07-12
Solution Design	Thu 01-10-09	Tue 31-07-12
1st Group of Experiments - Service Identification	Thu 01-10-09	Wed 30-06-10
2nd Group of Experiments - Service Levels Identification	Wed 01-09-10	Sun 31-07-11
3rd Group of Experiments - Dynamic Service Levels Evaluation	Thu 01-09-11	Fri 31-08-12
CAT Report	Sat 01-01-11	Fri 30-12-11
Dissertation	Wed 01-02-12	Wed 31-10-12

Figure 59 – Thesis work plan

Appendix C – Proposal Process Structure Diagrams

This appendix presents the remaining Process Structure Diagrams of our proposal to dynamically negotiate SLAs.

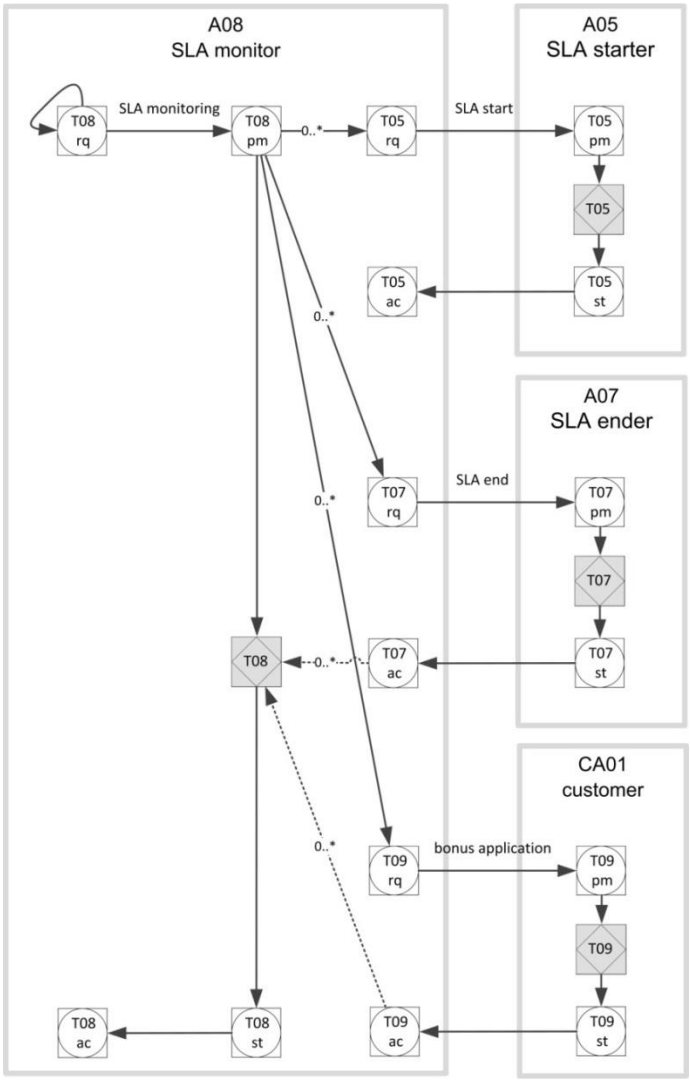


Figure 60 – Proposal Process Structure Diagram (PSD) of transactions T08 to T09

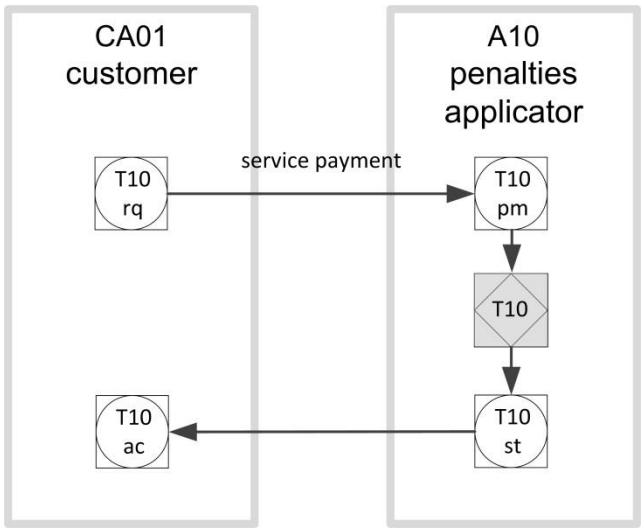


Figure 61 – Proposal Process Structure Diagram (PSD) of transaction T10

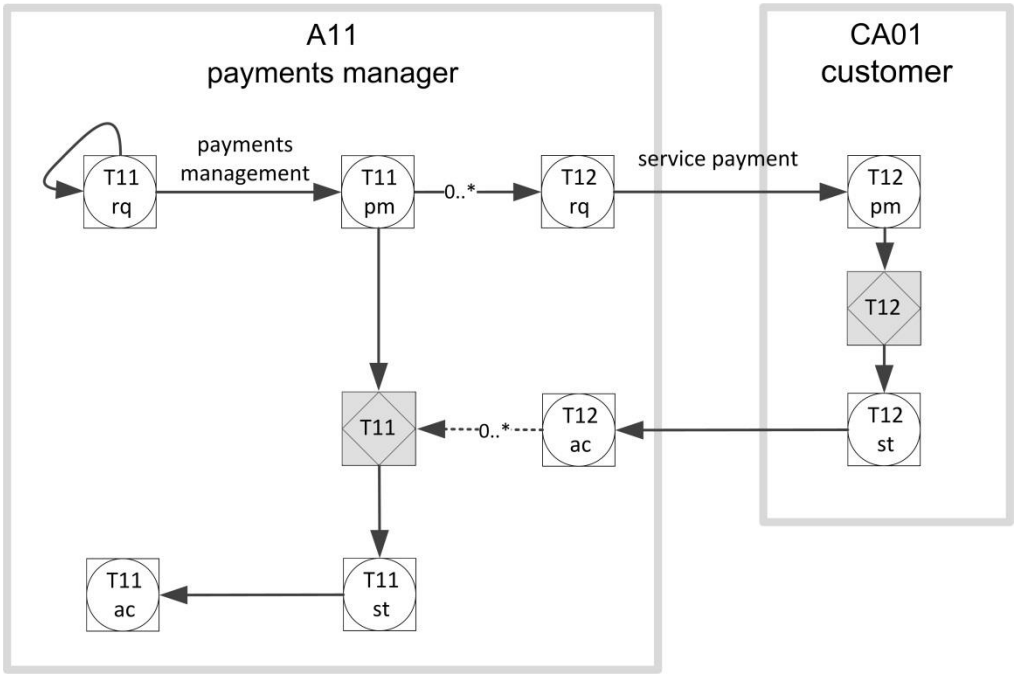
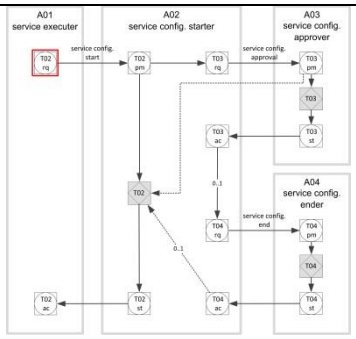
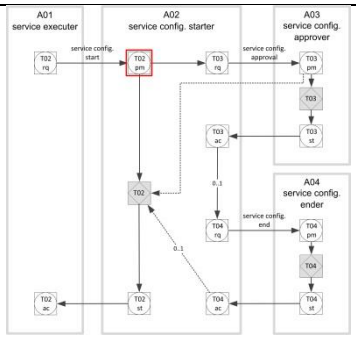
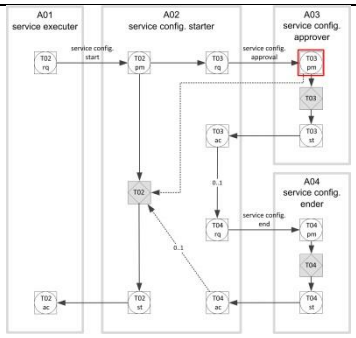
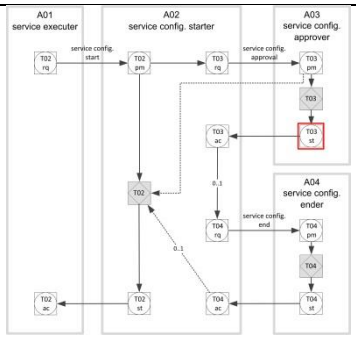
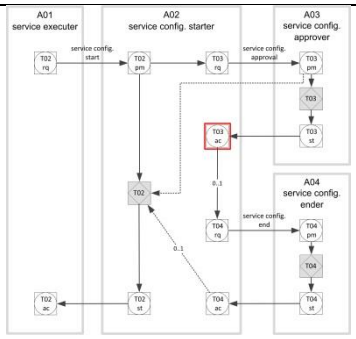
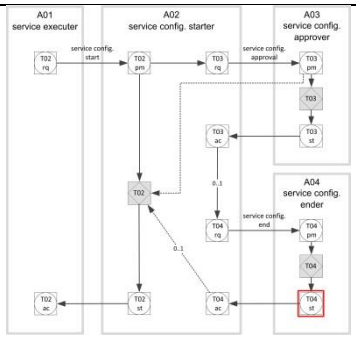


Figure 62 – Proposal Process Structure Diagram (PSD) of transactions T11 and T12

Appendix D – Proposal action rules

The following tables define the action rules of our proposal for the actor roles A02, A03, A04, A05, A06, A07, A08, A10, and A11. The second column of each table present a sample of the Process Structure Diagram (PSD) containing the coordination act that the action rule corresponds (highlighted with a red square).

#	PSD	Action Rule
1		**when** start of [ServiceConfiguration] is requested **if** <canFulfillRequest> **then** start of [ServiceConfiguration] must be promised **else** fulfilment of [ServiceConfiguration] must be declined
2		**when** start of [ServiceConfiguration] is promised **then** approval of [ServiceConfiguration] must be requested

#	PSD	Action Rule
3		**when** approval of [ServiceConfiguration] is promised **if** <needsEndServiceConfiguration> **then** start of [ServiceConfiguration] must be executed **then** start of [ServiceConfiguration] must be stated
4		**when** approval of [ServiceConfiguration] is stated **if** <approval of [ServiceConfiguration] is acceptable> **then** approval of [ServiceConfiguration] must be accepted
5		**when** approval of [ServiceConfiguration] is accepted **if** <needsEndServiceConfiguration> **then** end of [ServiceConfiguration] must be requested
6		**when** end of [ServiceConfiguration] is stated **if** <end of [ServiceConfiguration] is acceptable> **then** end of [ServiceConfiguration] must be accepted

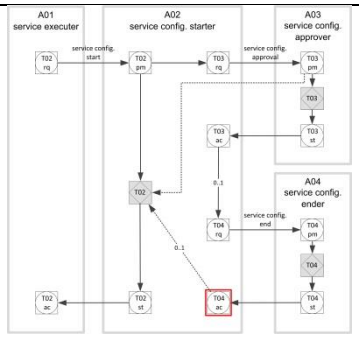
#	PSD	Action Rule
7		**when** end of [ServiceConfiguration] is accepted **then** start of [ServiceConfiguration] must be executed **and** start of [ServiceConfiguration] must be stated

Table 37 – Action rules for elementary actor role service configuration starter (A02)

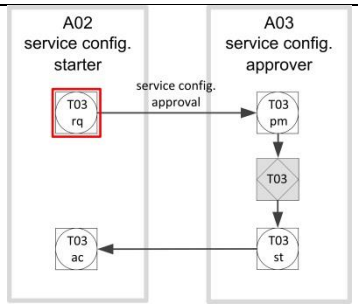
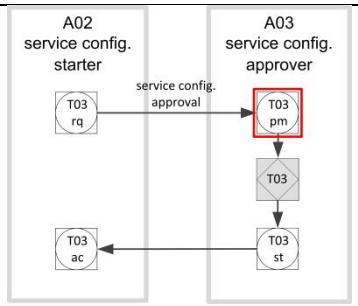
#	PSD	Action Rule
1		**when** approval of [ServiceConfiguration] is requested **if** <canFulfillRequest> **then** approval of [ServiceConfiguration] must be promised **else** approval of [ServiceConfiguration] must be declined
2		**when** approval of [ServiceConfiguration] is promised **then** approval of [ServiceConfiguration] must be executed **and** approval of [ServiceConfiguration] must be stated

Table 38 – Action rules for elementary actor role service executor (A03)

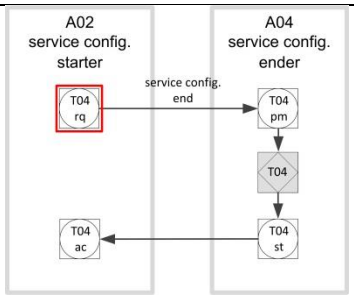
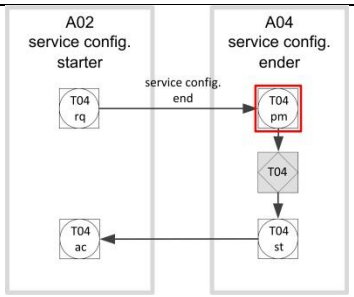
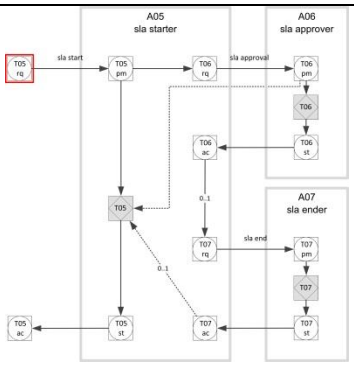
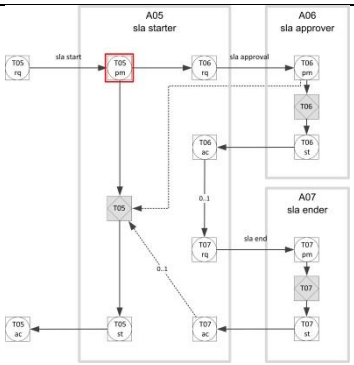
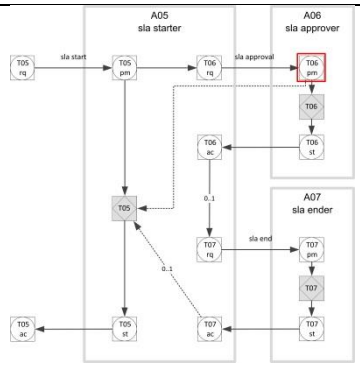
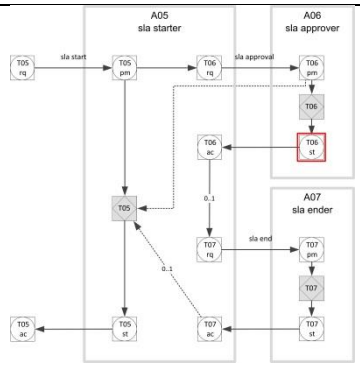
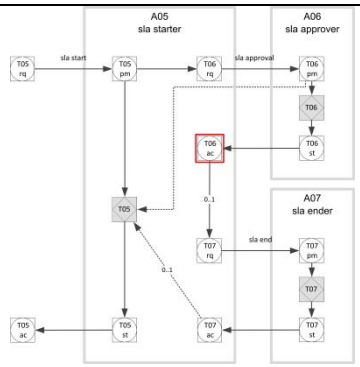
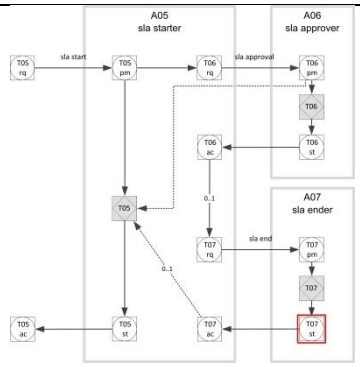
#	PSD	Action Rule
1		**when** end of [ServiceConfiguration] is requested **if** <canFulfillRequest> **then** end of [ServiceConfiguration] must be promised **else** end of [ServiceConfiguration] must be declined
2		**when** end of [ServiceConfiguration] is promised **then** end of [ServiceConfiguration] must be executed **and** end of [ServiceConfiguration] must be stated

Table 39 – Action rules for elementary actor role service configuration ender (A04)

#	PSD	Action Rule
1		**when** start of [SLA] is requested **if** <canFulfillRequest> **then** start of [SLA] must be promised **else** start of [SLA] must be declined
2		**when** start of [SLA] is promised **then** approval of [SLA] must be requested

#	PSD	Action Rule
3		**when** approval of [SLA] is promised **if** <needsEndSLA> **then** start of [SLA] must be executed **and** start of [SLA] must be stated
4		**when** approval of [SLA] is stated **if** <approval of [SLA] is acceptable> **then** approval of [SLA] must be accepted
5		**when** approval of [SLA] is accepted **if** <needsEndSLA> **then** end of [SLA] must be requested
6		**when** end of [SLA] is stated **if** <end of [SLA] is acceptable> **then** end of [SLA] must be accepted

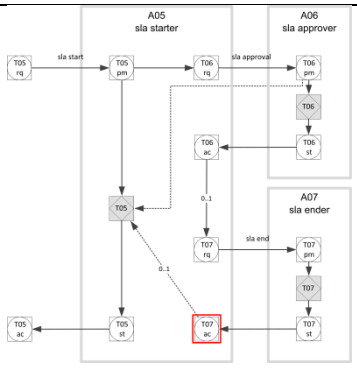
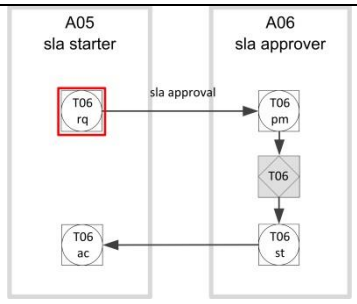
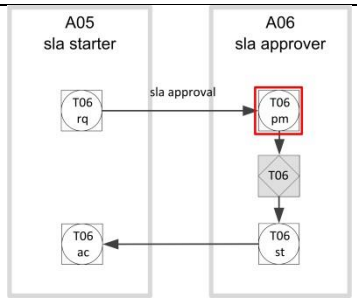
#	PSD	Action Rule
7		**when** end of [SLA] is <u>accepted</u> **then** start of [SLA] must be <u>executed</u> **and** start of [SLA] must be <u>stated</u>

Table 40 – Action rules for elementary actor role SLA starter (A05)

#	PSD	Action Rule
1		**when** approval of [SLA] is <u>requested</u> **if** <canFulfillRequest> **then** approval of [SLA] must be <u>promised</u> **else** approval of [SLA] must be <u>declined</u>
2		**when** approval of [SLA] is <u>promised</u> **then** approval of [SLA] must be <u>executed</u> **and** approval of [SLA] must be <u>stated</u>

#	PSD	Action Rule

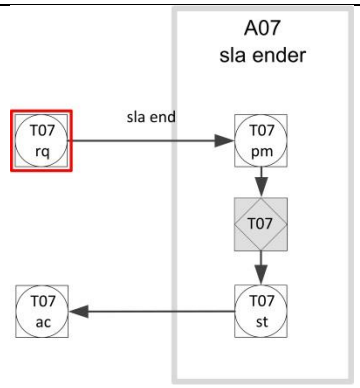
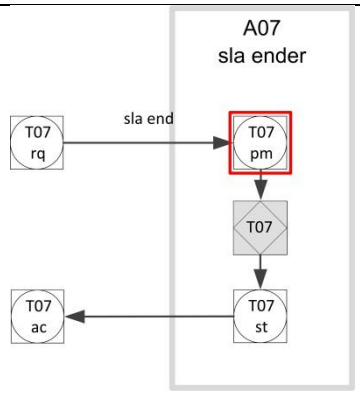
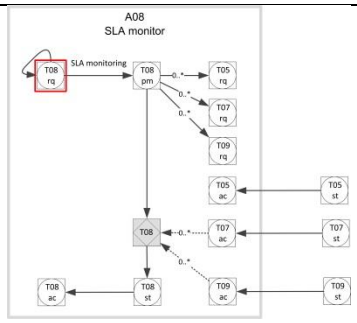
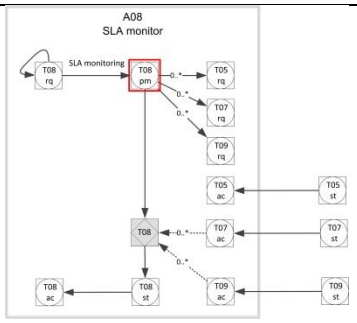
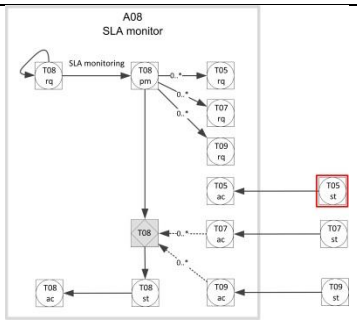
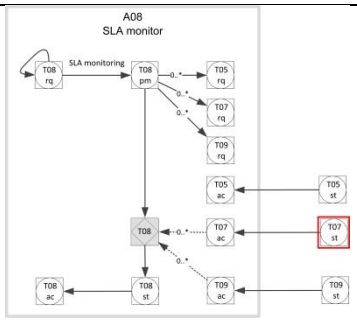
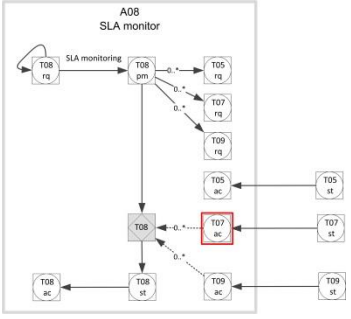
#	PSD	Action Rule
1		**when** end of [SLA] is requested **if** <canFulfillRequest> **then** end of [SLA] must be promised **else** end of [SLA] must be declined
2		**when** end of [SLA] is promised **then** end of [SLA] must be executed **and** end of [SLA] must be stated

Table 42 – Action rules for elementary actor role SLA ender (A07)

#	PSD	Action Rule
1		**when** SLAs monitoring of [Period] is requested **then** SLAs monitoring of [next Period] must be requested **and** SLAs monitoring of [Period] must be promised

#	PSD	Action Rule
2		**when** SLAs monitoring of [Period] is promised **for each** [SLA] in [Period] if <newSLAneeded> **then** start of [SLA] must be requested if <endSLA> **then** end of [SLA] must be requested if <bonusApplied> **then** application of [Bonus] must be requested
3		**when** start of [SLA] is stated if <start of [SLA] is acceptable> **then** start of [SLA] must be accepted
4		**when** end of [SLA] is stated if <end of [SLA] is acceptable> **then** end of [SLA] must be accepted
5		**when** end of [SLA] is accepted if <there are not bonus applications to be accepted> **then** SLAs monitoring of [Period] must be executed **and** SLAs monitoring of [Period] must be stated **and** SLAs monitoring of [Period] must be accepted

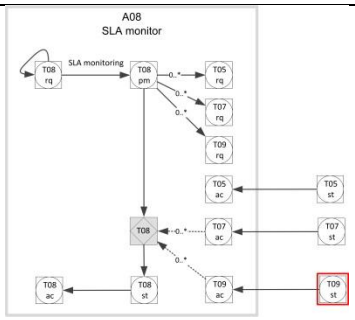
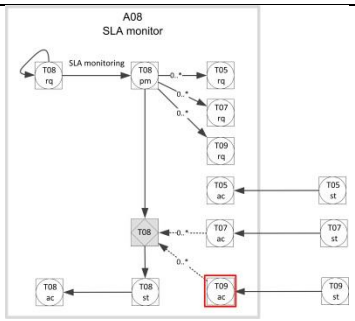
#	PSD	Action Rule
6		**when** application of [Bonus] is stated if <application of [Bonus] is acceptable> **then** application of [Bonus] must be accepted
7		**when** application of [Bonus] is accepted if <there are not SLA ends to be accepted> **then** SLAs monitoring of [Period] must be executed **and** SLAs monitoring of [Period] must be stated **and** SLAs monitoring of [Period] must be accepted

Table 43 – Action rules for elementary actor role SLA monitor (A08)

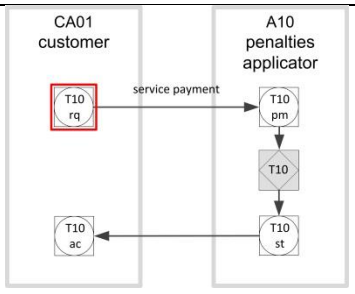
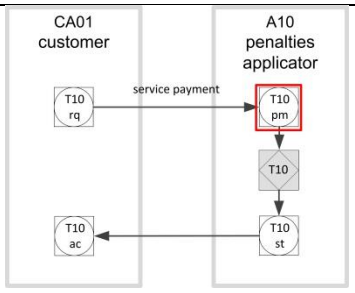
#	PSD	Action Rule
1		**when** payment of [Penalty] is requested **if** <canFulfillRequest> **then** payment of [Penalty] must be promised **else** payment of [Penalty] must be declined
2		**when** payment of [Penalty] is promised **then** payment of [Penalty] must be executed **and** payment of [Penalty] must be stated

Table 44 – Action rules for elementary actor role penalties applicator (A10)

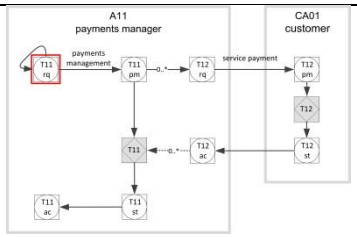
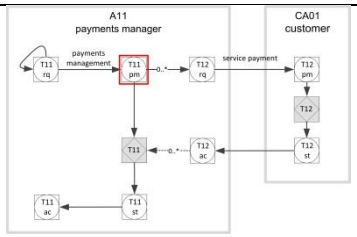
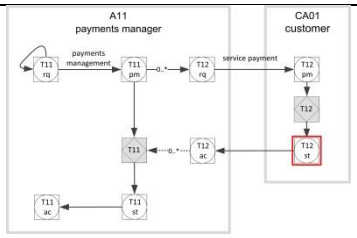
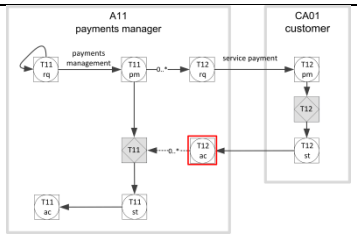
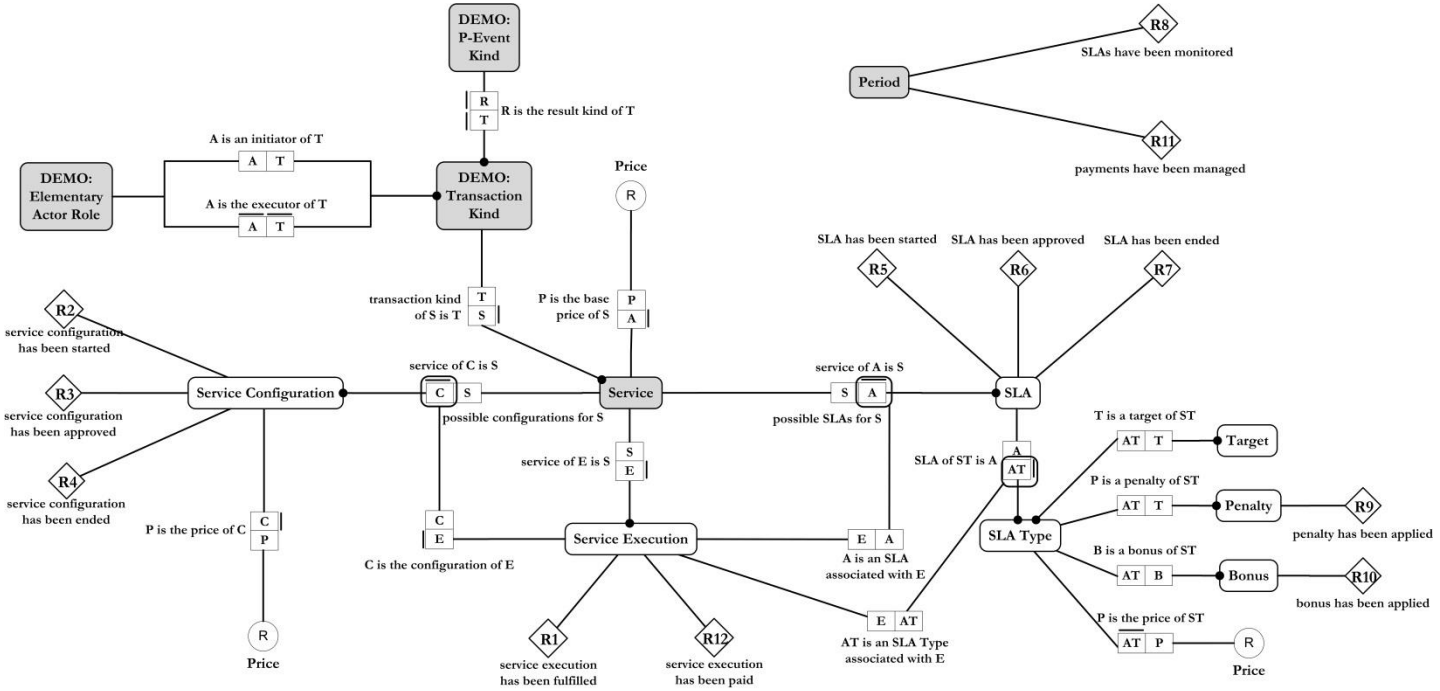
#	PSD	Action Rule
1		**when** payments management of [Period] is requested **then** payments management of [next Period] must be requested **and** payments management of [Period] must be promised
2		**when** payments management of [Period] is promised **for each** [ServiceExecution] in [Period] if <[ServiceExecution not paid]> **then** payment of [ServiceExecution] must be requested with *service_execution_price*([ServiceExecution])
3		**when** payment of [ServiceExecution] is stated if <payment of [ServiceExecution] is acceptable> **then** payment of [ServiceExecution] must be accepted
4		**when** payment of [ServiceExecution] is accepted if <all [ServiceExecution] in [Period] are paid> **then** payments management of [Period] must be executed **and** payments management of [Period] must be stated **and** payments management of [Period] must be accepted

Table 45 – Action rules for elementary actor role service executor (A11)

Appendix E – Proposal Object Fact Diagram



Appendix F – Proposal Information Use Table (IUT)

The next table defines the IUT of our proposal for dynamically negotiate SLAs.

Object class, fact type, or result type	Process steps
Service (S)	T01/rq, T02/rq, T03/rq, T04/rq, T05/rq, T06/rq, T07/rq
P is the base price of S	T01/rq, T12/rq
Service Configuration (C)	T01/rq, T02/rq, T03/rq, T04/rq
service of C is S	T02/rq, T03/rq, T04/rq
P is the price of C	T01/rq, T02/rq, T03/rq, T12/rq
SLA (A)	T01/rq, T05/rq, T06/rq, T07/rq
service of A is S	T05/rq, T06/rq, T07/rq
Service Execution (E)	T01/rq, T09/rq, T10/rq, T12/rq
pre_conditions_price	T01/pm, T12/rq
service_execution_price	T01/rq, T12/rq
service of E is S	T01/rq
C is the configuration of E	T01/rq
A is an SLA associated with E	T01/rq
SLA Type	T01/rq, T05/rq, T06/rq, T07/rq
SLA of ST is A	T05/rq, T06/rq, T07/rq
AT is an SLA type associated with E	T01/rq
Target	T05/rq, T06/rq, T07/rq
T is a target of AT	T05/rq, T06/rq, T07/rq
P is the price of AT	T05/rq, T06/rq, T07/rq
Penalty	T05/rq, T06/rq, T07/rq, T09/rq
P is a penalty of AT	T05/rq, T06/rq, T07/rq
Bonus	T05/rq, T06/rq, T07/rq, T10/rq
B is a bonus of AT	T05/rq, T06/rq, T07/rq

Table 46 – Information Use Table (IUT)

Appendix G – Proposal Interstriction Model (ISM)

The Interstriction Model (ISM) constitutes the 'right hand side' of the CM (see Figure 10). It is expressed in the Bank Contents Table (BCT) and the Actor Bank Diagram (ABD). In order to construct the BCT, one should indicate in the State Model for every event type the actor role that creates it. Then, one should copy this actor role indication to the object classes and fact types that are the domains of the event types, unless they are external. In case of multiples actor roles choose the first that uses the object classes. All remaining fact types, event types, and property types are external. They are contained in (external) aggregate production banks. One may freely devise these banks.

Object class, fact type, or result type	P-bank
service execution E has been fulfilled	PB01
Service Execution	
service of E is S	
C is the configuration of E	
A is an SLA associated with E	
AT is an SLA type associated with E	
pre_conditions_price (*)	
service_execution_price (*)	
service configuration C has been started	PB02
Service Configuration	
P is the price of C	
service of C is S	
possible configurations for S	
service configuration C has been approved	PB03
service configuration has been ended	PB04
SLA A has been started	PB05
SLA	
service of A is S	
possible SLAs for S	
SLA Type	
SLA of ST is A	
T is a target of AT	

Target	
P is the price of AT	
P is a penalty of AT	
Penalty	
B is a bonus of AT	
Bonus	
SLA A has been approved	PB06
SLA A has been ended	PB07
SLAS have been monitored for period P	PB08
penalty P has been applied	PB09
bonus B has been applies	PB10
payments have been managed for period P	PB11
service execution E has been paid	PB12
DEMO: P-Event Kind	CPB01 - DEMO data
DEMO: Transaction Kind	
R is the result kind of T	
DEMO: Elementary Actor Role	
A is an initiator of T	
A is the executor of T	
Service	CPB02 - GSSF data
transaction kind of S is T	
P is the base price of S	
Period	CPB03 - Period data

Table 47 – Bank Contents Table (BCT)

We decided to group these remaining fact types, event types, and property types in three composite production banks: CPB01 - DEMO data, CPB02 - GSSF data and CPB03 Period Data.

Next, to model the ABD, one should copy the ATD and add the external production banks identified before. Afterwards, one should draw dashed lines (information links) between every internal actor role and the banks it has to have access to.

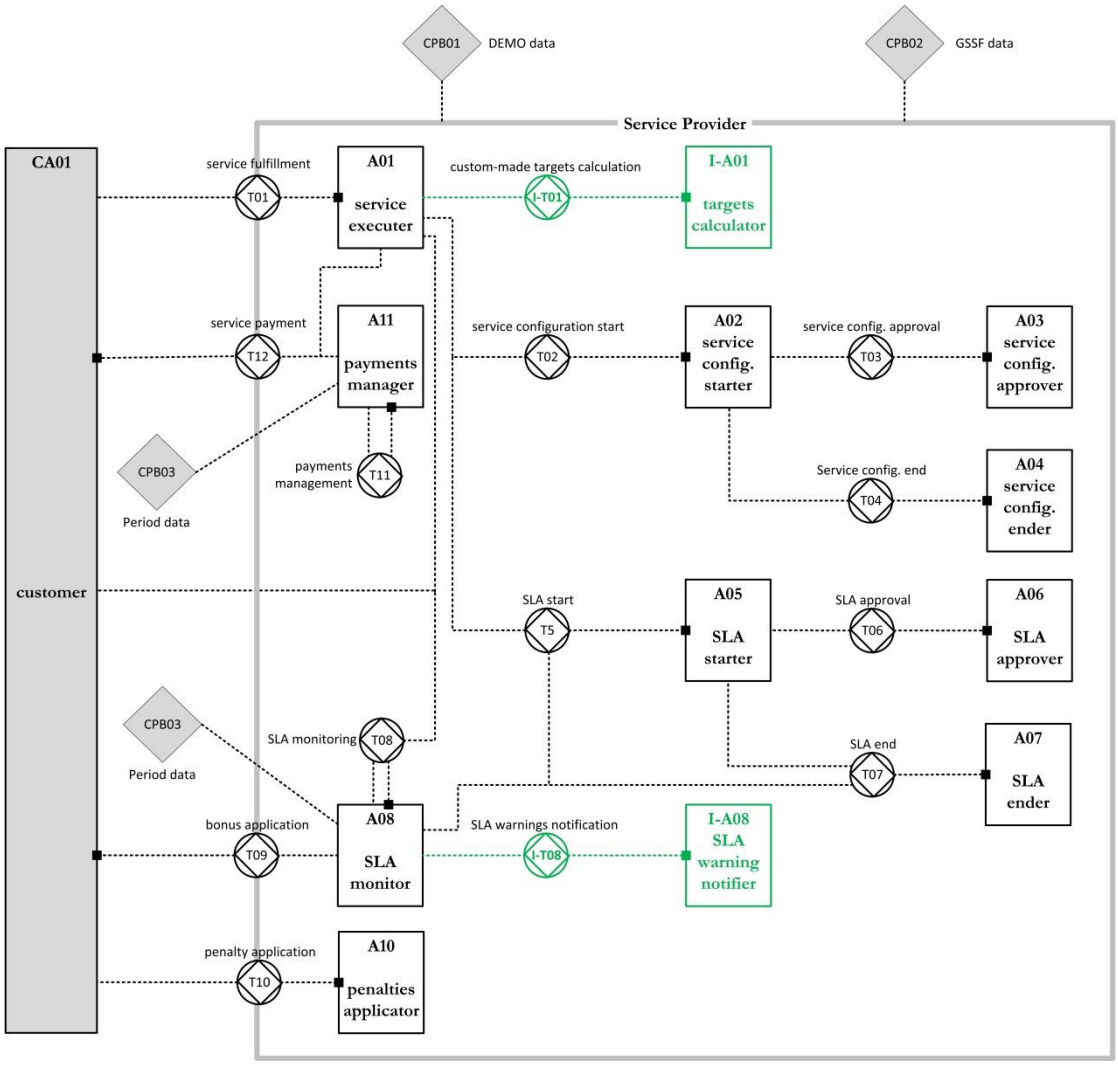


Figure 63 – Proposal Actor Bank Diagram (ABD)

The ABD may also be constructed as an extension of the ATD. This results into the Organization Construction Diagram (OCD).

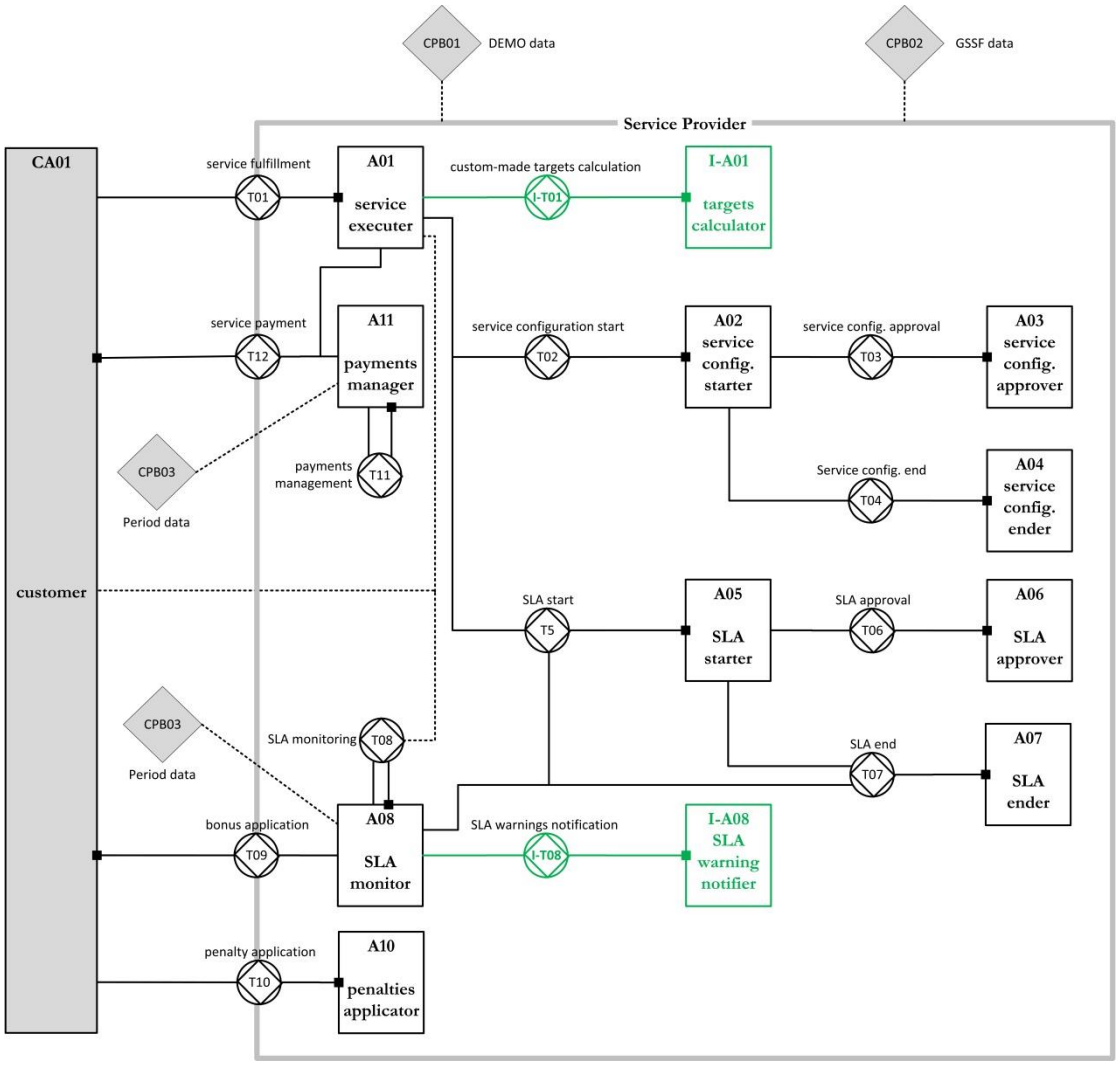


Figure 64 – Proposal Organization Construction Diagram (OCD)