Factors Affecting Reusability in Service Oriented Architecture

An Empirical Investigation in Three Companies

Ershad Ebrahimi

Department of Computer and Systems Sciences

Degree project 30 HE credits
Degree subject computer and Systems Sciences
Degree project at the master level
Spring term 2012
Supervisor: Elin Uppström
Co-Supervisor: Jelena Zdravkovic
Reviewer: Hercules Dalianis
Factors Affecting Reusability in Service Oriented Architecture

An Empirical Investigation in Three Companies

Ershad Ebrahimi

Abstract

Service Oriented Architecture (SOA) is an architecture which provides a single topology for business applications. SOA provides architecture for integration of several applications within or outside of organizations by reusing application components and services. Reusability is one of the mostly emphasized principles and benefits of SOA. Code reusing has been a concern for application developers. However, another way of reusability is reusing software components and services in SOA. By reusing services, a service can be used in more than one business process of service application.

By considering the significance of reusability in SOA, a number of studies have been performed, to theoretically identify the factors affecting reusability in SOA. However, a research is needed to perform an empirical research to identify a set of factors affecting reusability in real business situation.

The empirical study is performed by an investigation in three different companies in Sweden, which have been customers of a system integration company and have implemented SOA projects. The investigation is done by performing 5 semi-structured interviews with correspondences of those companies, to find more factors affecting reusability from their perspective.

As a result, a set of factors derived from the empirical study is identified which reflects the ideas of practitioners in business. By performing a qualitative approach and comparing the factors derived from the empirical study to factors derived from literature study, it is concluded that academic factors constitute just around 40 percent of factors found in business, and they are mostly technical factors. The rest of factors are new factors based on experience of practitioners in business. The new factors are mostly business oriented factors, and have emerged since 2008, when earlier works are performed. Due to the recent maturity of SOA in organizations, the factors affecting reusability have changed, especially during the last four years.

Keywords

SOA, Service Oriented Architecture, Reusability, Service
Acknowledgements

First and foremost, I would like to provide the greatest appreciation to my supervisor, Elin Uppström, PhD student of Stockholm University for all her consultant, help and guidance for supporting this master thesis. This thesis would not be possible without her guidance and constructive comments.

I would like to thank Jelena Zdravkovic, PhD, MBA at Royal Institute of Technology in Stockholm, for all her constructive comments and guidance on my master thesis.

Moreover, I would like to express my deep gratitude to Joakim Lundin, team leader and consultant at Dynabyte Company for helping me to organize interviews and spending time to validate the result of this thesis.

I wish to thank all the interviewees and experts including Architects and Consultants of companies consisting of: two major Telecommunication Companies, electronics selling companies and Dynabyte Company for their valuable input and support for this thesis.
Table of Contents

List of Tables .................................................................................................................. 6

1. Introduction .................................................................................................................. 7
   1.1 Background .............................................................................................................. 7
   1.2 Problem .................................................................................................................. 7
   1.3 Research Question ................................................................................................. 8

Extended Background ....................................................................................................... 10
   1.4 Service Oriented Architecture ............................................................................... 10
      1.4.1 Services ........................................................................................................... 11
      1.4.2 Analogy ........................................................................................................... 12
   1.5 Common Principles of SOA .................................................................................... 12
      1.5.1 Reusability of Services .................................................................................. 12
      1.5.2 Loose Coupled Services ............................................................................... 13
      1.5.3 Composition of Services .............................................................................. 14
      1.5.4 Discoverable Services .................................................................................. 14
      1.5.5 Statelessness of Services .............................................................................. 14
      1.5.6 Related Work .................................................................................................. 15

2 Method ........................................................................................................................... 17
   2.1 Choice of Method .................................................................................................... 17
   2.2 Application of Method ............................................................................................ 19

3 Results ............................................................................................................................ 23
   3.1 Factors Affecting Reusability Derived from Literature Study ......................... 23
      3.1.1 Using Consistent Business Terms to Describe Services ............................... 24
      3.1.2 Managing New Service Creation and Discovering of Existing Services .... 24
      3.1.3 Providing Assurance to Both Service Users and Owners ............................ 25
      3.1.4 Managing Service Enhancements and Changes .......................................... 25
      3.1.5 Using Standard Superior Interfaces ............................................................... 25
      3.1.6 Business Commonality .................................................................................. 25
      3.1.7 Modularity ....................................................................................................... 26
      3.1.8 Adaptability ..................................................................................................... 26
      3.1.9 Standard Conformance .................................................................................. 26
      3.1.10 Discoverability .............................................................................................. 26
      3.1.11 Separation of Processes ............................................................................... 26
      3.1.12 Message Exchange with Multiple Patterns ............................................... 27
      3.1.13 Stateless Services ......................................................................................... 27
      3.1.14 Cost of Reuse vs. Benefits of Reuse ............................................................. 27
   3.2 Derived Factors from the Empirical Study ............................................................... 27
      3.2.1 Documentation ................................................................................................. 28
4 Analysis .................................................................................................................. 35
4.1 Combining Derived Factors from Empirical Study ................................................. 35
  4.1.1 Managing and Governing Services ................................................................ 35
  4.1.2 Standard Conformance .................................................................................... 35
4.2 Mapping Factors from Empirical Study to Literature Study ................................. 37
  4.2.1 Using Consistent Business Terms to Describe Services ............................... 38
  4.2.2 Managing New Service Creation and Discovering of Existing Services ......... 39
  4.2.3 Managing Service Enhancements and Changes ............................................. 39
  4.2.4 Standard Conformance .................................................................................... 40
  4.2.5 Political Factors ............................................................................................... 40
  4.2.6 Making Services as General as Possible ......................................................... 40
  4.2.7 Using Common Message Model ..................................................................... 40
  4.2.8 Documentation ................................................................................................ 40
  4.2.9 Having Generic Information Model ............................................................... 41
  4.2.10 Having Common Information Model ............................................................ 41
  4.2.11 Business Process and Information Model Adherence ................................... 41
5 Discussion ................................................................................................................. 44
  5.1 Conclusion ........................................................................................................... 45
  5.2 Limitations and Future Research ....................................................................... 46
6 Bibliography ............................................................................................................ 48
7 Appendices .............................................................................................................. 50
  7.1 Appendix A: Interview’s Questions .................................................................... 50
  7.2 Appendix B: Table of factors driven from literature review .............................. 51
  7.3 Appendix C: Interviews ...................................................................................... 52
    7.3.1 Interview 1 .................................................................................................... 52
List of Figures

Figure 1: An early incarnation of SOA (Erl, 2005, p. 75) ................................................................. 10
Figure 2: A reusable service exposes reusable operations (Erl, 2005, p. 293) .............................. 13
Figure 3: The Update Everything operation encapsulating a service composition (Erl, 2005, p. 301) 14
Figure 4: Mapping of key features and derived quality attributes (Choi & Kim, 2008, p. 295) ....... 15
Figure 5: Managing and governing services ..................................................................................... 35
Figure 6: Standard conformance ........................................................................................................ 36
Figure 7: Mapping factors from empirical findings to theoretical findings .................................. 38
Figure 8: Using consistent business terms to describe services ....................................................... 38
Figure 9: Managing new service creation and discovering of existing services ....................... 39
Figure 10: Managing service enhancements and changes ............................................................... 39
Figure 11: Standard conformance ..................................................................................................... 40

List of Tables

Table 1: interviewees information .................................................................................................. 21
Table 2: Factors derived from literature review ............................................................................ 24
Table 3: Factors derived from empirical study .............................................................................. 28
Table 4: Factors derived from interviews after combining ............................................................ 37
Table 5: List of empirically driven factors after mapping with factors derived from literature study 42
1. Introduction

1.1 Background

Organizations need to stay aligned with technology in order to gain the competitive advantage in the market. Hence, they need to replace legacy systems with new systems which are based on recent information systems. The term “legacy systems” refers to existing information systems which are developed in past and are handling critical business processes in an organization; therefore, they are of significant business value for organizations (Lewis, et al., 2005). The migration to new systems may be too expensive for organizations because they should discard obsolete applications and purchase new application to adapt with new systems. To overcome this problem, an architecture is needed that can reuse older application components and integrate them with new applications in order to align with business goals of organizations; therefore, organizations expose all parts of their application as services to be able to integrate them (Schepers, et al., 2008). A service is a software entity that can have interaction and communication with other services and applications (Lewis, et al., 2005). A Service Oriented Architecture (SOA) is a group of principles for developing new applications from the integration and interconnection of services and software components that are building up the fundamental of business applications. SOA provides a solution for complexities achieved during development and integrations of enterprise applications according to business goals of enterprises and structure of organizations (Schepers, et al., 2008).

Migrating to new information systems architecture becomes cheaper and easier with SOA, because there is no need to discard or make any significant changes to current applications and services. They can be reused in the new architecture of organization (Smith, 2007). However, implementing SOA architecture can be a complex task, especially when services are expected to be used in a limited environment (Smith, 2007). SOA provides the possibility to reuse functionality of existing services and applications rather than creating them from scratch. In order to provide an integration solution, there are distributed systems and services which have dependencies on the functionality of other systems. SOA provides a solution to overcome these dependencies of distributed systems with a loosely coupled structure. (He, 2003)

Reusability is one of the most important advantages of SOA (Chappell, 2006). It provides some benefits for faster application integration with lower costs, because new applications can be created based on current applications by reusing components of current applications (Chappell, 2006). Reusing application components and services prevents enterprises from reworking to produce new services, and provides the possibility to reuse services which are already being used and interconnects them to the new information systems. In other words, maximization of reuse of services improves flexibility of business and return of investment. So, successful reusing of services and software components is highly emphasized and has gotten significant importance in SOA. Reusability has attracted the interest of software engineers because it increases productivity of software and has a great potential for making benefits for enterprises. (Uday, et al., 1990)

1.2 Problem

Information systems are important resources and business key competitions for organizations, because information systems are significant resources in conducting organizations to gain competitive advantages in the market (Brynjolfsson & Hitt, 1996). According to Berg et al (2007), the next chapter
of migration to IT in organizations is performed by Service Oriented Architecture (SOA) (Berg, et al., 2007, p. 17). Due to the SOA profits and advantages for businesses, organizations have an interest in investment on implementing SOA for their business (Berg, et al., 2007, p. 14).

In addition, one of the benefits of SOA is its ability to reuse services and software components. Migrating to SOA solutions requires fundamental changes in legacy systems and services. Without making services reusable, migrating to SOA is very costly and time consuming therefore, reusing software components and service will save cost for development and testing of new applications (Lewis, et al., 2005). Achieving a higher degree of reusability is a concern for system integrators when implementing service oriented architecture, because in many cases businesses are highly dependent on application components and services which are being used (Lewis, et al., 2005).

Reusability is an important criterion in service oriented architecture, some reasons for the importance of reusability consist of: successful reuse of services achieves agility to business, saving money and return of investments. The importance of reusability has made many researchers study on reusability in SOA (Dan, et al., 2008, Street & Gomaa, 2008, Vegter, 2009, Choi & Kim, 2008). These researchers have performed studies on evaluation of challenges and issues affecting reusability in SOA. They have found different factors which are important in reusing services and software components. Dan et al (2008) have performed a research which “details key challenges in achieving reusability of services and advocates taking a pro-active position in addressing these challenges”(Dan, et al., 2008, p.25). Choi & Kim (2008) have proposed a comprehensive quality model to evaluate service reusability. They have defined metrics for each quality attributes, and finally they perform a case study to measure the reusability of quality metrics in a real domain (Choi & Kim, 2008). Street & Gomaa (2008) have “explored software architectures and pattern for SOA using Unified Modeling Language (UML)” (Street & Gomaa, 2008, p.1). They have specifically studied issues related to composing applications from reusable services (Street & Gomaa, 2008). Vegter (2009) has discussed the critical success factors in SOA and have tested the factors by performing a case study in a financial sector (Vegter, 2009).

In all the previously mentioned studies, there is a lack of empirically supported research in a certain context to identify new factors affecting reusability in a real business situation rather than prioritization or testing of factors. In other words, these researchers have found factors theoretically. Just Vegter (2009) has focused on testing the existing theoretical factors in a real case; therefore, no research has been performed in a context by focusing on finding new factors in real cases from the perspectives of those practitioners who have adapted SOA in their organizations. On the other hand, studies regarding to deriving factors affecting reusability are at least four years old; as a result, a research is needed to find the new factors appeared in the business during last four years.

1.3 Research Question

The main goal of this thesis is to find factors affecting reusability by performing an empirical research in three companies in Sweden. The context of this research is a system integration company (Dynabyte AB) which provides system integration solution for other organizations. Therefore, the empirical study is performed by investigating experiences from three different organizations that are clients of Dynabyte AB. The reason for choosing these companies is that, they are the only customers of Dynabyte AB that have implemented SOA project completely in their company. Moreover, these three companies are accessible for performing the empirical study of this master thesis. The organizations are big companies in Sweden and consist of two major telecom companies and a leading electronic seller company.
To achieve the above mentioned goal and fulfill the lack of research mentioned in the problem section, the following principal research question should be answered:

What factors affect reusability in SOA implementation in companies under investigation?

The objective of this research is to find the factors affecting reusability in companies under investigation, and then compare and analyze them in regard to possible factors found by literature study to elicit the new factors which are absent in previous works.

The expected result would be a list of factors which are derived from investigations in three companies that have adopted SOA. Therefore, the finding of this master thesis helps practitioners to get insights about factors in reusability in real business situations. In addition it may guide them toward more efficient reusability when they want to adopt SOA and reuse services and application components in their organizations. The result can also be of interest for the telecom industry, because two out of three studied companies are major telecom companies. So, the factors, which are common for these two telecom companies, might be applied to other telecom companies.

In addition this thesis will contribute knowledge to the existing research within the information system field. Therefore, this research contributes to science by adding to academic materials which are available in factors affecting reusability in SOA. Since the previous works have been performed before 2008, this work reflects differences emerged in factors affecting reusability in organizations during four years.

This thesis is organized in six sections: In section 2, an extended background is presented. In this section literature review about service oriented architecture, its benefit and reusability is described. In section 3 methodology used for this study is presented. In section 4, factors derived from the literature review is also listed and elaborated and then, the empirical findings from interviews are presented. Analysis of factors affecting reusability is described in section 5. Finally in section 6 we present discussion, conclusion and future work of research.
Extended Background

1.4 Service Oriented Architecture

SOA was first described by Gartner in 1996 (Natis, 2003). SOA is an architecture that builds up the topology of whole applications as a single topology of interfaces. In Natis (2003) it is a so called "Interface Oriented Architecture." (Natis, 2003)

Services are part of applications called through an interface, and the service consumers are the applications that use the interface of that service. Web services are a technology specification and can act as an interface for SOA solutions; however SOA is a principle for designing software (Natis, 2003).

SOA in general is providing a single general architecture for business applications. Nowadays, on one hand, companies have their own applications with customized interfaces and new projects are coming up to reuse these applications and business logics behind them. On the other hand, there are lots of clients with different conditions, such as different operating systems, different devices which want to use the same back-end business applications. Therefore, SOA, as a loosely coupled architecture, provides a solution for reusing the back-end business applications by different clients through different channels (Natis, 2003).

Other important criterion in system integration is the reuse of old services and application components to provide composite applications in SOA architecture. Composite application is a software program which is built out of other software applications. Composite applications have appeared as a new method for developing applications. Many new applications, which are developed recently, are dependent on external applications; therefore, they compose composite applications. Composing and integration of different application components including old and external applications is the aim of the SOA solution (Natis, 2003). When enterprises purchase an application package, they want it to be adaptable with their existing applications. Therefore, most of the applications vendors are designing their application in a way that supports SOA interfaces. (Natis, 2003)

According to Erl (2005) service oriented architecture has three basic elements. These elements are: service provider, service requester and service registry which are shown in figure 1:

![Figure 1: An early incarnation of SOA (Erl, 2005, p. 75)](image-url)
According to this figure, the service provider supplies a web service and publishes the interface and description of service to the service registry using Web Service Definition Language (WSDL). WSDL is an XML-based language which is used for describing functionalities provided by web services. Service registry is a repository of information and description about services which can be used by the service provider or the service requester. A standard service registry format can be created by Universal Description, Discovery and Integration (UDDI). UDDI is a registry and a mechanism for registering and locating web service application. After publishing service descriptions and interfaces by service provider, the service requestor sends a request to a service registry to find and retrieve required WSDL. After that, service requestor binds to service provider to invoke the service. The connection between service requestor and service provider is by exchanging Simple Object Access Protocol (SOAP) messages. SOAP is a protocol for exchanging information in web services. (Erl, 2005)

### 1.4.1 Services

Service is a fundamental concept in SOA. The architecture of SOA is based on services. It means that all the application components are converted to services in SOA. Therefore, service is an important concept in SOA.

Service is not a technology term; the concept of services firstly has been used in business. There are many services which are being provided by a business such as cleaning services and parking services. For each of these services someone or an organization is offering a service to its environment and consumers use and benefit from the services (Open Group, 2009). The concept of software services has adopted the idea of services and applied them to technology. It also has provider and consumer of services. Consumers are customers (people or organizations) that use the benefits of software services, and providers are those who are responsible for running the software services. These services can be provided over a network or internet, which has a human interface. Software services can be created inside an organization over their internal network, and another program running on a different machine can access a service and use it. This is how software as services is organized, and it also creates the essence of SOA. (Delgado, 2010)

Using services can provide following benefits for organizations:

- Comparing to large enterprise applications, using grained software services makes a better possibility for information movement between or within organizations, because integration of large enterprise applications is very costly, but SOA makes integration cheaper (Open Group, 2009).

- Moving to software as a service rather than large enterprise applications provides the possibility to reveal the system functionality externally. This can provide business value by bringing visibility to the business (Open Group, 2009).

- The business processes inside organizations are highly dependent on software and applications which are being used. Changes in business processes require modifications in the supporting software. Therefore, service based software architectures is more flexible to be changed in compare to large enterprise applications and provides greater flexibility to the business process of organizations and makes changes easier (Open Group, 2009).
1.4.2 Analogy

Exemplifying helps better understanding the concept of SOA. In this example, an airline company wants to expand its services to let customers book flights, hotels and make car reservations at the same time. In this new system, the airline wants to reuse and build services from other airline companies, hotels and car renting companies (Kraflzig, et al., 2004). SOA provides architecture to unify business processes of organizations by structuring a big airline system from a number of internal small services and external services from other airline companies, hotels and car renting companies. SOA architects a new application by mixing services from a pool of services (Dejavu, 2009). Creating this airline system from existing services of other businesses is much easier and more deployable. The ability to reuse existing services decreases the cost of developing new services for flight, car and hotel reservations and maximizes agility for developing new services.

1.5 Common Principles of SOA

According to Erl (2005) there is not an official list of principles for SOA. However, Erl (2005) has derived a list of common principles which are most related to SOA. This list contains eight principles: reusability of services, services share a formal contract, loosely coupled services, services abstract underlying logic, composition of services, autonomous services, stateless services and discoverable services. (Erl, 2005)

However, of these eight principles, five which are more relevant for this research and useful for the rest of the study are elaborated in following. Principles consisting of: services share a formal contract, services abstract and autonomous services are not elaborated because these three principles have no direct relationship with the scope of this research which is focusing on reusability principle of SOA. The five principles, which are discussed here, have a direct relation with the reusability principle and contribute in factors affecting reusability in SOA.

1.5.1 Reusability of Services

Reusability is one of the mostly emphasized principles and benefits of SOA (Stevens, 2002). Reuse of code in software development has been a concern during the last decades. However, due to the variety of languages and platforms it has been very hard to achieve reuse in software development. Another way to achieve reusability is reusing application components or services. Reusing services and binding to a service is very easy, such as finding a service in a list and bind to it. So developers do not worry anymore about the incompatibilities that happen during code reuse (Stevens, 2002). Using current application components and services rather than writing new software will decrease the cost of development and testing of application which can provide a great saving and return of investments (Open Group, 2009).

According to Erl (2005), reusability of services is always emphasized by SOA regardless if it is immediately required for reuse. By conforming to the standards when designing services, the chance of services to be being adapted in the future with minimum development is highly increased. Figure2 illustrates a reusable service that exposes the reusable operations. (Erl, 2005)
Figure 2: A reusable service exposes reusable operations (Erl, 2005, p. 293)

In this figure, the operations: get account, update account and add account are reused by multiple service requestors such as an account service. (Erl, 2005)

In addition to the design and nature of services, messaging can also support reusability of services indirectly. A SOAP envelope can contain metadata details in the header of a package of data into a single package. The metadata contains business rules and process instructions for recipients that inform the recipient how messages should be processed. Embedding message processing information in the message makes the services less specific to the activity. This makes operations of services more generic. Generic services have a higher degree of reusability in compare to specific services. (Feuerlicht & Lozina, 2007)

1.5.2 Loose Coupled Services

Loose coupling is another benefit in SOA. The concept of coupling refers to the dependencies between modules. Coupling consists of two parts: Loose coupling and tight coupling. Modules, which are loosely coupled, have just a few dependencies which are known and under control. Tightly coupled modules are those that have many unknown dependencies. Software architectures try to achieve more loosely coupled modules rather than tightly coupled. SOA advances the concept of loose coupling and limits well-known dependencies between software modules. This means finding approaches for decreasing dependencies between service providers and service consumers (McGovern, et al., 2003).

SOA implements loose coupling via contracts and bindings. A Service consumer connects to the registry of service provider; the registry provides a list of services which are available and matches to the request from the service consumer. The service consumer select services among the offered services based on descriptions that exist in the registry bind to it and then execute a method. So the service consumer has a dependency only on a contract over the service which is provided. Since the dependency between the service provider and the consumer is online, a loosely coupled dependency has been emphasized by SOA (McGovern, et al., 2003).
1.5.3 Composition of Services

Service Composition is another benefit of SOA that builds new services by composing other services. Service modular structure of applications creates service composition. Modular structure provides services to be composed and being reused in new applications. Reusing existing and tested services improves the quality of systems and return of investments (McGovern, et al., 2003). In figure 3, an operation which is named update everything is illustrated. This operation encapsulates three other service compositions (Erl, 2005).

![Figure 3: The Update Everything operation encapsulating a service composition (Erl, 2005, p. 301)](image)

In this figure, the service update everything is composed of other services: update history, update account and update log. In order to enable “update everything” service, all the participant services must be enabled. Composition can be considered as another form of reuse, in order to maximize composition opportunities, operations must be according to standard manner and sufficient level of granularity. (Erl, 2005)

1.5.4 Discoverable Services

Discoverability prevents accidental creation of redundant services. Since each service provides a reusable processing logic, the description and metadata of a service should describe all the functionality and purposes of the service to make it discoverable (Papazoglou, et al., 2007). According to Erl (2005) discoverability on SOA level means to create discovery systems to make the design of new services as discoverable as possible. Discoverability systems are service registries or directories that keep metadata about services. (Erl, 2005)

1.5.5 Statelessness of Services

Services must avoid keeping state data that they process and limit the time which they hold it. State data means the data about the current operation of a service. When a service receives a message to process, the service keeps the state. After processing the message, the service should not keep any state information of the message. In other words, service must be unaware of the message after
finishing the processes. If the service keeps the state information of activity for a longer period of time, it loses availability to other requests (Srinivasan & Treadwell, 2005). Erl (2005) considers statelessness of services as an important condition for reusability and scalability of services. (Erl, 2005)

1.5.6 Related Work

Because of the importance of the reusability principle of SOA, a number of studies have been performed on reusability issues in SOA for example, (Dan, et al., 2008, Narayanan, 2009, Street & Gomaa, 2008, Vegter, 2009, Choi & Kim, 2008, Branca & Atzori, 2011, Newcomer & Lomow, 2005, Dunn & Knight, 1993, Fichman & Kemerer, 2001). In these studies, reusability in SOA, issues and challenges that may affect on service reuse are discussed. Some of these studies which have focused on factors affecting reusability in any way and those which can be used as a reference for result of this study are discussed in the rest of this section.

Choi and Kim (2008) have evaluated current literatures and listed eleven key features of services in SOA. However, according to definition of the reusability Choi and Kim (2008) have mapped those key features to five quality attributes that can cover all the key features of services (Choi & Kim, 2008). These five derived quality attributes are considered to be relevant factors and are mentioned in table 1. Branca & Atzori have also used these derived factors for their survey of SOA technologies in NGN network architectures (Branca & Atzori, 2011). In figure 4, the mappings of the key features and derived quality attributes of Choi and Kim (2008) is shown:

![Diagram](image)

**Figure 4: Mapping of key features and derived quality attributes (Choi & Kim, 2008, p. 295)**

In this figure, solid arrows represent a strong derivation and dashed lines represent weak derivation (Choi & Kim, 2008) where:

**Business Commonality**: Shows how common it is to consumers in a certain domain to use functionality of services (Choi & Kim, 2008).
**Modularity:** Means to have independent functionality and do not have any dependency on other services (Choi & Kim, 2008).

**Adaptability:** Degree in which a service can be adapted to different service users (Choi & Kim, 2008).

**Standard Conformance:** Degree in which a service is conformed to the industry standards (Choi & Kim, 2008).

**Discoverability:** Degree in which a service is found easily and correctly by service consumers (Choi & Kim, 2008).

Dan et al. (2008) have focused on key governance aspects for service reusability and identified four aspects of SOA that need to be governed in order to improve reusability in SOA (Dan, et al., 2008). The key governances consist of: governing enterprise-wide use of consistent business terms, governing new service creation and discovery of existing services, governing service entitlement and governing service enhancement (Dan, et al., 2008).

Vegter (2009) has also used those four factors produces by Dan et al (2008) in his research to measure critical success factors for a SOA implementation (Vegter, 2009).

Street and Gomaa (2008) have considered reusability of services and software as a goal, to achieve this goal, they constituted some criterion to take into account (Street & Gomaa, 2008). However, some of the factors mentioned by Street and Gomaa (2008) overlap with factors mentioned by Den et al (2008).

We use all the above mentioned studies to derive and elicit possible factors affecting reusability from the theoretical point of view. The factors derived from these studies are presented in the result section.
2 Method

2.1 Choice of Method

The purpose of information systems within organizations is to increase the efficiency and productivity of organizations (Hevner, et al., 2004). Factors affecting reusability in SOA is in the field of information systems because, this research aims at improving the application of information technology in organizations and managerial processes of information systems in organizations (ISR, 2002).

Hevner et al (2004) argues that to accomplish these kinds of research, two paradigms exist, behavioral science and design science. They are two separate paradigms, but they complement each other (March and Smith 1995, Hevner et al 2004). "The behavioral-science paradigm has its roots in natural science research methods. It seeks to develop and justify theories (i.e., principles and laws) that explain or predict organizational and human phenomena surrounding the analysis, design, implementation, management, and use of information systems" (Hevner, et al., 2004, p. 76). Behavioral science focuses on the interactions between technology and human or organizations that are managed with the help of information systems (Hevner, et al., 2004). March & Smith (1995) expressed that natural science intends to understand the reality. Many theories and principal explanation of phenomenon are results of natural science (March & Smith, 1995).

Design Science is the second paradigm identified by Hevner et al (2004). Behavioral science, as an alternative method, has its roots in natural science, but design science has roots in engineering disciplines and sciences of the artificial. Behavioral science aims at development and justification of theories while design science aims at utility and creating innovations (Hevner, et al., 2004). Design science "is fundamentally a problem solving paradigm. It seeks to create innovations that define the ideas, practices, technical capabilities, and products through which the analysis, design, implementation, management, and use of information systems can be effectively and efficiently accomplished" (Hevner, et al., 2004, p. 76). Design science is the study of artifacts and the environment surrounding the artifact, as it is under development or being used by human to solve practical problems (P.Johannesson & E.Perjons, n.d.).

This research is using design science method, because the aim of this paper is to provide a solution for a practical problem by creating new artifacts. This artifact is the list of factors affecting reusability in SOA which is found by investigation in three different companies.

This study is not behavioral science, because it does not have roots in natural science and justification of theories. Behavioral science, as an alternative method, could be used if it was intended to study a human behavioral in response to artifacts and interaction of users and organizations with artifacts.

Johannesson and Perjons defined six research strategies as the most well established ones in the field of design science. They are surveys, experiments, case studies, action research, and ethnography. A survey is conducted by collecting data from different group of experts, organizations and systems (P.Johannesson & E.Perjons, n.d.). Survey has its drawbacks: accessing to appropriate people or organizations is not always possible; however, in this study a number of experts are accessible at Dynabyte AB and their customers who have implemented SOA and have experience in reusability in SOA. As a result, a qualitative survey seems to be an appropriate research strategy for this study.
Case study is an alternative method for the research strategy. A case study provides detail consideration of a single case of a general object which is under consideration. It provides a deeper understanding of a general phenomenon (P. Johannesson & E. Perjons, n.d.). To better understand reusability in SOA and evaluating the factors affecting reusability in SOA, a case study seems to be applicable for this research. However, accessing the information about implemented projects done in the context of SOA is very difficult and not possible for this research; therefore, survey fits best to be the research strategy.

This research can be performed both qualitatively and quantitatively. A quantitative research, as an alternative, is related to the collection and analysis of data in numeric format. It requires relatively large scale of data, and it's harder and more time consuming to collect this data and analyze it (Hughes, 2006). We choose qualitative research method because collecting and analyzing of data can be performed by as many forms as possible in non-numeric format (Hughes, 2006). It requires a small number of cases, it can be focused on cases which seem to be more interesting and related (Hughes, 2006). This method seems to fit more to this study because the number of instances for this study is limited; therefore quantitative method is not applicable.

As mentioned before the aim of this research is to find factors affecting reusability in companies under investigation. To learn more about these factors and to get closer to answer of the research question, a literature study should be carried out. According to Bhattacherjee (2012), the purpose of literature review is three-fold: (1) to make a survey on the current state of knowledge, (2) to identify significant authors, articles and findings in the context of study, (3) to identify gaps in knowledge in that context of study (Bhattacherjee, 2012). Literature study helps to build up a conceptual framework of SOA and to understand more about possible factors. This also helps to design interview questions. As a result of the literature study, fourteen potential factors are derived and elaborated which could be the ones that we are looking for in companies under investigation. By comparing the factors elicited by literature review and empirical study, we can elaborate the new factors that exist in real business situation and new factors that emerged since 2008 that most of the previous studies on factors affecting reusability are performed.

Johannesson and Perjons have mentioned several methods for data generation and analysis. The most important ones consist of: "interviews, group discussions, questionnaires, observation studies, and document studies" (P. Johannesson & E. Perjons, n.d.). Since qualitative survey is chosen to be the research strategy of this research, interviews or questionnaires are appropriate tools for data generation and analysis. We choose interview method for data generation and analysis because interview is a more personalized method for data collection than questionnaires (Bhattacherjee, 2012). According to Bhattacherjee (2012), interviews can be organized in three forms: face-to-face interview which is the most typical form of interview, focus group where a group of respondents are interviewed and finally telephone interviews which respondents are contacted over phone (Bhattacherjee, 2012). In this study, face-to-face interviews with related experts in the industry are performed. The aim of the interviews is to get more insights about reusability in SOA and to find factors affecting reusability in companies under investigation.

Document study is an alternative method for data generation. In this method, existing documents, user manuals or system specifications are studied (P. Johannesson & E. Perjons, n.d.). Since this research studies factors affecting reusability in SOA, and is neither a case study nor dependent on any product, document study is not applicable alternative method in this research.
Another alternative method is observation study. In an observation study, one observes and investigates the natural environments and real cases (P.Johannesson & E.Perjons, n.d.). In this study, no real case is running currently to be observed; therefore, this method is not appropriate.

There are several methods in qualitative research methods for sampling such as purposive sampling, quota sampling, and snowball sampling. The alternative methods of sampling are purposive sampling and quota sampling. In purposive sampling, a number of participants are collected “according to preselected criteria relevant to a particular research question.” (Mack, et al., 2005) Quota sampling which is sometimes considered as a type of purposive sampling is a method which researcher decides on the number and characteristics of the population. The difference between purposive and quota sampling is that quota sampling is more specific to the size and proportions of samples. Finally, snowball sampling is a method in which, “participants or informants with whom contact has already been made use their social networks to refer the researcher to other people who could potentially participate in or contribute to the study.” (Mack, et al., 2005) Snowball sampling is an appropriate method for this study because in this method participants who are not easily accessible can be accessed and contacted. (Bhattacherjee, 2012)

The ethical consideration of researches has got great importance because, "the researcher’s own ethical responsibility forms the basis of all research ethics" (Swedish Research Council and Uppsala University, 2011). It means that the researcher is responsible for the research to have a good level in the case of quality and morality. Good level is achieved by following some norms. The general norms are altruism, organized skepticism and universality (Swedish Research Council and Uppsala University, 2011). The researcher is obliged to find appropriate literatures to review, and cite them appropriately when it is needed. The researcher must avoid fabricating or falsifying of data. Since interview is one of the methods used in this research, it is important to take into consideration the ethical issues related to human being. The research might not put the participant into risk of harm. They must be informed about the procedures, aim and risks of research (William M.K. Trochim, 2006). The consent must be given voluntarily. The involved persons must have authorities and committees to be consulted (O’Brien, 1998).

In this research, interviewees are informed about the purpose of this study, the audience and publication of this study. They are informed before the interview. Furthermore, we tried to let them know about what kind of information are required and to get their consent before starting the interview. In this research, the author is very careful about privacy of interviewees; however they are asked if they want to take the responsibility of their statements by their real names or company name. Questions are also asked about confidentiality of their statements and to check if they want to publish them.

2.2 Application of Method

As mentioned in previous section, survey is chosen to be research strategy for data generation and analysis. Literature review and interview are used as methods for gathering data. The purpose of literature review in this research is firstly to make an understanding of SOA and reusability in SOA, secondly to derive possible factors affecting reusability. This was done by studying a number of journals, conference publications, textbooks and many other scientific literatures. The keywords for finding these literatures were SOA, services, reuse and synonym words. After providing background knowledge about SOA and reusability in SOA, based on literature study, a list of factors that affect reusability in SOA is created.
In order to make an investigation in companies that have implemented SOA, and to find more factors affecting reusability in SOA, five face to face, semi-structured interviews are organized. Semi-structured interviews are usually a data collection method for qualitative researches; they are designed and scheduled before the interviews. They consist of open-ended questions and questions that are produced from the conversation between interviewer and interviewee (DiCicco-Bloom & Crabtree, 2006). Semi-structured interview is applicable to provide comparable and reliable qualitative data (Cohen & Crabtree, 2006). This kind of interview allows interviewees to express their ideas freely by their own terms (Cohen & Crabtree, 2006). The interviews take 50-70 minutes. They are recorded and transcribed later. The transcriptions of interviews are presented in Appendix C.

As it is mentioned in previous section, snowball sampling is used as a method for finding correspondence for interview. Since this study is done in a context, the best way to find the appropriate companies is to investigate in customers of Dynabyte AB to find those companies that implemented SOA. Regarding the constraints and time limitation of this study, three companies are found to be relevant for interviewing. Hence three consultants associated to these companies are contacted and interviewed. The consultants used their social networks to refer the author to relevant participants in selected companies to interview.

The main goal of these interviews is to find more insight about reusability in SOA in their organizations and to find factors affecting them when reusing. Currently, Dynabyte AB has just three customers that have implemented SOA completely. Therefore, five interviews take place with these three companies consisting of two major telecom companies in Sweden and one electronic selling company in Sweden. Dynabyte AB is a system integration company that provides and implements integration solution to their customers.

For the first major telecom company two interviews are organized consisting of one interview with two senior architects and another interview with a corresponding consultant. The first interviewee is an information architect and team leader of information architect who has two years experience in that company. The second interviewee is an integration architect and team manager, who has 6 years experience in that company, and the third interviewee, is a consultant and information architect who has 8 years experience at Dynabyte and is currently working for the leading telecom companies.

In the second telecom company two interviews with an architect and a consultant are organized. The first interviewee in this company is a senior integration architect in broadband services with 15 years experience and the second interviewee is a Dynabyte consultant who has 18 years experience and currently is working in the major telecom company.

For the electronics selling company one interview is organized with corresponding Dynabyte consultant who has two years experience, because the consultant has the highest information about SOA projects implemented in this company.

All the selected correspondences have at least two years experience in system integration and information architect and currently they are dealing with SOA projected in related companies. The information about the interviews is presented in table 1.
<table>
<thead>
<tr>
<th>Interview</th>
<th>Interviewee</th>
<th>Name</th>
<th>Position</th>
<th>Company</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Architect1 &amp; Architect2</td>
<td>Maria &amp; Sousa</td>
<td>Information Architect and team leader of information architects / Integration Architect</td>
<td>Leading Telecommunication Company</td>
</tr>
<tr>
<td>2</td>
<td>Consultant 2</td>
<td>Sam</td>
<td>Dynabyte Consultant</td>
<td>Major Telecom Company</td>
</tr>
<tr>
<td>3</td>
<td>Architect 3</td>
<td>Javier</td>
<td>Senior Integration Architect</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Consultant 1</td>
<td>Johan</td>
<td>Dynabyte Consultant</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Consultant 3</td>
<td>Jesper</td>
<td>Dynabyte Consultant</td>
<td>Electronics Selling Company</td>
</tr>
</tbody>
</table>

*Table 1: Interviewees Information*

The interview questions have three parts. The first part is introductory questions and the second part there are general questions about SOA projects they have implemented and importance of reusability for them. In the third part, questions about factors affecting reusability in SOA are asked. The purpose of these questions is to derive new factors which have emerged during the last four years and are absent in literatures. The list of questions for interview is presented in Appendix A.

Validation of the result often can be performed in form of experiment, interview, questionnaire, survey and etc. (Yin, 2003). However, artifact evaluation researches can be evaluated by review. Using this type of validation, reflection on result and usefulness of evaluation will be gained (Yin, 2003). Therefore a list of questions is created for reviewer to conduct the validation of this thesis. The questions are about validity of result and the thesis. Therefore, the validation is not only about the result of the research, comments about all part of the research are considered. The questions are sent to correspondent by email as a convenient way of communication. Since this research is performed in a system integration company, the correspondent is selected from the consultants of this company which is familiar with concepts of SOA and reusability of SOA. Since this correspondent is from the same company, he is familiar with the factors therefore, the validation is done precisely. The questions and result of validation can be found in appendix D.

In order to analyze the result, cross-case analysis method is used. According to Bhattacherjee (2012) in such analysis, similar concepts and patterns between different cases are discovered and relationships are described (Bhattacherjee, 2012). In this method, the research can choose categories and find similarities and differences within or between several categories (Bhattacherjee, 2012). In this research factors derived from literature study and factors derived from empirical study are compared in order to find similarities and differences. In addition, the factors are categorized according to companies in order to analyze them and find similarities and differences across the companies.

According to Guba (1981), soundness and trustworthiness in qualitative research can be evaluated by four constructs, consisting of, credibility, transferability, dependability and confirmability (Guba, 1981). Therefore these constructs are chosen to be analyzed rather than analyzing reproducibility, reliability and validity of research method. Because the approach of this study is qualitative method, henceforth, analyzing credibility, transferability, dependability and confirmability is an appropriate alternative for analyzing research method.

"In addressing credibility, investigators attempt to demonstrate that a true picture of the phenomenon under scrutiny is being presented. To allow transferability, they provide sufficient detail of the context of the fieldwork for a reader to be able to decide whether the prevailing environment is similar to another situation with which he or she is familiar and whether the findings can justifiably be applied to
the other setting. The meeting of the dependability criterion is difficult in qualitative work, although researchers should at least strive to enable a future investigator to repeat the study. Finally, to achieve confirmability, researchers must take steps to demonstrate that findings emerge from the data and not their own predispositions.” (Shenton, 2004, p. 63)

The credibility of a research can be evaluated by considering if a true picture of the phenomenon is captured (Shenton, 2004). The survey of this master thesis is performed in a consultant company, where experts from three different companies participated in the survey. Therefore, the data is collected from perspectives of three different companies in order to capture a comprehensive understanding of the phenomenon under study.

According to Meyers (2009), the transferability construct is adhered by providing a thorough description of the context of the research (Myers, 2009). Dynabyte AB is chosen as context of this master thesis; because, it is a system integration company which provide integration solution for many large companies. Therefore, performing the research in this context makes it easier to access the clients that have implemented SOA to organize the interviews.

The dependently of this thesis work is adhered by providing an in depth description of the research process and procedures of performing the survey to strengthen the dependability criteria in order to make it easier to another researcher to perform the same study (Shenton, 2004). It is expected that, the same research will most probably lead to the same result as this research (Myers, 2009).

The confirmability criterion of this master thesis is dealt because, all the literatures reviewed are selected from databases that provide articles that are published in famous journals and are recognized by the university. Moreover, it is tried to find articles which are not obsolete and are more up to date. Five interviews are organized, and all the interviewees have at least 2 years experience in SOA implementation in their own companies. Three of the interviewees are senior consultants, which are expert in system integration and SOA and currently are working with a SOA projects. The rest of three interviewees are selected by snowball sampling method. Using this method ensures that the selected interviewees in three different companies are appropriate persons for this research work. Furthermore, the companies under investigation are counted as large companies in Sweden with multiple branches all over the world therefore; companies have good experience of implementing SOA in each branch. In addition, the result is validated by performing an expert review. Reviewing is done by a senior consultant who has experience about SOA and is not interviewed. The thesis and list of question for validation is forwarded to the expert by email. He answers the questions and verifies the validity of result and thesis. Therefore, the risk of affection of authors own personal opinions is minimized in this research.
3 Results

3.1 Factors Affecting Reusability Derived from Literature Study

The list of derived factors is presented in table 2. These factors are derived from literature study according to the following definition of reusability:

Reusability of services is defined as degree in which a service can be used in more than one application or business process with minimum overhead to discover, configure and invoke the service (Choi & Kim, 2008).

In this section, more sources are searched for each factor in order to verify the validity of factors. Some factors are broadened to cover closely related factors from different articles. Some factors are common among different studies, and some of them are produced of combination of closely related factors from different studies.

<table>
<thead>
<tr>
<th>No.</th>
<th>Factor</th>
<th>Explanation</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Using consistent business terms to describe services</td>
<td>Using consistent business terms of existing services in order to match with required services</td>
<td>(Dan, et al., 2008)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Vegter, 2009)</td>
</tr>
<tr>
<td>2</td>
<td>Managing new service creation and discovering of existing services</td>
<td>To have a catalog of existing services enabled to reuse and map it to the plan of future implementation</td>
<td>(Dan, et al., 2008)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Vegter, 2009)</td>
</tr>
<tr>
<td>3</td>
<td>Providing assurance to both service users and owners</td>
<td>Managing service entitlement to provide assurance about the quality of service and control over the services by users</td>
<td>(Dan, et al., 2008)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Vegter, 2009)</td>
</tr>
<tr>
<td>4</td>
<td>Managing service enhancements and changes</td>
<td>Taking into account further enhancement and changes of services.</td>
<td>(Dan, et al., 2008)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Street &amp; Gomaa, 2008)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Vegter, 2009)</td>
</tr>
<tr>
<td>5</td>
<td>Using standard superior interfaces</td>
<td>Using standard interfaces provides flexibility for modifications in implementations over the time</td>
<td>(Narayanan, 2009)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Street &amp; Gomaa, 2008)</td>
</tr>
<tr>
<td>6</td>
<td>Business commonality</td>
<td>How common it is to consumers in a domain to use functionality of services.</td>
<td>(Choi &amp; Kim, 2008)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Branca &amp; Atzori, 2011)</td>
</tr>
<tr>
<td>7</td>
<td>Modularity</td>
<td>To have independent functionality and not having any dependency on other services.</td>
<td>(Choi &amp; Kim, 2008)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(Branca &amp; Atzori, 2011)</td>
</tr>
<tr>
<td>8</td>
<td>Adaptability</td>
<td>Degree in which a service can be</td>
<td>(Choi &amp; Kim, 2008)</td>
</tr>
</tbody>
</table>
adapted to different service users. (Branca & Atzori, 2011)

| 9 | Standard conformance | Degree in which a service is conformed to the industry standards such as OASIS WS-standards | (Narayanan, 2009) (Choi & Kim, 2008) (Branca & Atzori, 2011) |
| 10 | Discoverability | Degree in which a service is found easily and correctly by service consumers. | (Choi & Kim, 2008) (Street & Gomaa, 2008) (Branca & Atzori, 2011) |
| 11 | Separation of processes | Separation of Process logic and Business logic to ensure that business services are not dependent on a specific process. | (Street & Gomaa, 2008) (Newcomer & Lomow, 2005) (Dunn & Knight, 1993) |
| 12 | Message exchange with multiple patterns | to support multiple processes by services, it is important that they exchange massages with multiple patterns | (Street & Gomaa, 2008) (Newcomer & Lomow, 2005) |
| 13 | Stateless services | Data is stored on the client side and it is sent to the server on each request. It increases load balancing, reliability and scalability of services | (Erl, 2005) (Street & Gomaa, 2008) |
| 14 | Cost of reuse vs. benefits of reuse | cost of reusing services must be less than benefits of reusing service | (Fichman & Kemerer, 2001) |

Table 2: Factors derived from literature review

### 3.1.1 Using Consistent Business Terms to Describe Services

It is very difficult to reuse existing services, if the business items of existing service do not match with required services. Business terms consist of name, structure, textual description of business entities that they support, and their semantics. Using consisting business terms enables data integrity and using common language and semantics in applications and data sources. In order to manage the use of consistent business terms, organizations must control creation of new business terms or make changes to the current terms to match with new requirements. Moreover, organizations should control development team to use specified business terms in related services. SOA can help implementation of this factor by creating a repository of business terms, named as business glossary, and enabling tools for managing, navigating, defining and exporting terms to other service tools. (Dan, et al., 2008)

### 3.1.2 Managing New Service Creation and Discovering of Existing Services

In order to reuse services and application components, "we need to understand the various roles in the solution lifecycle who work with services as artifacts, in accessing, modifying, and publishing the services." (Dan, et al., 2008, p26) This requires some pre requisites to support the roles in the reuse solution process.
Establishing a central point of management is the first point. It defines the required tools, repository and communication policies for development of services. SOA Competency Center is responsible for managing central point. This is "a point of business, development and architectural coherence for SOA in the organization, supporting the various stakeholders." (Dan, et al., 2008, p26) This tool enables each role to be supported with related tools and have access to the central registry and repository. In other words, this managing function provides technical and business SOA guidelines for teams to understand architectural guidelines for managing the design and implementation of services. It is important in an organization to have a catalog of existing services enabled to reuse and map it to the plan of future implementation. (Dan, et al., 2008)

3.1.3 Providing Assurance to Both Service Users and Owners

Another issue in reusing services, which are developed by another owner, is the lack of assurance about the quality of service and lack of any control over the services by users. Therefore, to accommodate reusability, it is very important to manage service entitlement, because a high workload on a specific business processes may effect on all other business process that share this service. Service entitlement can be defined as for example, who can use the service, what is expected quality of service, and maintenance issues such as, limiting time for development of a new version or modifying infrastructure, protocols for notifying users about unexpected errors. Providing assurance on service can be done in many ways such as establishing contract. Service level agreement is well known concept for managing the terms of agreement. Users are intended to reuse services if assurance about the service is provided. (Dan, et al., 2008)

3.1.4 Managing Service Enhancements and Changes

When a service is first developed, it is not possible to consider all future requirements and changes of business processes. Therefore, further enhancement and changes of services are always required to improve the reusability (Dan, et al., 2008). Taking into account the variability is useful; because, with a single design, future functionality or user changes will be covered. This factor assures that the designed service is suitable for future new functionalities (Street & Gomaa, 2008). Without service enhancement, the service will not have the possibility to be reused in the future. On the other hand, it is very important to do changes not only for development time or investment considerations, but also for minimizing the impact on users and business processes. Therefore, there should be a methodology for managing the enhancement according to requirements of each user and business process tools for scoring these requirements. The methodology should be supported by: tools for managing requirements for each user, tools for collecting related data and scoring new functionalities, tools for testing new requirements and tools for managing the service changes and versions in registry. (Dan, et al., 2008)

3.1.5 Using Standard Superior Interfaces

To reuse services, they must have the flexibility to be modified in implementation over the time. Therefore, using standard interfaces provides flexibility for modifications in implementations over the time (Narayanan, 2009). Advanced interfaces with detailed implementation and functionality make services possible to be reused by several applications (Gomaa, 2000).

3.1.6 Business Commonality

Business Commonality defines how common is to consumers in a domain to use functionality of a services. The reason for defining this criterion is that, the functionalities provided by services must be
common for service users. For a service to be reused widely, it is necessary to provide widely applicable functionalities rather than just specific functionalities. A service can provide many advanced functionalities, but it may not be used by many consumers if those functionalities are not common for consumers. (Choi & Kim, 2008)

3.1.7 Modularity
Modularity in services means that the services must have independent functionality and do not have any dependency on other services. A reusable service has modularized functionality. Modularity is enabled by high cohesion and a loosely coupled dependency. This factor accommodated reusability because, concerns about service interfaces and implement can be separated by implementing this factor (Choi & Kim, 2008). If a service has high dependency on other services, reusing the service requires reusing of related services. This increases the complexity and increases the cost of integration. (Choi & Kim, 2008)

3.1.8 Adaptability
Adaptability means the degree in which a service can be adapted to different service users. Service requirements are different from service consumer point of view; therefore, adaptability can be done through variation point binding. This factor increases the reusability because, in order to reuse a service in SOA, a service must be adaptable to various service context and requirements. (Choi & Kim, 2008)

3.1.9 Standard Conformance
Standard conformance factor means the degree in which a service is conformed to the industry standards for example, OASIS WS-standards. The reason for this factor is that, in order to reuse a service in industry, the service should conform to the industry standard to be composed easily by service consumers. If a service does not conform to standards, the probability of being reused by consumers decreases. In other words, one can say that a service with higher standard conformance has the higher probability of reusability. (Choi & Kim, 2008) It is very important to validate the service interfaces by using tools such as WSDL analyzer in order to verify that the services are usable (Narayanan, 2009).

3.1.10 Discoverability
Discoverability factor means the degree in which a service is found easily and correctly by service consumers. In order to find an appropriate service, service consumer compares the description of required service with existing services. Therefore, for a service to be discovered effectively, the specification of services must be specified in an understandable manner and can be easily and correctly understood with service consumers. Henceforth, for a service to be discovered easily, the service should expose its necessity to its environment and this makes service available to be reused. A highly discoverable service enables higher reusability of services. (Choi & Kim, 2008) This factor also helps developers to locate correct services quickly and make reusability easily (Street & Gomaa, 2008).

3.1.11 Separation of Processes
There are two kinds of logics for software architectural: Process logic and Business logic. Process logic is the order of levels which are processed and Business logic is functionality of business. Separating these two processes is an architectural strategy for reusing services and creating
applications because, in this case business services are not dependent on a specific process (Newcomer & Lomow, 2005). Process flow can act as coordination of services which controls usage of business services. This minimizes the coupling between services which is important for reusability (Dunn & Knight, 1993). Business Process Execution Language (BPEL) can implement process separation (Street & Gomaa, 2008).

### 3.1.12 Message Exchange with Multiple Patterns

As mention above separation of processes enables reuse of business processes in different applications. However, in order to support multiple processes by services, it is important that they exchange massages with multiple patterns (Newcomer & Lomow, 2005). The pattern of message exchange must be fully defined in the service interfaces description (Street & Gomaa, 2008).

### 3.1.13 Stateless Services

Stateless services mean that when no information about previous requests is stored on server, data is stored on the client side, and it is sent to the server on each request. Therefore, the requests of a client can be responded with many servers. This improves reusability because services will not be specified to any specific process or request; therefore, service can be reused by other requests and processes too. Stateless increases load balancing, scalability and reliability because messages can be distributed between servers and if one server stops, the others can handle the requests (Erl, 2005). As a result, reusability is improved because of load balancing, reliability and scalability characteristics of stateless services (Street & Gomaa, 2008).

### 3.1.14 Cost of Reuse vs. Benefits of Reuse

The significance of this factor appears when cost of reusing services is more than benefits of reusing service. In this case, developing new services may cost less; therefore, reusing services is not beneficial. Fichman and Kemerer have mentioned several factors for cost of reuse: Domain and cost analysis cost, locating and using reusable services cost, cost of integration of services, service modification cost, maintenance and testing cost, obtaining reusable service cost (Fichman & Kemerer, 2001). If all these costs are higher than benefits gained by reuse and cost of developing new services, then new service should be developed instead.

### 3.2 Derived Factors from the Empirical Study

The following section presents the results of interviews with tree different companies that have implemented SOA projects. In this section, factors affecting reusability produced and derived from interviews are presented.

The summary of the result is presented in Table3. In this table, an overview of factors which have been found during interviews are illustrated. These factors are derived from the interviews with various roles in organizations. Some of the factors are mentioned with more than one interviewee. So for each derived factor related interviewees are determined in table 3.
<table>
<thead>
<tr>
<th>No.</th>
<th>Factor</th>
<th>Interviewee</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Documentation</td>
<td>[Architect1] [Architect2] [Consultant2]</td>
</tr>
<tr>
<td>2</td>
<td>Having generic information model</td>
<td>[Architect1] [Architect2]</td>
</tr>
<tr>
<td>3</td>
<td>Having common information model</td>
<td>[Consultant2]</td>
</tr>
<tr>
<td>4</td>
<td>Business process and information model adherence</td>
<td>[Consultant1] [Architect1 and Architect2]</td>
</tr>
<tr>
<td>5</td>
<td>Governance of services</td>
<td>[Consultant2]</td>
</tr>
<tr>
<td>6</td>
<td>Having catalog of services</td>
<td>[Architect3]</td>
</tr>
<tr>
<td>7</td>
<td>Having a library of services</td>
<td>[Architect1] [Architect2]</td>
</tr>
<tr>
<td>8</td>
<td>Having map of all services</td>
<td>[Consultant3]</td>
</tr>
<tr>
<td>9</td>
<td>Services according to standard compliance</td>
<td>[Consultant1] [Architect3]</td>
</tr>
<tr>
<td>10</td>
<td>Using standard technology</td>
<td>[Consultant2] [Consultant3]</td>
</tr>
<tr>
<td>11</td>
<td>Managing service changes over time</td>
<td>[Consultant1] [Architect3]</td>
</tr>
<tr>
<td>12</td>
<td>Making services as general as possible</td>
<td>[Architect3]</td>
</tr>
<tr>
<td>13</td>
<td>Using common message model</td>
<td>[Architect3]</td>
</tr>
<tr>
<td>14</td>
<td>Using open and well known API</td>
<td>[Consultant2]</td>
</tr>
<tr>
<td>15</td>
<td>Using standard industry terminology</td>
<td>[Architect1] [Architect2]</td>
</tr>
<tr>
<td>16</td>
<td>Having rules and guidelines for creating services</td>
<td>[Architect3] [Consultant3]</td>
</tr>
<tr>
<td>17</td>
<td>Political factors</td>
<td>[Architect1 and Architect2]</td>
</tr>
</tbody>
</table>

Table 3: Factors derived from empirical study

### 3.2.1 Documentation

Documentation is a factor which is mentioned to be important by two interviewees. Architect1 and Architect2 mention documentation as an important factor which makes reusability easier and lack of documentation of services and legacy systems has been counted to be biggest challenge for them. As they mention:
"We need service to be documented, we cannot reuse services which are not documented, and most of the legacy systems have no documentation, so it is very hard to reuse something cannot be found and is not documented well enough, that is probably the biggest challenge we have." [Architect1 and Architect2]

In addition, Consultant2 also states that the documentation around services is very helpful. Consultant2 mentions that the documentation must be in UDDI registry for being managed and accessible by everybody. According to interviewee:

"Documentation around the service helps a lot, when you have a UDDI registry, you should write about the service, document the service, documentation is always important for reusability." [Consultant2]

### 3.2.2 Having generic information model

Service based on a generic information model is another factor which affects reusability. Two of the interviewees agree on this factor and believe that this increases reusability of services. Architect1 and Architect2 agree that services which are based on an agreed and generic information model and structure have higher degree of reusability because they exist on their own and are not influenced by other systems. This makes it possible for services to connect to other systems where they are required. Architect1 and Architect2 agree on:

"I think usually you will succeed better reusing of services if you base your services on an agreed and generic enough information model and structure. You should not be too influenced by systems around it, you should try to have services that exist on their own and make it possible to connect to different and several systems if they have the same functionality requirement based on functional need and not based on system API.” [Architect1 and Architect2]

### 3.2.3 Having common information model

Consultant2 argues that having a common information model for all the services across the organization such as information model which is based on XML, is an important aspect for improving reusability.

"For example the information model should be based on xml and also you should try to use the same information model for all the services that used schema, the same entities, and the same composite and so on, because that is another important aspect. If you have a common information model that you can distribute among the company than you will have a common denominator when it comes to information so everybody know what we are talking about every time.” [Consultant2]

### 3.2.4 Business process and information model adherence

Services which are adhered to business process and information model are more reusable. This factor is mentioned by three interviewees. Consultant1 believes that if services are not based on information model and business processes, we cannot achieve reusability because to know how services work, we must know how they are related to business process and business processes are relied on information systems:

“I would like to have the factor of the business process and information model adherence as a very important factor […], if you do not take this factor deeply into account when you
produce the services you do not get a lot of reusability, you need to know which problem
you solve with services. And if you are going to uphold the catalog of how services work
you need to relate them to how you produce your value that is the business process, and
business process will rely heavily on information models, so you need to have those clearly
stated.”[Consultant1]

Architect1 and Architect2 state that services which are in new areas should be modeled in generic way
and adhered to business processes:

“First of all base service on the information model if it is a new area where we have not
defined the information structure then we try to model it in a generic way and adhered to
business processes.”[Architect1 and Architect2]

3.2.5 Governance of services

Service governance and management is another factor that is mentioned by one interviewee.
Consultant2 mentions this factor to be important for reusability because having information about
current services helps to reuse them. Managing services can be implemented by having UDDI
repositories, to keep services and information about services. Consultant2 believes that organizations
may have dedicated people for managing all the services, otherwise people tend to produce their own
services and do not reuse existing services:

“You need to put the service in a for instance repository registry, so it can be looked by the
others who want to reuse that service. That’s when you use an UDDI repository to keep
services, that is very important and also to have people dedicated to work with service to
manage all the services. The governance part is very important for reusability, because if
you do not have any dedicated person in this area then people usually tend to create their
own services.” [Consultant2]

3.2.6 Having catalog of services

Having a catalog for keep services information is a factor mentioned by Architect3. He counters lack of
a catalog for keeping data about services as a problem in his organization. He believes having this
catalog helps to find which services exist in the organization and what they do:

“Something that we are lacking here and trying to have is that we do not exactly have a
specific catalog for the service to find which service we have today, what they are doing.”
[Architect3]

3.2.7 Having a library of services

Architect1 and Architect2 have library of reusable services in their organization to manage the
services and enhance reusability of services. They agree on:

“We have a tool for management of services; we have a library of reusable external
components and services that we import it in to our systems.” [Architect1].

3.2.8 Having map of all services

Consultant3 also believes that somewhere in the organization, information about services should be
written. This makes reusability easier. In Electronic Selling Company they have a big map of all
services and also information about coupled services which makes changes of services easier:
"Somewhere in the company you have to write down or have all the services that are using. […] then development is easier and it is also easier to reuse. […] In this company we have a big map of plan that all the services are available. And we can see all the systems that are coupled to each other, then you can get an overview and also if you want to change something it is easy to see what happens.” [Consultant3]

**3.2.9 Services according to standard compliance**

Making services according to standard compliance is also mentioned to be important factor by two interviewees. They state that services which are built according to standards are more reusable, because the service will be applicable for larger and broader systems. Building the services specific for a particular system need, reduces the reusability of system. Consultant1 states that a service which is designed for a specific system is not a service; it is more like an application. He continues that the reusable services should be built to standard compliance, this standard can be external standards:

“Services built to standard compliance, are often well reusable, services developed for particular system needs have less reusability. Standard can be an external standard. If you build your services according to standards, they become reusable, but if you build the service to just fulfill a specific system need than it is not a service it's just application.” [Consultant1]

Architect3 finds services that are according to standards more reusable. He believes that if services can be changed to conform to standards, they become more reusable:

“The service must be according to standards; […] I would say that if you are able to change the service as expresses in a standard way they are more reusable. We try to express services in standard way.” [Architect3]

**3.2.10 Using standard technology**

Using standard technology is factor which accommodates reusability according to two interviewees. Consultant2 also mentions that to have reusable services, standard technologies must be used to provide a broader availability of services for different systems. He states JMS and web services as two standards which services can be built on them:

“To have a really reusable service, […] you should use technology that is standard, usually in integration to make integration or service available for broader stand on different systems you should use either JMS or web service, both of them are standards. Going proprietary is not a good idea when it comes to reusability.” [Consultant2]

Consultant3 discusses standards and standard technology as a good factor for reusability. He exemplifies the standards with JMS, EMS, FTP, HTTP and SOAP. Consultant3 discuses that using these factors makes the reusability easier because it provides possibility to communicate with new systems:

"It is good to use techniques that are standards and simple, like using JMS and EMS standards. We also use FTP transfer and we are also using JDBC but it is down to data base and in the web based integration this company is using HTTP and SOAP. When it is standards it is easy to communicate with new systems. "[Consultant3]
3.2.11 Managing service changes over time

When designing services it is important to take into consideration the future business requirements and future services changes and enhancements. A service which is more changeable during time can cover long term needs and therefore is more reusable. This factor is mentioned by two interviewees. They considered services which are more changeable over time more reusable. Managing services changes must be taken into account at the development time. Consultant1 states that if a service cannot be changed, less degree of reusability is achieved and future business needs cannot be fulfilled by service. Therefore, handling differences over time in content of service is very important. To manage changes over time there should be a plan to how to evolve the service:

“When developing services it shall not take into account only what is done today by systems but also what future business process expect from that system and of course that cannot be done 100% correct first time, that’s also should be taken into account how we are going to change or evolve the service over time, because if you make service, you make it general and available, and try to reuse it but you can’t change it, than you make very less degree of reusability, so be able to handle differences over the time in the service content it is very important, you need to have a plan to how to evolve the service, therefore the service should be changeable.” [Consultant1]

Moreover, Architect3 mentions that to have higher degree of reusability services must be updateable and changeable in a life cycle because new need appear. Looking ahead and estimating future needs makes possibility for services to cover long term needs.

“You are able to update a service but you do not want to make large changes because new needs appear, you are able to update a service in a life cycle and the service should be changeable in order to adapt to future needs. But when you create a service from the beginning it is good to look ahead and see what can we have with that service to be able to make it as wide as possible to cover that long term needs.”[Architect3]

3.2.12 Making services as general as possible

If a service is general, it can be adapted by more businesses. Therefore service becomes more reusable. Architect3 mentions this factor several times. He states that services which are general are easier to be reused. General services mean those that are not product or business specific. If services are general and have general or natural interfaces, changes to the service will not affect the consumers of services.

"We make the services as general as possible. It makes it easier to be reusable […] If you are able to create a service as general as possible, changes should not affect that much consumers and providers. Because changes in a service mean that the systems providing functionality or information behind that service are changed some way. So the only thing is how you can keep the interface as natural or general as possible to not affect the other that are consumers of the service[…] With general services I mean i.e. no product or business line specific. If you are able to handle for example "Manage Workforce" that is not connected to a specific product, is general and therefore more reusable."[Architect3]

3.2.13 Using common message model

Using common message model in services is important factor that might influence reusability. Architect3 states that they express services in a common message model, support more messages and increase reusability of services. They use OAGIS as the common message model in their company.
"To use the common message model to express that service […] we try to express them in a common message model way, because services with common message models can support more message models and can be reused easier. We use OAGIS as our Common Message Model." [Architect3]

### 3.2.14 Using open and well known API

Services with open and well known Application Programming Interface are more reusable. Consultant2 mentions this factor that has effect on reusability of services because services with standard API are easier to be accessed and developed by developers:

"If it is an open and well know API, it is more reusable because it is easier for new developers to access it and develop it, because there are usually lots of researches for instance to create SOA services." [Consultant2]

### 3.2.15 Using standard industry terminology

Using standard business and industry terms is another factor mentioned by Architect1 and Architect2. The terminology used for expressing industry terms should be consistent and standard across the organization to make reusability easier. Architect1 and Architect2 state that they have created their own standards in their company:

"You must be agnostic in you terminology and trying to take industry standard terminology for your services if you can. […] There aren’t that many standards and we create our own standards" [Architect1 and Architect2]

### 3.2.16 Having rules and guidelines for creating services

Having rules and guidelines is important factor affecting reusability, rules about creation, expression, offering and documentation of services. According to Architect3, to make reusability easier there should be a common way of looking at services. Therefore some rules should be created to decide from the beginning on how to create, express, offer and document the services. These rules can be written as internal guidelines for organizations. Architect3 states:

"It is a little bit difficult to create reusability if you do not have a common way of looking at these services that is why it is good to be able to create a rule and decide from the beginning. Rules for how the services are going to be created, expressed, offered, described and documented. […] Rules mean the guidelines inside the company primarily." [Architect3]

In addition, Consultant3 states that having guidelines is important. Guidelines that describe the integration and preferences:

"Having some guidelines description for the integration in the company for example how you do the web services in the company. Having guideline that they can write how to use SOAP for example in the company and what they have for preferences." [Consultant3]

### 3.2.17 Political factors

According to Architect1 and Architect2 political factor is important for reusability. Services have their own policy therefore; one should be allowed to reuse the services:
"The political factor is important factor influences reusability. You have to be allowed to reuse a service. This needs to be budgeting in the projects for it."[Architect1 and Architect2]
4 Analysis

4.1 Combining Derived Factors from Empirical Study

In order to derive more consistent and reliable factors from the empirical study, in this section similar detailed factors are combined to constitute more reliable and consistent factors where it is possible. Some of the factors mentioned by correspondents have similar semantics; therefore, they can be merged to produce a factor which is more reliable, consistent and can cover all the corresponding sub factors. The factors are categorized in such a way that, factors with similar semantics are considered to be one factor for ease of management and analysis of factors. The combination is done through the definition of reusability in SOA (Choi & Kim, 2008). The new factors are mentioned in following:

4.1.1 Managing and Governing Services

Service management and governance means to keep metadata about services to make it possible to manage and access them easily. In order to manage and govern services, organizations need a central point of management that everyone can access it to find the appropriate service and also can reuse the existing services. As it is represented in figure 5, this factor can cover four factors derived from empirical studies.

![Managing and governing services](image)

According to Consultant2, governance of service means that organization must assign people to manage all the services and to use a UDDI registry to keep the services. According to Architect3, having a catalog of services is important to find which services exist in organization and what they do. Architect1 and Architect 2 mention having a library of services as a tool for managing services. Finally, according to Consultant3, having a map of all services and information about coupled services makes an overview of services and; therefore, reusability and development of services becomes easier.

All the above mentioned factors relate to governing and management of services. As a result, they are combined and constitute a more general service as Managing and governing of services.

4.1.2 Standard Conformance

Standard conformance factor means the degree in which a service is conformed to the industry standards and technology standards. The reason for this factor is that, in order to reuse a service in industry, the service should conform to the industry standards in order to be composed easily by
service consumers. Standard services can be applicable for broader systems and businesses. As it can be seen from the figure 6, this factor can cover two factors mentioned by correspondents:

![Diagram showing Services according to standard compliance and Using standard technology](image)

**Figure 6: Standard conformance**

According to Consultant1 and Architect3, standard compliance means that, building services not specific for particular system needs. Furthermore, according to Consultant2 and Consultant3, using standard technologies like web services, JMS, EMS and HTTP makes the services communicate easier with other systems. Henceforth, services will be more available to other services.

Both of these factors are related to conforming of services to standards. So, they are composed to produce a more generic factor which is Standard conformance.

After combining corresponding factors to more general factors, the number of factors derived from the empirical study decreases to 13 factors as can be seen in table 4. The newly produced factors from combining factors are highlighted in this table.

<table>
<thead>
<tr>
<th>No.</th>
<th>Factor</th>
<th>Interviewee</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Documentation</td>
<td>[Architect1]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Architect2]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Consultant2]</td>
</tr>
<tr>
<td>2</td>
<td>Having generic information model</td>
<td>[Architect1]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Architect2]</td>
</tr>
<tr>
<td>3</td>
<td>Having common information model</td>
<td>[Consultant2]</td>
</tr>
<tr>
<td>4</td>
<td>Business process and information model adherence</td>
<td>[Consultant1]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Architect1]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Architect2]</td>
</tr>
<tr>
<td>5</td>
<td>Managing and governing services</td>
<td>[Consultant2]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Architect3]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Architect1]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Architect2]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Consultant3]</td>
</tr>
<tr>
<td>6</td>
<td>Standard conformance</td>
<td>[Consultant1]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Architect3]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Consultant2]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Consultant3]</td>
</tr>
<tr>
<td>7</td>
<td>Managing service changes over time</td>
<td>[Consultant1]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Architect3]</td>
</tr>
<tr>
<td>9</td>
<td>Making services as general as possible</td>
<td>[Architect3]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Using common message model</td>
<td>[Architect3]</td>
</tr>
<tr>
<td>10</td>
<td>Using open and well known API</td>
<td>[Consultant2]</td>
</tr>
<tr>
<td>11</td>
<td>Using standard industry terminology</td>
<td>[Architect1]  [Architect2]</td>
</tr>
<tr>
<td>12</td>
<td>Having rules and guidelines for creating services</td>
<td>[Architect3]  [Consultant3]</td>
</tr>
<tr>
<td>13</td>
<td>Political factors</td>
<td>[Architect1]  [Architect2]</td>
</tr>
</tbody>
</table>

**Table 4: Factors derived from interviews after combining**

### 4.2 Mapping Factors from Empirical Study to Literature Study

In order to make a factor analysis, we compare the factors found by literature studying to factors found by empirical studying. There are many factors that overlap with each other. Discovering these overlapping factors helps to find most consistent and reliable factors and, this leads to produce a list of reliable factors. In figure 7, the connections of factors derived from the empirical study with factors derived from literature study are presented.
<table>
<thead>
<tr>
<th>Factors from the empirical study</th>
<th>Factors from literature study</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using standard industry terminology</td>
<td>Using consistent business terms to describe services</td>
</tr>
<tr>
<td>Managing and governing services</td>
<td>Managing new service creation and discovering of existing services</td>
</tr>
<tr>
<td>Having rules and guidelines for creating services.</td>
<td>Providing assurance to both service users and owners</td>
</tr>
<tr>
<td>Managing service changes over time</td>
<td>Managing service enhancement and changes</td>
</tr>
<tr>
<td>Using open and well known API</td>
<td>Using standard superior interfaces</td>
</tr>
<tr>
<td>Standard conformance</td>
<td>Business commonality</td>
</tr>
<tr>
<td>The political factors</td>
<td>Modularity</td>
</tr>
<tr>
<td>Making services as general as possible</td>
<td>Adaptability</td>
</tr>
<tr>
<td>Using common message model</td>
<td>Standard conformance</td>
</tr>
<tr>
<td>Documentation</td>
<td>Discoverability</td>
</tr>
<tr>
<td>Having generic information model</td>
<td>Separation of processes</td>
</tr>
<tr>
<td>Having common information model</td>
<td>Message exchange with multiple patterns</td>
</tr>
<tr>
<td>Business process and information model adherence</td>
<td>Stateless services</td>
</tr>
<tr>
<td></td>
<td>Cost of reuse vs. benefits of reuse</td>
</tr>
</tbody>
</table>

**Figure 7: Mapping factors from empirical findings to theoretical findings**

In this figure, in the left column, the list of factors derived from the empirical study is presented and in the right column, the list of factors found by literature review is shown. The arrow lines between these two columns show the connection of these two lists. Where a factor from the empirical study is semantically similar to the factor from the literature review, an arrow line maps these similar factors. In the following, mapping of these factors is explained in details.

### 4.2.1 Using Consistent Business Terms to Describe Services

**Figure 8: Using consistent business terms to describe services**
According to Architect1 and Architect2, the terminology used for expressing industry terms must be consistent and standard across the company. This factor semantically is the same as factor number 1 found by literature review. In this factor, using consistent business terms is emphasized to have data integrity and common language and semantics in applications (Dan, et al., 2008). Therefore, as it can be seen in figure 8, this factor is considered to be the same factor as factor number 1 derived from the literature review. Using standard terminology is mentioned by two architects in Leading Telecommunication Company. Den et al (2008) have also derived this factor in his research; therefore, this adds to the significance of this factor.

4.2.2 Managing New Service Creation and Discovering of Existing Services

![Figure 9: Managing new service creation and discovering of existing services](image)

Managing and governing services is a factor which is mentioned by five correspondences. According to correspondences, organizations should have tools such as UDDI registry, catalog of services, library of services and map of all services to be able to manage and govern the services in an organization. The factor: having rules and guidelines for creating services, is mentioned by Consultant3 and Architect3. They mentioned that, it is important to have rules for creation, expression, offering, documentation, and integration of services. Managing new service creation and discovering of existing services is a factor which is derived from the literature review. This factor means that companies should establish a central point of management to manage services; moreover, they should have SOA guidelines for teams to understand architectural guidelines for managing the design and implementation of services (Dan, et al., 2008). As a result, factor number 2 of literature review can support both factors from the empirical study, as it can be seen in figure 9. Consequently, both factors can be combined to produce a new factor which has overlap with the factors from the literature review. This is a both managerial and technical factor, and it is mentioned by all three Architects and two of Consultants which means that it has been an issue for all of them. This factor proves that the management around the services has high significance for reusability of services in an organization. Companies must know what kind of services exists in their company, and they should manage them in order to prevent recreation of existing services and to maximize the reusability. They must also have a plan and rules for creation of future services in order to manage future creations.

4.2.3 Managing Service Enhancements and Changes

![Figure 10: Managing service enhancements and changes](image)

Managing service changes over time is factor mentioned by Consultant1 and Architect3. They state that services must be changeable and updatable in a life cycle to cover long term needs of business. However, managing service enhancements and changes is factor found by literature review. Further enhancement and changes of services are always required to improve the reusability (Dan, et al.,
2008), because with a single design, future functionality or user changes will be covered. This factor guarantees that the designed service is suitable for future new functionalities (Street & Gomaa, 2008). Finally, it is concluded that these two factors are related to each other, and they can be considered as same factor.

### 4.2.4 Standard Conformance

![Standard conformance](image)

Standard conformance is a factor which is mentioned by four correspondents and three researchers. According to interviewees, services which are built according to standards are more reusable, because the service will be applicable for larger and broader systems. JMS and web services are examples of standards that services can be built on them. Three studies have also mentioned this factor in their researches (Narayanan, 2009, Choi & Kim, 2008, Branca & Atzori, 2011). According to Choi & Kim (2008), services that are conformed to industry standards are more reusable. As a result, one can say that standard conformance factor, mentioned by interviewees is the same factor derived from the literature review. However, this factor is mentioned by correspondents from all companies which means that it has affected all of them in reusability of services. This factor shows that, developing services according to standards increases the reusability of services. So when developing new services it is important to take into accounts the conformance to standard of services.

### 4.2.5 Political Factors

The political factor in mentioned by two interviewees in leading Telecommunication Company. This factor does not have anything with technical perspective of services; this is more of business perspective of factors. This factor proves that, in order to be able to reuse a service, one must be allowed to use that. Permission to reuse a service is an important criterion.

### 4.2.6 Making Services as General as Possible

According to Architect3, General services mean those that are not product or business specific. If services are general and have general or natural interfaces, changes to the service will not affect the consumers of services. This factor is important for major Telecom Company, and it was not mentioned in the literature review.

### 4.2.7 Using Common Message Model

Having common message model is a factor mentioned by Architect3. If a service has common message model, it can support more messages. This is a factor which is important in Major Telecom Company. They try to use messages with a common model.

### 4.2.8 Documentation

Documentation is a factor which is mentioned by three correspondents from Leading Telecommunication Company. Lack of documentation has been the biggest challenge for them. Since this factor is mentioned by all the correspondences of a telecommunication company, one can say that documentation factor is an important factor for a telecommunication company.
4.2.9 Having Generic Information Model

There are factors mentioned by correspondences regarding information model of business processes. Architect1 and Architect2 agree on this factor and mentioned that having generic information model is important, and increases reusability because in this case, services are not influenced by other systems. This is experience of two architects from a leading telecommunication company.

4.2.10 Having Common Information Model

Another factor regarding the information model of business processes is to have a common interface. This is also mentioned by Consultant2 of leading Telecommunication Company. He argues that if companies have the same information model, they can distribute the information model across the company.

4.2.11 Business Process and Information Model Adherence

This is a factor which is mentioned by three correspondences from both telecommunication companies. Consultant1 believe that to understand how services work, it should be understood that, how services are related to the information model. Architect1 and Architect2 agree that services, which are in new areas, should be modeled in a generic way and adhered to business processes. This factor is also related to the information model of business processes which affects reusability. As a result, one can say that information model issues and dependencies of services on the information model are very important criteria in telecommunication companies. Because, all the interviewees of telecommunication companies have mentioned factors related to the information model which affects reusability of services. Therefore, in a SOA solution, information model has a significant impact on reusability of services in telecommunication companies.

Finally, after mapping and combining related factors from the empirical study to factors from the literature review, a list of factors with 12 factors is created. In table 5, the list is presented.
<table>
<thead>
<tr>
<th>No.</th>
<th>Factor</th>
<th>Interviewee</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Documentation</td>
<td>[Architect1] [Architect2] [Consultant2]</td>
</tr>
<tr>
<td>2</td>
<td>Having generic information model</td>
<td>[Architect1] [Architect2]</td>
</tr>
<tr>
<td>3</td>
<td>Having common information model</td>
<td>[Consultant2]</td>
</tr>
<tr>
<td>4</td>
<td>Business process and information model adherence</td>
<td>[Consultant1] [Architect1] [Architect2]</td>
</tr>
<tr>
<td>5</td>
<td>Managing new service creation and discovering of existing services</td>
<td>[Consultant2] [Architect3] [Architect1] [Architect2] [Consultant3]</td>
</tr>
<tr>
<td>6</td>
<td>Standard conformance</td>
<td>[Consultant1] [Architect3] [Consultant2] [Consultant3]</td>
</tr>
<tr>
<td>7</td>
<td>Managing service enhancements and changes</td>
<td>[Consultant1] [Architect3]</td>
</tr>
<tr>
<td>9</td>
<td>Making services as general as possible</td>
<td>[Architect3]</td>
</tr>
<tr>
<td>9</td>
<td>Using common message model</td>
<td>[Architect3]</td>
</tr>
<tr>
<td>10</td>
<td>Using standard superior interfaces</td>
<td>[Consultant2]</td>
</tr>
<tr>
<td>11</td>
<td>Using consistent business terms to describe services</td>
<td>[Architect1] [Architect2]</td>
</tr>
<tr>
<td>12</td>
<td>Political factors</td>
<td>[Architect1] [Architect2]</td>
</tr>
</tbody>
</table>

Table 5: List of empirically driven factors after mapping with factors derived from literature study.

In the above table, the factors are denominated with synonym names from literature study, to have a consistent name for the rest of the study.

As it can be seen from table 5, out of 12 factors elicited by the empirical study, five factors are common with factors found by literature study, and they form the basis of factors affecting reusability. Factors such as *managing new service creation and discovering of existing services* and *standard conformance* are emphasized by many practitioners today and many researchers four years ago. However, they are not sufficient factors for today’s implementation of SOA. As a result, seven new factors are emerged in the companies under investigation. Most of the factors derived from literature study are from studies performed in 2008; whereas, this research is performed four years later. Accordingly, some new factors emerged during four years. The reason for appearing new factors is that firstly, SOA has been more matured in companies recently, and more companies are migrating to
SOA since 2004. Secondly, the new factors are appeared in practice and in the phase of implementation of SOA; therefore, they might have been hidden from theoretical researchers.

However, the factors elicited by empirical study can be classified from different perspectives. As it turns out from the factors driven from the empirical study, six of the factors are emphasized on adjectives such as standard, general, common and synonym words. These factors consist of:

- Having generic information model
- Having common information model
- Making services as general as possible
- Using common message model
- Standard conformance
- Using standard superior interfaces

Standardization applies to the technical aspect of services such as interfaces and technology. Generalization applies to the information model and services. Finally, commonality applies to the information model and massage model. It can be understood that in these companies reusability is accommodated by achieving standardization, generalization and commonality. These three issues construct six factors that affect reusability in these companies. In conclusion, one can say that to improve reusability of services in companies no unique way of implementing technology, information model and message model is recommended. The importance of these factors for these companies in Sweden is that because, these companies have branches over the world so, uniqueness of services prohibits them to reuse them in other branches. Therefore, standardization, generalization and commonality are important for them.

Information model is an issue which plays an important role in factors affecting reusability. Three of driven factors are about dealing with the information model and its effect on reusability. These factors are:

- Having generic information model
- Having common information model
- Business process and information model adherence

It turns out that information model is a vital part of services and dependency of services to the information model is an important issue in reusability of services in these companies. Generalization and commonality of the information model is a criteria which plays an important role in reusability in one hand, and on the other hand, services must be adhered to the information model and business process.

It is worth mentioning that, the factors related to the information model are absent in the theoretically driven factors. This could be because theoretically driven factors are more of technical perspective. However, in real life business oriented factors such as information model issues are important in reusability.

The analysis of this study is implemented based on the factors by combining derived factors and mapping them to the factors derived from literature study. The limitation is that the application of factors in different type of industries is not deeply discussed. Therefore, another alternative for discussing the result of this study could be discussing the factors based on type of companies that have mentioned the factors. Another limitation of this discussion is that importance of each factor, in regard to number of correspondences and authors mentioned the factor, is not deeply discussed.
5 Discussion

As mentioned earlier in chapter 2, a literature review is performed, and some factors are derived by literature review in order to learn more about the possible factors affecting reusability in SOA. These factors are listed in table 2. In addition, an empirically study is carried out to find applicable factors in three companies under investigation. The empirically driven factors are listed in table 3. In chapter 4, the relations between the theoretical findings and the empirical findings are discussed.

The findings show some divergences between the literature study and the empirical study. This is because, the literature emphasizes on reusability in established standard definition while, empirical study results in more practical needs based on business support. In other words, earlier studies have focused on factors affecting reusability from the theoretical perspective. However, this study focuses on reusability from both perspectives of the literature and practitioners in business. Henceforth, divergence between this study and previous studies may appear. Besides that, earlier studies are performed in 2008, and during four years, SOA is more matured in business than before. Therefore, it seems that some new factors appear during four years which are different from factors found in 2008.

The empirically driven factors can be classified into two perspectives including technical perspective and business perspective. Technical perspective refers to those factors which have a direct relation to the information system technology and its boundaries in SOA. On the other hand, business perspective refers to the business oriented issues of SOA. Moreover, the interviews of this study are conducted with people with both technical and business backgrounds. It is worth mentioning that, the classification of factors is performed according to the knowledge and understanding of the author. The business oriented factors are:

- Having generic information systems
- Having common information model
- Business process and information model adherence
- Making services as general as possible
- Using common message model
- Using consistent business terms to describe services

The technical factors are:

- Documentation
- Managing new service creation and discovering of existing services
- Standard conformance
- Managing service enhancements and changes
- Using standards superior interfaces

Out of five technical factors, four are common with factors found by literature review. This may show that technical factors are more well-known factors than business factors. In other words, technical factors, which are found in this investigation, were mentioned in literatures which are reviewed; therefore, they may be considered as well-known and permanent factors. Moreover, as it can be seen from table 5, technical factors are mentioned by bigger number of correspondences in comparing to number of correspondences for business oriented factors. This is another reason that proves technical factors are widely-known.
On the other hand, business oriented factors are almost new and not mentioned by any literature. The reason for appearing these factors is that the companies under investigation are highly business oriented companies. Therefore, the new factors found in this survey are mostly business oriented factors. As it can be seen from table 6, two major telecom companies in Sweden are those which have focused on business oriented factors as new factors. Another reason for appearing these factors is that firstly, earlier works have not performed an empirical study to derive factors. Secondly, SOA is more matured today in compare to four years ago; therefore, it turns out that the earlier works have focused on the technical factors.

As it can be seen from the factors, the reason for implementation of SOA and reusability in SOA in these companies is mostly because of business perspective advantages of SOA rather than technical perspective; because, the majority of the factors are business oriented and most of the technical factors are almost well-known factors which interviewees were expected to mention them.

The previous knowledge of the author about system integration and SOA impacts on the research by influencing on formulating the research question, designing interview questions and conducting the interviews. In addition, the knowledge of the author helps to elicit more factors from interviews transcriptions. The knowledge of the author could also have a negative influence in factors derived by the interviews in a way that, the derivation of factors that are elicited from interviews might have influenced by previous knowledge of the author, which might not be the same as what the interviewees exactly meant. (Myers, 2009)

The ethical and social impact of this study consists of two perspectives: organizational perspective and individual perspective. From the organizational point of view, this study derives the factors to improve implementation of SOA. This research aims at providing value for businesses by reusing services and application components. Therefore, it provides a win-win situation for the organizations by saving the cost of developing new services and applications. Hence, it is valuable for society and provides the expected benefit requested from research participant.

From the individual perspective, Dynabyte AB was volunteered in the research, provided the research environment and organized the interviewees. The participants on interviewees were informed about the procedure and consequences of research. Therefore, the ethical and social impact of study is fulfilled by defining primary intention of this research and considering consent of Dynabyte AB and the participants of this study.

5.1 Conclusion

Since reusability is one of most emphasized principals of service oriented architecture, this master thesis focuses on primary empirical research, seeking to find factors affecting reusability in SOA, reflecting the current state of three companies in Sweden. The research has been performed by studying a number of literatures regarding reusability in SOA, identifying 14 factors affecting reusability in SOA and then through an empirical study in three companies 12 factors are elicited. Furthermore, through a cross-case analysis factors derived from the literature review are compared to factors derived from interviews to find the similarities and differences of perceptions of factors affecting reusability in SOA. By comparing factors from perspectives of literature review and practitioners, this research can support effective reusability of services for practitioners in SOA implementation by deriving possible factors in literature and in real business situations.

From 12 factors derived from the empirical study, fives are common with factors derived from literature study. Seven new factors are formulated based on the practitioner’s contributions, meaning
that academic factors constitute just around 40 percent of factors found in business. The rest of factors, which are found in this research, are new factors based on experience of practitioners. The new factors are mostly business oriented factors which have emerged in the companies since 2008.

As a result, it is advised that company’s effective reusability of services in SOA will most often be accompanied by balancing factors that pull together management of services, issues related to information model, standardization and generalization factors. An effective reuse of services yields a good return of investment, agility and rapid implementation of SOA.

This master thesis differs from similar works (Dan, et al., 2008, Street & Gomaa, 2008, Choi & Kim, 2008) firstly, by performing an empirical investigation in three different companies in Sweden to find the factors affecting reusability in real business situation. Secondly, this study finds the new factors appeared since 2008, when earlier works have been performed on reusability, and reflects the differences appeared in factors affecting reusability during last four years.

This study can be concluded by summarizing its research contribution as follows:

Firstly, this research is undertaken as one of the first piece of work aiming to identify factors affecting reusability in SOA by empirical investigation in three companies in Sweden. Although some studies have been performed on factors affecting reusability in SOA, this thesis identifies and defines a set of factors in a business context to find the differences from earlier works, and to answer the research question of what factors affect reusability in SOA implementation in companies under investigation? This research question is answered by identifying and defining a set of factors and comparing them to factors from previous theoretical studies.

Secondly, this thesis contributes by capturing differences and similarities between perception of factors from literature on one side and from investigation in companies on the other side. As a result, practitioners will have a sense of which factors are relatively more important than others, and they will be guided toward more efficient reusability of services and application components when they implement SOA.

The conclusion of this master thesis does not have any large ethical impact on society. Despite, it is useful for businesses by providing agility, return of investment and rapid implementation of SOA.

5.2 Limitations and Future Research

The limitation of this research is that the findings of this research is limited to three companies in Sweden and cannot be generalized to all companies with different business objectives in other countries. In other words, the factors found and analyzed in this study are based on ideas and experience of experts and consultants working for these companies. Therefore, it might not cover all the organizations that have reusability in SOA adoption. Moreover, all the factors in companies under investigation are not elicited. Because, it is not possible to organize interviews with all experts of the companies to find all the possible factors; meanwhile, the interviewees may have not expressed all the possible factors during interviews.

As a future research, since there have been limitations on a number of companies and people under investigation, this research have found factors affecting reusability in SOA qualitatively, it can be feasible as a future research to perform this research quantitatively by involving a larger number of companies and people.

Furthermore, it would be interesting to undertake a research in the form of quantitative study, to find out how generalizable the factors are. Organizations adapted SOA can be asked to rank a selection of
factors including the ones acknowledged in this research, and also the factors identified in academic literatures.
6 Bibliography


Vegter, W., 2009. Critical success factors for a SOA implementation. s.l., 11th Twente Student Conference on IT, Enschede, June 29th,2009 Copyright 2009, University of Twente, Faculty of Electrical Engineering, Mathematics and Computer Science..


7 Appendices

7.1 Appendix A: Interview´s Questions

Introduction Questions:

1. Company name:
2. The company which you have done SOA project.
3. Position at the company:
4. Duration of employment:
5. Do you want this interview to be treated confidentially?
6. Can you provide some information about your company?
7. Can you provide some information about SOA projects implemented in your company?
8. Why did you decide to adapt SOA in your organization?

General Questions:

9. What kind of benefits did you expect from SOA adaption?
10. To what degree reusability was important for you?
11. To what extend your SOA implementation was successful in terms of reusability?
12. To what extend did you consider the capability of legacy systems and services to be reused in SOA solution?
13. How do you measure the ability of applications to be reused?

Factor Related Question:

14. What challenges did you face when you wanted reuse legacy system and services?
15. What key features of services do you consider to be important in reusability? Or what kinds of services are more reusable?
16. What should be taken into consideration in developing new services to be reused later?
17. What do you think about benefits of reusing services? Cost of reuse vs. developing new services
18. What significant changes can be applied to services to make them reusable?
19. What other factors do you think affect reusability in SOA?
20. How important do you find these following factors (Appendix B)??
### 7.2 Appendix B: Table of factors driven from literature review

<table>
<thead>
<tr>
<th>No.</th>
<th>Factor</th>
<th>Explanation</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Using consistent business terms to describe services</td>
<td>Using consistent business terms of existing services in order to match with required services</td>
<td>(Dan, et al., 2008) (Vegter, 2009)</td>
</tr>
<tr>
<td>2</td>
<td>Managing new service creation and discovering of existing services</td>
<td>To have a catalog of existing services enabled to reuse and map it to the plan of future implementation</td>
<td>(Dan, et al., 2008) (Vegter, 2009)</td>
</tr>
<tr>
<td>3</td>
<td>Providing assurance to both service users and owners</td>
<td>Managing service entitlement to provide assurance about the quality of service and control over the services by users</td>
<td>(Dan, et al., 2008) (Vegter, 2009)</td>
</tr>
<tr>
<td>4</td>
<td>Managing service enhancements and changes</td>
<td>Taking into account further enhancement and changes of services.</td>
<td>(Dan, et al., 2008) (Street &amp; Gomaa, 2008) (Vegter, 2009)</td>
</tr>
<tr>
<td>5</td>
<td>Using standard superior interfaces</td>
<td>Using standard interfaces provides flexibility for modifications in implementations over the time</td>
<td>(Narayanan, 2009) (Street &amp; Gomaa, 2008)</td>
</tr>
<tr>
<td>6</td>
<td>Business Commonality</td>
<td>How common is to consumers in domain to use functionality of services.</td>
<td>(Choi &amp; Kim, 2008) (Branca &amp; Atzori, 2011)</td>
</tr>
<tr>
<td>7</td>
<td>Modularity</td>
<td>To have independent functionality and do not have any dependency on other services.</td>
<td>(Choi &amp; Kim, 2008) (Branca &amp; Atzori, 2011)</td>
</tr>
<tr>
<td>8</td>
<td>Adaptability</td>
<td>Degree in which a service can be adapted to different service users.</td>
<td>(Choi &amp; Kim, 2008) (Branca &amp; Atzori, 2011)</td>
</tr>
<tr>
<td>9</td>
<td>Standard conformance</td>
<td>Degree in which a service is conformed to the industry standards such as OASIS WS-standards</td>
<td>(Narayanan, 2009) (Choi &amp; Kim, 2008) (Branca &amp; Atzori, 2011)</td>
</tr>
<tr>
<td>10</td>
<td>Discoverability</td>
<td>Degree in which a service is found easily and correctly by service consumers.</td>
<td>(Choi &amp; Kim, 2008) (Street &amp; Gomaa, 2008) (Branca &amp; Atzori, 2011)</td>
</tr>
<tr>
<td>11</td>
<td>Separation of processes</td>
<td>Separation of Process logic and Business logic to ensure that business services are not dependent on a specific process.</td>
<td>(Street &amp; Gomaa, 2008) (Newcomer &amp; Lomow, 2005) (Dunn &amp; Knight, 1993)</td>
</tr>
<tr>
<td>12</td>
<td>Message Exchange with Multiple Patterns</td>
<td>to support multiple processes by services, it is important that they exchange massages with multiple patterns</td>
<td>(Street &amp; Gomaa, 2008) (Newcomer &amp; Lomow, 2005)</td>
</tr>
<tr>
<td>13</td>
<td>Stateless Services</td>
<td>Data is stored on client side and it is sent to server on each request. It increases load balancing, reliability and scalability of services</td>
<td>(Erl, 2005) (Street &amp; Gomaa, 2008)</td>
</tr>
<tr>
<td>14</td>
<td>Cost of reuse vs. benefits of reuse</td>
<td>cost of reusing services must be less than benefits of reusing service</td>
<td>(Fichman &amp; Kemerer, 2001)</td>
</tr>
</tbody>
</table>
7.3 Appendix C: Interviews

7.3.1 Interview 1

Introduction Questions:

1. **Company name:** Leading Telecommunication Company.

2. **Position at the company:** Information Architects and team leader of information architects/Integration Architects and team manager

3. **Duration of employment:** 2 years / 6 Years

4. **Do you want this interview to be treated confidentially?** Yes

5. **Can you provide some information about your company?**

   It’s a telecom company that resides in 12 or 13 countries around in large Europe. In these countries we have wide verity of portfolio in just countries just prepaid mobile telephony while other countries have both paid mobile telephony and prepaid mobile telephony. Broadband in different kinds, TV offers and business products they normally classes as alternative telecom operator.

6. **Can you provide some information about SOA projects implemented in your company?**

   This company is renewing its application portfolio in the major areas. We are installing Sibo for example and bringing new systems and our new architecture is based on using SOA services to join all the systems together and keep them decoupled so, through that, project which is now going on for almost 3 years we involved in large number of reuse services they are used to roll out this project into different countries and increasing scope of the project and adding new products to it and services support different counties, different products.

7. **Why did you decide to adapt SOA in your organization?**

   I have been working in integration for ten years the first few years from about 2001 to 2005 SOA didn’t even exist. We did enterprise application integration as it was called at the time. For so far without having very large management role of new products, a lot of integration initiatives tend to be a little bottom up, you might say, even this company role of SOA services and we see that by using SOA, it gives us a way to get a step back if our goal here is reuse of services you want to be able to get step back to the project we working in, we do not want to deliver always with the projects scope even if that is actually our corporate way of working and through the development and concept of SOA services we are able to broaden the scope if we do at least a little bit and obtain a lot more about what we use and more worth to the company.

General Questions:

8. **What kind of benefits did you expect from SOA adaption?**

   Simple one is of course reduces cost by reusing existing generic services, fast deployment more flexible integrations and integrations that are not system dependent, this SOA services as we work with them are structured in a generic language or generic description and generic structure that does not necessarily show the exact APIs and or their formats of the underlying systems and then also of course, reducing technical complexity for some systems that have difficulties in talking JMS of whatever and of course always in the bottom you have improved stability the integration platform that we run this SOA service and provides an extra level of stability and security for integration platform.

9. **To what degree reusability was important for you?**
Maybe 90 to 80 percent. Reusability was the main focus.

10. To what extend your SOA implementation was successful in terms of reusability?

My own personal estimate is about 75%. One of in four services is new today and rests are reuse. It means that for every 4 services we implement 1 of them is newly developed that the rests are services which are just reused.

11. To what extend did you consider the capability of legacy systems and services to be reused in SOA solution?

Sometimes people go out and they do an inventory of existing services of the company to identify which of those are candidates of being SOA services, but we did not do that. We had no integration platform before at all, so we do have some native services that we use that maybe legacy systems has an interface and we can reuse their interface, we couple them almost explosively to this generation SOA services we have.

12. How do you measure the ability of applications to be reused?

It’s hard to say, it depends on business case, I have no statistics on it. We have the policy that if a legacy system has existing interface, especially in our box interface that we should reuse that and map it to SOA service. But how much they have had to adjust to services I am not sure. I see other way around, in how many instances have those systems are connecting to us implemented our services because they had nothing on their own choice implemented in their interface and now we have very few, we try not to do that, we would like for systems to connect to us to only on interfaces as well. So we have never measured it, since it is totally new in this company.

We have excel spreadsheet where we have the integrations and we define them to what reuse level they comply. You have KPI for reuse up to we reuse a service, So I have a service called manage incident and I am able to reuse manage incident, but if I reuse manage incident version 1 and I also reuse manage incident version 2 in production, for us version of that service doesn’t matter, they are built on each other, major investment has been done on the first one. So we want to reuse based on service name not service name and version so that is one issue than you have reuse of integration, I have two systems connected to service and I can maybe start putting new products on it new information to it, So I have new usage of that without having to rebuild because my service is so flexible and that is a level of reuse, and the third level of reuse which means I have the provider in a service and I am going to add more consumers then I am using half of the integration, so we can have four or five level of reuse which you can actually measure and all of them have different cost price tag attached.

Factor Related Question:

13. What challenges did you face when you wanted reuse legacy system and services?

We have to get them to except integration platform that is a challenge, we used to always get them integrated directly to other system, they are not used to talking to the integration center, and in this case the organizational change can be hard to affect. We need service to be documented, we cannot reuse services which are not documented, and most of the legacy systems have no documentation, so it is very hard to reuse something cannot be found and is not documented well enough, that is probably the biggest challenge we have.

14. What key features of services do you consider to be important in reusability? Or what kinds of services are more reusable?
I think usually you will succeed better reusing of services if you base your services on an agreed and
generic enough information model and structure. You should not be too influenced by systems around
it, you should try to have services that exist on their own and make it possible to connect to different
and several systems if they have the same functionality requirement based on functional need and not
system API, something we grantee is that having information model behind that is very important. The
service as you design as a service not as an API generally is more reusable, API is application
interfaces which we usually see it a lot in online areas looks a lot like a database. So if you have or
you cannot reuse our interfaces project specific scope also consider if you put naming in your service
of your API. We have two providers of the service to systems that are travel ticket and the other is
incident ticket. And if you create your API with the work incident ticket in it the system that causes
travel ticket will not be happy when you talk with them because that is their interface are not mine, so
you must be agnostic in you terminology and trying to take industry standard terminology for your
services if you can. So you can stand the politics of the different back end systems. There aren’t that
many standards and we create our own standards, we have the standards how the XML should look
like and what kind of types within the standards are allowed, information model that you build on.

15. What should be taken into consideration in developing new services to be reused later?

First of all base it on the information model if it is a new area where we have not defined the
information structure then we try to model it in a generic way and adhered to business processes. Then
we go to build of the XML structure of the service, there we got very good things you have reusable
XML objects that you use across all services they also use the value list, we have some tools, we have
a tool for management of services, we have a library of reusable external components and services that
we import it in to our systems. We reuse our service, we reuse our integration. but if I sit down and
talk about the reuse there is different level to it, there is reuse within xml schemas of objects, so if I
modeled address one, then I know this is a good way to model the address then I create an address
XML schema and that is then imported in every service that needs to define address so we have tones
of those that we reuse and build the services upon. what is fun with that is we do it cross domain so
normally in normal organizations without SOA we have CRM system and people responsible for
CRM systems build APIs and integrate to other systems and their address looks one way and then you
have the people for delivering installation addresses and service addresses and they model addresses
another way when you need to exchange information between the two addresses and we impose the
same structure address across domains which gives the company a common language, so trust the
information model is very important. The information model is our bases for canonical massages and
canonical massages are our standard for creating SOA services.

16. What do you think about benefits of reusing services? Cost of reuse vs. developing new services

Maybe the quick short term financial benefits to reusing a service properly, but there should be a long
term quality performance lose when you have to make changes to in. we have a tendency and I have
five services and I do not want to place a new order to my vendors and I want to reuse all these five
services for everything and you get of this famous value per services which is just one big value per
and a document that describes what it does, those are not beneficial. It is very troublesome in support
scenarios when something goes wrong when a system we feel that we do not have right system
information on the system or the system did not accept the attempt to integrate and how do you try to
find the source for problem if it solves any. It is all actually in memory on it to way through it is not
even substantiated in a format it is in a big value list, you can easily correlate even the field name with
the value so it is very flexible form legacy system perspective that gets lots of reuse of interfaces, you
have placed few orders but for integration point you try to make it effective. So we try to have clear,
concise precise services that need to represent clear functionality, so they are not just be called such customers, such agreements, such address and then you have to be careful with optional mandatory parameters, we keep the mandatory parameters to look to a common core which is a key factor for reusability. You cannot have many mandatory fields you have to cover up for the optional, it can be one provider have the optional the other provider has the mandatory but it has to be optional in the interface and then we take care with the native adapter level. So there are very few elements which are mandatory. Because when we come around a project at the first time or we want to do a service or integration for the first time you have that project requirements in your head ok it seems clear that we want to have this mandatory and this mandatory, so we are adding business data rules to the service, then we come around the second time to reuse that service it could be within the same integration but from other functionality or between other systems then we will realize this use case really does not fit that this fields are mandatory, they need to be optional, that is what we have learnt, the hard way. To sum up, clear services of what business functionality are they solving, clear and precise data structures or information structure within it and that also means proper and good type and element names very few mandatory elements and loads of key values. Strategic use of service values to increase the service life.

17. **What significant changes can be applied to services to make them reusable?**

   This question is covered in previous questions

18. **What other factors do you think affect reusability in SOA?**

   The political factors. You have to be allowed to reuse a service. This needs to be budgeting in the projects for it. It can be challenge to get people to discuss proper scope for new services, they are very project focused. We want to transfer blue pen between the system where as we say well we need to have service that transfers all pens of all kinds and then they put in it that they do not have any knowledge on any kind of pen because that will increase the scope. we are very tight to our platform of course we have a web method based platforms and it is not like we can pick up sharp and move to another platform so, and since it should be decided to not use web methods anymore and then of course we are very tight we have to rebuild something but we even done it so our service definitions and our xml structures can be moved to other platforms.

19. **How important do you find these following factors (Appendix B)?**

   We covered factor number 1, we covered factor number two but for our systems not the other systems. And we have a repository to li those services. For factor number 3 we do not have that policy controls in that sense. We do this, but we do it through SLA policy and follow up terms that you used. Factor number 4 we try very hard to do it, it is not easy, but we do try. For factor number 5 we can say that in this company we have quite a big various of business services to different consumers, so it is very common to have one provider of services and several consumers so all that like online service channels can be several different kind of sale channels and we use the same web service towards all of them. So that would be business commonality. We do have modularity of services to high degree. We have Adaptability of services. (Standard conformance)We have our own and we use that standard. We follow web services standard but the rest we follow our standards. We do not allow our services to be discoverable. All of our integration is done within our intergradations center as a team. And since we are trying to use native services as much as possible, in order to have a discoverable service, then we have an implication there that the system calling it has implemented major in the interface. We do not use discoverable services but information model every system connected to us must be a named system which they have their own entry point. We do separation of processes. We exchange messages using multiple patterns, we expose the same service, we have the same SOA service, and they can be
exposed as stored procedure, JMS, FTP. Statelessness is one of our main policies is ICC that integration platform does not maintain state. Cost of reuse vs. benefits of reuse. Money vs. SOA never wins. It depends on how to finance the first user of the service, and the service creation, then you have the reuse, you are diminishing the reuse cost but is still tends to play roll around 60%. You still have to make workshops, you still have to go out and verify the service, most of our hours go not for development, it goes for test verification, test service development or in this case workshops to verify service usage on agree to what to integrate and all those costs happen if it is reuse, so the actual development cost is the minor. In real life there are so much projects, politics, department politic, things that mess up and all of those add to cost are the costs related to SOA service, at the end how do you calculate the cost, you calculate the cost for eight department which also includes those bad factors which can really affect, or do you calculate what we have just spent on development which could be more true cost calculation.

Sit support is very expensive also, sit support is that if I have new application going into production someone is going to test it in the test environment for the production and someone has to support them in the test environment and answer questions, going into logs and make corrections. These make costs and are part of reuse.

7.3.2 Interview 2

Introduction Questions:
1. Company name: Dynabyte
2. The company which you have done SOA project: A Leading Telecommunication Company.
3. Position at the company: Consultant Information architect
4. Duration of employment: 8 years
5. Do you want this interview to be treated confidentially? Yes
6. Can you provide some information about your company?
   This company is a telecom company that sells a lot of different services; the main focus area is obviously mobile telephone services, since that was the core business when they started. But they also have regular telephony, they also sell broadband services as an ISP and I think that they sell content as television. They are moving toward content providers.

General Questions
7. Can you provide some information about SOA projects implemented in your company?
   It was service instance for all the different projects going on in this company, so there were different projects and I as an information architect that was responsible to ensure that all the services that we made had well known API within an information model that everybody could understand. So it would be reusable and easy to implement.

8. Why did you decide to adapt SOA in your organization?
   When I did my work in that company that was also SOA maturity in that company so they had a service oriented view on how to do service implementation and how to leverage the SOA to get the least time for development to reuse. So they already had in place from before.

General Questions:
9. What kind of benefits did you expect from SOA adaption?
What you gain is that you have reusability and you have a well-known API, that you can govern which is also very important in SOA governance and if you develop a service it is very easy to add to it and therefore you have decreased development time and faster in market. So you could publish your service faster.

10. To what degree reusability was important for you?

For the information model we tried to have everything reusable, when it came to information model we built up entities that were quite small and then added to them to have composite and then too service message, by doing this, all the services as I said had well know API and they could be reused. Everything we did we tried to make them not complex to be understandable and 100% reusable. But usually you cannot make everything 100% reusable. You should always aim for 100% reusable.

11. To what extend your SOA implementation was successful in terms of reusability?

For this company I heard that from the beginning it was not that much services reuse but when the company became more SOA matured, when they had this SOA thinking, thing started to get better so, there are always services that you cannot reuse from another services but those services that I was responsible for were like 50% reusable from previous services. It was not that much high, there were of course other services which were higher. This was because they had used an old information model that was kind of replicated, so we had to do lot of refactoring and things like that.

12. To what extend did you consider the capability of legacy systems and services to be reused in SOA solution?

We did not have any legacy systems. In my other project I had legacy systems which had to wrap in a service. In this company we had systems that were quite new and they were still adding functionality. It was not legacy systems. When I talk about legacy systems it means systems that are running for a long time that you do not do anything other than maintenance on. All the services in this company were web services that they used. They also had wrappers in front of the services to be able to adapt more easily, if there were changes in requirements on either systems so you could take care for changes without changing the actual KPI every time. But the basic way for invoking the service was SOAP web service.

13. How do you measure the ability of applications to be reused?

We did not do any measurement about; there were other people that had KPI responsibility to see how good we were at implementing and how well we could reuse things. The whole departments they had different responsibilities. There were some people who actually measured how much was reused and how fast we could do that. Because they obviously also have requirements that they should decrease the development time for service in the development area and the only way to be able to decrease those time was actually focusing on reusability, faster and cheaper. we were a whole team that worked with the services and they already had an information model, they had already adapt the rules, how web services should look like and how it should be invoked, so the framework was already there, so I was just one of the workers. So, I wasn’t responsible for this company’s SOA maturity.

Factor Related Question:

14. What challenges did you face when you wanted reuse legacy system and services?

The challenges when you get change request for services obviously always what other systems will be affected, how can we make minimum impact on the systems and as I said before there are different ways you can do that in this company they used wrappers. So, wrap a service to be able to not change the actual service that often since it is a lot of planning needs to be done but most SOA systems you
can have 2 or 3 versions of the same service available at the same time but as I said you also have to have plan to make system go for the new versions of the service, so you do not have to take care of many services because it gets a bit complicated if you have many variants of the same services, so you have to only have 2 or maximum 3 version of same service.

15. What key features of services do you consider to be important in reusability? Or what kinds of services are more reusable?

To have a really reusable service, the couple of things that you really must adhere to are: first one is that you should use technology that is standard, usually in integration to make integration or service available for broader stand on different systems is to use either JMS or web service, both of them are standards. Going proprietary is not a good idea when it comes to reusability. That is the first main thing, you should have an open standard like web service or JMS and of course it is the content, the information that you want to put in or get from the service, should also be an open standard, then XML is most widely used one which is good, because when you have the schema you have well defined information model that you can validate that, you know that both sides that uses the services complies to it. So for example the information model should be based on xml and also you should try to use the same information model for all the services that used schema, the same entities, and the same composite and so on, because that is another important aspect. If you have a common information model that you can distribute among the company than you will have a common denominator when it comes to information so everybody know what we are talking about every time, they know what an address is and how it should look like for instance.

16. What should be taken into consideration in developing new services to be reused later?

First of all we have to use standard protocols as I said previously, web service or JMS, with XML with a known information model, then of course you need to put the service in for instance repository registry, so it can be looked by the others who want to reuse that service. That’s when you use an UDDI repository to keep services, that is very important and also actually to have people dedicated to work with service to manage all the services the governance part is very important for reusability, because if you do not have any dedicated person in this area then people usually tend to create their own services, so it is always better to have department responsible for handling the governance of service to be able to say this is our repository, these are the services that exist, you want this type of service, this type of service we have, this service is very similar and we can probably make a composite out of it to give you the last 10% you need.

17. What do you think about benefits of reusing services? Cost of reuse vs. developing new services

Not really, usually there are always couple of services that won't be reusable they are too special. Sometimes it is best to just leave them as is and not make them as a service, usually there are some point to point integration that you really do not need to move to SOA architecture because it is just consistent, and it won't be reusable by any other system but I would say for at least 80% of services that you make it would be reusable. It is always important that not reuse where it is not really necessary to make it into a service, it would be easier to have just point to point like RMI. Therefore sometimes it is better to leave a service and develop another one because when you are in phase of looking for a service you should always have that in mind that is this such a generic service that it will be reusable for others because in the beginning it takes a little bit longer to make it SOAP service for instance, there are some special requirements within the information model and everything, so it can take longer to develop as a service than if you would not just make the direct call to database or something like that.
18. What significant changes can be applied to services to make them reusable?

The service in itself is well known and well specified entity and therefore it is reusable by itself and that is the whole idea, when you have a service that conforms to standards than you do not change it, the only time that you change it is when the requirement changes and I said before than you need to go to governance cycle and to see if you should develop a new version of service directly or you should have different version running at the same time depends on what the other systems requirements are. If you have an old function then you can make it into service of course. If there exist service and you want to change the information provided or the way it works than it is just normal change management. You can also make it to composite if you want to value added information for instance, you have the basic information that you receive from this service but if you create another sub service and make it into composite it will develop more information and perhaps it is more beneficial, but I mean usually that is looked on case by case, usually if you can leave the service as it is, it is the best. If you wrap it inside another service to change so that mean this is easiest way trying to wrap the service into wrapper and add things, because you never know, if you do not have 100% control of the system, than if you change this service, it can affect unknown systems. Therefore usually better to do wrapper.

19. What other factors do you think affect reusability in SOA?

If it is an open and well know API, it is more reusable because it is easier for new developers to access it and develop it, because there are usually lots of research for instance to create SOA service. The information should be based on XML as I said before. Documentation around the service helps a lot, when you have a UDDI registry, you should write about the service, document the service, documentation is always important for reusability, supporting organizations is also very important for reusability you should have a single point of contact that the whole company knows where to go to fetch information about service, because if such an organization does not exist people tend to know what already exist and develop new, so there is also an organization matter in SOA and it is very important, they usually call it center of excellent that handles all the SOA matters, so documentation and organizations apart from technical aspects are very important for SOA.

20. How important do you find these following factors (Appendix B)?

Factors number 1 and 2 are those which we talked about. Providing insurance is very important that I didn’t mention, that is to gain the trust from the users is very important, because if they do not trust you that you will provide them with adequate information they won't use you. It is always the matter of trust. If people do not trust you they won't work with you and things will not be reused. You should have center of excellent, you should have an organization around SOA and having that organization will provide assurance for service users and owners. Managing service enhancement is the governance as I talked about earlier it is that usually comes later. They usually have SOA maturity and I would say managing service and enhancement changes ranks very high. I do not understand the business communality factor. For me this means that there are certain things in the business that is common for all departs in the company, say for instance customers, what is customer? So customer is common business object, product is the common business objects. All those services that are used for business objects throughout the company should always be made into a service. Search for a customer, search for a product. Things like that. Standard definition is very important. That object has all the property attributes so that everybody can use that service mean, not all products at builds are important for sales, for manufacturing then it is the same product but they need to know different things about the product. For sales prices are more important than the materials in the product that is more important for manufacturing for example. For delivery it is more important to know the specification when it comes to weight and how big things are, so that business object must be able to take care of all
different business units. Modularity is when you think of a service and then you think of composite a service is a self can be quite small maybe only couple of functions. We can then use them as small modules and build better services more complex services around it. So it is important to have modularity. Adaptability is of governance and how you should actually want to manage your service. It does not always make sense to change the service per say if there is new requirement you may have to wrap it because it will disturb the functionality of other systems. I think it is more of governance part and also part that I said that the business objects that you provide the information about should have considered the different uses within an organization. Standard conformance is one of the top ones. You should always keep standards because it will be easier to maintain and to access from new systems. Discoverability is UDDI when it comes to SOAP services. But you may have the services which are not based on SOAP services so you should be able to look it up anyway so the documentation part and a bit about it is assurance, you should know where to look for it. Separation of processes, usually the business logic is in the back end system somewhere, when it comes to integration you do not want to have the business logic, but if you look at BPM process that are relying on SOA architecture you actually have some business logic in the process logic there. So you may want to look at number 10 to make it clearer. I think something is mixed here. Usually when you talk about patterns when it comes to services basically there are four kinds of patterns. Stateless service, is absolutely correct, you should never save the state in the services. It is more of implementation, how to implement service. For the business they always look at cost. You should always be able to do more with less all the time. If the available services do not give you the information or the value, then reusing and adapting existing service maybe more than just creating a new service but if you have an organization that knows everything about the services they should be able to tell you when it is more beneficial to make a new service then reuse.

7.3.3 Interview 3

Introduction Questions:

1. **Company name:** Major Telecom Company
2. **Position at the company:** Senior integration architects in broadband services
3. **Duration of employment:** 15 years
4. **Do you want this interview to be treated confidentially?** Yes
5. **Can you provide some information about your company?**
   This company is the largest telecom company in the Nordic counties; we are operators not only in Nordics, also the other parts of Europe. In Sweden we have 9000 employees.
6. **Can you provide some information about SOA projects implemented in your company?**
   There has been different kind of projects where we established some SOA projects because of reuse of our services. But the establishment of reusable services which maybe are replacing, point to point integration when you want to achieve decoupling systems and a base for creating reusability. This is maybe more considered as infrastructure project that in this time when you have not so much money to do this kind of projects, infrastructure projects are not easy to sell, there is a name on this kind of projects because in long term you can see benefits but your business case should be good to be able to sell that, now there are some SOA projects going on. SOA projects have been going on for last many years, 10 years or more. This company was the first telecom operator in the world using Tibco at that time. SO it was a big thing at that time. It was the BUS that have had different names but another most famous was the TMB, and it was more base on transport messages and in some cases you can say that,
you are not doing things right all the time, maybe sometimes you can see that the platform on the bus was used just to create point to point integrations through this kind of infrastructure. Now things have been changed and SOA in this case has not been easy. Everybody wants to achieve services and reusability but it is not only just some architectural requirements, it is more like a whole package. You need also the organization changes to make sure that you are able to secure SOA governance, I mean who decides which are the services are going to be created and what is life cycle for a service and in which process all these services being used or reused. This is not fully achieved here but we are doing the best to make everybody aware of that.

7. Why did you decide to adapt SOA in your organization?

We have a large experience of bad integrations in this company and this hard dependency in current integrations from legacy system makes it very costly and sometimes very difficult to establish new product and it takes long time. In some cases establishment of new products has required establishment of new system flows, so you create new systems just to be able to sell new products and that is wrong, in that case you were creating new integrations and these integrations were in SOA that time. There was no reusability thinking in that time, you wanted just to connect two systems to information. So this means that when you want to change current flows or when you want to sell a new product it is a big pain because of the dependencies and that is why there are some needs of making this better, that is why integration in a good manner is a positive factor to achieve this. And of course reduce number of systems. If you were able to achieve good integration, you also will be able to get rid of some of the systems in an easy way.

General Questions:

8. What kind of benefits did you expect from SOA adaption?

If you are able to have services that you are able to reuse, they will be easier to lock it to new products. You want to reduce some of business processes that are consuming some on these services as activities in some way. So instead of finding new ways of doing things in every project we reuse what you have. Your services would be able to grow in the life cycle but if you do not write your implementation, you won't be able to make your changes in services without affecting on many different systems. That is why we decouple the systems, and we make the services as general as possible. It makes it easier to be reusable of course.

9. To what degree reusability was important for you?

We have discussed about how to measure reusability and unfortunately we have not come to that point on the measurement of reusability, you know that we have some examples. We have service called manage address to be able to get address information from a master address system and we make that project to be able to offer address information to one system and then there were other systems that need the same kind of information and it was very easy to add these systems to consume that service so how do you measure that in comparison of trying to make other integrations from these systems to these master systems. There are some figures which you can do estimations but there are many factors which are very difficult to count. So measurement in reusability is difficult. I think that what is important is to have the possibility to offer that in an easy way and general way as it is the management of the service, it was easy to enable a new system to consume that we have another service called management of the order for when you are ordering goods from source system. So we created this service and we changed current integration from CRM systems to this order handling system and then when we had a service, there was mobility portal and they just connect to service and consume the service. It was easy at that time, so it is easy to do it when you have well described how
the service will work and operations enabled and what kind of information to trigger and how the exception chandelling will be handled. With general services I mean i.e. no product or business line specific. If you are able to handle for example "Manage Workforce" that is not connected to a specific product, is general and therefore more reusable.

10. To what extend your SOA implementation was successful in terms of reusability?

   In some cases it has been very good and for the services we have created we see a good percentage of reusability, we have another system called manage workforce for workforce management, when you are ordering service orders or delivery orders, something that is done manually is we were able to connect two different consumers and they were three different providers of that service and there was a new system that needed that kind of coupling to consume send service order to sub-contractors and all the clients that are going to reuse order to manage workflow service. It fits well described and explain how that will work. The only problem which I see is to decide an agreement of kind of working methods and what I see, difficulty is: we have different areas in enterprise architecture map and the responsible for this area should decide which are capabilities you want to offer from our area, so no matter which is the system provided that capability, this is what we would like to offer and this capabilities can be activities in this kind of processes. So I do not see any problem in creating services and reusability with them, I see the problem in designs which are priorities services we would like to create. And in what level services are to be created.

11. To what extend did you consider the capability of legacy systems and services to be reused in SOA solution?

   Everything we have is legacy systems beside some new ones. You need to use the legacy systems because they are the one providing functionality behind your service, the only thing is that sometimes you do not want to affect those systems, because you are establishing these services, but at the same time, maybe you have a need of some updates in the interface to the systems to be able to create reusability in a way of how the response for a service are going to be addressed to. Depending on what kind of service you are providing if it is a more synchronous service or unsynchronous service. You need to know where to address a reply in every request service. And in some cases you need the system that is providing that functionality to be able to be audited in the interface to handle some kind of address in parts that are going to be sent back in your reply for services. Sometimes interfaces that some legacy systems are very obsolete and not so good. You do not want to affect or change systems because you want to establish new services and at the same time some changes could be necessary, so we try to reuse them as they are, but in some cases some updates are needed.

12. How do you measure the ability of applications to be reused?

   This question is answered before

   Factor Related Question:

13. What challenges did you face when you wanted reuse legacy system and services?

   The challenges could be more related to in what way these legacy systems have been used before, or how they were developed, as I told you in some cases we have created flows for specific products and you want to put more in general context you have these flow dependency in some of systems and you need to break down that to be able to do in more general, so it could be that how some of systems have been more specific for some kind of products. Sometimes it is how you want to package this concept of a service and some of services we are pointing now, and the characteristics we have seen from other services that we are providing into enterprise service bus. Preferably you are able to access that service
in a couple of different ways through directory or through JMS, as a backbone we are using for messaging services or as a web service and the characteristics also are, this services express in common message model where using ESB, and what is behind the service is also handled as part of the integration solution connected to the service, to be able to do routing to end points and transformation of messages to specific messaging format.

14. **What key features of services do you consider to be important in reusability? Or what kinds of services are more reusable?**

Composite services are in some cases difficult to reuse, of course lower level, get address details, get point id, it is easy to reuse because it is maybe more general need that you can have in different part of the flow. Reusability of composite services sometimes can be difficult depending on in what purpose composite services where created. Because sometimes there are some needs that is a little specific for one project or one specific kind of product. It is difficult to create higher level services that are so general. We have not done that much regarding composite services. Gather information services that you are getting address details, like we have some common services in Finland for credit check. Reusability is more in the way of systems providing the functionality behind the service or the information behind the service. For instance in the manage workforce there are three different systems that could provide that, and it is depending on what kind of product areas that are covering the orders to be sent to. These are not only gather information services manage workforce for instance, it is a create update and cancel order service, the only thing is that who is the system providing that capabilities. What I can see from high level, if you create an order service that an order service could be maybe different in the ways which are underlying services to be called in order service maybe depending on what kind of product you are going to create, and you may be able to trigger a service with necessary information, to do the service so intelligent that it is able to call different services, underlying services depending on the input you are triggering with.

15. **What should be taken into consideration in developing new services to be reused later?**

The service itself might have kind of mean in high level, and to make it more reusable is to follow the principles of the services we have already done, and being able to offer it in numerous different ways regarding what kind of transport you use, to use the common message model to express that service to express what is triggering that service and to have it well described and defining in the service contract so you see what you get when you want to use that service, and to get some kind of approve from the people responsible for the area providing that service, because you want to create a service as general as possible or as future prove as possible, you are able to update a service but you do not want to make large changes because new needs appear, you are able to update a service in a life cycle and the service should be changeable in order to adopt to future needs. But when you create a service form the beginning it is good to look ahead and see what can we have with that service to be able to make it as wide as possible to cover that long term needs. We use OAGIS as our Common Message Model

16. **What do you think about benefits of reusing services? Cost of reuse vs. developing new services**

Depending on how big change to service could mean and the impact that could have for some of the consumers or the providers that you want to avoid and depending on if it is the same kind of functionality behind that service, we can wonder why do you need a new or parallel service that is going to provide you almost the same functionality, that has to be analyzed depending on the impact for the consumer, but I guess if you are able to create a service as general as possible, changes should not affect that much consumers and providers. Because changes in a service mean that the systems providing functionality or information behind that service are changed some way. So the only thing is how you can keep the interface as natural or general as possible to not affect the other that are
consumers of the service, as I see it, a service is an interface that you trigger sending some kind of information or data and then getting back other kind of data responsible for that, and if you are able to make the service as transparent as possible, your service is expressed in a large container that you just need to trigger with some data, and the data is one that is triggering the system or systems behind that, if you are sending more information in that container and the container is transparent, it is just how you solve integration to that system, sending the right data to be able to trigger the specific functionality to get the affect you need and get the response you need, if it is initiating an order or if is just getting address information on where it could be.

17. What significant changes can be applied to services to make them reusable?

Maybe you have a service that just gathers the information from system A and if you want to be able to gather information from system A or System B you need to be able to address where to pick up that information and depending on who is requesting that from the service you must be able to address responses from your service calls back. If I do not know why a service is not reusable I cannot tell you what to do to make a service reusable. The service must be according to standards, It depends on how you call the service and how the service is expressing, what kind of format to call that service, I would say that if you are able to change the service as expresses in a standard way they are more reusable. We try to express services in standard way and we try to express them in a common message model way, because services with common message models can support more message models and can be reused easier, we are using wages as a framework for this common message model anyway it is a large container on how your pieces objects are described in a structured way and if you have a service that is more compact way expressed, you have problem of making reusable because to be able to reuse it you need to change the service and not the new system that is maybe consuming the service changes and that would mean that you need to change the first system that consumes the service and as I see it you need to express the service in a general way, based on standard, that is able to make the service as general and transparent as possible.

18. What other factors do you think affect reusability in SOA?

We are dealing with a complex system in this company, and with heritage from different kind of influences from broadband services, mobility services, enterprise and so on, when you are looking almost at the same thing from different points of view, it is a little bit difficult to create reusability if you do not have a common way of looking at this services, that is why it is good to be able to create a rule and decide from the beginning. Rules for how the services are going to be created, expressed, offered, described and documented. Something that we are lacking here and trying to have is that we do not exactly have a specific catalog for the service to manage the existing services, to find which service we have today, what they are doing and, they are using different kind of tools in another level to describe using Aris tools to describe some of the systems and functions they are having, but not exactly the services we are providing and we maybe are looking to wiki where we can go in there but I do not think that it is so clear reusable when new projects are coming, they are not able to go there, you have to know when new project are coming they are not able to go there, you have to know what you have when new project comes with their needs and say ok, we can solve the needs with this current services, if you understand the specific needs that project have or we will need a new service for this so maybe as I said to create roles on how they are created how they are expressed how they are documented. Rules mean the guidelines inside the company primarily. Because you have a large company and they are doing things in different ways in every corner. during the years maybe the same systems have provided similar functionality from many different kind of interfaces, the same application and it is because new projects have new requirements and very similar to old ones but they
did not look at what already exist today, so the people that was developing those systems maybe just think that it is ok, we can do that, they have something similar but it does not work. So they have the power to decide and understand how things are to be done, implemented, documented and maintained.

19. How important do you find these following factors (Appendix B)?
I have mentioned factor number 2 before. That we are able to decide which services we have and you want to create new services just because a new consumer comes and wants to do almost the same as you have in one service. Factor number 1 is also important. Facto number 3 as I told you before if you want to make a change in a service you do not want to affect that much other consumers that are using that service that is part of reusability positive part that changes in a service because of one consumer does not need to affect other consumers of that service and how to achieve that of if you are adding other providers or functionality of that service it is more like an address or address issue. You do not want to affect consumers because you are adding a new provider, how to do that in an easy way.
For modularity, if you have a higher level service or a composite service that is consuming other underlying services you are able to create that kind of services but maybe the reusability of that service can vary, depending how you have built that service and how specific is for some kind of product. I do not think that is as important as first or second factors. It is important to have modular services but depending on what level you are going to create composite services. Standard conformance is important too. Stateless services are also important. All the services we are doing are stateless and it affects reusability because state full services would mean you need to store data on services and when you want to change the services you need to change data in it, and that would add a little more complexity.

7.3.4 Interview 4
Introduction Questions:
1. Interviewee: Johan Tuvstedt
2. Company name: Dynabyte
3. The company which you have done SOA project: A major telecom company
4. Position at the company: consultant
5. Duration of employment: 18 years
6. Do you want this interview to be treated confidentially?
   Yes
7. Can you provide some information about your company?
   This company is main telecom company in Nordic countries and the biggest wireless and fixed telephony vendor in Nordic countries and have about 30 000 employs all over the world.
   Can you provide some information about SOA projects implemented in your company?
   This company has two branches, which one is for mobility and one is for broadband and difference between them is that, mobility side is based on wireless services and broadband side is based on fixed telephony and fixed data services. These two branches work very independently. They are so independence that an integration competence center which is residing only in broad band side and this company is residing only on broadband side, there should be corresponding organization with wireless or mobility side but there is not. In the same way there is no CIO office which is on enterprise level,
it’s different CIO office depending whether you work with mobility or broad band. The only connection point is only channels or CRM applications which deal with customer and I work in the integration competency center, for broad band which is a split operation between Sweden and Finland. We make the guidelines and run the actual operations. There are currently 800 systems which are integrated over integration platforms. The policy is to use SOA solution to integrated systems from all parts of company. Some of the systems are master systems which publish or maintain business objects such as customers or install based or finical systems and publish this for systems to subscribe to if they need to keep their on data set up to date which the master system data feed. There are 40 systems which utilize customer objects.

8. Why did you decide to adapt SOA in your organization?

The company itself can answer this question better however, there is a policy board called BBC architecture who decides on various policies and how the things are to be done, and the results of BCC architecture were written in an IT hand book and the IT hand book, should hold the policies at one point in time. I assume that it says that you should always do your integration in SOA way. There are between 50 to 100 reusable services which can be dealt with SOA. The customer object and order object can be those services.

General Questions:

9. What kind of benefits did you expect from SOA adaption?

When you implement an object which you mean to be reusable, you would expect that it would be conformed to the business process. If you have common information model you can start publishing objects according to how model is built in such a way you are building a puzzle which is more and more covering a particle which does not leave holes. Which means that if someone wants to fulfill a process need there will be a process which supports that and the process should be modeled according to information model that would implicate the service which I need either it is already there or can be easily adapted, so I get what I need from it, or if it’s not there, it should be built to fulfill the hole in the puzzle but if you do not start with processes with information model but with systems start sending, I need this, then I need this, you get the whole system integration but you get very little process support. I expect benefits that if you build everything correctly, all the business needs that are concerning that information object would be fulfilled with such an object.

10. To what degree reusability was important for you?

In project of CRM orders to different backend systems. The CRM was the channel to the customer and provides data for web and this CRM system places orders to multiple systems. Those orders are formed on group of services that uses atomic services on the previsioing systems. For me it was important to make sure that CRM did not have to make point to point integration to all the previsino systems. But rather could send general orders with lots of specifications and parameters in them then they would be handled by different systems depending on the order content.

11. To what extend your SOA implementation was succesful in terms of reusability?

Services such as product availability, and production object order could be reused to very high degree. Say more than 90 percent. But the deviation on continues change of the business process and hands the products and offerings differ over time because the information model was less then optimal when the first services were done, they needed to be added parameters to handle different kinds of particular objects that were known at the time of creation of the services. But the services in themselves were built in such a way that they were extendable and the envelope containing the business object
standard envelope which is produced by an open application group integration specification. And say that well if we have an order for a product these things needs to be in there so the object that we use is very easily extendable and it's also very complete to any kind of business object. So the reusability of the service is very high but there needs to be additions to the data set or information model both form provisioning system sides and from the offering side from CRM because of the business change, the reason why most mobile subscriptions are filtrate is because, then you do not have to change the IT systems sitting behind them just because you want to sell other service. It's so expensive to change the systems that help you sell this services that is more beneficial to just keep them filtrate and add whatever services are expected from customers. So the reusability is very high. The project is still ongoing and we could achieve 100 percent reused and its only data set which is evolving over the time.

12. To what extend did you consider the capability of legacy systems and services to be reused in SOA solution?

The focus is to make sure that on the systems that sit in production side actually does the composition of services to fulfill one or multiple user need. That means that we do not consider the capability of legacy systems to be that great we say that let them do what they do, then we identify one of them and then fortify that substantially to make sure that is the one that produces the SOA services.

13. How do you measure the ability of applications to be reused?

The reason for reusing of for focusing on one system rather than others is that because if you have many integration points that derives cost very quickly, but if you want to identify one system to say that this is the one which is going to produce the service and is going to rely on other subservices is not so much form capability perspective is more from perspective of which system is in fame, which systems are not domed due to the life cycle and due to license parameters, so the decision is not made on particle utility of the system rather made on the based on which is the vendor? Is it cut system? or if it is a home build, is it in the target application architecture? If you want to quickly deploy some kind of new product, to a SOA service you need to take into account whatever legacy systems you have and what they can produce otherwise it takes you forever to do the actual production. But we never make systems just particularly for the integration or for the SOA publication, relying on another systems, we always identify one the systems saying that, this is our new target system and we need to have that to get produced from other systems.

Factor Related Question:

14. What challenges did you face when you wanted reuse legacy system and services?

The biggest challenge is the lifecycle, because if you are to change an existing service you need to take into account all its consumers. And if you need to change the service you need to affect all the consumers potentially and then you need to make sure that you can release synchronized and you can test the synchronized and you can handle the changes synchronized. Otherwise there is no reuse, otherwise you just reproduce and republish an existing service but for a new user, other challenges are usually connected to IT environments such as java versions ability to handle different transport paradigms and transport mechanisms. Physical capability of systems is kind of challenges. If the system or service from the beginning is built towards filling the gaps in information model I discussed earlier, there will not be difficulties with that because it means the service has its home somewhere in the information model it supports some kind of business processes.

When you connect a new consumer to an existing service, ideally you should not change the service, you should reuse it. But that means you implicate the consumer of the service actually can handle the
language and dialog patterns and use case produced, by the service, and if there is a difference between what the consumer expect and what the system produces, then in my word that’s a good reason to actually consider changing the service. You still have the reuse of service but you still need to do some work with it. There is no such a thing as plug and play, it never happens.

15. What key features of services do you consider to be important in reusability? Or what kinds of services are more reusable?

Services built to standard compliance, are often well reusable, services developed for particular system needs have less reusability. Standard can be either an external standards. If you build your services according to standards, they become reusable, but if you build the service to just fulfill a specific system need than it’s not a service it's just application.

16. What should be taken into consideration in developing new services to be reused later?

All the expected users of the service should be taken into account, it is easy to say but what does this actually mean? Right now we are in the scope of producing services for handling service incidents and system or network changes so its two services that we are developing as we speak and if we take for instance service incident, today it’s handled by maybe 10 systems in Sweden or Finland, and they all have their personal touch. That’s why we say service orientation initiative that saying when we that new incident service, that is going to handle all service requests and handle incidents to tell the customers that we are working on the problem. It’s closely related to the produced functions of the system but when doing this it is very important to take into account both information models so the service becomes as complete as possible and you also need to take into account what is the best practice for actually producing this kind of service, from different standard organizations. When developing services it shall not take into account only what is done today by systems but also what future business process expect from that system and of course that cannot be done 100% correct first time, that’s also should be taken into account how we are going to change or evolve the service over time, because if you make service, you make it general and available, and try to reuse it but you can’t change it, than you make very less degree of reusability, so be able to handle differences over the time in the service content it is very important, you need to have a plan to how to evolve the service, therefore the service should be changeable, it should be built on the top of information model, it should be built according to the business processes it should be taken into account a target information landscape systems. But unless you can change it over time it won't work.

17. What do you think about benefits of reusing services? Cost of reuse vs. developing new services

The cost of service over the life time is more than 90% in the operation and management of it rather than the development. Development cost is very little; it is between 5 to 10 % of total cost. And rest is operation cost. So if you do not reuse the service you most compare, what is the cost of changing existing ones so it works for its current users, and the new user, and you need to compare that cost to the 10 times of the cost of developing new ones. Operation cost is fixed cost for maintaining CPUs environment software, service content, business object and so on.

The life time cycle cost for one machine to machine connection, according to Gartner 2009 is between 10 and 100 USD per year. So you can keep the connection points as few as possible you save a lot of money. That why it is very important to make sure that the services you are going to produce is reusable.

18. What significant changes can be applied to services to make them reusable?
Adherence to standards and adherence to standard models, and to uphold information model within your enterprise which is up-to-date with the business processes. If you do the service according to the way you do the business, and to the way you treat your information the changes are handle able, which mean that if you evolve your information model, by for instance saying that we can have multiple customers to one offer instead of one offer per customer because we change our business model, but if that change is handle able through information model, the service is built on that information model, and you can change your service, but if you just develop your service to fulfill systems requirement but not your information models than you need to take big steps every time to change them. Because you do not only change providing server, or service you need to change all the consumers as well because if you want to reuse you have to build them so that they can be reused. Even though you can change the service the recipients might not be able to handle that.

19. **What other factors do you think affect reusability in SOA?**

I believe that businesses are very much alike; they work in very similar ways, which means that if you sell cars or you sell electricity, most of the services can be reused between those services except for services which you have actually for producing the goods which are inherently different. But with that in mind I would say that whatever you can do to adhere to existing standards for how business is made on general level and make your service more reusable and vice versa. If you start to build your services at your particularities of your production object or your solution or your service level agreement for electricity, they will not be very much reusable, of course the car industry services cannot be directly adapted by power industry, but there is more because information content of service are very much focused toward this particular product or this particular business model. If you can keep the service anonymous of the particular use of it than you have more reuse possibility. So I am very much for using standards framework for building the service competence. When you use an external framework you anonyms your information model from the service and does the service easier to adapt to changes if you take your particular system needs, in this system needs these five objects, and you produce them in custom object, and you need to change either parts of this chain than, you will have to redo entire integration. Or if you have one CRM system delivering customer particularly for the financial system, that financial system would need to know like social security number, the name and surname, but that customer object will not be reusable for the systems that are going to deal with logistics. Because that needs likely name which is then reusable but if you do not have the address you have no use of the name for the logistic systems. So if you can fulfill and external specification for an information model or for business objects instead of trying to fulfill your particular system need when you do the service it will become reusable. And you inhabit reusability by doing in system specific.

20. **How important do you find these following factors (Appendix B)?**

Standard conformance is the most important quality attributes. Discoverability is very important because the service that no one knows about does not even have potential to reuse. But if you make your service connected to a particular standard anyone that can relate to that standard can use it. You might not use entire object or you might need to combine objects with something else, but it gives you the context so that you understand how this works, if you do not conform to any kind of standard there is less need in the declaration of the service to actually say that this and this data means this, which means that it gives you few objects which does not actually have metadata which means the understandability of the service utility is very difficult for user, so that is definitely the most important one.

I think most of these factors are important. Any kind of project undertaking, likely tends to be more and more pragmatic to closer to the time bracket you end up. So when you run out of time in the
project or when you need to deliver particular functionality you need to shortcut few things and I think factor Separation of processes, is very often has to give way to other prioritizations. It is very common to have the service dialog agnostic. To keep term catalogs consistent is also very important. To hit your term catalog up-to-date and to not deviate from the purpose of a service when evolving it over time is important. if you have different purpose but you think you might reuse part of another service it is very important to actually keep the business decision of the business case for doing it to make sure that you do not corrupt someone else working services just to fulfill an extra need which might not be important enough to actually handle as a separate service. I like the factors and I would like to have the factor of the business process and information model adherence as a very important factor which I cannot see here, if you do not take this factor deeply into account when you produce the services you do not get a lot of reusability, you need to know which problem you solve with service. And if you are going to uphold the catalog of how services work you need to relate them to how you produce your value that is the business process, and business process will rely heavily on information models, so you need to have those clearly stated. that our product looks like this and our customers look like this and that is business specific knowledge but still needs to be reflected in the service. Stateless services is the important factor, service itself or service implementation must never be allowed to hold state data because then it becomes an application and it would need to consume huge of amount of CPU if they hold states.

7.3.5 Interview 5

Introduction Questions:

1. Company name: Jesper
2. The company which you have done SOA project: Electronics Selling company
3. Position at the company: Dynabyte Consultant
   Duration of employment: Almost two years
4. Do you want this interview to be treated confidentially? Yes
5. Can you provide some information about your company?
   This company sells electronic devices all over the world. It is a big company and established in many countries in Europe and also in USA. And I work for the company in Sweden and also in Netherland. They are selling from small electronic parts to big electronic parts, so it is very big company and sell a lot of products.
6. Can you provide some information about SOA projects implemented in your company?
   This company is the best company that uses SOA, because there are many companies talking about SOA but they do not implement it right but in this company they have done SOA implementation that they have different countries and all the companies are using the same integration for handling the information between the websites and their backend system so they use the same product in all countries and its deployed on a lot of service and they are using the same web service for handling this and also the implementation between the products information management system and also the economy system. This is integrated with the same integration not in all countries yet but it goanna be. And I am using JMS between product information management system and economy system. We reuse these adapters in all the countries of the same and we are using canonical data model because we are going to reuse it and all the countries have some other fields and additional fields. We add it to canonical data model and then we can reuse it.
7. Why did you decide to adapt SOA in your organization?

Because they are big companies and if they are big companies, they have to use SOA however in some case not, but the two companies I worked for are using SOA in some way. Because they want to have reusability and also they want to have the flexibility so they can add the systems and take away others.

General Questions:

8. What kind of benefits did you expect from SOA adaption?

The reusability in SOA is important and the reusability have impacts on flexibility and the flexibility is that they can still have the backend system and working with new front end systems. This makes the company faster because, when they are going to do a change, it is going fast because the backend that have a web service they can just change the front end system fast and reuse the backend system. So I think that is the best thing with SOA in this case because this is also lower cost for the company, because they can reuse as I said, because the implementation we have done in this company in Sweden is that we reuse the same adapters in all the companies and they also have adapters that communicates with front end systems and they can reuse those adapters for all the countries which is good, because they do not have to pay the development for every company. They have paid for Swedish solution that is what we have done and now they are going to implement it in all the others with the same adapters.

9. To what degree reusability was important for you?

We reused the services very much. Because the services make the things flexibility because if we change the services you have to change the other part, so you are using the same services for all systems just to make a quick change and if you reuse the services, you do not have to develop a new one when you are going to adapt another.

10. To what extend your SOA implementation was successful in terms of reusability?

I was successful. I have done it for just two counties now. But the reuse was good. Because we are reusing one platform that is the same and also the same systems and because the both adapters are the same and the services are the same and reusability will be fine, and we are going to develop in other country in short term.

11. To what extend did you consider the capability of legacy systems and services to be reused in SOA solution?

We didn´t have any legacy systems. And the services which we had were absolutely reusable.

12. How do you measure the ability of applications to be reused?

We did not have any measurement tool to measure the reusability of services.

Factor Related Question:

13. What challenges did you face when you wanted reuse legacy system and services?

When we developed the first service, because we adopted and developed the first one we have to think about all the challenges we have, we are going to deploy it to the next country to think one step longer all the time, to do the reusability easier next time that is why we implement two adapters one for ESB and one for canonical data model. To come a long time to develop this integration but when we are going to reuse it, it takes much short time, the first project took about six to seven months and the integration took about one month. So there you have the saving of the cost and saving of the money because it is not so long project. But you have to think when you develop the first one just to make it reusable to next integration, so that was challenge for me first time to do that.
14. What key features of services do you consider to be important in reusability? Or what kinds of services are more reusable?

It is good to use techniques that are standards and simple, like using JMS and EMS standards. We also use FTP transfer and we are also using JDBC but it is down to data base and in the web based integration this company is using HTTP and SOAP. When it is standards it is easy to communicate with new systems. So that is good and easy to manage them.

When you develop the adaptors or integration you have to develop it that you are using like a canonical data model. In company number 2 they have canonical data model for everything. If you are using that, it is easy to add new systems. Because you have canonical data model that is transporting the data and it is easy then to change the systems on which they reside and then you do not have to do anything else. Like in another project I am working right now we have this small project where we have system error in about one year we are going to add new system. We are sending the information with canonical data model and we are using adaptor before this and when this system takes away, you have just the queues and canonical data model. Canonical data model is a place for data.

15. What should be taken into consideration in developing new services to be reused later?

15:34 a service have to be standard that makes it easy to reuse and just to handle information. That means you are not using any specialties or something like that, you are using the normal interface and normal service standard and do not do any specialties around it so that you are using like a SOAP and have it like a SOAP and do not make it harder than that but the best thing is to use the standards. Standard means for example JMS and standard interfaces as well.

16. What do you think about benefits of reusing services? Cost of reuse vs. developing new services

No answer

17. What significant changes can be applied to services to make them reusable?

No answer

18. What other factors do you think affect reusability in SOA?

Somewhere in the company you have to write down or have all the services that are using, it is good to have a template that you are working on, then development is easier and it is also easier to reuse. On the other hand using the same standard for all the transports and maybe Having some guidelines description for the integration in the company or how you do the web services in the company, what you transport in or if you have canonical data model, so when new consultants attending to new products they have guideline that they can write how to use SOAP for example in the company and what they have for preferences

19. How important do you find these following factors (Appendix B)?

For me using standard interfaces is important, And it is good to have, because all the new consultants will use the same standard for interfaces and business terms, so nothing goes wrong there. Factor number two is good because it is good way to reuse the services because they are in the same place or it is good way to reuse like template or you can use them again and also look at the code and maybe the same thing again for reuse. In this company we have a big map of plan that all the services are available. And we can see all the systems that are coupled to each other, then you can get an overview and also if you want to change something it is easy to see what happens. Factor number 3 I do not have any experience about that. Factor number 4 is absolutely important, because when you develop a service you have to think about the future, what is going to happen. Is it going to be changed in the
future or do you have any plan for changing. If you have that you have to develop it more open and not so specified so to be reused in future. Factor number 5 is important because if you want to reuse, you have to provide common services. Modularity is also important because it is like loose coupled system. Adaptability is good to have but I do not know much about it. Standard Conformance is very important as I said. If you are using SOA internal it is easy to use standards but it is better if you are using SOA B2B and external because it is easier to communicate without the system all of them or many of them are using standards. Discoverability is very important because if you do not find them you cannot reuse them so it is very important that you know which system are coupled and how they are coupled with which services. Separation of processes is the main thing here, if you separate processes it is easier to reuse and it is very important that they are not dependent on a single process flows, and it is better also in security point that you have or error points if you separate processes. In this project we have one pattern for message exchange but in other projects we are using multiple patterns. Stateless service is best practices for integration. Because the integration should be stateless because it is better in error, if there are errors it is better for integration to not keep any state, and not save any information about product data.

7.4 Appendix D: Validation´s Questions:

Please introduce yourself (Name, Company, Position and a description of knowledge about topic of this master thesis.)

Please write a review based on your opinions about the result obtained in this thesis. Your review can be based on the following questions:

- Do presented results answer the question of this research?
- Are all the collected data presented in results?
- Are the results obtained according to the methods mentioned in this thesis?
- Are obtained result correct?
- Do you find obtained results credible?
- Are the results presented clearly, logically, accurately and systematically?
- Do you find the result useful?
- Do you find the result relevant?
- Is the quality of results good enough?
- Are there any comments or suggestion?

My name is Joakim Lundin and I am a team leader and consultant at Dynabyte where I work with enterprise application integration. I have nearly five years of experience working with integration related tasks and daily get in contact with SOA oriented questions.

The thesis presents a thorough analysis of the factors that affects reusability of services in a software oriented architecture. This is achieved by creating a list of factors derived from literature which is later merged with a list created from a qualitative analysis of interviews with senior integration architectures. The author also makes a comprehensive description of the basic concepts and principles of SOA which makes it easier for the reader to understand how the factors actually affect reusability. The results are neatly presented in a complete list of factors affecting usability in SOA around which he continuous with an analysis and discussion.
The list of interviewees is composed of an impressive crowd of senior integration architects and consultants with a wide range of experience from different companies and organizations in between them. The majority of the factors derived from the interviews have two or more persons supporting them which make them easy to accept as valid. The author highlights the most valid factors in the results which are the ones confirmed by both the interviews and the literature analysis.

I support the reasoning and the conclusion the author makes in the analysis and discussion. During the thesis the author displays a good understand of SOA and how different elements can weigh in and affect the effectiveness of the architecture.

The results in the thesis are relevant both academically and professionally. Professionally the list of relevant factors will help an architect to ensure an implementation which will allow for reuse. Additionally it will serve as a tool for making priorities of what efforts to make in a SOA project. It would have been interesting to see an extended analysis where the interviewees are confronted with each other’s results and asked why they had neglected certain factors and whether or not they deemed them relevant for reusability. I would also have been interested in an attempt to rate the factors to see which one affects reusability more than the others. However, that might be outside the scope of this thesis and may be more relevant as future work.