University of Crete
School of Sciences and Engineering
Computer Science Department

Analyzing Service Networks from different perspectives using the Service Network Analysis & Prediction Tool (SNAPT)

by Giorgos Stratakis

Master's Thesis

Heraklion, November 2010

University of Crete
School of Sciences and Engineering
Computer Science Department

Analyzing Service Networks from different perspectives using the Service Network Analysis & Prediction Tool (SNAPT)

by Giorgos Stratakis

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science

Author: _____
Giorgos Stratakis, Computer Science Department

Board of enquiry:

Supervisor: _____
Christos Nikolaou, Professor - University of Crete

Member: _____
Dimitrios Plexousakis, Professor – University of Crete

Member: _____
Kostantinos Maggoutis, Researcher – FORTH-ICS

Approved by: _____
Angelos Bilas, Professor – University of Crete, Chairman of the Graduate Studies Committee

Analyzing Service Networks from different perspectives using the Service Network Analysis & Prediction Tool (SNAPT)

Giorgos Stratakis

Master's thesis

University of Crete – School of Sciences and Engineering
Computer Science Department

Abstract

Over the past years, the services industry has claimed a huge share of the income in the worlds most advanced countries. Services have become essential for the economic viability of modern enterprises. Businesses aim to offer services by providing the appropriate resources for the benefit of another party, in exchange to consume another service. Within this context, businesses are interconnected with each other to form complex service systems. There is a large community of stakeholders and business experts that study the dynamics of these service systems from different perspectives. Business analysts study complex service systems in terms of strategic decision making and business performance management. From a more technical viewpoint, Service-Oriented Architecture (SOA) can provide the implementation infrastructure for developing, monitoring and optimizing entire service systems. Service Networks offer an abstract way of viewing complex service systems as a set of independent entities that interact with each other to deliver services and serve their customers. In order to bridge the gap between business analysts and SOA (IT) infrastructure, Service Network Analysis & Prediction Tool (SNAPT) was designed with the objective of becoming a universal platform for developing, analyzing, monitoring, and optimizing Service Networks. To facilitate the different needs of each business expert, we extended SNAPT to support two methods for analyzing Service Networks from different perspectives. The first approach deals with the semi-automatic transformation of service networks to the corresponding abstract business processes. The second approach focuses on the value created by the service exchanges between the nodes of the network, applying a more economical analysis on Service Networks.

Αναλύοντας τα Δίκτυα Υπηρεσιών από διαφορετικές οπτικές γωνίες με τη χρήση του εργαλείου σχεδίασης και ανάλυσης Δικτύων Υπηρεσιών SNAPT

Γιώργος Στρατάκης

Μεταπτυχιακή εργασία

Πανεπιστήμιο Κρήτης - Σχολή Θετικών & Τεχνολογικών Επιστημών
Τμήμα Επιστήμης Υπολογιστών

Περίληψη

Τα τελευταία χρόνια ο κλάδος των υπηρεσιών αξιώνει ένα τεράστιο μερίδιο του εισοδήματος των προηγμένων χωρών του κόσμου. Οι υπηρεσίες έχουν καταστεί απαραίτητες για την οικονομική βιωσιμότητα των σύγχρονων επιχειρήσεων. Οι επιχειρήσεις έχουν ως στόχο να προσφέρουν υπηρεσίες, παρέχοντας τους κατάλληλους πόρους προς όφελος κάποιου, με αντάλλαγμα τη κατανάλωση κάποιας άλλης υπηρεσίας. Στο πλαίσιο αυτό, οι επιχειρήσεις είναι διασυνδεδεμένες μεταξύ τους, σχηματίζοντας πολύπλοκα συστήματα παροχής υπηρεσιών. Υπάρχει μια μεγάλη κοινότητα ενδιαφερομένων και εμπειρογνωμόνων πάνω στις επιχειρήσεις, οι οποίοι μελετάνε τη δυναμική αυτών των συστημάτων υπηρεσιών, κάτω από διαφορετικές οπτικές γωνίες. Οι αναλυτές των επιχειρήσεων μελετάνε τα πολύπλοκα συστήματα παροχής υπηρεσιών όσον αφορά τη λήψη στρατηγικών αποφάσεων και τη διαχείριση επίδοσης των επιχειρήσεων. Από μία πιο τεχνική σκοπιά, αρχιτεκτονικές συστημάτων που προσανατολίζονται στις υπηρεσίες (SOA), μπορούν να προσφέρουν την κατάλληλη υποδομή υλοποίησης, για την ανάπτυξη, εποπτεία και βελτιστοποίηση ολόκληρων συστημάτων παροχής υπηρεσιών. Τα Δίκτυα Υπηρεσιών προσφέρουν μια αφηρημένη οπτική γωνία για τα συστήματα παροχής υπηρεσιών, καθώς τα θεωρούν ως ένα πλήθος από ανεξάρτητες οντότητες που αλληλεπιδρούν μεταξύ τους προκειμένου να παρέχουν υπηρεσίες και να εξυπηρετούν τους πελάτες τους. Για να γεφυρωθεί το χάσμα μεταξύ των αναλυτών επιχειρήσεων και των υποδομών με αρχιτεκτονικές συστημάτων που προσανατολίζονται στις υπηρεσίες, το εργαλείο σχεδίασης και ανάλυσης Δικτύων Υπηρεσιών SNAPT αναπτύχθηκε με σκοπό να καταστεί μια γενική πλατφόρμα για την σχεδίαση, ανάλυση, εποπτεία και βελτιστοποίηση Δικτύων Υπηρεσιών. Για την ανταπόκριση στις διαφορετικές ανάγκες του κάθε εμπειρογνώμονα πάνω στις επιχειρήσεις, επεκτείναμε το SNAPT για να υποστηρίξει δυο μεθόδους για την ανάλυση Δικτύων Υπηρεσιών, από διαφορετικές σκοπιές. Η πρώτη μέθοδος αφορά τον ημι-αυτόματο μετασχηματισμό Δικτύων Υπηρεσιών στις αντίστοιχες αφηρημένες επιχειρησιακές διαδικασίες. Η δεύτερη επικεντρώνεται στην αξία που δημιουργείται από τις ανταλλαγές των υπηρεσιών μεταξύ των κόμβων του δικτύου, εφαρμόζοντας μια πιο οικονομική ανάλυση πάνω στα Δίκτυα Υπηρεσιών.

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to express my gratitude to my supervisor, Professor Christos Nikolaou for his assistance and support during the process of this thesis. I would also like to express my appreciation to the other members of my committee, Professor Dimitrios Plexousakis and Kostas Maggoutis for their valuable comments and feedback.

This thesis would not have been possible without the help of my friend and colleague Pantelis Petridis. His useful advice and consistent encouragement provided assistance to all levels of this research work.

Very special thanks to all of my close friends for their patience and support during the last years. They did not stop encouraging me, even in difficult times.

Finally, I would like to dedicate this work to my family. Their love and endless support have earned my respect.

1. Introduction

Services industry has become very dominant in the global economy over the last decades. In most of the worlds advanced countries, services represent between 60% and 80% of the Gross Domestic Product (GDP) [1]. The great interest in service economy and the need to study its implications has led to the creation of *Service Science* [2]. However, several different concepts exist in an attempt to define what services are. Traditionally, they are thought as a type of output product or the value added to the product as a result of interactions with the customer owning or using the product. Lately, there is a need to shift from a product-oriented to a service-oriented logic in order to better understand the term of service under the concept of service science. Within this context, services are considered to be the process of providing resources for the benefit of another party, in exchange for a service provided by the involved party [3]. These service processes are executed by service providers under complex relationships built with customers, altogether forming a *service system* [4].

As service systems connect to other service systems, they constitute networks of complex interactions, also known as S*ervice Networks*. Service networks offer an abstract way to model complex business networks, whose nodes represent the business partners, while the arcs indicate the interactions between the partners in terms of services provided or consumed by the partners. Several research challenges regarding the dynamics of service networks exist, such as designing and analyzing the interactions between service network partners, monitoring the behavior of network partners over the progress of time and finally optimizing the network to achieve better performance for the involved partners.

Nowadays there are several works that analyze service network dynamics on different perspectives. From the economical viewpoint, approaches [5], [6], [7] focus on analyzing how and why partners of the network co-operate with each other to produce value of some type. The purpose of such approaches is to configure service networks in a way to efficiently enable value co-creation and increase the overall value of the network. From the SOA and Business Process Management (BPM) perspective, service networks are studied in terms of organizational structure [8]. The focus is on bringing service networks to the level of business processes and make services available on service

delivery platforms [9]. In this context, services are the trends in the evolution of distributed computing over the web, based on open standards, such as RosettaNet[1] standards and are widely known as *Web Services* [10].

The main problem in these traditional approaches is that the BPM perspective cannot address human interactions, while the value-centered approaches cannot address business processes. Although SOA and web service technology were mainly focused on the perspective of computer science, the need to address organizational and strategic perspectives in order to explain SOA in practice has arose [11]. Thus, no approach has been accomplished yet to bridge the gap between service network models and service system implementations.

The primary contribution of this thesis is that it provides comprehensive methodologies to analyze service networks from both BPM and economical perspectives. To support these methodologies we use Service Network Analysis & Prediction Tool (SNAPT) [12] as the central platform for designing service networks, based on a notation model that represents the interactions between the network's partners and their services. Extending this notation with modeling constructs to represent how services are delivered as a result of other services composition within the context of service networks, has been the first task of this essay. We have also designed and implemented several extension mechanisms on SNAPT that will later on assist us in the development of the proposed methodologies for analyzing service networks from different viewpoints. These methodologies enable the SNAPT user to analyze the interactions between the network partners and assist the decision making in an attempt to optimize service networks.

The first step towards bridging the gap between business analysts and IT system implementations is to discover the business processes that take part within service networks. This thesis presents a solution that automates this step through a set of proposed transformation rules that map service network modeling constructs to business process modeling constructs specified using BPMN. To implement and illustrate these mappings, we have enabled the interaction of SNAPT with two different software tools. The SNAPT user is enabled to convert any modeled service network to the corresponding Abstract Business Processes supported by either the eclipse-based BPMN modeler or IBM's WebSphere Business Modeler Advanced. Moreover, we propose an approach to transform service networks to Verna Allee's value networks [7] and generate a report to depict the results of the Value Network Analysis (VNA) (see section

[1]RosettaNet, available online: http://www.rosettanet.org/

5.1) applied to the transformed value network. By conducting this approach the business analysts using SNAPT can discover the capabilities and the bottlenecks of the network through powerful business performance indicators.

The rest of the thesis is organized as follows. In chapter 2 we discuss the background theory and approaches of Service Networks and Business Process Modeling Notation (BPMN). Chapter 3 presents an extended service network model, which will be the basis of our approaches. We also show some extension mechanisms and new implemented components in SNAPT. Chapter 4 proposes a solution towards the alignment of service network level with business process level. In chapter 5 we introduce the design and the implementation of a qualitative, report-based analysis applied on service networks. Finally, chapter 6 concludes the thesis by summarizing the work accomplished and discussing ways of extending it in the future.

2. Background

Nowadays, many companies have abandoned the logic of producing and selling products to customers and focus on the service-oriented logic of assisting the customers in their value creation process [13]. This emerging and dynamic service-oriented economy requires finding ways to design and deliver services to meet the new needs [14]. To address these needs, service science communities established new modern business models, which are approached from many different aspects and under many labels such as *Service Systems* [14], *Service Ecosystems* [15], *Value Networks* [7] [16], *Service Value Networks* [17] or *Service Networks* [18]. The general idea of such business service ecosystems is depicted in figure 1, where people, technology, service systems connected via value propositions and shared resources cooperate to produce value. In this thesis we focus on *Service Networks* to study such business models. In brief, *Service Networks* are independent organizations that interact with each other to deliver services and serve their customers. In the next section we discuss about the theoretical background of service networks that constitute the basis of modern analysis approaches.

Figure 1: Business Service Ecosystem

2.1. Defining Service Networks

Service networks theory came into existence from Porter's original concept of value chains [19]. Value chain model was originally used as a strategic analysis tool to define and examine the value adding activities across an organization, for the purpose of pursuing a competitive advantage. In service networks, business analysts take into consideration both social and technical resources, when studying these networks and evaluate their existence. Today, many theories on service networks exist, each one with a different business analysis perspective. At this point we will discuss two traditional theoretical approaches suggested by Clayton Christensen and Stabell – Fjeldstad respectively, which despite the fact of having different viewpoints they can be applicable to many same situations.

2.1.1. Clayton Christensen's definition

In Clayton Christensen's term, a Service Network is defined as "the context within which a firm identifies and responds to customers' needs, solves problems, procures input, reacts to competitors, and strives for profit" [20]. Whenever a new technology or innovation arises, the competitive strategy of each organization in the network is determined by its past choices in the market. So, it is of great importance for an organization to preserve the value of a key product, by fulfilling any emerging demand in the specific market and rather ignoring any changes made in a different market. Thus, in Christensen's Value Networks, organizations are trained to be dedicated to the customer's need, by finding and carrying out any factor that derives market demand.

2.1.2. Stabell and Fjeldstad's definition

Based on Michael Porter's Value Chain [19], Stabell and Fjeldstad define Service Networks as a value creation logic derived from the relationships and interactions between the organization-nodes of the network [21]. Their network is composed of a set of customers, a service that is consumed by the customers enabling interaction between them, an organization that offers this service and a set of contracts that gives access to the service. Phone – telecommunication is an indicative example of such networks, which include the phone company that provides the service and a set of customers that sign a contract with the company and therefore grant access to communicate with all the other customers of the company. According to Stabell and Fjeldstad, this interaction between the customers is the main reason of value creation, which assists the progress of the network. The network focuses on a customer's unique problem and configures its process and activities in a manner to solve this problem. Although, Stabell and

Fjeldstads's approach may be considered "old thinking", it has been applied in many cases until today; one such is the comparison website sector in the UK [22]. The authors explain the growth of comparison websites using Stabell and Fjeldstads's Value Network and suggest a general way to use their approach for e-commerce markets.

2.1.3. Overview

Comparing these two theories, one could state that Stabell and Fjeldstat focus on the enterprise perspective, while Christensen takes into consideration the fact that the enterprise is part of a system of enterprises that grow relationships between each other. Another characteristic that comes in contrast to Stabell and Fjeldstat's work is that Christensen's value networks are centralized in products and services rather than in the organizations that take part in the network. Following in this section we present various research studies that apply a modeling language to design and analyze service networks and how organizations evolve in these networks of inter-organization exchanges.

2.2. Modeling Service Networks

In this section, we will focus on research conducted on the field of Service Networks and expand on relevant methodologies. Some of the presented approaches apply the most to economists and business managers while others consider Service Networks from an ICT perspective. As a result, some researchers define and interpret Service (Value) Networks in such a way that makes it impossible to converge with others. Please, also note that some approaches refer to Service Networks as "Service Value Networks" or just "Value Networks". We remark declining views on value as well.

2.2.1. ValueNet Works™ Analysis

Within the context of ValueNet Works™ methodology, Service Networks are referenced as Value Networks. The purpose of such networks is to create the most benefit for every organization involved in the network by modeling the generation of value, both tangible and intangible. In this concept, tangible value is created from every transaction between two organizations that includes exchange of goods, services or revenue. For example, any payment or invoice or contract is considered to generate tangible value. On the other hand, any exchange that includes knowledge or benefit produces intangible value. The most common example of intangible value is when an organization requests a group of volunteer for their expertise in exchange for the benefit that outrages from the prestige of the specific affiliation. Organizations that make such transactions over the

network intend to build trustful and long term relationships that will keep everything run stable. Verna Allee was a leader in discovering the importance of knowledge management for the organizations business strategy and the conversion of knowledge into financial value [23].

Verna Allee applied Value Network Analysis (VNA) [7], as a way to achieve value conversion in a business and accomplish greater value for the business itself. By value conversion, Verna Allee means the know-how of a business to convert any tangible or intangible asset in a negotiable form of value, which influences the forthcoming existence or success of the business as a whole. VNA is an approach to model, visualize and analyze any model of business interactions, which is composed of three basic elements: participants, transactions and deliverables [24]. Participants can either be individuals or groups of humans that make any business decision. Transactions represent any tangible or intangible interaction between the participants of the Value Network. Any tangible or intangible asset that flows from one participant to another corresponds to the deliverables. The analysis begins by defining and visualizing not only all traditional tangible business transactions, but also all critical intangible knowledge transactions that exist between the nodes of the network. Nodes within a Value Network can either represent roles or the actual participants such as individuals or business units. Once all basic elements of the Value Network are specified, several analysis methods are applied to provide useful insights to the business analysts that will assist the business in finding new value opportunities. The first analysis approach also known as Exchange Analysis is applied to depict whether the Value Network's transactions are healthy and able to exist. When applying this analysis method, several performance metrics are taken into consideration such as the balance between the overall tangible and intangible transactions of the network, the overall pattern reciprocity, meaning whether a single participant that offers a large number of transactions, at the same time receives a fair reciprocation. The Exchange Analysis is the appropriate tool to analyze the Value Network in a whole and trace any lacking step in a key process of the network, featuring the intelligence of the network. VNA certifies that every participant of the Value Network receives positive value for each tangible or intangible input by the Impact Analysis. Every input of a participant causes a corresponding response that can either increase or decrease a set of tangible or intangible assets, including financial, assets business relationships, competence and structure. By analyzing every single input together with the cost or benefit that it brings to the participant and consequently to the network, one can depict new techniques to handle each input. Thus, the Value Network will make better decisions, build stronger and trustful relationships and increase its

overall financial image. Finally, the Value Creation Analysis focuses on analyzing how a particular participant enhances value to other participants of the Value Network, by adding new tangible or intangible value for each value output. Usually one participant can "think" of many ways to extend its value outputs, primarily by taking advantage of the intangible value that has been developed. The Value Creation Analysis is pertinent to the Impact Analysis, since the participant must also consider the impact an output has on the participant that receives it. Value Network Insights is a web-based application that provides analytics reports for measuring the performance of the Value Network, based on the VNA methodology. We will discuss more on Verna Allee's VNA in chapter 5, where we introduce a methodology to convert a Service Network model to Verna Allee's Value Network and apply a qualitative analysis on Service Networks, through the Service Networks Analysis & Prediction Tool (SNAPT).

2.2.2. E3value Modeling

The e3value methodology, as firstly introduced by Jaap Gordijn and Hans Akkermans in [25], is another methodology for designing and modeling Service Networks, using ontology and conceptual modeling techniques. In short terms, e3value explores the organizations that take part in offering a particular service and analyzes the economic sustainability for each of these organizations with the use of profitability sheets. As stated in [26], it displays "who is offering and exchanging what with whom and expects what in return". To determine the network of organizations the e3value methodology applies the following modeling constructs. An Actor is an economically independent entity equivalent to Verna Allee's participant. A Value Object is anything that depicts an economic value for at least one actor. A service, a product or money are significant examples of Value Objects, which are exchanged between the Actors. Value Ports model these exchanges, when an Actor requests or offers a specific Value Object. Value Interface groups many in-going and out-going Value Ports together, modeling at the same time the concept of bundling. Whenever actors offer a Value Object, they must receive something of equivalent economic value in return, showing the economic reciprocity of the network. Value Exchange connects two Value Ports together, displaying which actors are trading Value Objects with each other. Market Segment groups actors that perceive the value of each object in the same way. Value Activity indicates the actions that each actor performs to obtain an expectable profit from it. To explain and estimate the economic value of the network, the e3value methodology uses dependency paths. A dependency path is composed of start and end stimulus dependency boundaries, connections between the interfaces of the same actor and

complex dependency elements such as AND/OR gateways, which examine the customer's needs based on specific use case scenarios. The e3value methodology is a graphical approach, which uses the previously mentioned modeling constructs to design any network of enterprises and analyze the profitability of the network. An open source software tool has been developed for this reason [27]. The first stage that will initialize the use of e3value methodology is the discovery of an innovative service idea based on the customer's needs and market information. Once the idea is formulated, the service providers that will co-ordinate with each other to deliver the final service must be determined, according to value-based requirements engineering [26]. The business model can now be designed graphically, using the previously described graphical notation. After visualizing the e3value model, the business analyst can take advantage of the tool's features and generate profitability sheets for the model. These sheets identify the way the value flows between the actors, indicating the strengths and weaknesses of the model and whether there is commercial success for each actor involved in this network of enterprises.

The e3value framework is widely used for business modeling in the context of e-commerce, abstracting from the process details, making it easier for the business analysts to aim attention at economic viability [28]. E3value focuses on business models, limiting the analysis to the "what" of the e-business idea and leaving out the "why". It seems to be an ideal approach for analyzing rather small business models. As enterprises evolve, matters such as market analysis, competitiveness and strategic policy are of special interest for the business analysis. The lack of e3value in focusing in strategic analysis reveals the weakness of the specific methodology. To handle e3value's limitations, Hans Weigand proposes an extended approach of e3value, called c3value, which we will present next.

2.2.3. C3value Modeling

As introduced in [29], c3-value is a business model focusing on the strategic analysis. Based on e3-value's modeling techniques, c3-value proposes an extended approach that supports deep analysis evaluating three factors: competition, costumer and capability. In competition analysis, c3-value takes into account secondary value objects that strengthen the value of the primary e3-value objects exchanged between the actors. Any manual for technical support of a product is an example of complementary value object that distinguishes the competitive actors offering this product. Customer analysis measures the value of the offering value object based on the customer's key resources, in an attempt to fulfill the customer's current and future needs. In capability analysis an

actor has to identify the capabilities of a specific value activity that will offer him a competitive advantage and the way to sustain this advantage. To support this strategic analysis, the authors of [29] have enriched e3-value model and introduced three new e3-value "views" (Customer Value Model, Capability Resource Model, Competitive Value Model). However these three models seem to be quite abstract at the current time as they are not supported by a well-established and evaluated methodology. Moreover, the focal point of the c3-value model is the competitor and the customer actor along with the composition of the value objects they exchange, ignoring the strategic motivation that derive from the inter-dependencies of the network perspective.

2.2.4. IBM's WRC Value Network Model

Alain Biem and Nathan Caswell of IBM's Watson Research Center (WRC) presented a new framework for modeling networks of inter-organization exchanges [30]. Their view of a value network consists of a set of economic entities and a set of offerings that connect the economic entities with each other, whose purpose is to deliver a common value proposition to an end consumer. An economic entity can either be a firm, a business unit or an individual and is responsible for creating value by transforming input offerings to output offerings. An offering is transferred from one economic entity to another and can either be a product, a service, knowledge or brand. In this model, an offering is transferred either as an input or as an output. The transformation of input offerings to output offerings, through the economic entities' valuation functions, is the way value is created. Every such transformation of offerings may also include some transfer of revenue between the economic entities. The end consumer has the special role of evaluating and accepting the value proposition of the whole network, through his own valuation function. A value proposition clearly refers to the profits, not necessarily financial, that the end consumer will gain from the consumption-transformation of all offerings of the network. Biem and Caswell present a detailed methodology to achieve a two perspective strategic analysis. The first perspective they propose is to target the creation of value and the transformation of offerings towards optimizing the value proposition at the end consumer node not to immediate customer nodes. In this way, the whole network must remain healthy to appreciate the overall value proposition and all every node of the network can identify a strategic plan. A firm applying this perspective can have financial profits from the coordination of the network in a competitive environment. The second perspective supports that "the flow of offerings and flow of financials are separated", meaning that financial revenue is not exclusively created via transferring offerings but by the "strategic use of the whole network". An

offering may or may not include financial profits for an economic entity, which may choose the strategy of "volunteering" for a specific offering. Open-source-based companies are a typical example of such cases. Bien and Caswell's methodology is very important for analyzing value networks as they take essential issues into consideration such as network stability and health. However their methodology has not been applied and validated through real cases studies.

2.2.5. B. Blau's Service Value Networks

In [31], Benjamin Blau presented a formal framework for modeling Service Value Networks. Blau consider Service Value Networks as networks of service providers that cooperate to compose complex service deliveries based on customer's requests. The formal model is composed of a set of service providers that provide a set of service offers. Service offers that are in consistent with each other, depending on their inputs and outputs reveal a composition relation. They are connected directly with each other in a graph-based structure, starting from a source node and ending at a termination node. Each possible connected set of service offers within this graph forms a path, processing a specific customer request, which consists of business functionalities. Each of these functionalities can be fulfilled by a variety of different potential service offers which are candidate nodes for the path to be obtained. Moreover, each service provider can provide more than one service offer implied by an ownership relation. Each service offered exhibits a service configuration, which differentiates the capabilities of each service in the network. The network's ability to compose a complex service offer matching the customers' needs is the main factor that determines the overall value of the network. Based on the previously described model, Blau introduces a mechanism implementation that enables the composition of services to form a complex service request [32]. The allocation of service components is based on the customer's needs expressed in Quality of Service (QoS) characteristics of the complex service. In [33], strategic behavior of the service providers is studied in means of creating trustful relationships between the customer and the providers. The main goal that must be accomplished in such networks is to determine price for the final complex service by methodically compose service offers in accordance to customer's service requests. The current mechanism has been applied in two simple strategies showing that collusion can be effective for the service providers. Further research, in the sense of identifying additional strategies and comparing the findings with alternative existing mechanisms, should provide interesting and beneficial results.

2.2.6. Overview

Modeling Approach	Key concepts	Business Process	Human interactions	Purpose
ValueNet Works	Participants, Transactions, Deliverables	No	Yes	Generate both tangible & intangible value in existing networks
E3value	Actors, Value Objects, Value Exchanges, Value Ports, Market Segments, Value Activities, Dependency Paths	Yes	Yes	Create benefit for all actors of the network
C3value	E3value's key concepts, Secondary Value Objects	Yes	Yes	Support strategic analysis in e3value modeling
IBM WRC	Economic Entities, Offerings, Financials, End Customer, Value Proposition	No	Yes	Optimize value by strategic use of the networks
B. Blau	Service providers, Service Offers, Customer Requests, Business Functionalities, Ownership relations, Service Configurations	Yes	No	Deliver and determine price for complex services

Table 1 : Overview of different Service Networks modeling approaches

In table 1 we provide an overview of the different approaches on modeling Service Networks, which are already presented in this section. The first column illustrates the key concepts that describe the service network model of each approach. The next two columns show whether each modeling approach typically shows business process and human interactions respectively. The last column summarizes the purpose that each modeling approach focuses on.

3. Extending the Service Network Analysis & Prediction Tool (SNAPT)

The first contribution of this thesis, which is presented in this section, deals with the design and the implementation of extending mechanisms regarding the Service Network Analysis & Prediction Tool (SNAPT). SNAPT is a software tool which allows the business analysts to model complex Service Systems and analyze their commercial viability [12]. It is an open-source, eclipse-based implementation, based on a service network model, which we discuss in section 3.1. In our approach we extend this service network model in an attempt to bridge the gap between Service Networks and Business Processes. We also present an architecture provided by SNAPT to support extensions and the way we use them to implement various facilities of the tool. Furthermore, we implement a Key Performance Indicator (KPI) Repository, based on service-oriented architecture (SOA). In this way, each SNAPT application can anytime synchronize the local KPI Library with the KPI Library of the repository.

3.1. Modeling Service Networks: a Transformation Services Laboratory (TSL) approach

The formal model introduced in [5], was the first result of the research made by TSL[2] and IBM Watson Research Center (WRC)[3] in an attempt to represent how service is delivered through complex service systems. It focuses on an economy point of view, computing the value that is created as a result of the relationships that are developed between the partners of the service system, taking into account the partner's satisfaction. The approach presented in chapter 4 has a different perspective, focusing on the organization of service systems and the business models that derive from them. The first step towards automating the procedure of discovering the business processes that take place in a complex service system is to find a way to model these service systems, also known as Service Networks. Service Networks, i.e. networks offering services, which are derived from the composition of other services delivered by a set of service providers, are represented by the following modeling constructs.

[2] Transformation Services Laboratory Website: http://www.tsl.gr/
[3] Watson Research Center Website: http://www.watson.ibm.com/index.shtml

A *Service Network* is composed of several *Business Entities*, which may offer or consume one or more *Services*. A *Business Entity* is an economic independent entity, equivalent to e3value's Actor and Verna Allee's participant, which has revenues and costs deriving from the services offered and the services consumed respectively. Depending on their specific characteristics, each Business Entity can be further recognized as a *Service Sub-Network*, a *Participant*, an *End Customer* or an *Enabler*. A *Service Sub-Network* has a double role in a Service Network. It can, not only be a part of a Service Network inheriting the features of the Business Entity, but also be a Service Network itself. It enables the concept of internal Service Networks, providing an abstract way of modeling Service Networks. An *End Customer* is a special class of Business Entity that has zero total revenue as it can only consume a Service and is not able to deliver one. They are basically used to recognize the final Services that are delivered within a specific Service Network. *Enablers* have a specific role in a Service Network. They are not considered strategic partners in the network, but they offer specified type of Services, which "enable" the delivery of a specific Service. The delivery of a product undertaken by a business entity different from the one that sells the specific product is an indicative example of such services. A *Participant* is the ad-hoc Business Entity of a Service Network, which is not a Service Sub-Network, an End Customer or an Enabler. *Role* is used to model a set of Business Entities that share common properties, meaning that they offer and consume the same Services, as a single node in the Service Network. A *Service* represents either a product or a service and is either offered or consumed by Business Entities, producing respectively revenue and cost to the Business Entities. If a Service has zero revenues, then the type of service offered is equivalent to the intangibles as introduced by Verna Allee in [34]. Key Performance Objectives are associated with a specific Offering or Consumption, depicting in certain values the objectives of the Business Entities in financial and operational performance indicators, regarding the offering or the consumption of a single Service. Examples of such indicators, also known as Key Performance Indicators (KPIs) [35], are the overall execution time of a service delivery or the percentage of requests fulfilled for a specific service. Based on this model, Service Network Analysis & Prediction Tool (SNAPT) was designed and implemented as presented in [12]. We will discuss about SNAPT in more details, together with some extension mechanisms that it supports, in the next section.

3.2. SNAPT Extension Mechanisms

Service Network Analysis & Prediction Tool (SNAPT) enables the business users to visualize Service Networks and analyze their behavior. It is a tool developed on the

Eclipse platform, based on the concept of plugins. Plugins are independent components that provide additional function to the system. A plugin may include code, documentation or other data that can be used by other plugins. To integrate plugins with each other, the developer is enabled to define extension points in each plugin. An extensions point is a mechanism, which defines a type of contract that the interested plugins must agree with. Any interested plugin must implement this contract in order to achieve the integration, even if they know nothing for the plugin being extended beyond the scope of the extension point. Taking advantage of these extension mechanisms provided by the Eclipse platform we have built several plugins to extend SNAPT's functionality.

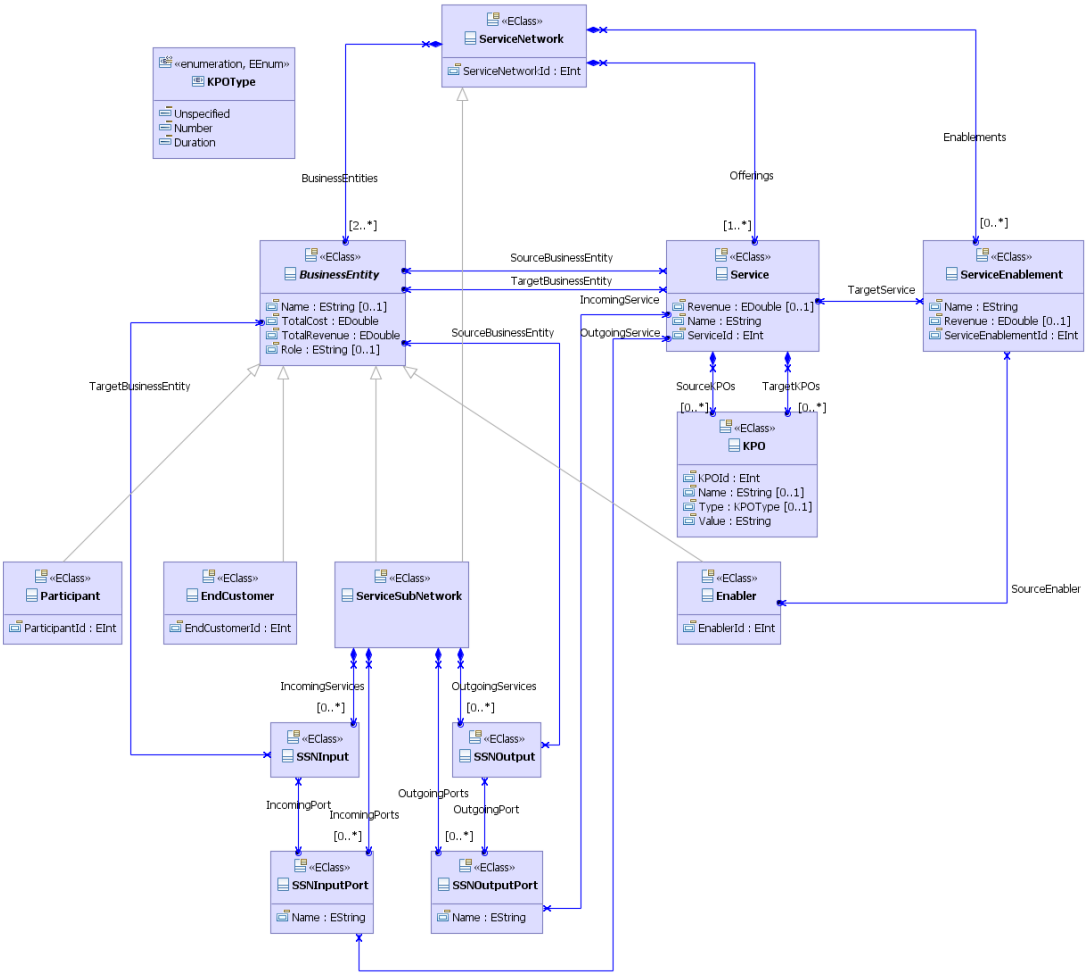


Figure 2: The Ecore Service Network meta-model

A set of *sequencing* plugins are responsible for extending the SNAPT's core meta-model. SNAPT is an Rich Client Platform (RCP) [36] application developed on the Eclipse

Graphical Modeling Framework (GMF)[4]. GMF provides code generation framework that enables developers to implement graphical editors for modeling languages built on top of the Eclipse Modeling Framework (EMF) [37], which provides modeling tools for Eclipse applications based on structured data models. The EMF framework encloses a meta-model, also known as Ecore, which describes the SNAPT application data. In figure 2, the Ecore Service Network meta-model is displayed as a UML diagram, based on the model described in section 3.1. In the next section, we will present the *sequencing* plugins, which extend the service network meta-model in the context of SNAPT. These extended modeling constructs are used to depict the services that are obtained by composing other services. Another plugin developed in the scope of this thesis and discussed in section 3.4 is the *kpi-repository*, which enables the business user to assign KPIs to any service network designed in SNAPT, selected from a huge online KPI repository. Furthermore, we implemented two plugins (Eclipse Modeler, WBM) that enclose the transformations from service networks to business processes for two different software tools as presented in chapter 4. Finally we have also extended the SNAPT functionality to support a qualitative analysis on service networks (see chapter 5). Figure 3, displays the SNAPT architecture, including the plugins that extend the SNAPT's core functionality.

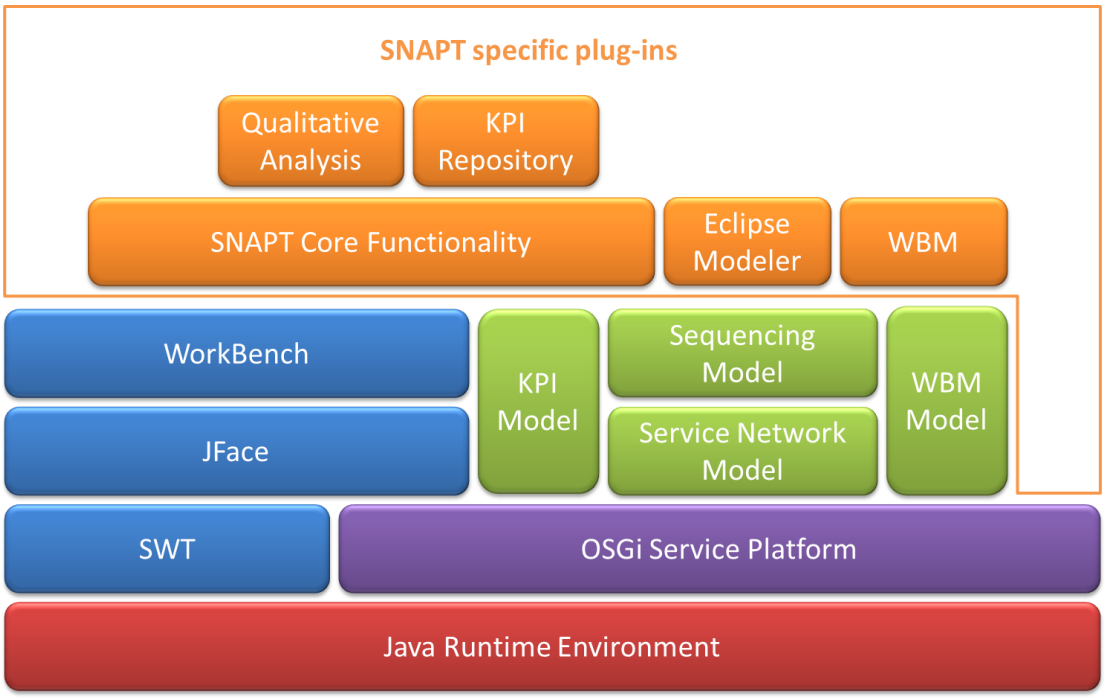


Figure 3: The extended SNAPT architecture

[4] Eclipse Graphical Modeling Framework (GMF), available online: http://www.eclipse.org/gmf

3.3. Extending the Service Network core meta-model

After establishing the meta-model that will be used to depict the Service Networks, the next step to discover the business processes that occur in a Service Network is to find a mechanism that will describe the way some services are composed of other services. The EMF framework is very powerful, allowing to define base models and then to extend them. Using the Service Network Ecore of section 3.1 as a base model, we have taken advantage of the EMF's extending mechanism to easily support the sequencing behavior in the Service meta-class. The specific methodology can be used for any future extension in the SNAPT data model. Figure 4 shows the new extended Service Network Ecore model as a UML diagram. Not all classes from the core Ecore model are depicted in the figure to accommodate the display of the meta-model classes. Only the Service meta-class is displayed, because of its connection with the new modeling constructs. All the objects of the core service network meta-model are illustrated in figure 2.

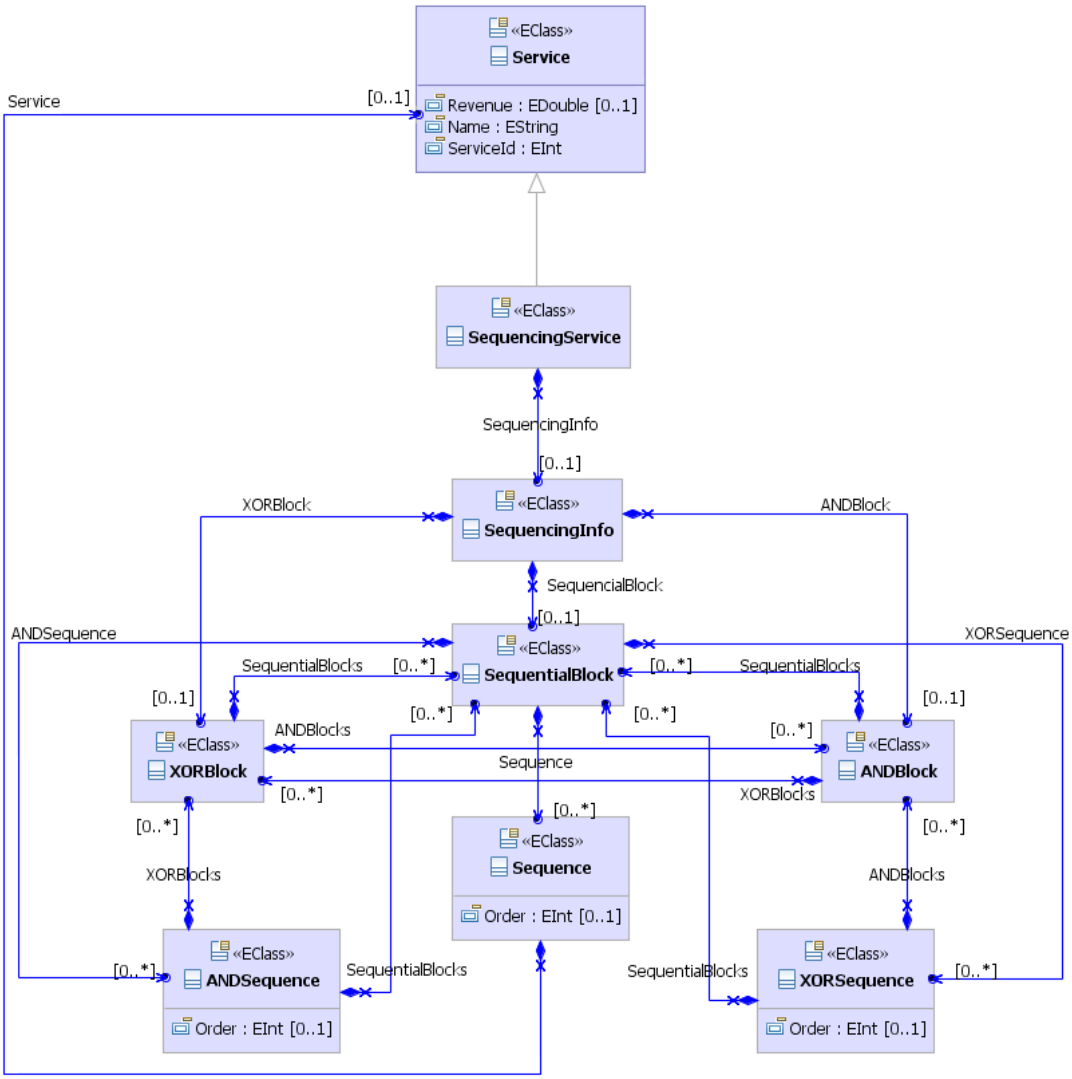


Figure 4: The extended Ecore Service Network meta-model

3.3.1. The Sequencing Ecore model

The reason for providing this extended meta-model is the need to describe how some services delivered in a Service Network are composed of other services. This kind of information is stored in the service object itself distinguishing it from the other services that are not a result of any kind of service composition. Such a service is called *SequencingService* and owns the attributes inherited by the core Service meta-class plus some extra sequencing information that describes in detail what services and in what sequence they must be delivered to enable its own delivery. This information is depicted by the *SequencingInfo* meta-class, which is the root element in a tree-based data structure that models the sequencing details of each Service. The SequecingInfo object must have exactly one of the following children: a *SequentialBlock*, an *ANDBlock* or a *XORBlock*. The SequentialBlock implies that any of its children elements is delivered in series, one after the other. All elements of an ANDBlock are required to be delivered in parallel, while exclusively one of the total elements of a XORBlock is chosen to be delivered. Both ANDBlock and XORBlock must have at least two children elements. A SequentialBlock must have one or more children elements of *Sequence*, *ANDSequence* or *XORSequence* kind. An ANDBlock must have two or more children elements of Sequence, or XORSequence kind and a XORBlock must have two or more children elements of Sequence, or ANDSequence kind. Sequence is the final element of the sequencing information structure, which has a specific service and an integer indicating the order in which this service must be delivered, in case the Sequence object is contained in a SequentialBlock. The ANDSequence and the XORSequence meta-classes have the same properties as ANDBlock and XORBlock respectively, plus the order in which they must be executed, when included in a SequentialBlock. These modeling constructs describe fully any kind of sequencing information a service may have in the context of Service Networks. So to sum up, any SequencingService will finally be decomposed to several Services that will be delivered in a sequence defined by the described gateways.

3.3.2. Defining Constraints in the Sequencing Ecore model

Defining constraints in the Ecore model is not a generic action in eclipse. To ensure the data integrity of the Sequencing Ecore model, we have integrated Object Constraint Language (OCL) in Ecore. OCL [38] is a powerful language that enables the construction of query expressions and constraints on object-oriented models and other object modeling artifacts. It's a simple to read and write declarative language, and it is rather convenient as the constraint is based on simple assertions on an object's attributes. To enable the embedded OCL functionality the appropriate delegate functions must be referenced by the annotation within the Ecore model. These functionalities are defined in the Ecore model with the following XML code:

```
1. <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore">
2.         <details key="settingDelegates" value="http://www.eclipse.org/emf/2002/Ecore/OCL"/>
3.         <details key="invocationDelegates" value="http://www.eclipse.org/emf/2002/Ecore/OCL"/>
4.         <details key="validationDelegates" value="http://www.eclipse.org/emf/2002/Ecore/OCL"/>
5. </eAnnotations>
```

The setting delegate function is responsible for handling computation of features. It is a class implementing an interface with methods that are called by the EMF runtime to access the feature's value. Via the invocation delegate, one can define a certain behavior as an annotation in the Ecore operation, which will be called by the EMF runtime to execute the operation's behavior. Validation delegate is implemented as java methods defining constraints and invariants, which are called by a specific validator to perform validation. We have defined several invariants that are responsible for validating the sequencing Ecore model. Such invariants are represented as annotations in the Ecore model by defining the name of the constraint together with the OCL expression associated with the invariant. The following XML code defines the invariant named "SequencingInfoMustHaveOneChild" and the OCL expression with body "(self.SequentialBlock->size() + self.ANDBlock->size() + self.XORBlock->size()) = 1", indicating that each SequencingInfo object must exclusively have one SequentialBlock, or one ANDBlock or one XORBlock as a child to be valid:

```
1. <eClassifiers xsi:type="ecore:EClass" name="SequencingInfo">
2.         <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore">
3.                 <details key="constraints" value="SequencingInfoMustHaveOneChild"/>
4.         </eAnnotations>
5.         <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore/OCL">
6.                 <details key="SequencingInfoMustHaveOneChild"
7. value="(self.SequentialBlock->size() + self.ANDBlock->size() + self.XORBlock->size()) = 1"/>
8.         </eAnnotations>
9.         ...
10. </eClassifiers>
```

In the same way, we have also defined invariants for the ANDBlock, XORBlock, ANDSequence and XORSequence meta-classes to validate the same constraint of having at least two children elements for each of these meta-classes. We present the OCL body of the specific invariant, expressed by the following Ecore annotation as XML code for the ANDBlock meta-class:

```
1. <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore/OCL">
2.     <details key="ANDBlockMustHaveAtLeastTwoChildren"
3. value="(self.SequentialBlocks->size() + self.XORBlocks->size()) >= 2"/>
4. </eAnnotations>
```

These OCL expressions are queried by the EMF runtime environment, ensuring the creation of valid sequencing data objects. Next we will present how a SNAPT user can create such objects via the sequencing interface we have implemented using eclipse mechanisms.

3.3.3. Implementing a SNAPT view for the Sequencing Information

Prior to this section, we have presented an extended meta-model for Service Networks, supporting a way to depict the sequencing information of a single Service object, i.e. how this service is composed of several other services. We have also provided a mechanism to retain the integrity of the sequencing model via OCL invariants. In this section we will

demonstrate how the user of SNAPT is capable of viewing and editing the sequencing information via a user interface we have implemented.

SNAPT or any Eclipse-based RCP application contains a set of workbench windows. Each workbench window contains one or more pages, and each page contains a set of editors and views. Views are visual compartments that enable users to display or navigate data objects of any type. A view's responsibility is to demonstrate data from a domain model, grouping similar types of objects into the view. We have implemented such a view in the SNAPT to allow the user to create and edit data objects from the sequencing meta-model. Figure 5 shows the graphical representation of this view in SNAPT.

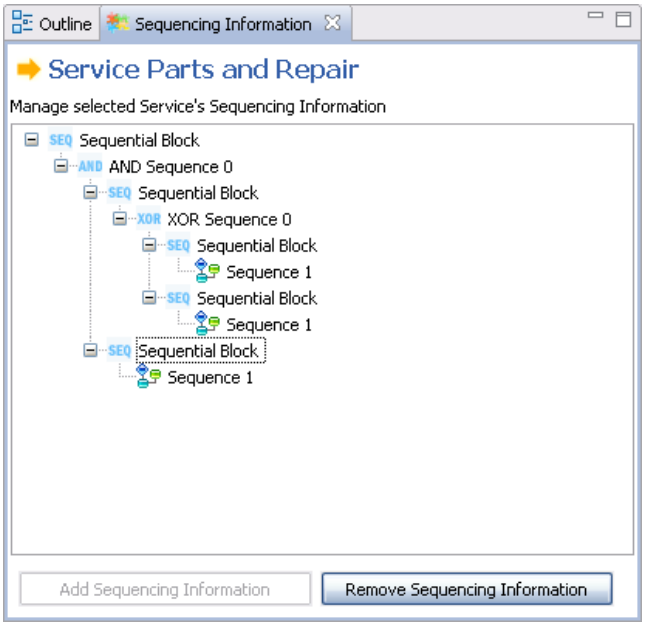


Figure 5: The Sequencing View

Apart from creating the view via the extension mechanisms provided by Eclipse and designing its graphical representation, the actions associated with this view must also be defined together with an interaction with the other existing views of the SNAPT. The sequencing view like any new view that is added in SNAPT, co-exists with other views and selection within the sequencing view may affect the input of another. The mechanisms that control the selection in a view are the *SelectionListener* and the *SelectionProvider*. The first one will listen for the selection of objects within other views, while the latter will fire selection events that happen within the sequencing view. If the sequencing view is interested in the selections that take place within other views, it must register this interest within the class that implements the SelectionListener. In the same way, any other view that is interested to the selection of objects within the sequencing view must also register their interest to the sequencing view's SelectionProvider. Through the sequencing view the user may define the sequencing information for any service provided in a Service Network. When assigning the sequencing information of a specific service delivered by a Business Entity, the user can choose only from the set of services that are consumed by the same Business Entity. Figure 5, for example displays the sequencing information of service named "Service Parts and Repair" that is provided by the Business Entity Dealers to the Business Entity

Car Owners, as illustrated in the service network of figure 6, inspired by the Car Repair Industry. In this case, the sequencing information indicates that "Parts and Repair" service is a result of the following composition of services: the "Parts" service provided either by the Car Original-Equipment-Manufacturer (OEM) or the Third Party Suppliers, and the "Repair" service provided by the Technicians.

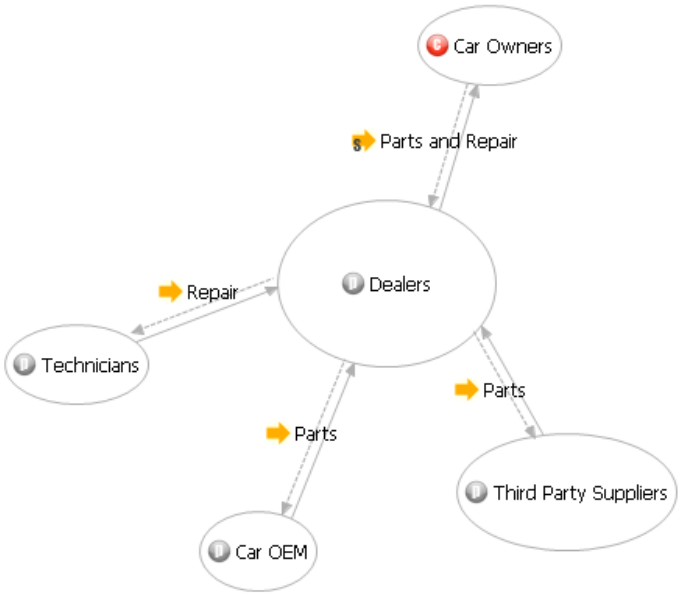


Figure 6: Simple Car-Repair Service Network

3.4. An online Key Performance Indicators (KPIs) repository

As already stated, KPIs are numeric variables measuring the business health and performance. They enable the business users to measure essential activities of the companies and monitor how these activities influence the business results. SNAPT provides a KPI editor to support the definition of new KPIs and the creation of KPI library for each standalone SNAPT application. The SNAPT's user may manually define any kind of KPI concerning the business goals of the service network participants. However, there is a large number of KPIs defined in various industries making it very difficult for the user to handle the persistence of the KPI library within the SNAPT application. To release the user from doing so, we have implemented a REST-based SOA repository, which provides a highly available storage service for any location independent SNAPT application. This KPI repository is based on APQC's Process Classification Framework (PCF) [39]. APQC is a non-profit organization that provides business measurements for hundreds of companies worldwide from all industries. PCF organizes operating and management processes that apply to almost any business, regardless of industry, size, or location. Thus, there are almost 800 KPIs stored in the SOA repository, grouped under various industry-based categories and subcategories similar to the public open KPI database of [40].

3.4.1. Implementing the KPI Repository

To implement the KPI Repository we used the following technologies: MySQL [41] database in order to store the KPI data, the Java Hibernate framework [42] to map database tables to Java objects so that the database tables can be queried and used as Java objects and a RESTful [43] architectural design that is responsible for managing the KPI data and building a service for any SNAPT application client that needs to synchronize the local KPI Library with the KPI data stored in the repository. Figure 7 illustrates the overall architecture of the KPI repository. Representational State Transfer (REST) approach to building Web Services is associated with the interactions between the client and the services that are limited to the resource exchanged between them. To access theses resources, the client, in our case any SNAPT RCP application, must achieve communication via the HTTP protocol. Due to its lightweight nature, in contrast to Remote Procedure Call (RCP)-based Web Services, they enable building high performance and reliable systems.

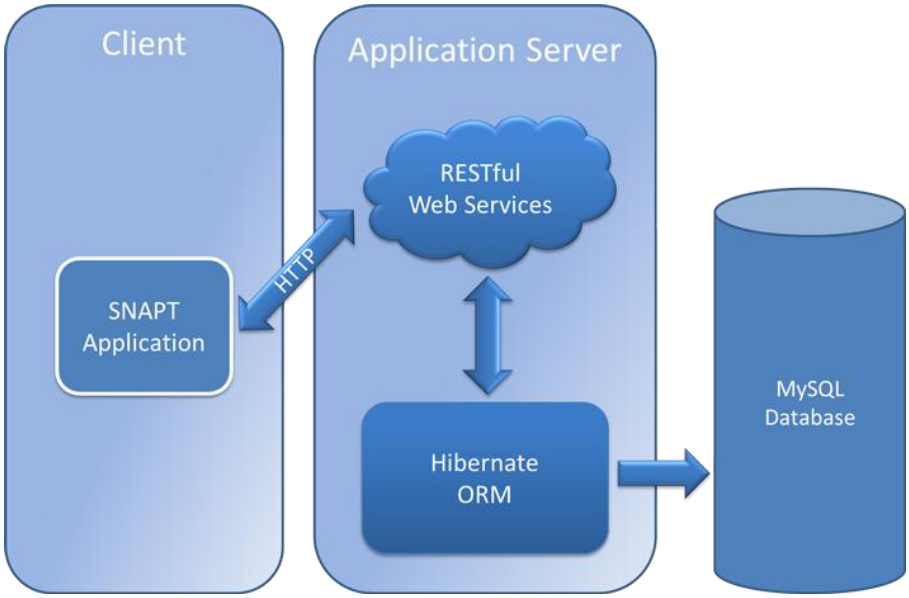


Figure 7: The KPI repository architecture

Figure 8 displays the database schema of the KPI data. KPIs are organized under categories and sub-categories, based on the industry-specific Process Classification Framework (PCF). There are almost 800 KPIs under 17 categories and 75 sub-categories available in the repository. To implement the mapping of the database tables of figure 8 to Java objects we used Hibernate, a widely accepted and powerful framework for Object Relational Mapping (ORM) and object persistence. The first step was to appropriate configure Hibernate in order to establish a connection to the MySQL database. Consequently, the REST Web Services were enabled to execute queries straight via Hibernate Query Language, which automatically creates the SQL query and executes it against the Hibernate configuration. The following REST-based Web Services were implemented to "PUT" the data and update the database: *addCategory* service creates a new category in the database given a new category id and name, *addSubcategory* service creates a new sub-category under an already existing category given a new sub-category

id and name, *addKPI* service creates a new KPI belonging in specific category and subcategory given a new KPI id, name and description. The only implemented Web Service that is responsible to "GET" data from the database to the client is the *getKpiLibrary* service, which returns all the KPIs of the database and their corresponding categories and sub-categories. In the next section, we show how SNAPT takes advantage of the getKpiLibrary service to preserve the local KPI library synchronized with the KPI Repository.

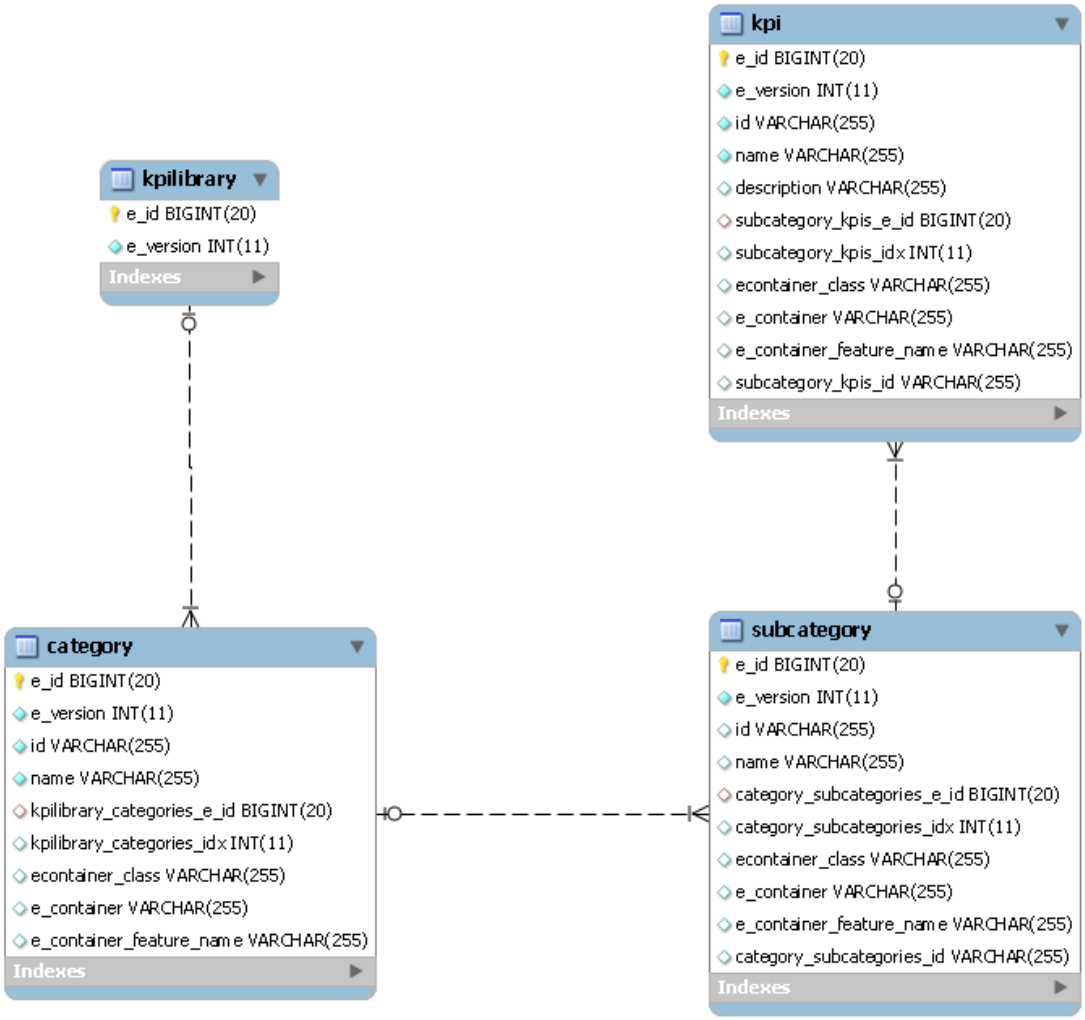


Figure 8: KPI repository database schema

3.4.2. Implementing a client for the KPI Repository

KPIs are of great importance for the business analyst as they measure the success of any firm [44]. They are critical business measurements as they are directly related to the firm's strategy and decision making [45]. KPIs assist the company to evaluate the business goals that has set for itself and they can act as motives for better performance. Thus, it is crucial for the businessmen using SNAPT to be able to easily find the appropriate KPIs to analyze specific Service Networks. In this section, we will describe how any SNAPT application interacts with the KPI repository to get all existing KPIs in the local KPI Library. To support this task, we developed a client that communicates via

HTTP requests with the *getKpiLibrary* REST service of the KPI repository. The following XML code exhibits the response message that is returned from the getKpiLibrary service. The users of SNAPT application can grant access to the KPI Repository via an action associated with the KPI Editor. The synchronize action is an extension to the already defined actions of the KPI Editor, which uses the REST client to get all the available KPIs from the repository and save them in the local KPI Library. Thus, the SNAPT user is enabled to browse almost 800 KPIs categorized in an industry-based structure, via the KPI editor.

4. Analyzing Service Networks: a Business Process Management (BPM) viewpoint

This section studies Service Networks, focusing on the business connections and the service exchanges that exist between the nodes – participants of the network. Figure 9 depicts the overall picture of how Service Networks can be developed, monitored, analyzed and measured through the different layers of the Business Process Management (BPM) stack. In the first level, service networks are developed to specify an abstract view of the relationships between the participants, taking out the operational details, meaning the way these relationships between the participants take place. This kind of information is depicted in the business process level, where service networks are transformed to the corresponding abstract business processes modeled in BPMN. The abstract business processes cannot be executed until the service composition layer refines them as composition of real software services [46], which are then deployed at the services layer. Service composition can be specified either by service orchestration or service choreography [47]. In service orchestration a single service is responsible to coordinate the execution of different operations accomplished by the services involved in the composition. On the contrary, in service choreography all the involved services of the composition know exactly what to perform and with whom to interact, without relying on a central coordinator. Orchestrations are specified using process flow languages such as Web Service - Business Process Execution Language (WS-BPEL)[5], while choreographies are specified using notation languages like BPEL4Chor [48]. In the services layer the services included in the service networks are realized using SOA specifications like SOAP, WSDL etc. [49]. At this point the evaluation of the service interactions included in the service network level is performed, in terms of business metrics such as Key Performance Indicators (KPIs). The measured value of these indicators can be the basis of decision making regarding the optimization of Service Networks.

[5] WS-BPEL 2.0 specification is available online: http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.html

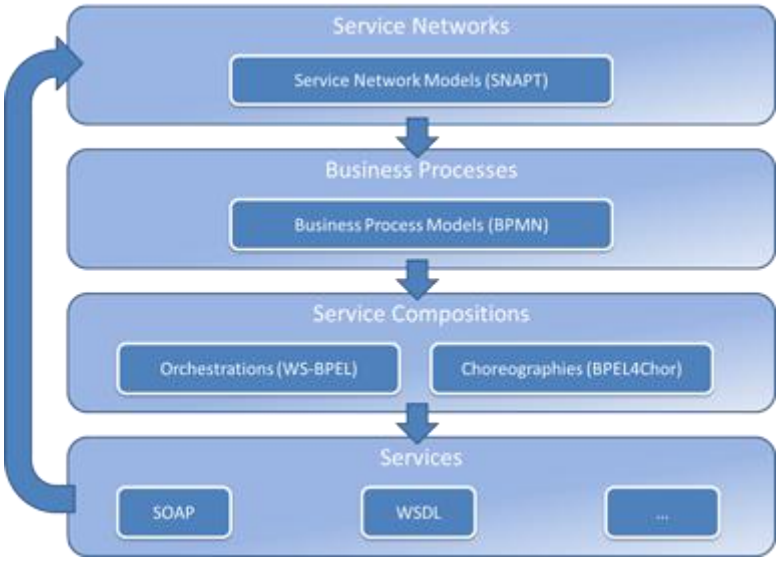


Figure 9: Service Network & BPM Layering Stack

Our approach contributes to the first step towards such Service Network analysis, providing the appropriate mappings to transform a service network model to an abstract business process model and the implementation of such transformations in two different SNAPT plugins. The first one consists of the creation of BPMN choreographies depicted by the BPMN modeler tool of eclipse, based on the service networks modeled in SNAPT. In the second implementation, we discover the business processes that take place in a Service Network in terms of the IBM's WebSphere Business Modeler meta-model. In the next section we discuss the basic principles around Business Process Modeling Notation (BPMN) that will be the basis for our abstract business processes.

4.1. Business Process Modeling Notation (BPMN)

The Business Process Management Initiative (BPMI) has developed Business Process Modeling Notation (BPMN), which in recent years has become a widely accepted open standard graphic notation for process modeling and service-oriented architecture (SOA). The primary goal of BPMI workgroup was to design a standard that will be easy to use and to understand to all users involved, starting from the business analyst that will create the initial abstract process, the IT developers responsible for implementing and deploying processes and the business users that will monitor the results of those processes. Thus, BPMN has managed to bridge the gap between the process design and process implementation, via the mapping of BPMN diagrams to Business Process Execution Language (BPEL). Furthermore, BPMN allows for the modeling of orchestrations and choreographies, providing wide support for modeling the process patterns. The version of BPMN presented and applied in this work is BPMN 1.2, which was published in January 2009 and is the established standard for modeling the operational aspects of business processes. BPMN 2.0 was available in BETA version at

the time this study was undertaken and there was no software tool implemented to support modeling of business processes in BPMN 2.0.

4.1.1. BPMN language constructs

Basic elements of BPMN as defined in the official specification [50] are categorized in four groups as depicted in figure 10: Flow Objects, Connecting Objects, Swimlanes and Artifacts. Flow objects and connecting objects are used in the definition of the process, whereas swimlanes and artifacts are used to supply additional information in the process model for the user.

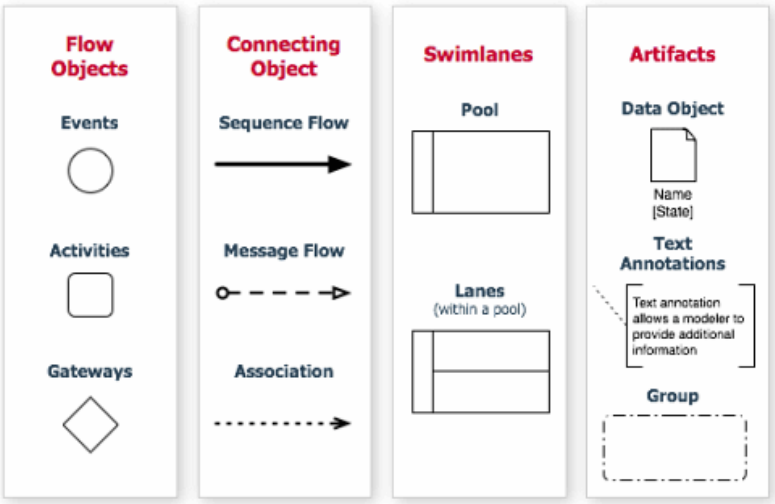


Figure 10: BPMN core elements

4.1.1.1. Flow Objects

Flow Objects are Activities, Events and Gateways that define the behavior of a business model. Activities are work items used to visualize the tasks that are performed within a process either by humans or machines. There are atomic and compound activities, which can be composed of many other activities. Activities are distinguished in three categories: processes, sub-processes and tasks. Events take place during the business process and can have a trigger and a result. Examples of event triggering can be an incoming message, a timer, a defined rule, or a composite trigger involving two or more of them. There are three types of events, depending on the time they take place within the business process: start, intermediate and end events. Gateways control the flow of a process, depending on their type. There are five different types: parallel (AND), where all paths are created or joined, data-based exclusive (XOR) and event-based exclusive (XOR), where exactly one path is created or joined, inclusive (OR) and complex type, where multiple paths are created or joined.

4.1.1.2. Connecting Objects

Flow Objects are connected to each other via Connecting Objects with three different ways: Sequence Flow, Message Flow and Association. A Sequence Flow models the order that flow objects will be carried out in a business process. There can be exactly one sequence flow between the same flow objects. Message Flows are used to model the information exchange between two participants of a process via messages. Participants that communicate through messages are visualized by pools (see section 4.1.1.3) in BPMN. Thus, activities of the same pool cannot communicate with each other through Message Flows. Message Flows can also be used to synchronize or trigger the start of a process. Associations are used to associate data objects in form of artifacts (see section 4.1.1.4) with flow objects, showing that a data object is either an input or an output for a flow object.

4.1.1.3. Swimlanes

There are two constructs used to group the modeling elements of BPMN: Pools and Lanes. As already mentioned, pools are used to represent the participants or organizations or different economic entities within a process. They are used as container for activities that are executed by different organizations. Pools can further be divided into lanes, each one of them depicting a different department of the organization.

4.1.1.4. Artifacts

Artifacts provide additional information about a process without affecting the sequence and message flow of the process. They are distinguished into data objects, groups, and text annotations. Data objects model information such as documents or messages that are inputs or outputs in a specific flow object. Groups are used to group flow objects together depending on the characteristics they have in common. Text annotations assist the reader to better understand the business process diagram, by providing additional descriptive information to the connecting and flow objects.

4.1.2. BPMN Models

BPMN supports various types of modeling, allowing the construction of business processes at different levels of precision. Depending on the goal that must be achieved through the business process modeling, there are three basic types of BPMN models that can be developed: Internal (Private) Business Processes, Abstract (Public) Processes and Collaborative (Public) B2B Processes. The usual strategy followed when modeling a

business process is to begin with designing high-level activities and then attaching further details in the process to lower levels within separate diagrams. Although BPMN is independent of any specific process modeling methodology, there may be several levels of diagrams, formed in sub-processes, based on the methodology applied by the model developer.

In this work we will focus on the Abstract (Public) Processes model, in our attempt to map Service Networks to business process modeled by service Choreographies which are expressed in BPMN 1.2. Service Networks do not include many details about the business process itself, so we will show in an abstract way the processes that are "hidden" in a specific Service Network.

4.1.2.1. Internal (Private) Business Processes

Internal business processes are the typical types of workflow processes, focusing on one specific business organization. The activities that are designed within the internal processes have private scope, meaning that they are invisible to the public. However, internal processes usually also show interactions with external organizations. A Pool represents the boundaries of a single organization in BPMN and activities can be categorized within swimlanes. Thus, the Sequence Flow of the business process is exclusively contained in the Pool, whereas the Message Flow can cross the boundaries of the Pool to show the existing interactions between separate internal business processes. Consequently, a single Business Process Diagram can provide multiple internal business processes.

4.1.2.2. Abstract (Public) Processes

Abstract (public) processes focus on the message flow between separate business organizations, showing the interactions between an internal (private) business process and at least one other organization taking part in the Business Process Diagram. Such processes contain the activities that are designed outside the Pool of the internal process and exchange messages with the activities of the internal Process, plus the desired flow objects of the abstract process.

4.1.2.3. Collaborative (Public) B2B Processes

A collaborative (Public) B2B process shows the interactions between two or more business organizations, i.e. they can be depicted as two or more abstract processes communicating with each other. With this model one can show the interactions between the organizations from a global point of view. The focus here is not on the activities of

one specific organization, but on the sequence of activities and message exchange patterns that take place between the participants. In a collaborative B2B process all interactions that are visible to the public for every single organization are defined. Although this kind of model cannot be mapped to BPEL, is also very efficient as it can be mapped to various collaboration languages such as ebXML BPSS [51] or RosettaNet.

4.1.3. BMPN 2.0

In this thesis we do not use this version of BPMN because it was not released as a stable version. The most resent stable version of BPMN at this time is 1.2. Upcoming version 2.0 of the BPMN specification provides several extensions in the BPMN 1.2 specification [52]. The major changes are associated with graphical constructs and diagram types. Extensions of graphical elements allow the modeling of specific task types, meaning that the action performed by the task can be pre-defined. For example, Service Tasks can be used to specify the execution of web services or automated applications. Extensions of diagram types such as Choreography Tasks are used to visualize Choreographies. In such diagrams one can depict the interactions and message exchanges between the organizations. Choreography can be modeled either as a stand-alone diagram or in between two pools of a collaboration process. The latter can also be used in the current version of BPMN 1.2 and in this way we model Choreographies in this work. So our approach of mapping Service Networks to BPMN business processes could also be accomplished with BPMN 2.0. Moreover, according to [53], BPMN models in the new version will be completely mapped to BPEL, transforming a BPMN model into an executable process.

4.1.4. BMPN Software Implementations

At the moment this thesis was written, there were 61 current BPMN implementations listed and four planned. Several famous companies have produced an implementation for BPMN 1.2, such as Fujitsu's Interstage Business Process Manager 7.1, Sun Microsystems' Studio Enterprise Edition and IBM's WBI Modeler. At this time there is no commercial product or open source software supporting BPMN 2.0. We can expect to see tools implementing this specification in the near future. For the requirements of this work we chose one commercial product and one open-source tool supporting BPMN 1.2, to model the Service Choreographies in our attempt to map Service Networks to Business Processes. We chose IBM's WBI Modeler as the commercial tool and eclipse-based BPMN modeler as the open-source one. The work of this master thesis is based on an eclipse-based tool, so the decision for the open-source BPMN modeler was straight-

forward. WBI Modeler was chosen because it is a widely used tool for modeling business processes, offering the possibility to manage and monitor the business processes over certain periods of time, which can be the base of future work towards the mapping of Business Process to Service Networks resulting even in the emergence of new Service Networks.

4.2. An algorithm for managing Sequencing Information in Service Networks

In this section we present the algorithm used to manage the sequencing information of a service network designed in SNAPT. To complete the first step towards transforming a service network model to abstract business process model, the stakeholder using SNAPT must design the Service Network diagram and enrich it with the appropriate sequencing information as discussed in appendix B. In any Service Network, there is at least one final service that is being delivered. Such services are usually delivered to the end customer nodes in a service network diagram and are considered to be the leaf nodes of the network. However, this is not applied for any case of service network. There are many examples of business models that do not include an end customer. In these business models the customers also play the role of service providers, as they co-operate with the main service provider to co-produce higher value for the final service. Such a business model is applied in the eBay Service Network and is further discussed in section 5.3. Whatever the case may be, the SNAPT user needs to distinguish any final service delivered in the service network and start assigning its sequencing information. As soon as the whole sequencing information is added in the service network model, the user is enabled to export the abstract business processes that take part in the service network, according to the logic controlled by the following algorithm.

The first rule followed by the algorithm managing the sequencing information, concerns the number of abstract business processes that it generates. For every service in the service network model that is not part of another service's sequencing information, a new abstract business process is created. From now on, we will refer to such services as final services. The details of the processes depend on the sequencing information the user has added on each service, based on the sequencing model that we presented on section 3.3. The algorithm starts from creating the abstract business process from the sequencing information that is associated with each final service and carries on the same procedure for all services that are included in the sequencing information of the final service i.e. the services that compose the final service. The management of the sequencing information for each final service is described by the following pseudo-code.

```
//generates abstract business process for the final service Service
FUNCTION manageSequencing(Service)
        //examines all blocks included in Service's sequencing information
        FOR each Block in Service.SequencingInfo
                //For every SequentialBlock create sequence flows that are connected in series
                IF Block EQUAL SequentialBlock THEN
                        CALL createAbstractBusinessProcessInSeriesFlows(Block)
                //For every ANDBlock or ANDSequence block, create multiple parallel flows, that all of
                //them must be executed
                ELSE IF Block EQUAL ANDBlock OR Block EQUAL ANDSequence THEN
                        CALL createAbstractBusinessProcessParallelFlows(Block)
                //For every XORBlock or XORSequence block create multiple parallel flows, that only one
                //of them must be executed
                ELSE IF Block EQUAL XORBlock OR Block EQUAL XORSequence THEN
                        CALL createAbstractBusinessProcessExclusiveFlows(Block)
                //For the service that is contained in a Sequence block either create a generic flow or
                //recursively manage its sequencing information, if any
                ELSE IF Block EQUAL Sequence THEN
                        IF Block.Service INSTANCEOF SequencingService THEN
                                CALL manageSequencing(Block.Service)
                        ELSE
                                CALL createAbstractBusinessProcess(Block.Service)
                        ENDIF
                ENDIF
        ENDFOR
END FUNCTION
```

In order for every final service to be delivered to a service consumer, a set of services must be delivered firstly, in a specific way specified by the sequencing information of each service. The above pseudo-code is responsible for managing the sequencing information for every final service within a service network and producing the appropriate abstract business processes. The procedure starts with examining all building blocks of the sequencing information. There are six different blocks that can be included within a service's sequencing information: *SequentialBlock, ANDBlock, XORBlock, ANDSequence, XORSequence* and *Sequence.* A *SequentialBlock* can be composed of *XORSequence*, *ANDSequence* or *Sequence* blocks, whose generated process flows are connected in series, in the order specified by the SNAPT user. An *ANDBlock* can be composed of at least two *SequentialBlock* or *XORBlock* blocks, whose generated process flows are connected in parallel with a gateway indicating that all flow paths must be executed. A *XORBlock* can be composed of at least two *SequentialBlock* or *ANDBlock* blocks, whose generated process flows are connected in parallel with a gateway indicating that only one flow path must be executed. The *ANDSequence* and the *XORSequence* blocks generate the same abstract business process flow as the ANDBlock and the XORBlock blocks, respectively. The difference between these blocks and the ANDBlock, XORBlock blocks is that the ANDSequence, XORSequence blocks can be included in a SequentialBlock, as they contain an additional information of the order in

which they are executed. The *Sequence* blocks are the simplest building blocks within the sequencing information and are decomposed to the services that form the composition of any final service. Thus, the total number of Sequence blocks within the sequencing information of a service indicates the total number of services that take part in the composition of this service. For every *Sequence* block there are two possible cases. If the SNAPT user has not added any sequencing information for the service associated with the Sequence block, then a simple process flow regarding the delivery of this service is generated. On the opposite case, the same procedure is recursively executed in order to manage the sequencing information of the block's service. To sum up, the algorithm discussed in this section generates the abstract business processes for all the final services of a service network, by managing the sequencing details that the SNAPT user gives as input, individually to every service that takes part in the composition of the final services. In the next sections we will present the abstract business processes that are generated from the service network model, supported by two different software tools.

4.3. From SNAPT to Eclipse BPMN Modeler

BPMN Modeler is an open-source business process diagram editor targeting business analysts. Like SNAPT, it is based on an EMF model represented graphically by the GMF project. The editor implements fully the BPMN 1.2 specification via the EMF model objects and creates BPMN diagrams. The tool also provides an Application Programming Interface (API) that enables other software to interact with BPMN diagrams. We used this API to develop an extension in SNAPT that will export a BPMN diagram for the eclipse-based BPMN Modeler based on a specific Service Network modeled in SNAPT. The BPMN diagram is composed of abstract business processes that specify the sequencing of message exchanges taking place between the services of the network. Such business process models are not executable, but only describe how participants of the original Service Network carry out their tasks to deliver services. Further service composition refinement needs to take place in order to transform the abstract business processes to executable ones, which is not part of this work, as we focus only on the business process layer in this approach.

4.3.1. Mapping SNAPT modeling constructs to BPMN constructs

In the business process layer the focus digresses from the operational details and is centralized on the communications of the participants that are result of service compositions that take part in the service network level. In section 3.3, we provided an

extended Service Network meta-model to describe such composition of services within the context of Service Networks. Based on this extended meta-model that includes the sequencing information of the provided services within the network, we have mapped the service network model constructs to BPMN constructs for the eclipse BPMN Modeler EMF model, as illustrated in table 2. The BPMN model behind the Eclipse BPMN Modeler consists of BPMN modeling constructs as discussed in section 2.3.1.

SN constructs	BPMN constructs
Business Entity	Pool
Service	Generic workflow
ServiceEnablement	Generic workflow placed on request or on delivery of the Service to be enabled
SequencingService	Complex workflow depending on its sequencing information
SequentialBlock	Generic or Complex workflows connected in series
ANDBlock, ANDSequence	Generic or Complex workflows connected via a BPMN parallel gateway
XORBlock, XORSequence	Generic or Complex workflows connected via a BPMN exclusive data-based gateway

Table 2: Mappings between Service Network and BPMN constructs

The general rule that stands for transforming a Service Network modeled in SNAPT to the business processes that take place within it, is that for every Service that is not part of the SequencingInfo of another SequencingService a new BPMN process diagram is created. More specific mappings for transforming service network constructs to BPMN constructs are the following. Every Business Entity of the Service Network is mapped to a different BPMN pool. Each Service object provided by a source Business Entity and delivered to a target Business Entity, as illustrated in figure is mapped to the workflow objects in BPMN as shown in figure 11. The target Business Entity starts its process by requesting the Service from the source Business Entity, which receives the request via a message event starting a new process. In the source Business Entity pool, an empty sub-process is created, specifying the logic for realizing the service and afterwards the result of the Service is returned to the target Business Entity, which consumes the Service ending the process. We will refer to such a mapping of service to workflow in BPMN, as a generic workflow. A single ServiceEnablement object is also mapped to the generic workflow. The only difference is associated to where these workflow objects are placed in the business process. The business analyst using SNAPT has two options to place the ServiceEnablements workflow objects in the business process, either on request or on delivery of the Service that is being "enabled" by the ServiceEnablement. For example, if the previously described service required a ServiceEnablement on request to be delivered to the target Business Entity as shown in figure 11, the workflow objects

referring to the ServiceEnablement object will be placed just after the task of the target business entity that requests the service that needs to be enabled. Based on our research on various case studies of Service Networks, such kind of services only facilitate the delivery of other services either on request or on delivery of these services. A typical example of such enablement service is the payment services provided by eBay. The payment of the product can either be processed online on request of the product or in cash on delivery of the product to the customer. Until now, we described how simple kinds of services are mapped to BPMN workflow objects. In case a service is being composed of other services, i.e. a SequencingService object, the mapping to BPMN workflows is achieved in a more complex way, depending on the service's sequencing information. The sequencing information of a service is mapped to both generic and complex workflows, depending on the Service or SequencingService objects that is composed of. These workflows are processed in a specific manner, depending on the sequencing constructs used in the sequencing information. Objects that are contained in a SequentialBlock are mapped to workflows that are processed in series, in the order indicated by the Sequence, XORSequence or ANDSequence objects. Objects that are contained in an ANDBlock are mapped to workflows that are connected via a BPMN parallel gateway, while objects contained in a XORBlock are mapped to workflows that are connected via a BPMN exclusive data-based gateway.

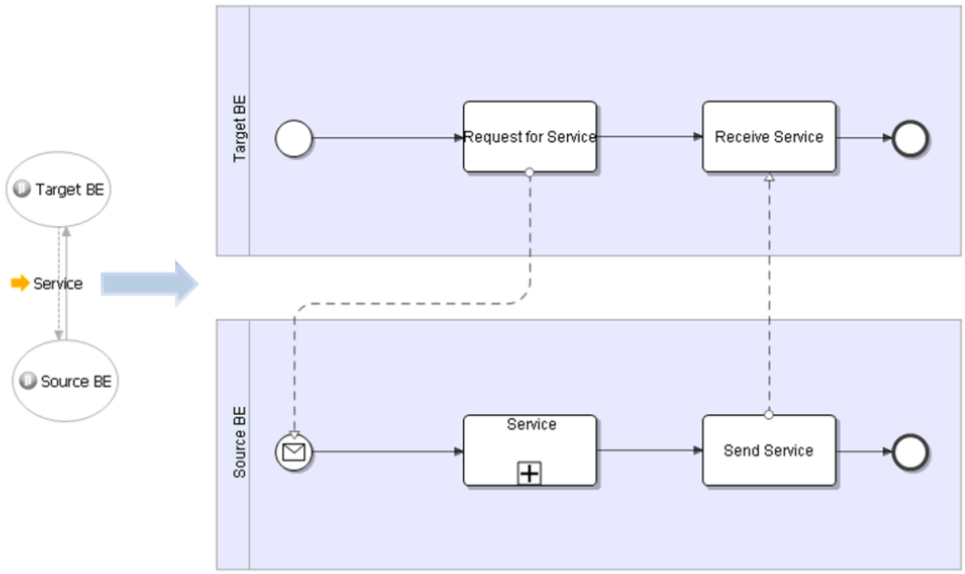


Figure 11: Mapping a single service delivery to a generic BPMN workflow

4.3.2. Export from SNAPT to Eclipse BPMN Modeler

To develop the previously discussed mappings from Service Network model to Eclipse BPMN Modeler's model we used an API provided by the BPMN Modeler eclipse plugin to generate the appropriate BPMN diagrams. The SNAPT user is enabled to export such diagrams via an export wizard, called "Eclipse BPMN Modeler". This wizard is registered as a SNAPT export wizard using the org.eclipse.ui.exportWizards extension point. The wizard is represented by a class named ExportEclipseBPMNModelerWizard that extends the Wizard class and implements the IExportWizard interface. The wizard is composed of three wizard pages. In the first one called SelectResourceWizardPage represents a page that enables the user to select which of the following SNAPT diagrams to export as a BPMN Modeler diagram and to set the file path of the BPMN diagram. The second wizard page, named BPMNSelectSSNWizardPage is responsible for selecting the Service Sub-Networks that the user is willing to include in the Service Network model to be exported. In the last wizard page, called EnablementElementInfoWizardPage the user chooses whether each Service Enablement found in the Service Network model is provided on request or delivery of the Service to be enabled. The user finally exports BPMN diagrams for Eclipse Modeler via the finish button of the last wizard page.

4.4. From SNAPT to IBM's WebSphere Business Modeler (WBM)

IBM's WebSphere Business Modeler[6] [54] is a worldwide accepted modeling tool that enables the business analysts to model and analyze the behavior of business processes by defining the organizations and the resources that take place within the processes. In our approach we have developed a SNAPT extension that transforms a service network model to an abstract business process model in such way that it can be imported in the WebSphere Business Modeler. Apart from the fact that it is a tool of high acceptance, another reason that forced us to choose IBM's application infrastructure is that WebSphere Business Modeler interacts with the WebSphere Business Monitor [54], which facilitates the monitoring of the business processes based on business metrics such as KPIs, assisting the decision making that will optimize the business processes and thus the Service Network itself.

4.4.1. The WebSphere Business Modeler's XML schema

The WebSphere Business Modeler (WBM) provides an XML schema defining the XML file structure that can be imported in the tool as a business process enriched with the

[6] IBM, WebSphere Business Modeler Advanced Version 7, available online: http://www-01.ibm.com/software/integration/wbimodeler/

appropriate artifacts. In appendix A, we exhibit the part of the XML schema that represents the content of a process flow. The Eclipse Model Framework (EMF) tools use simple mapping rules to create EMF models from XML schemas. We took advantage of this feature and generated the WBM EMF model and the associated Java implementation of the model that assisted as in the creation of business processes that can later be exported in an XML file from any SNAPT application. The EMF model code enabled us to easily create the appropriate model elements that can be imported in the WBM. Elements like business items, notifications, resources and organizations can be defined by the imported data structure of the XML schema. Our focus is on the following modeling elements of the WBM.

4.4.1.1. Business Items

Business items are business documents or work products that are exchanged within the context of business operations. Anything that is produced, modified or evaluated in a business process can be modeled as a business item. They represent the objects used in a process and can be passed from one process step to another. Typical example of business item is a customer order or an invoice. Further information describing a business item, can be defined in WBM with the use of attributes. By assigning specific values for each attribute, instances of the business items are created.

4.4.1.2. Organization Units

Organization units model specific groups of people that work together to achieve the same business objectives. Enterprises, companies, departments, or any independent economic entity are examples of organization units. Once created, the organization units can be assigned to specific tasks within the process in order to define which organization is responsible to carry out which business operations.

4.4.1.3. Processes

Processes represent real-time business processes. They are composed of business operations, control mechanisms that decide when these business operations are executed and the appropriate resource for the execution of the process. Processes are modeled as a sequence of business tasks connected by control and data flows, also known as process flow. A process may contain other processes, also known as sub-processes, or any of the following elements: Tasks are the main building constructs of a process that represent business operations in a process model. Each task performs a specific work action. They are the lowest level of sequence flow as they cannot be further decomposed in contrast to the processes. Each task can associated with input and output objects and organization units that are responsible for the completion of the task. Tasks are connected with each other or with sub-processes and flow objects to form the overall process flow. Flow objects that may be included in a process are start and end nodes that identify the beginning and the end of a process respectively, decisions that determine which one of several alternative paths will be executed, forks that split the process flow into at least two alternative process paths,

merges and joins that combine multiple process paths together into a single process flow and connections that link all the previously described elements with each other.

4.4.2. Mapping SNAPT modeling constructs to WBM constructs

Similar to the approach discussed in section 4.2.1., we have mapped the modeling elements of service networks to the BPMN modeling constructs of WBM, as illustrated in table 3. The general rule that stands for transforming a Service Network modeled in SNAPT to the business processes that take place within it is the same, as for every SequencingService that is not part of the SequencingInfo of another SequencingService, a new process is created. More specific mappings for transforming service network constructs to WBM constructs are the following. Each business entity of a service network is mapped to a different organization unit in WBM and is associated with specific tasks of the business processes. Service objects are mapped to generic process flows as depicted in figure 12, which represents a single service delivery as shown in figure 11. Whenever a service is mapped to this process flow, two business items are created in WBM: the Service Request and the Service Item. In the specific example, two organization units will be created in WBM, the Source BE and the Target BE that will be later on assigned with specific tasks of the process. The process begins from the task performed by the organization unit Target BE, which makes a request for the service, passing the business item Service Request to the next task. The organization unit Source BE is responsible for the three following tasks as it receives the service request as input, executes the activities that are hidden in the sub-process and realize the logic of the service and finally sends the service in its third task as a business item to the next task of the process. In the last task the organization unit Target BE receives the service from the Source BE and the process ends.

SN constructs	WBM XML Schema constructs
Business Entity	Organization unit
Service	Generic process flow
ServiceEnablement	Generic process flow placed on request or on delivery of the Service to be enabled
SequencingService	Complex process flow depending on its sequencing information
SequentialBlock	Generic or Complex process flows connected in series
ANDBlock, ANDSequence	Generic or Complex process flows connected via fork and merge elements
XORBlock, XORSequence	Generic or Complex process flows connected via decision and join elements

Table 3: Mappings between Service Network and WBI XML Schema constructs

Every ServiceEnablement object is mapped to the generic process flow of figure 12 and placed after either the task that receives a request or the task that sends the service to be enabled by the specific ServiceEnablement. A SequencingService is mapped to a complex process flow depending on the objects of its sequencing information. Any objects contained in SequentialBlock elements form a process flow of elements that are connected in series, in the order indicated by Sequence, XORSequence or ANDSequence objects. Objects contained in ANDBlock or ANDSequence elements form at least two different process paths that are

split by a fork element, executed in parallel and then combined again together to a single process path via a merge element. Objects contained in XORBlock or XORSequence elements form at least two different process paths that are connected through a decision element, which determines which of the process path will be executed and then combined again together to a single process path via a join element. Each service of a service network that is not contained in the sequencing information of another service, meaning that is not part of a service composition, represents a new process in WBM.

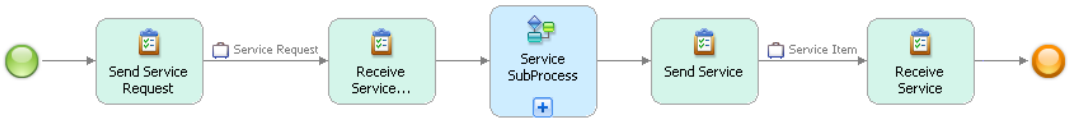


Figure 12: Mapping a single service delivery to a generic workflow in WBM

4.4.3. Export from SNAPT to WebSphere Business Modeler (WBM)

Following the transformations presented in the previous section we developed an extension in SNAPT that given a specific service network, it creates the corresponding objects of the EMF model generated from the WBM's XML schema. Firstly, the EMF elements, representing the catalogs within a WMB project, are created. These elements are containers that embody other elements of the same kind and facilitate the organization of processes and modeling objects within the context of WBM. The objects created group together all Business Items, all Resources, all Organizations and all Processes in four different catalogs, respectively. Afterwards, all business entities are mapped to the corresponding organization units and lastly all process flows are modeled. The created EMF model is exported as XML file via an export wizard, called "IBM WebSphere Modeler XML". This wizard is also registered as a SNAPT export wizard using the org.eclipse.ui.exportWizards extension point. It is represented by a class named ExportIBMModelerWizard that extends the Wizard class and implements the IExportWizard interface and is composed of the same three wizard pages as the ExportEclipseBPMNModelerWizard is described in section 4.2.2. The SNAPT user selects which of the SNAPT diagrams to export as a WBM XML file, sets the path of the XML file to be saved, selects the service sub-networks to be included in the Service Network model to be transformed and finally chooses whether each service enablement that exist in the service network model is provided on request or on delivery of the service to be enabled. The user created the WBM XML file by pressing the finish button of the last wizard page.

4.5. The Car Repair Case Study

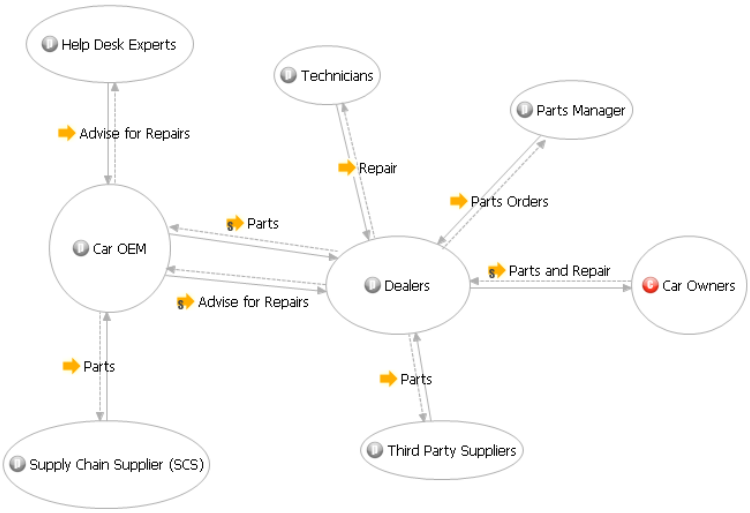


Figure 13: Car Repair Service Network

Inspired from the Car Repair service system presented in [5], we present how a more simplified example if the Car Repair Service Network modeled in SNAPT is transformed to the corresponding business processes in the Eclipse BPMN modeler and IBM's WebSphere Modeler. The Car Repair Service Network studied in this section consists of eight business entities: *Dealers*, *Car Original Equipment Manufacturer (OEM)*, *Help Desk Experts*, *Supply Chain Supplier*, *Technicians*, *Third Party Suppliers*, *Parts Manager* and the *Car Owners* that are the end customers consuming the final service of the service network. The whole network involving the service deliveries between the partners is depicted in figure 13. The purpose of this network is to efficiently deliver to the *car owners* the service of "Parts and Repair". The first step towards applying the BPM perspective analysis to the Car Repair Service Network is to discover the business processes that take part within it. In order for the Dealers to deliver the "Parts and Repair" service to the Car Owners, they must first order the parts with the help of the Parts Manager and then consume one of the "Parts" service delivered by Car OEM or the Third Party Suppliers, together with the "Repair" service provided by the technicians and taking into account the "Advice for Repairs" service delivered by the CAR OEM. The CAR OEM delivers the "Parts" and the "Advice for Repairs" service after consuming the corresponding services from the Supply Chain Supplier and the Help Desk Experts, respectively. This information is assigned to the appropriate services of the network from the sequencing view of SNAPT. The SNAPT user is now enabled to export this service network as business processes for the Eclipse BPMN Modeler and IBM's WebSphere Business Modeler as presented in the next sections as illustrated in figures 14 and 15 respectively. In the scope of this thesis, we only present the results of applying the proposed transformation rules to map service networks to business processes. The next step towards analyzing service networks through BPM perspective, is to find a semi-automatic way to refine these abstract business processes to real software service compositions and monitor their performance though business metrics such as Key Performance Indicators (KPIs).

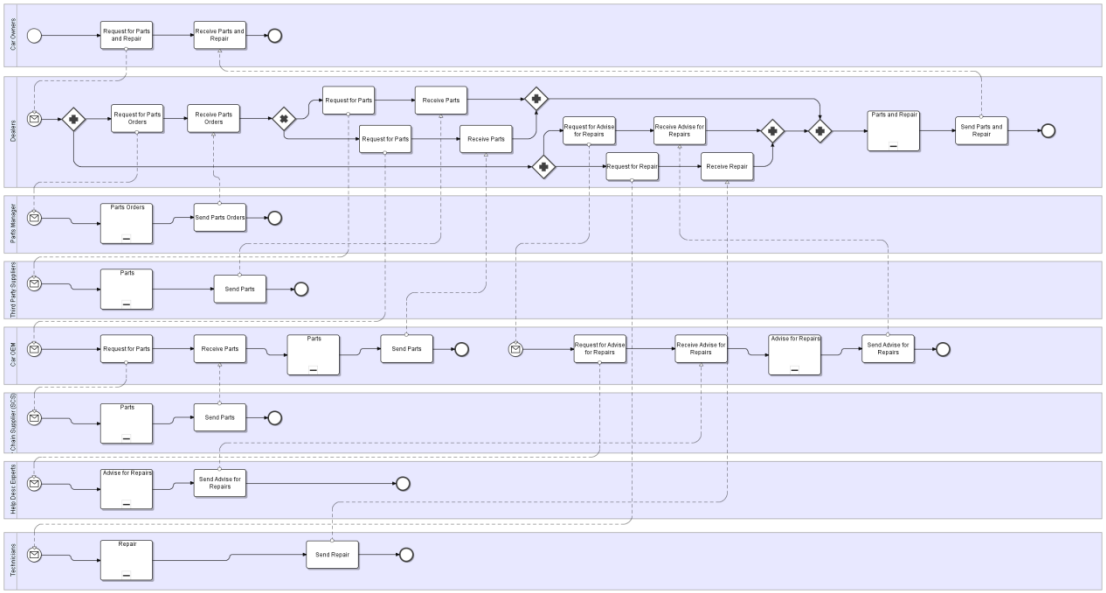


Figure 14: Car Repair – Abstract Business Processes in Eclipse BPMN Modeler

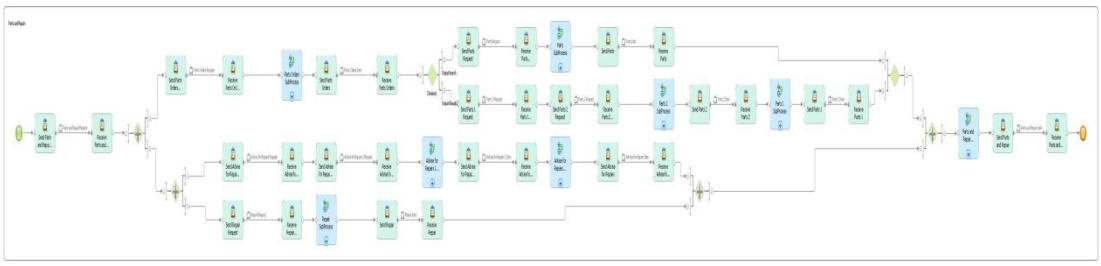


Figure 15: Car Repair – Abstract Business Processes in IBM's WBM

5. Analyzing Service Networks: a value-centered viewpoint

Currently, there are many research studies on Service Networks and most of them focus on their economical perspective. The objective in such economic-centered approaches is the optimization of the networks through the study and the analysis of the value created from the relationships that exist between the nodes of the networks. Such networks are also known as value networks. In section 2 we presented methodologies that analyze value networks focusing on the economic value generated within the networks. In this work we aim our attention at Verna Allee's analysis on value networks. Verna Allee's view of value networks is any "web of relationships that generate both tangible and intangible value through complex dynamic exchanges between two or more individuals, groups or organizations" [55]. Tangible value is created through the exchanges of products or services that produce revenue. Contracts and invoices are also considered to generate tangible value when included in network transactions. On the contrary, intangible value is created through knowledge exchanges and network transactions including benefit. Typical example of intangible value is the offering of volunteer work or expertise to someone in exchange of the benefit of prestige that can build strong relationships based on the concept of trust. To model and analyze tangible and intangible transactions of value networks, Allee developed the Value network analysis. It is an approach for understanding the creation of value within complex value ecosystems that can lead to the perception of trouble spots in the networks and become a basis for taking actions to change and optimize value networks [56]. In this section we will discuss in more details Verna Allee's Value network analysis and how it is conducted to provide indicators of how value is created through the realization of intangible assets in an attempt to improve the business performance. We also introduce an approach for transforming the service networks model of section 3.3 to Allee's value networks model and carry out a qualitative analysis on Service Networks. The results of this analysis are displayed through comprehensive indicator reports generated in SNAPT.

5.1. Value Network Indicators

Value Network Analysis (VNA) is a business modeling approach for defining, visualizing and finally analyzing network of interactions with the objective to discover

opportunities to improve business performance through an organizational structure change. We have already presented in section 2 the basic elements of VNA: participants, transactions and deliverables. VNA follows three basic steps: definition of the core participants, deliverables and transactions of the network, execution of typical analysis methods and use analysis findings to redefine participants or value flows. In this work we skip the first step as we create the value network elements through transformation rules applied on service networks created in SNAPT. VNA approach provides a solution to the valuation problem, meaning how to measure the value of intangible assets based on social aspects of business. Within a value network people interact with each other to organize exchanges and value flows. Humans can understand the emergence of value through such exchanges. When someone offers a deliverable to another person, the recipient may perceive it as a positive or a negative action, i.e. as having a positive or a negative value. In case people feel they have benefits from undertaking this action, the perceived value is positive. The more positive this value is then the most likely it is for the action to be repeated. Thus, trustful relationships are built between the participants of the networks that strengthen the network's stability. On the other hand, if the perceived value is negative, then the recipient person may be confused and discouraged to pay money or respond to recognize additional prestige to the other person, in a future same transaction. Apart from the emergence of value through exchanges within the networks, value can also convert from one form to another. Value conversion is basically achieved via intangible transactions. People manage to convert tangible or intangible assets to more negotiable forms of value in ordered to be delivered to others. For example, someone can pay for market analysis reports, which is a tangible asset and convert it to an intangible asset by increasing the market competency of his business. Various indicators arise from VNA that assist to understand the value conversion and address a comparative analysis on three different networks: the network including only tangible transactions, the network including only intangible transactions and the network including both tangible and intangible transactions. The following indicators and insights can be used to address the efficiency of value networks, which are the same addressed by the qualitative analysis implemented in SNAPT.

5.1.1. Resilience

The balance between the transactional (tangible) and the non-financial (intangible) data is a critical indicator for the resilience of the network. Thus, the ratio of tangible to intangible transactions indicates the network's capacity to respond to changing conditions. An appropriate ratio recognizes the fact that strong relationships exist

within the network in terms of better negotiations and trustful transactions. In case there are more tangible than intangible transactions in a high percentage, then the network has a high degree of transparency through formal structure. However, it could also indicate a low level of trust and knowledge sharing between the participants of the network. In business process-based networks a higher percentage of tangible to intangible transactions is a healthy situation. If the percentage of intangible is higher than the tangible transactions, then it may indicate that the network has a high level of flexibility and strong relationships. On the other hand, in case there is a big ratio of intangible to tangible transactions, it could mean that there are formal processes that are not working in a way they should or that more structure in needed. The appropriate ratio of intangible to tangible transactions varies with different industries.

5.1.2. Value Creation

VNA discovers the capacity of the network to create value, depending on how well participants utilize the assets to create both tangible and intangible value. If the value outputs of a participant decrease over time, it could be an indicator that productivity has declined. In the opposite way, the value productivity is improving. Comparing these indicators to the network's Perceived Value could reach to useful conclusions, as the network can achieve higher level value outputs, even if there are fewer transactions.

5.1.3. Brand Management – Perceived Value

Brand Management is associated with the human dimension of value, meaning how valuable a transaction in the eyes of the sender and the recipient is. Perceived value can be determined at the level of the transaction, the participant or at the network level. In order for value to be created the transaction must be perceived in a positive way from the recipient. In a different situation, both participants must wonder whether it was worth processing the transaction in the first place. The most interesting about the perceived value is when applied to intangible assets, as it is rather difficult to measure their value with a a financial quantity. Moreover, perceived value is an indicator of the network's stability: in case of high perceived value, the network is considered to be more stable than a network where people perceive low value.

5.1.4. Asset Management – Asset Impact

Asset Management is used to discover the asset impact of each transaction to the recipient. It identifies which type of assets will be impacted when a transaction is completed. There are four different types of assets: financial, business relationships,

competence and structure assets. VNA indicates which assets are affected by the participant's transactions or by the network as a whole.

5.1.5. Asset Management – Cost/Benefit

Once the asset types of the transactions are defined, the next step is to identify whether the transactions have a positive or a negative impact on the asset. Cost/benefit analysis of a transaction must include both financial and intangible asset impact. For example, completion of a knowledge transaction may have a positive benefit on competence asset and completion of a financial transaction may have a negative cost on financial asset.

5.1.6. Reciprocity

VNA studies the reciprocity of the network by measuring to what extend are the relationships between the participants symmetric. Asymmetric or unreciprocated relationships are considered to be unstable. Thus, a network with more reciprocated relationships than asymmetric ones has higher levels of stability, whereas the opposite situation may indicate a more heavily established structure.

5.1.7. Risk

Risk is a very interesting indicator in value network analysis. The first factor that can cause risk to the network is centrality. Centrality is measured by the total number of transactions for each participant. Although participants with the most relationships are considered to have high levels of prestige, they can become a threat to the entire network. If they fail for some reason to execute their transactions then they may negatively affect the network as a whole. Moreover, another case that increases the risk to the network is the variance between the connections of all the participants. High variance means that the network is composed of participants with many connections and others that have almost none, resulting in a weak distribution of network's power and a higher risk. Another risk factor is the speed in which communication is achieved between the participants. If a participant has not the appropriate resources to keep the value flow moving, it could be a bottleneck for the network by slowing down the value flow paths and even cause the network to break down.

5.1.8. Structure and Value

VNA also studies the way participants perceive or generate value from a structural viewpoint. We have already discussed about the indicator of centrality. "Central" participants have critical and the most privileged structural positions. Because of the

bigger number of relationships they have built, they have more alternative paths to choose in order to fulfill their needs for resources. Such participants have access to almost every resource of the network. To determine whether each participant is providing value to the network depending on his centrality, VNA defines the following indicators: the total number of incoming deliverables or transactions for each participant of the network indicates the value a participant gains from the network, whereas the total number of outgoing deliverables or transactions show the value a participant provides to the network.

5.1.9. Stability

Stability in a value network is very much dependent on the capacity of the network to generate value. Density, which typically indicates the communication speed of the network, and weak tie stability are also considered as important stability indicators for value networks. Density is measured as the number of actual connections between the participants divided by the number of potential connections. The maximum number of connections between two participants is one transaction. Density provides useful information about how quickly value flows among all participants. High density means that the participants of the network are well connected through trustful relationships that form the basis of high levels of network stability. Weak tie stability is measured as the ration between intangible to tangible transaction density. The higher the ratio is, the higher the density of intangible transactions and vice-versa. Perfect balance between densities of intangible and tangible transactions is achieved with a ratio equals to one. Weak tie stability can help understand the extent that the loss of connections between the participants will affect the network performance as a whole.

5.2. Designing and implementing a Qualitative Analysis in SNAPT

In this section we will present how we applied the Value Network Analysis (VNA) in an attempt to provide a qualitative analysis perspective in Service Networks with the goal of optimizing the overall performance of the business. To support the VNA in SNAPT we made the following steps: we established some transformation rules to automatically map service networks to value networks, we used the eclipse Business Information Reporting Tools (BIRT) project to generate the report template for showing the results of the analysis and lastly we implemented a wizard to enable the SNAPT user to assign the appropriate inputs to the value network model that will be the basis of the VNA and the parameters to the report template.

5.2.1. Transformation rules to map Service Networks to Value Networks

The first step towards applying VNA to Service Networks is to define the core elements of the value network model, meaning the participants, deliverables and transactions of

the network. To do so, we mapped service network model elements to value network model ones based on the proposed transformation rules depicted in table 4. Every service network business entity has a 1 to 1 mapping to a value network participant. Services and ServiceEnablements that provide revenue to the business entities offering them in the service network model are mapped to tangible transactions in the value network model, whereas all services with zero revenue are mapped to intangible transactions. Service revenues and ServiceEnablement revenues in Service Networks are also mapped to financial (tangible) transactions in Value Networks. Finally, every Services' name, ServiceEnablements' name and revenues are mapped to deliverable elements in Value Networks. These are proposed transformation rules that the user of SNAPT must be aware of, when needed to apply a qualitative analysis in service networks based on economical-centered indicators. In general, the analysts should choose this approach analysis depending on the scenarios depicted by the service network designed in SNAPT.

SN constructs	VN costructs
Business Entity	Participant
Service with positive revenue	Tangible transaction
ServiceEnablement with positive revenue	Tangible transaction
Service with zero revenue	Intangible transaction
ServiceEnablement with zero revenue	Intangible transaction
Service Revenues	Tangible transaction, deliverable
ServiceEnablement Revenues	Tangible transaction, deliverable
Service Name	Deliverable
ServiceEnablement Name	Deliverable

Table 4: Mappings between Service Network and Value Network constructs

5.2.2. Using Eclipse Business Information Reporting Tools (BIRT) to create the report template

The Business Information Reporting Tools (BIRT) project [57] is an open-source, eclipse-based reporting engine that integrated with Java applications to generate reports. We used BIRT's report designer, which is an eclipse component in BIRT, to create a report template that will be rendered to generate the final report of the qualitative analysis based on value network indicators (see section 5.1). The report designer is a tool that enables developers to easily create report templates by defining the appropriate data sets and associate them with a variety of charts and tables [58]. Using this tool, we designed different types of charts for each of the value network

indicators, based on the reports generated by the Value Network Insights [59] application. To illustrate the resilience in a value network, we defined one pie chart for the percentage of tangible/intangible deliverables and one bar chart for the number of tangible/intangible transactions. For the value creation indicator we designed two pie charts illustrating correspondingly the percentage of tangible and intangible deliverables generated by each participant of the network. In the brand management – perceived value, the first pair of bar charts show the overall perceived value of transactions in the value network, grouped by receiver and by sender. The next pair of bar charts shows correspondingly the overall receiver and sender perceived value in terms of both financial and non-financial assets of all the network transactions. Finally, there is a table displaying all transactions ordered by descending value perception of the receiver participant. The type of assets and the value perception of the participants are assigned to each transaction by the SNAPT user via a wizard page as described at the next section. The first pie chart in asset management - asset impact shows the overall asset impact of the network for all transactions. The second and third pie charts show the asset impact of intangible and tangible transactions, respectively. The other three charts indicate how asset impact is distributed by the participants of the network. In the asset impact - cost/benefit section, the first bar chart displays the overall cost/benefit impact of network transactions grouped by the asset type. The next three pie charts show the cost/benefit ratio for all transactions, for the intangible transactions only and for the tangible transactions only, correspondingly. Finally, the two bar charts illustrate the cost/benefit distribution and percentage by participant. To examine the reciprocity of the value network, we calculate and present the percentage of all reciprocated connections, all intangible reciprocated connections and all tangible reciprocated connections for all pairs of participants that have any connection. In the risk indicator we study the structural dependency of the network by calculating the highest, lowest and average number of connections per participant for all transactions, for intangible transactions only and for tangible transactions only. We also calculate the variance between the connections of all the participants of the network. We have already discussed in section 5.1.8 how we measure the centrality of the network in terms of incoming and outgoing connections. In the structure and value section, the pie charts depict the centrality indicators for all deliverables, for intangible and tangible deliverables only, distributed by participant. Finally, in the stability indicator we measure the weak tie stability, the overall density for all transactions and the density for every type of transaction separately.

5.2.3. Applying a Qualitative Analysis in SNAPT

To enable the business user to apply a qualitative analysis in service networks, we implemented the following extensions in SNAPT. Firstly, we defined a new eclipse action that triggers a wizard for the user to assign the appropriate inputs for the execution of the qualitative analysis. The wizard is composed of three wizard pages. In the first page, the user selects which of the existing service sub-network objects will be included in the service network that will be later transformed in a value network. In the second wizard page the service network has already been transformed to a value network, according to the transformation rules defined in section 5.2.1. In this page the user assigns the value network model the required information that will be taken into consideration in the analysis. More specifically, for each transaction between a sender and a receiver participant, the user needs to define the following information. Firstly the nature of the transaction must be defined, meaning whether the transaction offers a tangible or an intangible deliverable to the receiver. Moreover, every transaction contributes directly to a specific asset type of the value network, which must be chosen from the set of pre-defined asset types that includes financial assets, business relationships, competence and internal structure. Next the user must answer for every transaction whether it has a positive or negative impact on the previously defined asset. Lastly, the user must define the perceived value of the deliverable for the sender and the receiver, answering the question of whether each deliverable really provides value to the corresponding participant.

Based on the information provided by the user in the second wizard page, the qualitative analysis can now be applied to the value network. In the last wizard page the user sets the file path of the report generated by the BIRT engine in PDF. The report will contain the results of the qualitative analysis based on the template discussed in section 5.2.2. In the next section we present the results of such a qualitative analysis applied to the eBay service network and discuss how these results can assist in the optimization of the eBay's value network.

5.3. The eBay Case Study

In this section we apply the qualitative analysis to a specific case study by analyzing the eBay service network. We present the results of the analysis and show how the use of volunteers in the eBay can provide higher levels of stability to the eBay business model.

5.3.1. Modeling the eBay Service Network

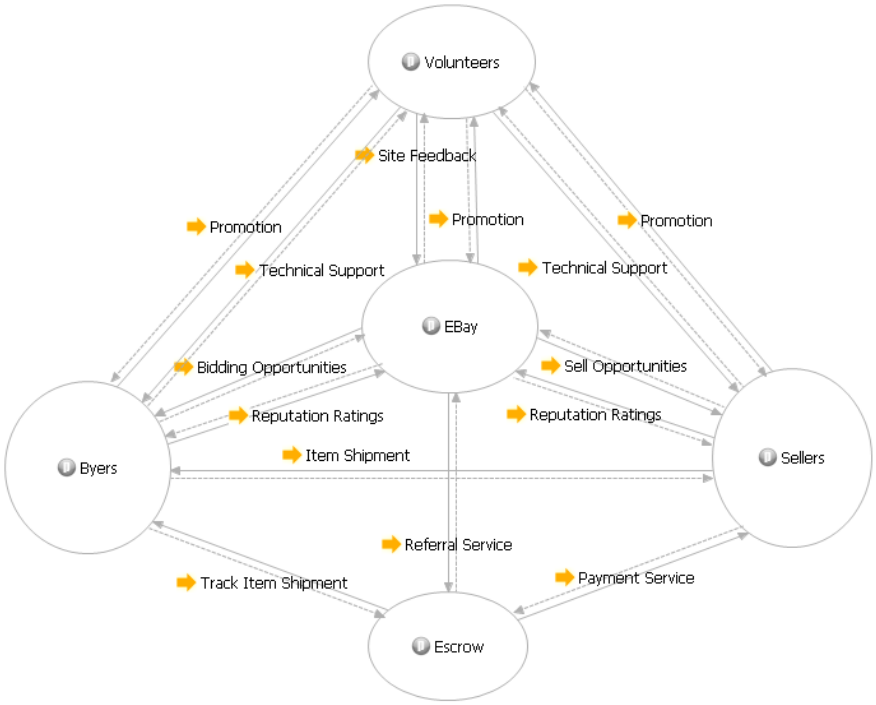


Figure 16: eBay Service Network

EBay is the biggest auction hosting company, owning one of the world's most popular web sites. In this section, our goal is to study eBay's business model and then try to define its service network. We chose this specific case study because eBay is a well-known company, so several studies considering its network have been established. At the same time eBay's model is not so complicated for the reader to understand as it involves only five business entities: eBay, byers, sellers, escrow company and volunteers group. The key of the company's success is a user-based community where both buyers and sellers use an interactive interface in order to accomplish their tasks. A prospective buyer is able to browse all listed items, search for an item that satisfies his own criteria and optionally bid for it. All these services are provided to the buyers for free. On the other hand a seller has to pay eBay one of the following fees in order to list an item for sale. Insertion fees ranges between 10 cents and $4.00, depending on the seller's starting or reserve price and the category that the item will be listed. Final value fee is charged to the seller at the end of the auction. It starts from 8.75% of the closing sale price for items sold under $25 and this percentage is gradually decreasing for bigger closing prices. Optional feature fees are charged to the seller depending on his will for additional features. These fees may include reserve fees, "buy it now" fees, listing upgrade fees etc. Moreover eBay has signed contracts with several escrow companies

that collect, hold, and send a buyer's money to a seller according to instructions agreed on, by both the buyer and seller. This is an additional income for eBay, as escrow companies are charged for transactions referenced by eBay. Finally, eBay has settled various "volunteer programs" that involve users with technical knowledge and expertise, wishing to share it with other community members. Volunteers, as their title suggests, have no income for providing their services, adding a zero-cost value to eBay's network. The eBay Service Network is modeled in SNAPT as illustrated in figure 16. In the next section we will apply the qualitative analysis twice to the EBay Service Network, with and without the volunteers and compare their results.

5.3.2. Analyzing the eBay Service Network

By applying the transformation rules of section 5.2.1 to the eBay service network, we can provide interesting results by comparing the value network indicators of two different Value Network Analyses. We firstly apply VNA in the eBay service network without involving the participant volunteers and afterwards we apply VNA to the whole eBay Service Network as depicted in figure 16. Table 5 provides a comparison of the most important value network indicators between the two different service network models of the same case study.

Value Network Indicator	eBay without Volunteers	eBay with Volunteers
Resilience	0.0	1.33
Reciprocity	all transactions: 0%, intangible transactions: 0%, tangible transactions: 0%	all transactions: 55%, intangible transactions: 33%, tangible transactions: 0%
Structural Dependency (Variance Transactions)	all transactions: 0.0, intangible transactions: 0.0, tangible transactions: 0.0	all transactions: 0.3, intangible transactions: 1.8, tangible transactions: 1.5
Weak Tie Stability	0	0.83
Density	all transactions: 100%, intangible transactions: 0%, tangible transactions: 100%	all transactions: 90%, intangible transactions: 50%, tangible transactions: 60%

Table 5: Comparing Value Network Indicators for the eBay Case Study

By comparing the value network indicators of table 5, one could reach to useful conclusions regarding the eBay service network. The resilience indicator measures the ratio of intangible to tangible interactions in the network. It depicts the ability of the network to respond to changing conditions through the balance of the formal structure (tangibles) to the informal knowledge sharing (intangibles). In the first case, the lack of intangibles is depicted in the absence of the volunteers. On the contrary, the eBay service network model including volunteers has resilience more than 1, meaning more intangible to tangible transactions that consequence to a high level of flexibility and strong relationships. Comparing the levels of reciprocity in the two different eBay service networks shows that the eBay with volunteers may be more stable as it has

more reciprocated relationships than the eBay without volunteers, which lacks in reciprocity indicating a more hierarchical structure. Structural dependency deals with measuring the centrality of the network, meaning how well are the transactions distributed over the participants of the network. In case there is a high variance of transactions in the network, there is a big possibility to find participants with many transactions and others with almost none. In both cases of eBay service network, the levels of transaction variance are low, minimizing the risk to the network. Density is another indicator showing the stability of the network. High density in the network shows that participants are well connected to each other, depicting a high level of social capital. This is applied to both cases of the eBay Network. The only difference in terms of density is the difference in the weak tie stability, which measures the ratio between intangible and tangible transaction density. The more close this indicator is to 1, a better balance between densities of tangible and intangible connections of exist. Thus, the service network of eBay with volunteers seems to be more stable as far as density is concerned. To sum up, the addition of the volunteers' participant to the eBay service network sets higher levels of network stability by better balancing the tangible to intangible exchanges and building flexible and trustful relationships between the participants.

6. Discussion & Future Work

Service Network Analysis & Prediction Tool (SNAPT) was designed with the objective of becoming a universal platform for developing, analyzing, monitoring, and optimizing services within the context of complex service systems. Service Networks form an abstract way of viewing these complex service systems. SNAPT targets business experts or stakeholders that study Service Networks, each one from his own viewpoint. To fulfill the different needs of each SNAPT user, we extended SNAPT to support two methods for analyzing Service Networks from different perspectives.

We presented an approach towards analyzing Service Networks in a Business Process Management (BPM) viewpoint, as depicted in figure 9. The thesis contributes to the first step of this approach by proposing several transformation rules to map service networks to the corresponding abstract business processes, in a semi-automatic way. We supported these transformation rules, by providing interaction between the SNAPT and two software systems that model business processes. Further work can be undertaken here to provide SNAPT with additional software interaction. More specifically, SNAPT could also support exporting BPMN process models from service network models, for the online Oryx editor tool [60]. The specific online BPMN editor supports the transformation of BPMN to BPEL4Chor choreographies through plugins implemented to Oryx [61]. This could bring the discussed approach one step forward, by enabling the refinement of the abstract business processes of BPMN to real software services through the service composition layer. At the time this thesis was undertaken, the specific Oryx BPEL4Chor plugin was not fully functional. Further research is necessary towards monitoring and optimizing the behavior of the services in terms of business performance metrics such as Key Performance Indicators (KPIs). In order to reach the level of actually measuring KPIs and monitor whether the Key Performance Objectives (KPOs) defined in the service network level are indeed achieved, the technological layers of the service networks must be further investigated together with their implementations. KPOs are defined in SNAPT at the service network level as objectives associated with specific KPIs. Within this context, it would be very interesting to associate KPIs and KPOs with Service Level Agreements (SLAs) [62]. SLAs define contracts between a service provider and a service consumer and approved by a third authority party, in terms of performance and quality characteristics of a service (QoS). Management change and optimization of service networks could be achieved by enhancing service network level using SLAs and an authority mechanism, responsible to monitor whether the involved parties conform to the contract.

Another contribution of this thesis concerns a value-centered approach for analyzing Service Networks based on the Value Network Analysis (VNA) developed by Verna Allee. We defined a set of rules to transform service network models to Verna Allee's value network models and apply a qualitative analysis on Service Networks. SNAPT supports

this value-centered analysis by generating reports that illustrate the analysis results, in terms of value network indicators. The specific approach can be a powerful mechanism for studying the value flows of the service networks and providing useful information that will assist the efficient change management of service networks. More extensions can be implemented in SNAPT to support different economical approaches on analyzing Service Networks. Improvement of business performance can be also achieved through the interaction of SNAPT with software platforms supporting system dynamics methods such as Vensim [63] or iThink [64]. A methodology that can be conducted in these tools is presented in [65], where service networks are simulated over certain periods of time in an attempt to increase the customer's satisfaction index [66] and as a result the overall value of the network. Steps towards implementing a SNAPT extension that will transform service network models to dynamic systems modeled in Vensim and will be able to simulate and predict service network's value taking into consideration factors like business competition, have already been conducted and will be presented in a future work.

Bibliography

1. **J. Spohrer, P. Maglio.** "Emergence of service science: Services sciences, management, engineering (SSME) as the next frontier in innovation", Nordic Service Innovation Workshop, Oslo, Norway, 2005

2. **H. Chesbrough, J. Spohrer.** "A Research Manifesto for Services Science", Communications of the ACM 49, 35-40, 2006

3. **S. Vargo, R. Lusch.** "Evolving to a new dominant logic of marketing", Journal of Marketing, 68(1), 1-17, 2004

4. **P. Maglio, S. Vargo, N. Caswell, Jim Spohrer**. "The service system is the basic abstraction of service science", Inf. Syst. E-Business Management, 7(4):395–406, 2009

5. **N. Caswell, C. Nikolaou, J. Sairamesh, M. Bitsaki, G. D. Koutras, G. Iacovidis.** "Estimating value in service systems: A case study of a repair service system", IBM Systems Journal, Vol. 47, 2008

6. **J. Gordijn, H. Akkermans.** "Designing and evaluating ebusiness models", IEEE Intelligent Systems, 16(4):11–17, 2001

7. **V. Allee.** "Value network analysis and value conversion of tangible and intangible assets", Journal of Intellectual Capital, 9, No. 1,:pp. 5–24, 2008

8. **L. Camarinha-Matos and H. Afsarmanesh.** "Towards a reference model for collaborative networked organizations", In Information Technology For Balanced Manufacturing Systems (BASYS'06), pages 193–203, 2006

9. **A. Barros, M. Dumas, P. Bruza.** "The Move to Web Service Ecosystems", s.l. : BPTrends, 2005

10. **M. Papazoglou**, "Web Services: Principles and Technology", Prentice Hall, 2007.

11. **H. Luthria, F. Rabhi.** "Service Oriented Computing in Practice – an Agenda for Research into the Factors Influencing the Organizational Adoption of Service Oriented Architectures.", Journal of Theoretical and Applied Electronic Commerce Research, 4 (1), 39-56, 2009

12. **P. Petridis.** "Towards a universal Service Network-centric framework to design, implement and monitor Services in complex Service Ecosystems: The Service Network Analysis & Prediction Tool (SNAPT)", Computer Science Department, University of Crete, 2010

13. **S. Vargo, R. Lusch.** "From goods to service(s): Divergences and convergences of logics", Industrial Marketing Management, Vol. 37, 2008

14. **J. Spohrer, P. Maglio, J. Bailey, D. Gruhl.** "Steps Toward a Science of Service Systems", IEEE Computer Society Press, Computer, Vol. 40, 2007

15. **P. Barros, M. Dumas.** "The rise of web service ecosystems", IT Professional, 8(5), 31–37, 2006

16. **J. Gordijn, H. Akkermans.** "Designing and Evaluating e-Business Models", IEEE, Intelligent Systems, Vol. 16, 2001

17. **B. Blau, J. Kraemer, T. Conte, C. van Dinther.** "Service value networks", Proceedings of the 11th IEEE conference on commerce and enterprise computing, July 20-23, 2009, Vienna, Austria. Los Alamitos (CA): IEEE Computing Society Press; 2009

18. **M. Bitsaki, O. Danylevych, W. van den Heuvel, G. Koutras, F. Leymann, M. Mancioppi, C. Nikolaou, M. Papazoglou.** "An Architecture for Managing the Lifecycle of Business Goals for Partners in a Service Network", 2008

19. **M. Porter.** "Competitive Advantage: Creating and Sustaining Superior Performance", Free Press, 1985

20. **C. Christensen.** "The Innovator's Dilemma", Boston: Harvard Business School Press, 1997

21. **C. Stabell, D. Fjeldstad.** "Configuring value for competitive advantage: On chains, shops, and networks", Strategic Management Journal 19, 1998

22. **D. Laffey, A. Gandy.** "Applying Stabell and Fjeldstad's Value Configurations to E-Commerce: A Cross-Case Analysis of UK Comparison Websites", the Journal of Strategic Information Systems, 18 (4), pp. 192-204, 2009

23. **V. Allee.** "The art and practice of being a revolutionary", Journal of Knowledge Management, Vol. 3 Iss: 2, pp.121 – 132, 1999

24. **V. Allee.** "Value network mapping basics", ValueNet Works™ Fieldbook, available online: http://www.alleevaluenetworks.com, 2006

25. **J. Gordijn, H. Akkermans.** "E3-value: Design and Evaluation of e-Business Models", In IEEE Intelligent Systems, Vol. 16(4):11-17, 2001

26. **J. Gordijn, P. van Eck, R. Wieringa.** "Requirements Engineering Techniques for e-Services", s.l.: MIT press, Readings in Service-Oriented Computing: the Web-Services Phenomenon, 2008

27. **e3value.** Tools for e3value, available online: http://www.e3value.com/tools/, August 2010

28. **J. Gordijn, J. Akkermans, J. van Vliet.** "Business Modeling is not Process Modeling", In Conceptual Modeling for E-Business and the Web, LNCS 1921, Springer-Verlag, 2000

29. **H. Weigand, P. Johannesson, B. Andersson, M. Bergholtz, A. Edirisuriya, T. Llayperuma.** "Strategic Analysis using Value Modelling – a c3 approach", In Prooceedings of the 40th Hawaii InternationaI Conference on System Sciences -2007

30. **A. Biem, N. Caswell.** "A value network model for strategic analysis", Proceedings of the 41st Hawaii International Conference on System Sciences, 2008

31. **B. Blau, J. Kraemer , T. Conte, C. van Dinther.** "Service value networks", Proceedings of the 11th IEEE conference on commerce and enterprise computing, July 20-23, 2009, Vienna, Austria. Los Alamitos (CA): IEEE Computing Society Press; 2009

32. **B. Blau, C. van Dinther, T. Conte, Y. Xu, C. Weinhardt.** "How to Coordinate Value Generation in Service Networks – A Mechanism Design Approach", Business and Information Systems Engineering (BISE) 1 (5), 343–356, 2009

33. **C. van Dinther, Benjamin Blau, Tobias Conte.** "Strategic Behavior in Service Networks under Price and Service Level Competition". Wirtschaftsinformatik (1): 599-608, 2009

34. **V. Allee.** "A Value Networks approach for Modeling and Measuring Intangibles", White Paper, 2002

35. **B. Wetzstein, O. Danylevych, F. Leymann, M. Bitsaki, C. Nikolaou, W. Van den Heuvel and M. Papazoglou.** "Towards Monitoring of Key Performance Indicators Across Partners in Service Networks", International Workshop on Service Monitoring, Adaptation and Beyond, 2008.

36. **Eclipse Foundation.** Rich Client Platform, availabel online: http://wiki.eclipse.org/index.php/Rich_Client_Platform, August 2010

37. **D. Steinberg, , F. Budinsky, M. Paternostro, E. Merks.** "Eclipse Modeling Framework", 2nd Edition, Pearson Education, 2008

38. **J. Warmer, A. Kleppe.** "The Object Constraint Language (OCL) - Getting Your Models Ready for MDA", Second Edition, Addison-Wesley Edition, 2003

39. **APQC.** "Process Classification Framework", available online: http://www.apqc.org/pcf, August 2010

40. **Mirror42.** KPI Library, available online: http://www.kpilibrary.com/, August 2010

41. **Sun Microsystems.** "MySQL 5.5 Reference Manual", available online: http://www.mysql.com, 2010

42. **Red Hat Middleware.** "Relational Persis-tence for Java and .NET", available online: http://www.hibernate.org, 2009

43. **L. Richardson, S. Ruby.** "RESTful Web Services", O'Reilly, May 2007

44. **R. Simmons.** "Performance Measurement and Control Systems for Implementing Strategy Text and Cases", Prentice Hall, 1999

45. **V. Kellen.** "Business Performance Measurement", available online: http://www.kellen.net/bpm.htm, 2003

46. **S. Dustdar, W. Schreiner.** "A survey on web services composition", IJWGS, 1(1):1–30, 2005

47. **A. Barros, M. Dumas, P. Oaks.** "Standards for web service choreography and orchestration: Status and perspectives", In Christoph Bussler and Armin Haller, editors, Business Process Management Workshops, volume 3812, pages 61–74, 2005

48. **G. Decker, O. Kopp, F. Leymann, K. Pfitzner, M. Weske.** "Modeling service choreographies using bpmn and bpel4chor", In Bellahsene and L´eonard, pages 79–93, 2008

49. **M. Papazoglou.** "Web Services: Principles and Technology", Prentice Hall, 2007

50. **Business Process Management Initiative (BPMI).** "Business Process Modeling Notation (BPMN)", available online: http://www.bpmn.org/, June 2010

51. **OASIS.** "Business Process Specification Schema (ebXML BPSS or ebBP)", Version 2.0.4, 2007

52. **OMG.** "Business Process Model and Notation (BPMN) Specification 2.0", OMG Document Number: formal/2009-01-03, Standard document URL: http://www.omg.org/spec/BPMN/1.2, pp 37-89, 2009

53. **J. de Jong.** "Could BPMN 2.0 Make BPEL a Moot Point", SD Times, Jul 2006

54. **IBM.** IBM Software - Websphere, available online: http://www-01.ibm.com/software/websphere/

55. **V. Allee.** "The Future of Knowledge: Increasing Prosperity through Value Networks", Butterworth-Heinemann, 2003

56. **Value Networks LLP.** "Value Network Adoption at Boeing and in Large Organizations", Publisher: Colabria, November 2007

57. **BIRT Project.** "Business Intelligence and Reporting Tools", Published by Eclipse Foundation, available online: http://www.eclipse.org/birt/, 2008

58. **D. Peh, N. Hague, J. Tatchel.** "BIRT: A Field Guide to Reporting", 2008

59. **V. Allee.** "Value Network Insights", available online: http://www.valuenetworks.com, August 2010

60. **D. Polak.** "Oryx - BPMN Stencil Set Implementation", June 2007.

61. **K. Pfitzner, G. Decker, O. Kopp, F. Leymann.** "Web Service Choreography Configurations for BPMN", Proceedings of the 3rd International Workshop on Engineering Service-oriented Applications: Analysis, Design and Composition (WESOA), Vienna, Austria, September 2007

62. **B. Wetzstein, D. Karastoyanova, F. Leymann.** "Towards Management of SLA-Aware Business Processes Based on Key Performance Indicators", 9th Workshop on Business Process Modeling, Development, and Support (BPMDS'08), 2008

63. **Ventana Systems.** VENSIM, available online: http://www.vensim.com/, August 2010

64. **iSee Systems.** iThink - Systems Thinking for Business, available online: http://www.iseesystems.com/softwares/Business/ithinkSoftware.aspx, August 2010

65. **M. Voskakis.** "Simulation and optimization of value in Service Networks", Computer Science Department, University of Crete, 2010

66. **Hill N., G. Brierley, and R. MacDougall.** "How to Measure Customer Satisfaction", Gower Publishing, Hampshire, 2003

Appendix A

XML Schemas of Domain Models

1. Sequencing Ecore Domain Model

```
<?xml version="1.0" encoding="UTF-8"?>
<ecore:EPackage xmi:version="2.0"
        xmlns:xmi="http://www.omg.org/XMI"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore" name="sequencing"
        nsURI="http://tsl.gr/snapt/sequencing" nsPrefix="sequencing">
 <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore">
  <details key="settingDelegates"
        value="http://www.eclipse.org/emf/2002/Ecore/OCL"/>
  <details key="invocationDelegates"
        value="http://www.eclipse.org/emf/2002/Ecore/OCL"/>
  <details key="validationDelegates"
        value="http://www.eclipse.org/emf/2002/Ecore/OCL"/>
 </eAnnotations>
 <eClassifiers xsi:type="ecore:EClass" name="SequencingService"
        eSuperTypes="../../org.eclipse.gmf.snapt/model/SNAPT.ecore#//Service">
  <eStructuralFeatures xsi:type="ecore:EReference" name="SequencingInfo"
        eType="#//SequencingInfo" containment="true"/>
 </eClassifiers>
 <eClassifiers xsi:type="ecore:EClass" name="SequencingInfo">
  <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore">
   <details key="constraints" value="SequencingInfoMustHaveOneChild"/>
  </eAnnotations>
  <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore/OCL">
   <details key="SequencingInfoMustHaveOneChild"
        value="(self.SequentialBlock->size() + self.ANDBlock->size() +
                self.XORBlock->size()) = 1"/>
  </eAnnotations>
  <eStructuralFeatures xsi:type="ecore:EReference" name="SequentialBlock"
        eType="#//SequentialBlock" containment="true"/>
  <eStructuralFeatures xsi:type="ecore:EReference" name="ANDBlock"
        eType="#//ANDBlock" containment="true"/>
  <eStructuralFeatures xsi:type="ecore:EReference" name="XORBlock"
        eType="#//XORBlock" containment="true"/>
 </eClassifiers>
 <eClassifiers xsi:type="ecore:EClass" name="Sequence">
  <eStructuralFeatures xsi:type="ecore:EReference" name="Service"
        lowerBound="1" Type="ecore:EClass
        ../../org.eclipse.gmf.snapt/model/SNAPT.ecore#//Service"/>
  <eStructuralFeatures xsi:type="ecore:EAttribute" name="Order"
        eType="ecore:EDataType http://www.eclipse.org/emf/2002/Ecore#//EInt"
    defaultValueLiteral="1"/>
 </eClassifiers>
 <eClassifiers xsi:type="ecore:EClass" name="ANDSequence">
  <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore">
   <details key="constraints"
        value="ANDSequenceMustHaveAtLeastTwoChildren"/>
  </eAnnotations>
```

```
  <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore/OCL">
   <details key="ANDSequenceMustHaveAtLeastTwoChildren"
          value="(self.SequentialBlocks->size() + self.XORBlocks->size()) >=
          2"/>
  </eAnnotations>
  <eStructuralFeatures xsi:type="ecore:EReference" name="SequentialBlocks"
          upperBound="-1" eType="#//SequentialBlock" containment="true"/>
  <eStructuralFeatures xsi:type="ecore:EReference" name="XORBlocks"
          upperBound="-1" eType="#//XORBlock" containment="true"/>
  <eStructuralFeatures xsi:type="ecore:EAttribute" name="Order"
          eType="ecore:EDataType http://www.eclipse.org/emf/2002/Ecore#//EInt"/>
</eClassifiers>
<eClassifiers xsi:type="ecore:EClass" name="XORSequence">
  <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore">
   <details key="constraints"
          value="XORSequenceMustHaveAtLeastTwoChildren"/>
  </eAnnotations>
  <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore/OCL">
   <details key="XORSequenceMustHaveAtLeastTwoChildren"
          value="(self.SequentialBlocks->size() + self.ANDBlocks->size()) >=
          2"/>
  </eAnnotations>
  <eStructuralFeatures xsi:type="ecore:EReference" name="ANDBlocks"
          upperBound="-1" eType="#//ANDBlock" containment="true"/>
  <eStructuralFeatures xsi:type="ecore:EReference" name="SequentialBlocks"
          upperBound="-1" eType="#//SequentialBlock" containment="true"/>
  <eStructuralFeatures xsi:type="ecore:EAttribute" name="Order"
          eType="ecore:EDataType http://www.eclipse.org/emf/2002/Ecore#//EInt"/>
</eClassifiers>
<eClassifiers xsi:type="ecore:EClass" name="SequentialBlock">
  <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore">
   <details key="constraints"
          value="SequentialBlockMustHaveAtLeastOneChild"/>
  </eAnnotations>
  <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore/OCL">
   <details key="SequentialBlockMustHaveAtLeastOneChild"
          value="(self.PlainSequence->size() + self.ANDSequence->size() +
          self.XORSequence->size()) >= 1"/>
  </eAnnotations>
  <eStructuralFeatures xsi:type="ecore:EReference" name="PlainSequence"
          upperBound="-1" eType="#//Sequence" containment="true"/>
  <eStructuralFeatures xsi:type="ecore:EReference" name="ANDSequence"
          upperBound="-1" eType="#//ANDSequence" containment="true"/>
  <eStructuralFeatures xsi:type="ecore:EReference" name="XORSequence"
          upperBound="-1" eType="#//XORSequence" containment="true"/>
</eClassifiers>
<eClassifiers xsi:type="ecore:EClass" name="ANDBlock">
  <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore">
   <details key="constraints"
          value="ANDBlockMustHaveAtLeastTwoChildren"/>
  </eAnnotations>
  <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore/OCL">
   <details key="ANDBlockMustHaveAtLeastTwoChildren"
          value="(self.SequentialBlocks->size() + self.XORBlocks->size()) >=
          2"/>
  </eAnnotations>
```

```
  <eStructuralFeatures xsi:type="ecore:EReference" name="SequentialBlocks"
          upperBound="-1" eType="#//SequentialBlock" containment="true"/>
  <eStructuralFeatures xsi:type="ecore:EReference" name="XORBlocks"
          upperBound="-1" eType="#//XORBlock" containment="true"/>
 </eClassifiers>
 <eClassifiers xsi:type="ecore:EClass" name="XORBlock">
  <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore">
   <details key="constraints"
          value="XORBlockMustHaveAtLeastTwoChildren"/>
  </eAnnotations>
  <eAnnotations source="http://www.eclipse.org/emf/2002/Ecore/OCL">
   <details key="XORBlockMustHaveAtLeastTwoChildren"
          value="(self.SequentialBlocks->size() + self.ANDBlocks->size()) >=
          2"/>
  </eAnnotations>
  <eStructuralFeatures xsi:type="ecore:EReference" name="ANDBlocks"
          upperBound="-1" eType="#//ANDBlock" containment="true"/>
  <eStructuralFeatures xsi:type="ecore:EReference" name="SequentialBlocks"
          upperBound="-1" eType="#//SequentialBlock" containment="true"/>
 </eClassifiers>
</ecore:EPackage>
```

2. XML schema representing the content of a process flow in IBM's WBM

```
<xsd:element minOccurs="0" name="flowContent">
    <xsd:complexType>
        <xsd:choice maxOccurs="unbounded">
            <xsd:element minOccurs="0" name="startNode">
                <xsd:complexType>
                    <xsd:sequence>
                                <xsd:element maxOccurs="unbounded" minOccurs="0"
                                        name="entryPoint">
                            <xsd:complexType>
                                <xsd:element maxOccurs="unbounded" minOccurs="0"
                                        name="entryPoint">
                            <xsd:complexType>
                                <xsd:attribute name="inputCriterion" type="xsd:string"
                                        use="required"/>
                            </xsd:complexType>
                        </xsd:element>
                    </xsd:sequence>
                    <xsd:attribute name="name" type="xsd:string" use="required"/>
                    <xsd:attribute name="flowOrder" type="xsd:int" use="optional"/>
                </xsd:complexType>
            </xsd:element>
            <xsd:element minOccurs="0" name="stopNode">
                <xsd:complexType>
                    <xsd:sequence/>
                    <xsd:attribute name="name" type="xsd:string" use="required"/>
                    <xsd:attribute name="associatedOutputCriterion" type="xsd:string"/>
                    <xsd:attribute name="flowOrder" type="xsd:int" use="optional"/>
                </xsd:complexType>
            </xsd:element>
            <xsd:element minOccurs="0" name="endNode">
                <xsd:complexType>
                    <xsd:sequence/>
                    <xsd:attribute name="name" type="xsd:string" use="required"/>
                    <xsd:attribute name="flowOrder" type="xsd:int" use="optional"/>
```

```
        </xsd:complexType>
    </xsd:element>
    <xsd:element minOccurs="0" name="endEventNode">
        <xsd:complexType>
            <xsd:sequence/>
            <xsd:attribute name="name" type="xsd:string" use="required"/>
            <xsd:attribute name="associatedOutputCriterion" type="xsd:string"/>
            <xsd:attribute name="flowOrder" type="xsd:int" use="optional"/>
        </xsd:complexType>
    </xsd:element>
    <xsd:element minOccurs="0" name="task" type="Task"/>
    <xsd:element minOccurs="0" name="receiveTask" type="ReceiveTask"/>
    <xsd:element minOccurs="0" name="decision" type="Decision"/>
    <xsd:element minOccurs="0" name="merge" type="Merge"/>
    <xsd:element minOccurs="0" name="fork" type="Fork"/>
    <xsd:element minOccurs="0" name="join" type="Join"/>
    <xsd:element minOccurs="0" name="notificationBroadcaster"
        type="NotificationBroadcaster"/>
    <xsd:element minOccurs="0" name="notificationReceiver"
        type="NotificationReceiver"/>
    <xsd:element minOccurs="0" name="observer" type="Observer"/>
    <xsd:element minOccurs="0" name="timer" type="Timer"/>
    <xsd:element minOccurs="0" name="map" type="Map"/>
    <xsd:element minOccurs="0" name="loop" type="Loop"/>
    <xsd:element minOccurs="0" name="forLoop" type="ForLoop"/>
    <xsd:element minOccurs="0" name="process" type="Process"/>
    <xsd:element minOccurs="0" name="businessRulesTask" type="BusinessRulesTask"/>
    <xsd:element minOccurs="0" name="humanTask" type="HumanTask"/>
    <xsd:element minOccurs="0" name="localRepository" type="LocalRepository"/>
    <xsd:element minOccurs="0" name="compensationActivity"
        type="CompensationActivity"/>
    <xsd:element minOccurs="0" name="callToProcess">
        <xsd:complexType>
            <xsd:complexContent>
                <xsd:extension base="Invocation">
                    <xsd:sequence>
                        <xsd:element minOccurs="0" name="correlationKeys">
                            <xsd:complexType
                                ecore:name="ProcessCorrelationKeysType">
                                <xsd:sequence>
                                    <xsd:element minOccurs="0" name="inputsKeys"
                                        type="InputCorrelationKeyType" />
                                    <xsd:element minOccurs="0" name="outputsKeys"
                                        type="OutputCorrelationKeyType" />
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                    </xsd:sequence>
                    <xsd:attribute name="process" type="ElementName"/>
                </xsd:extension>
            </xsd:complexContent>
        </xsd:complexType>
    </xsd:element>
    <xsd:element minOccurs="0" name="callToTask">
        <xsd:complexType>
            <xsd:complexContent>
```

```
                    <xsd:extension base="Invocation">
                        <xsd:attribute name="task" type="ElementName"/>
                    </xsd:extension>
                </xsd:complexContent>
            </xsd:complexType>
        </xsd:element>
        <xsd:element minOccurs="0" name="callToBusinessRulesTask">
            <xsd:complexType>
                <xsd:complexContent>
                    <xsd:extension base="Invocation">
                        <xsd:attribute name="businessRulesTask" type="ElementName"/>
                    </xsd:extension>
                </xsd:complexContent>
            </xsd:complexType>
        </xsd:element>
        <xsd:element minOccurs="0" name="callToHumanTask">
            <xsd:complexType>
                <xsd:complexContent>
                    <xsd:extension base="Invocation">
                        <xsd:attribute name="humanTask" type="ElementName"/>
                    </xsd:extension>
                </xsd:complexContent>
            </xsd:complexType>
        </xsd:element>
        <xsd:element minOccurs="0" name="callToService">
            <xsd:complexType>
                <xsd:complexContent>
                    <xsd:extension base="Invocation">
                        <xsd:sequence>
                            <xsd:element minOccurs="0" name="correlationKeys">
                                <xsd:complexType
                                    ecore:name="ServiceCorrelationKeysType">
                                    <xsd:sequence>
                                        <xsd:element minOccurs="0" name="inputsKeys"
                                            type="InputCorrelationKeyType" />
                                        <xsd:element minOccurs="0" name="outputsKeys"
                                            type="OutputCorrelationKeyType" />
                                    </xsd:sequence>
                                </xsd:complexType>
                            </xsd:element>
                        </xsd:sequence>
                        <xsd:attribute name="service" type="ElementName" />
                    </xsd:extension>
                </xsd:complexContent>
            </xsd:complexType>
        </xsd:element>
        <xsd:element minOccurs="0" name="connection" type="Connection"/>
        <xsd:element minOccurs="0" name="annotation" type="Annotation"/>
    </xsd:choice>
  </xsd:complexType>
</xsd:element>
```

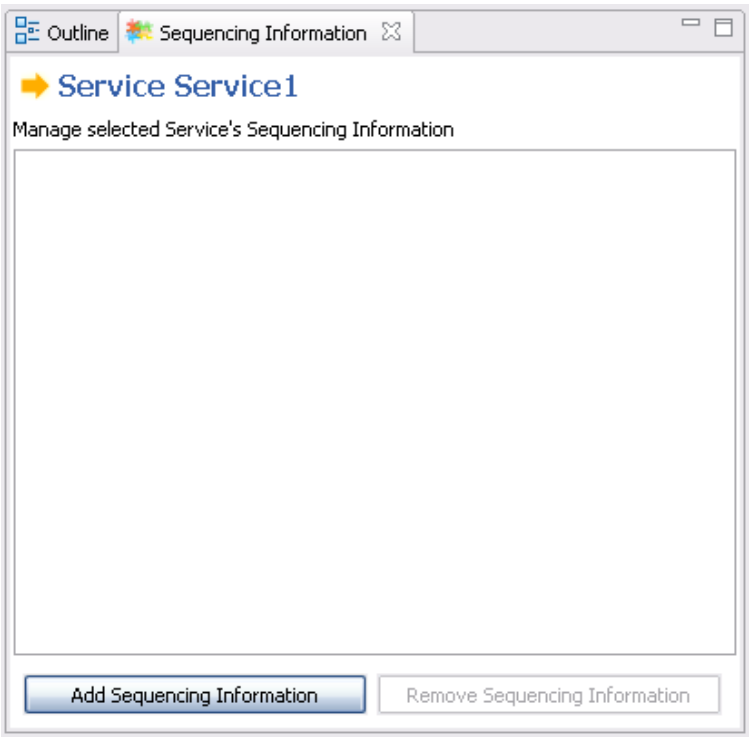
Appendix B

User Manual for managing Sequencing Information in SNAPT

Adding the appropriate sequencing details in a service network is the most important step towards generating the abstract business processes that take part within the service network designed by the SNAPT user. In this appendix, we present a general overview of how to manage sequencing information in SNAPT and we also use a simple example for demonstration purposes.

1. Adding new Sequencing building blocks

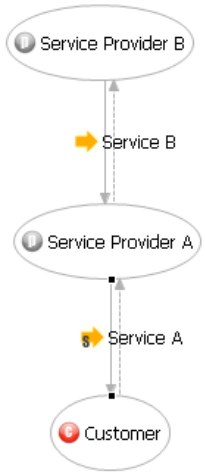
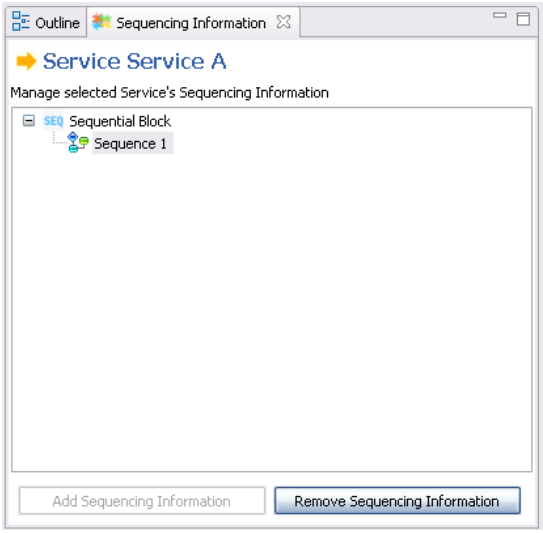
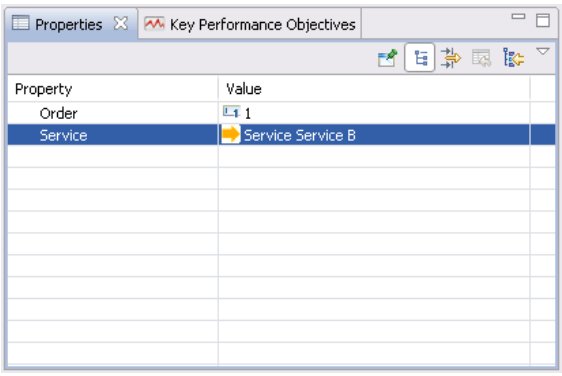
After the user has designed the desired service network in SNAPT the first step is to distinguish the final services that are delivered within the service network. Afterwards, the user must add the sequencing information for all final services and if necessary, for the services that are included in the sequencing information of the final services. The procedure of adding new sequencing information in a service is achieved through the "Sequencing Information" view of SNAPT as depicted in the figure below. The specific view is "activated" whenever a user selects a service from the service network diagram of SNAPT.



Sequencing Information View

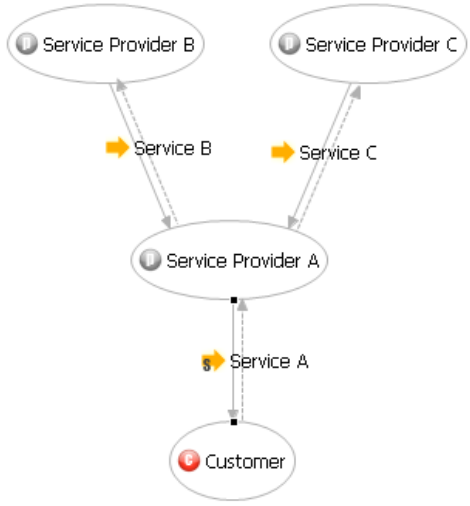
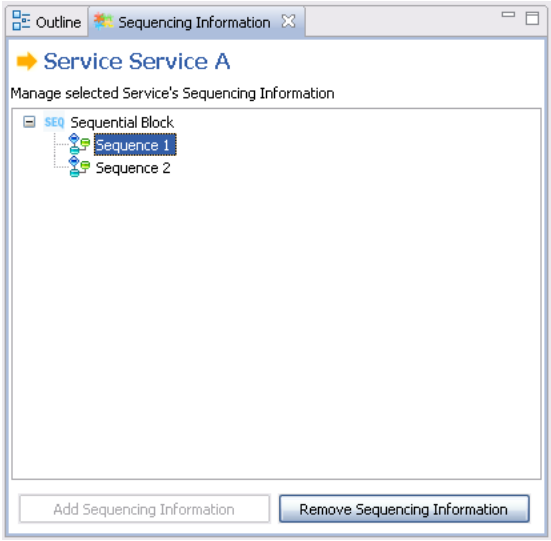
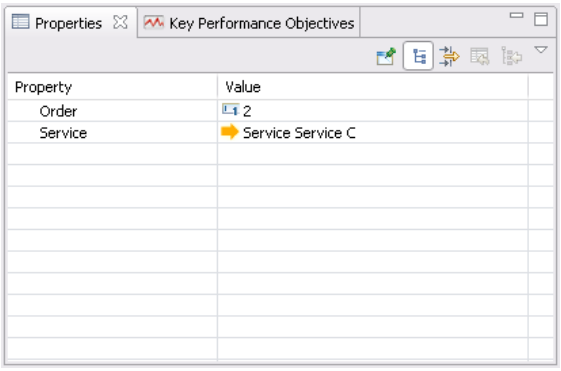
1.1. Adding new Sequence block

As already stated the Sequence block is the simplest building block within a service's sequencing information. Below we demonstrate how a final service is delivered through the simplest form of sequencing information that can be found in a service network, in three steps.

The first step consists of designing the service network diagram. The screenshot in the right depicts a simple service network as designed in SNAPT: the delivery of service A from service provider A to an end customer, requires the delivery of service B from service provider B to service provider A, which in this case consumes service B before delivering service A to the customer.	
In order to depict the previously described service network, the SNAPT user must enter the following building blocks in the sequencing information of Service A via the Sequencing Information view: the top building block is a *Sequential* block that consists of a *Sequence* block as illustrated in the adjacent figure. This is the simplest case of sequencing information that can be assigned in a single service.	
In the final step, the user must choose the service that is under the Sequence block via the Properties view. The available set of services to choose from include all the incoming services that are consumed by service provider A, in this case only service B. Thus, service B must be delivered to service provider A, in order for the final service A to be delivered to the customer.	

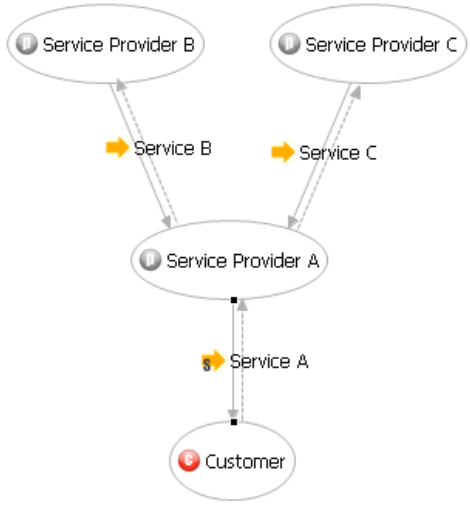
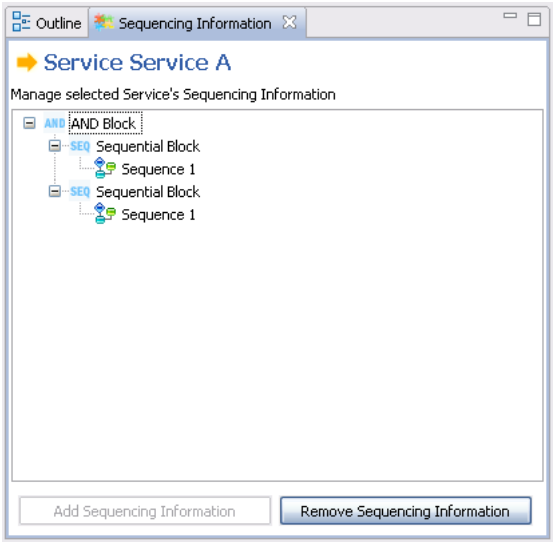
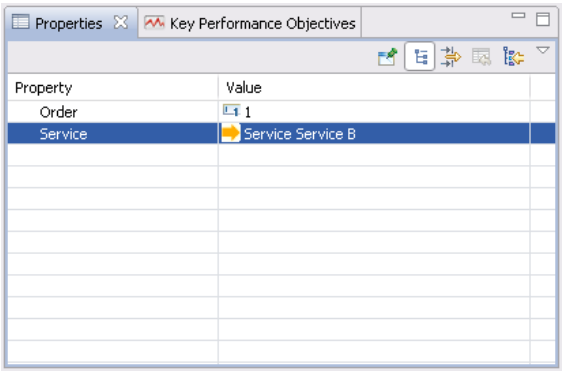
1.2. Adding new Sequential block

In this case we will study the delivery of a final service that requires the service provider to consume two other services in series, in the order specified by the SNAPT user. The same procedure can be followed for more than two services that are delivered in series.

The service network illustrated in the figure delivers the final service A to the customer. In order for the service provider A to deliver service A, he must first consume service B offered by service provider B and then service C offered by service provider C.	
The sequencing information that must be assigned by the user to service A consists of one *Sequential* block that contains two *Sequence* blocks as shown in the screen-shot. It indicates that the two Sequence blocks are executed in series in the order that is depicted by the number next to the Sequence blocks.	
Finally, the user must associate each Sequence block with service B and service C and determine the order in which they must be delivered to service provider A from the Properties view. In this case service C is delivered after service B and must be assigned with order 2, while service B has order 1.	

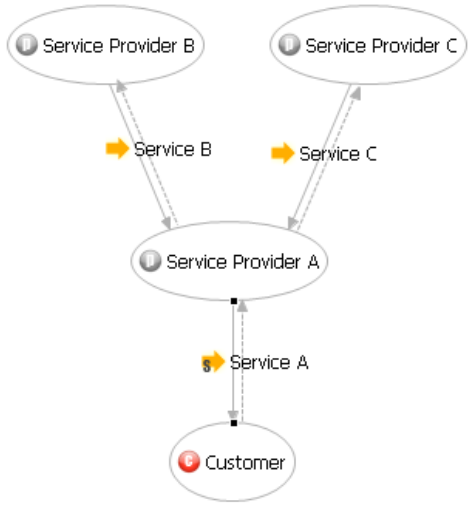
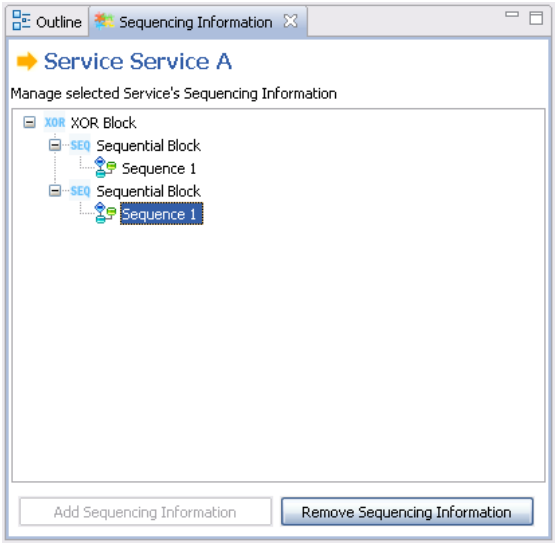
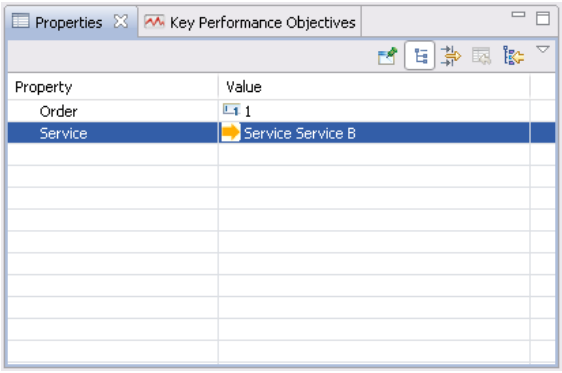
1.3. Adding new AND block

In this case we will study the delivery of a final service that requires the service provider to consume two other services in parallel. The same procedure can be followed for more than two services that are delivered-consumed in parallel.

The figure on the right illustrates the same service network of the previous case. The difference here is depicted on the way the final service A is delivered to the customer. In order for the service provider A to deliver service A, he must consume in parallel, both service B and service C offered by service provider B and service provider C, respectively.	
The sequencing information of service A has an *AND* block as the top building block that is further decomposed to two *Sequential* blocks, each containing a *Sequence* block. The number indicating the order in which the Sequence blocks are executed is not important in this case, as all blocks contained in AND blocks are executed in parallel.	
Finally, the user must assign from the Properties view the Sequence blocks to service B and service C, without determining the order in which they will be delivered to service provider A.	

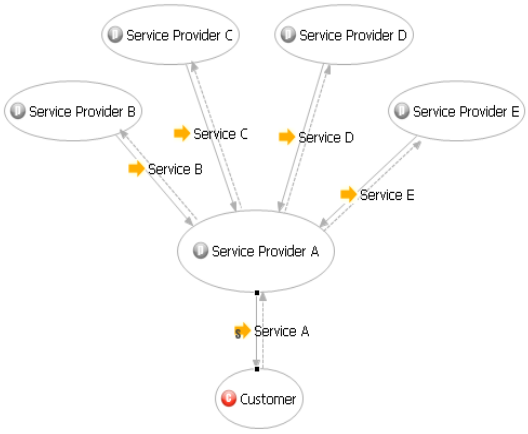
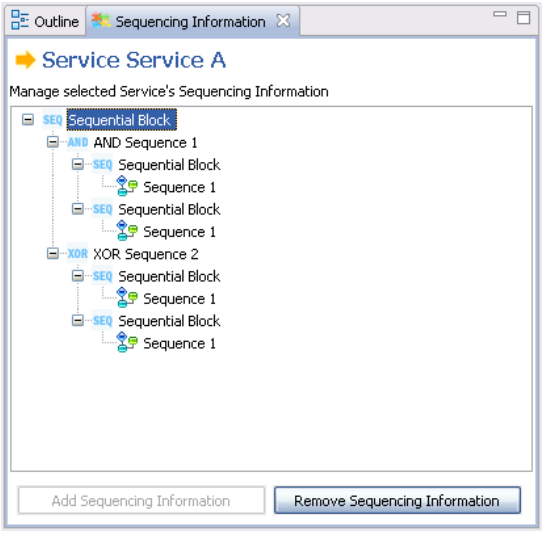
1.4. Adding new XOR block

Here we examine the delivery of a final service that requires the service provider to consume only one of two possible services that are delivered to the service provider. The same procedure can be followed when the service provider has more than two services to choose from, in order to exclusively consume only one of them.

The figure displays the same service network as in the two previous cases. Here, in order for the service provider A to deliver service A, he must consume either service B or service C offered by service provider B and service provider C, respectively.	
The sequencing information of service A has an *XOR* block as the top building block that is further decomposed to two *Sequential* blocks, each containing a *Sequence* block. The number indicating the order in which the Sequence blocks are executed is not important in this case, as exclusively only one of the blocks contained in XOR blocks is executed.	
Finally, the user must assign from the Properties view the Sequence blocks to service B and service C, without determining the order in which they will be delivered to service provider A.	

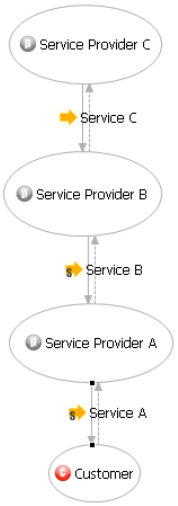
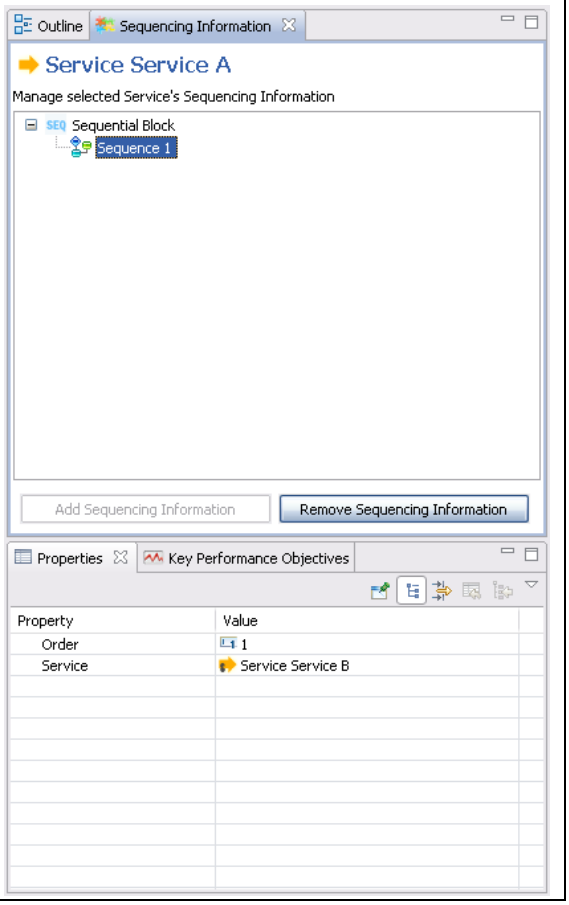
1.5. Adding new ANDSequence, XORSequence blocks

In this case, we examine the delivery of a final service that requires the service provider to consume in a specific way four services offered by four different service providers: he must first consume two services in parallel and then choose one service to consume from two possible services offered. The above sequencing behavior is achieved through the use of ANDSequence and XORSequence blocks.

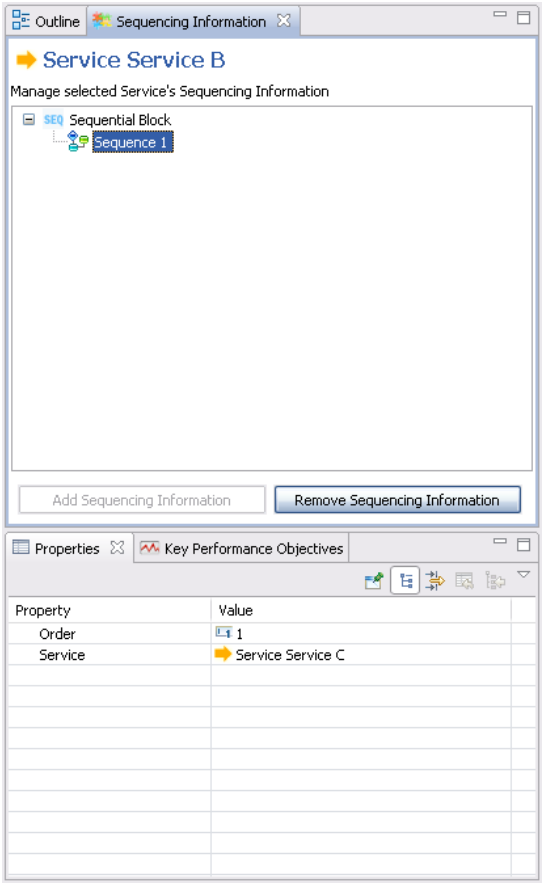
The service network shown in the figure displays the final service A to the customer with a specific way: in order for the service provider A to deliver service A, he must first consume both service B and service C offered by service provider B and service provider C respectively and afterwards choose to consume either service C from service provider C or service D offered by service provider D.	
The sequencing information that the user must assign to Service A is more complicated: the top building block is a *Sequential* block that is consisted of two blocks an *ANDSequence* and a *XORSequence*. The number next to these blocks indicates that the ANDSequence block is executed before the XORSequence block. Each of the ANDSequence, XORSequence block is decomposed to two *Sequence* blocks, same as the AND, XOR blocks are described in section 1.3 and 1.4 of the appendix, respectively. The Sequence blocks of the ANDSequence are associated with services B and C that must be consumed in parallel, while the Sequence blocks of the XORSequence are associated with services D and E, that exclusively one of them will be consumed by service provider A.	

1.6. Adding sequencing information in multiple services

In this case, we examine the delivery of a final service that requires several service providers to consume services until the final service becomes available to the customer. We will demonstrate the consumption of two services by different service providers, but the same procedure can be followed for multiple services consumed by several service providers in many possible ways.

The service network of the figure delivers the final service A to the customer. The delivery of service A requires the service provider A to consume service B offered by the service provider B, which in turn requires the consumption of service C from the service provider B. Thus, the SNAPT user needs to assign sequencing information to two different services: service A and service B as described below.	
The sequencing information that the user must assign to Service A consists of a *Sequential* block that contains a *Sequence* block, which is associated with the delivery of service B offered by service provider B. It indicates that service B must be consumed by service provider A in order for him to deliver service A to the customer.	

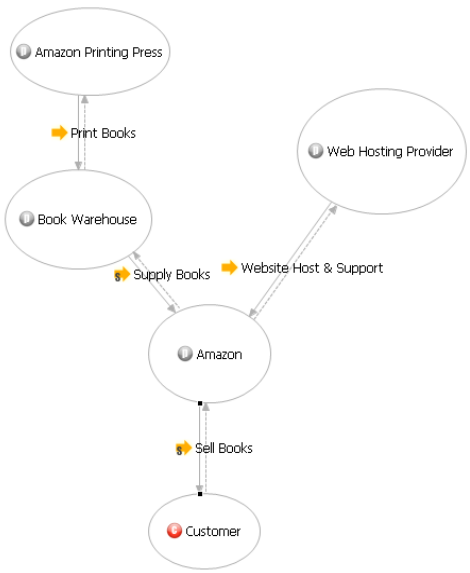
The sequencing information that the user must assign to service B is similar to the one assigned for service A. It consists of a *Sequential* block that contains a *Sequence* block, which is associated with the delivery of service C offered by service provider C. It indicates that service C must be consumed by service provider B in order for him to deliver service B to the service provider A.

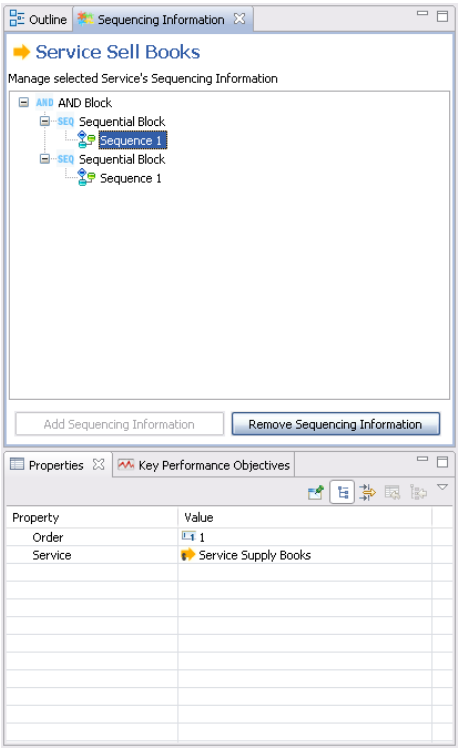


2. Adding Sequencing Information in a real-world Service Network

In this section, we present how to add the appropriate sequencing information in a simplified real-world case study. The case study discussed is based on the Amazon's service network that delivers the final service of selling books to the end customers.

As depicted in the figure the simplified Amazon's service network consists of four different service providers and the end customers that consume the final service of selling books provided by the Amazon. The Amazon needs to consume both the *Supply Books* service offered by the Book Warehouse and the *Website Host & Support* service offered by the Web Hosting Provider in order to deliver the *Sell Books* service. The Book Warehouse must consume the *Print Books* service offered by the Amazon Printing Press in order to provide the *Supply Books* service.



The user must assign the following sequencing information to the *Sell Books* service as described in section 1.3 of the appendix B. The *Sell Books* service requires the parallel consumption of two services offered by two different service providers. Thus, its sequencing information consists of an *AND* block that is decomposed to two *Sequence* blocks. Each Sequence block is associated to the *Supply Books* service offered by the Book Warehouse and the *Website Host & Support* service offered by the Web Hosting Provider respectively, as they form the composition of the *Sell Books* service offered by the Amazon.	
Another service that needs to be assigned with the appropriate sequencing information by the SNAPT user is the *Supply Books* service offered by the Book Warehouse. In order for this service to be provided to the Amazon, the books must be printed from the Amazon Printing Press, which is a service consumed by the Book Warehouse. Thus, the sequencing information of the *Supply Books* service is assigned according to the 1.1 section of this appendix. It is the simplest form of sequencing information that is associated with the single consumption of the *Print Books* service offered by the Amazon Printing Press in order for the Book Warehouse to be enabled to offer the *Supply Books* service to Amazon.