E-Business Interoperability Frameworks for SMEs

Supervisors: Professor Kari Smolander
D.Sc. (Tech) Ossi Taipale

Lappeenranta, 11th April, 2010

Md. Abul Kalam Azad
Härmälänkatu 30, apartment 10
33900 Tampere
md.azad@lut.fi
ABSTRACT
Author: Md. Abul Kalam Azad
Title: E-Business Interoperability Frameworks for SMEs
Department: Information Technology
Year: 2010
Master's Thesis, Lappeenranta University of Technology
81 pages, 25 figures and 3 tables
Supervisors: Professor Kari Smolander
D.Sc. (Tech) Ossi Taipale

Keywords: Frameworks, E-Business, Interoperability, SME, Standardization

Current e-business standards have been developed and used by large organizations to reduce clerical costs in business transactions by increased automation and higher level of business-to-business integration. Small and medium enterprises (SME's), however, cannot easily adopt these standards due to the SME's lacking the technical expertise and resources for implementing them. Still, large organizations increasingly require their business partners, most of which are SME's, to be able to interoperate by their chosen e-business standards.

The research question for the study was, first, which of the existing e-business technologies are most SME-adoptable, and, second, how could those e-business technologies be made easier for SME's to implement. The study was conducted as a literature study that evaluated the available e-business frameworks and SME-oriented e-business architectures based on the implementation complexity and costs incurred for the SME adopter. The study found that only few e-business solutions are SME-adoptable. The technological approaches used in the solutions need to be improved on a number of areas, the most important of which is implementation complexity. The study revealed that this also applies to the special, SME-oriented e-business architectures, which are also still too difficult for SME's to implement. Based on these findings, a high-level e-business interoperability framework concept was proposed as the basis for future research to overcome the found implementation complexities for SME's.
FOREWARD

This Master’s Thesis has been accomplished under the Software Engineering Laboratory of Information Technology department in Lappeenranta University of Technology, since May, 2009 to April, 2010. During this time I have received several internal and external cooperation and advices from different person. First of all, I would like thank to Tero Pesonen who was reviewer of my writing; if there are any standard in writing I will be very happy to pass the credit to him. I want to thank Mr. Kari Korpela, Project Manager of Lappeenranta Innovation Ltd. who has provided me several documents on their on going project and also he advised me for future research ideas. Finally, I want to say special thanks to my supervisors Prof. Kari Smolander and Dr. Ossi Taipale for their all supports and help.
# Table of Contents

1. **INTRODUCTION** ....................................................................................................................................... 6

2. **BACKGROUND** .......................................................................................................................................... 8
   2.1 **ELECTRONIC BUSINESS** ................................................................................................................ 8
   2.2 **SMEs** ................................................................................................................................................. 9
   2.3 **E-BUSINESS INTEROPERABILITY** ...................................................................................................... 9
   2.4 **INTEROPERABILITY SUPPORT AT SMEs LEVEL** .............................................................................. 12
   2.5 **XML** ................................................................................................................................................ 12
   2.6 **E-BUSINESS FRAMEWORKS** .......................................................................................................... 14
   2.7 **CLOUD COMPUTING** ..................................................................................................................... 15
   2.8 **SUMMARY** ......................................................................................................................................... 18

3. **E-BUSINESS INTEROPERABILITY BUILDING BLOCKS** ................................................................... 19
   3.1 **BUSINESS TO BUSINESS INTEGRATION** ...................................................................................... 19
       3.1.1 **B2Bi Approaches** .................................................................................................................... 19
       3.1.2 **CCTS & CCL** .......................................................................................................................... 21
       3.1.3 **UBL** ........................................................................................................................................ 23
       3.1.4 **Web Services** ........................................................................................................................ 24
   3.2 **SOA CONCEPTS** ............................................................................................................................ 26
       3.2.1 **SOAP** ...................................................................................................................................... 27
       3.2.2 **WSDL** ...................................................................................................................................... 28
       3.2.3 **UDDI** ....................................................................................................................................... 30
   3.3 **XML AND WEB SERVICES SECURITY** .......................................................................................... 31
   3.4 **SUMMARY** ......................................................................................................................................... 35

4. **E-BUSINESS INTEROPERABILITY FRAMEWORKS AND TECHNOLOGIES** ......................... 36
   4.1 **EDI-BASED E-BUSINESS FRAMEWORKS** .................................................................................... 36
   4.2 **XML-BASED E-BUSINESS FRAMEWORKS** .................................................................................. 40
       4.2.1 **RosettaNet** ............................................................................................................................. 40
       4.2.2 **ebXML** .................................................................................................................................... 42
       4.2.3 **OAGIS** .................................................................................................................................... 45
   4.3 **COMPARISON BETWEEN EDI-BASED AND XML-BASED EBUSINESS FRAMEWORKS** .............. 47
   4.4 **SUMMARY** ......................................................................................................................................... 49
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>eBusiness</td>
<td>Electronic Business</td>
</tr>
<tr>
<td>SME</td>
<td>Small and Medium Enterprise</td>
</tr>
<tr>
<td>MNC</td>
<td>Multinational Company</td>
</tr>
<tr>
<td>B2Bi</td>
<td>Business to Business Integration</td>
</tr>
<tr>
<td>B2G</td>
<td>Business to Government</td>
</tr>
<tr>
<td>B2I</td>
<td>Business to Intermediary</td>
</tr>
<tr>
<td>EDI</td>
<td>Electronic Data Interchange</td>
</tr>
<tr>
<td>XML</td>
<td>Extensible Markup Language</td>
</tr>
<tr>
<td>XSD</td>
<td>XML Schema Definition</td>
</tr>
<tr>
<td>HTML</td>
<td>Hyper Text Markup Language</td>
</tr>
<tr>
<td>SGML</td>
<td>Standard General Markup Language</td>
</tr>
<tr>
<td>DTD</td>
<td>Document Type Definition</td>
</tr>
<tr>
<td>UBL</td>
<td>Universal Business Language</td>
</tr>
<tr>
<td>BIE</td>
<td>Business Information Entity</td>
</tr>
<tr>
<td>ABIE</td>
<td>Aggregate Business Information Entity</td>
</tr>
<tr>
<td>ASBIE</td>
<td>Association Business Information Entity</td>
</tr>
<tr>
<td>BBIE</td>
<td>Basic Business Information Entity</td>
</tr>
<tr>
<td>ESB</td>
<td>Enterprise Service Bus</td>
</tr>
<tr>
<td>CC</td>
<td>Core Component</td>
</tr>
<tr>
<td>ACC</td>
<td>Aggregate Core Component</td>
</tr>
<tr>
<td>ASCC</td>
<td>Association Core Component</td>
</tr>
<tr>
<td>BCC</td>
<td>Basic Core Component</td>
</tr>
<tr>
<td>CCTS</td>
<td>Core Component Technical Specification</td>
</tr>
<tr>
<td>CCL</td>
<td>Core Component Library</td>
</tr>
<tr>
<td>BOD</td>
<td>Business Object Document</td>
</tr>
<tr>
<td>UDDI</td>
<td>Universal Description Discovery and Integration</td>
</tr>
<tr>
<td>SOAP</td>
<td>Simple Object Access Protocol</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>WSDL</td>
<td>Web Service Definition Language</td>
</tr>
<tr>
<td>EIF</td>
<td>European Interoperability Framework</td>
</tr>
<tr>
<td>PIP</td>
<td>Partner Interface Process</td>
</tr>
<tr>
<td>TPIR-PPIP</td>
<td>Trading Partner Implementation Requirements PIP</td>
</tr>
<tr>
<td>TPIR-PF</td>
<td>Trading Partner Implementation Requirements Presentation</td>
</tr>
<tr>
<td>RAE</td>
<td>RosettaNet Automated Enablement</td>
</tr>
<tr>
<td>RNIF</td>
<td>RosettaNet Implementation Framework</td>
</tr>
<tr>
<td>ebXML</td>
<td>Electronic business using eXtensible Markup Language</td>
</tr>
<tr>
<td>ebMS</td>
<td>ebXML Messaging Service</td>
</tr>
<tr>
<td>BPSS</td>
<td>Business Process Specification</td>
</tr>
<tr>
<td>UN/EDIFACT</td>
<td>United Nations Electronic Data Interchange for Administration, Commerce and Transport</td>
</tr>
<tr>
<td>UMM</td>
<td>UN/EDIFACT Modeling Methodology</td>
</tr>
<tr>
<td>UNTMG</td>
<td>UN/CEFACT Techniques and Methodologies Group</td>
</tr>
<tr>
<td>NDR</td>
<td>UN/EDIFACT Naming and Design Rule</td>
</tr>
<tr>
<td>OASIS</td>
<td>Advancing Open Standard for the Information Society</td>
</tr>
<tr>
<td>CPP</td>
<td>Collaboration Protocol Profiles</td>
</tr>
<tr>
<td>CPA</td>
<td>Collaboration Protocol Agreement</td>
</tr>
<tr>
<td>ISO</td>
<td>International Organization for Standardization</td>
</tr>
<tr>
<td>ICT</td>
<td>Information and Communication Technology</td>
</tr>
<tr>
<td>WWW</td>
<td>World Wide Web</td>
</tr>
<tr>
<td>W3C</td>
<td>World Wide Web Consortium</td>
</tr>
<tr>
<td>ABILITIES</td>
<td>Application Bus for Interoperability in enlarged Europe</td>
</tr>
<tr>
<td>SMEs</td>
<td></td>
</tr>
<tr>
<td>IETF</td>
<td>Internet Engineering Taskforce</td>
</tr>
<tr>
<td>OAGi</td>
<td>Open Applications Group</td>
</tr>
<tr>
<td>OAGIS</td>
<td>Open Applications Group Integration Specification</td>
</tr>
<tr>
<td>EIF</td>
<td>European Integration Framework</td>
</tr>
<tr>
<td>BPEL</td>
<td>Business Process Execution Language</td>
</tr>
<tr>
<td>------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>BPEL4WS</td>
<td>BPEL for Web Services</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

The concept of electronic business (e-business) was introduced over two decades ago to improve business process efficiency, and to reduce operational costs. The first standardized approach to e-business was Electronic Data Interchange (EDI), which was developed to reduce manual effort and clerical costs for large organizations, who started and drove the adoption of e-business technologies. Several e-business frameworks have since been developed to support different types of businesses. These frameworks now provide enterprise integration and electronic data exchange facilities within and between businesses. Today, RosettaNet, for example, is one of the e-business frameworks that provide seamless business operations and integration according to machine readable and pre-defined business processes. These modern e-business frameworks require proper resources and expertise from the business organizations involved. However, unlike Multinational Companies (MNCs), most of the Small and Medium Enterprises (SMEs) lack some or all of these resources and are hence incapable of adopting e-business technologies. As a result, a large part of the SME level business organizations are still relying on manual business operations, which causes a big challenge to implement enterprise interoperability at the SME level by harnessing e-business facilities.

Considering these given challenges, the main objective of this thesis work was to study and analyze the available e-business interoperability frameworks, standards and research projects. Then, the study intended to determine feasible solutions to recommend for SMEs so that they could adopt e-business framework and interoperability solutions with less expertise and fewer resources and costs.

The research questions became, thus,

1. What kind of e-business frameworks, techniques and standards are available in the market, and how do they answer to SME's requirements, which include implementability (only few resources needed), cost effective operation, security, and interoperability with other solutions?
2. Do we need to improvements or future research to provide more feasible solutions for SME e-business techniques adoption and interoperability enablement?

The research was conducted as a literature study, which analyzed and evaluated existing e-business frameworks and research projects. The structure of this thesis is the following: chapter 2 and 3 discuss background study of the problem area and the related issues and technologies. Chapter 4 and 5 analyze and compare results among e-business frameworks and research projects. Chapter 6 contains the answers of the research questions together with recommendations and justifications. Finally, chapter 7 summarizes the overall thesis work.
2. BACKGROUND

This background study mainly focuses on the basic concepts of eBusiness, SMEs and interoperability, and how they interconnect. E-Business interoperability in SMEs is the main problem area, so eBusiness frameworks and interoperability support at SME-level is also considered. Today, most of the E-Business interoperability frameworks use XML-based messaging, business documents and business processes, which are therefore also considered in this chapter. Finally, cloud computing, with a lot of future potential to become an effective E-Business solution for many enterprises in the form of software as a service, is studied as another interoperability approach.

2.1 Electronic Business

The specific definition for electronic business (eBusiness) is: “information and communication technology (ICT) is utilized to perform and automate business interactions within and between businesses” (Kotinurmi et al. 2006). Electronic commerce also considered as a part of eBusiness, though eBusiness is more than eCommerce. The idea of eCommerce is buying and selling products using ICT and the big picture of eBusiness activities and interactions are not only buying products from supplier and selling them to customer, but also cover all kinds of interactions and exchange of information with it’s business partners, such as distributing order forecasting information and so on. Moreover, a eBusiness functions, such as online-sales, purchases, demand forecasting, resource management etc. are also simple business functions in which company shares information with its business partners through computer-mediated networks, such as internet.

Though still it is not that simple to automate even simple business interactions between a small numbers of partners. Because they may differ from each other in many ways, for example, differences of using terms, mode of operations and which may cause of error when interaction between those incompatible information systems. However, to overcome this interoperability problem and to increase compatibility between companies, an eBusiness framework can facilitate a shared understanding between
many companies and their units because it brings order into the uncertainty by reducing variety in practice. (Kotinurmi et al. 2006; Nurmilaakso 2008)

2.2 SMEs

Micro, small and medium-sized enterprises are categorized together as SMEs and different kinds of parameters are used to define these kinds of enterprises individually. In addition, the number of staff and the financial condition are considered to categorize the different enterprises.

The European Union Commission has recognized the social and economic importance of SMEs: they represent 99% of enterprises, contribute to entrepreneurship and innovation and provide about 65 million of the jobs in the European Union (EU). Therefore, to avoid the inconsistency of the definition of the different size of enterprises between EU wide and national level, it is necessary to have a common interpretation of definition. The European Commission has defined the definition for SMEs to use in the single market under the European Union. Table 1 represents the data defined by the European Union based on the different sizes of the enterprises definitions. (European Commission 2003)

<table>
<thead>
<tr>
<th>Enterprise Category</th>
<th>Headcount</th>
<th>Turnover</th>
<th>Or</th>
<th>Balance sheet total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Medium</td>
<td>&lt; 250</td>
<td>&lt; or = 50 Million Euro</td>
<td>&lt; or = 43 Million Euro</td>
<td></td>
</tr>
<tr>
<td>Small</td>
<td>&lt; 50</td>
<td>&lt; or = 10 Million Euro</td>
<td>&lt; or = 10 Million Euro</td>
<td></td>
</tr>
<tr>
<td>Micro</td>
<td>&lt; 10</td>
<td>&lt; or = 2 Million Euro</td>
<td>&lt; or = 2 Million Euro</td>
<td></td>
</tr>
</tbody>
</table>

2.3 E-Business Interoperability

Interoperability is defined as the capability within and between business partners to interchange data and collaborate with each other efficiently, accurately and effectively. In the competitive global business area, services are provided to consumer 24 hours in
a day and 7 days in a week through the internet. However, to provide the services, an efficient and reliable interoperability e-business framework is required to ensure the quality of services, cost effective solution and effective business operations. Quality of service represents increasing customer satisfaction and retention, while cost effective solutions represent reducing operating cost and increasing gain. Finally, effective business operations enable the quicker execution of internal processes and improve supply chain integration. (Hoyer et al. 2006; IBIS 2007)

To achieve the interoperability within and between business organizations, it is important to focus on the different information systems which are used in the organizations, such as ERP (Enterprise Resource Planning), SCM (Supply Chain Management), CRM (Customer Relationship Management), PDM (Product Data Management) and related systems which are not yet sharing information between each other. Although data can be exchanged by human intervention, it has possibility of higher costs and lower efficiency. So integration techniques become popular between different information systems which means fully automated and interoperable.

![Figure 1: Model and Message flow in Interoperability System (Guglielmina et al. 2008.)](image-url)
Figure 1 is an abstract model for Interoperability System (IS) adopted from ABILITIES, one of the European Commission funded research project for SMEs interoperability. Above figure is the message flow of the IS where some repositories with necessary data are used at the run time, among them only two are included here: reconciliation rules and negotiation rules repositories. Message Event Handler wait for the incoming messages from the sender, when message arrive first it handle the message and deliver it to synchronization module. Synchronization module is capable to decide which message transformations are needed and then send it to either one or both of Reconciliation engine and Negotiation engine (if require). The Reconciliation engine performs semantic reconciliation, finds and loads semantic reconciliation rules, translates UBL business documents etc. Negotiation engine applies stored negotiation rules and transforms the message. At this point Synchronization engine deliver the message to receiver end Message Event Handler, which then convert the message to its appropriate format and deliver it to the actual receiver. (Guglielmina et al. 2008)

To ensure interoperability between business organizations, heterogeneous business partners need to first understand each other’s existing information systems, as well as what information need to be shared. With this knowledge, they can implement compatible eBusiness frameworks that operate on top of their back-end systems, which consume the exchanged data. (Kotinurmi et al. 2006)

Figure 2: Generalized interoperability framework view
Figure 2 is generalized from figure 1, in figure 2 the synchronization layer communicate one or more business processes to verify and ensure if any changes needed for the message to communicate with the specific business partner. The decision is based on the defined business process for a particular business partner and finally applies business rules from repository to ensure the business document is synchronized between sender and receiver.

### 2.4 Interoperability Support at SMEs Level

In last few years eBusiness concept adoption is tremendously increased around the world. However, still very limited options for the SMEs case to adopt such technology. As a consequence of this situation, too many manual inputs are required in SMEs business operations and transactions, as they can not effort for available costly enterprise solutions; however most of those information system solutions are affordable for the MNCs. As a consequence SMEs business transaction has a very high possibility to generate severe errors.

i2010 (Europe's Information Society 2009) is an important initiative by European Commission which is an investment in R&D and innovation of SMEs interoperability launched in June 2005. A number of research issues funded by the European Commission in enterprise interoperability focused on the SMEs are: ATHENA, TrustCoM, Interop, FUSION, ABILITIES, GENESIS. (Guglielmina et al. 2008)

### 2.5 XML

To manage industrial documents IBM has started to develop General Markup Language (GML) in 1969. After IBM, ANSI recognized the importance of GML from different perspective and for electronic document management they have started to develop Standard General Markup Language (SGML). In 1986 SGML achieved the international standard from ISO. Probably the mostly used of SGML is Hypertext Markup Language (HTML) for the internet publishing of information and still this is one of the important uses of the internet based application development. HTML was
developed at the European Organization for Nuclear Research (CERN) in 1989 and it was the beginning of the World Wide Web (WWW). HTML achieved the standardization form Internet Engineering Taskforce (IETF) in 1995. Although, still HTML has drawbacks for its fixed structure and elements type, World Wide Web Consortium (W3C) realized it for flexibility of electronic document format. In 1996 W3C has taken the initiative to develop a standard to overcome the complexity of SGML/HTML and to achieve the flexibility of defining structure and elements for electronic business documents. As a result, Extensible Markup Language (XML) is developed and in 1997 XML achieved W3C recommendation. In 1998 W3C published XML. (Kotinurmi et al. 2006; Nurmilaakso 2008)

XML was designed to structure, transport and store data, focusing on the meaning of data rather than how data is presented. XML is a platform independent format, where data can be expressed as plain text format with self defined structure and interchange between applications. XML based Application Programming Interfaces (APIs) can parse the information from the self descriptive XML data structure. Figure 3 presents an example of XML format for email information which is just plain text and an application need to be written to use this information to send or receive emails. (W3Schools 2009)

```xml
<email>
  <header>header information</header>
  <from>sender address</from>
  <to>receiver address</to>
  <date>received date</date>
  <subject>email subject</subject>
  <body>email texts</body>
</email>
```

**Figure 3:** Basic XML format for email information.

DTD (Data Type Definition) is a part of XML since the beginning of the XML specification, which is derived from SGML. Though, DTD has comparatively less expressive ability to define advanced elements and structures of XML documents. Therefore W3C recommended more expressive and powerful XML schema in 2001
which has improved capability of DTD for expressing more advanced elements and structures. Since, DTD and XML schema provide different levels of XML document specification, it is important to decide which schemas are needed to be used in e-Business. (Kotinurmi et al. 2006)

2.6 E-Business Frameworks

Since late 1960s EDI (Electronic Data Interchange) has been used for facilitating application to application and business to business integration and exchange of standard business documents within and between companies. When World Wide Web Consortium (W3C) has introduced the Extensible Markup Language (XML) that became more easy to use and many interoperable XML based frameworks has been defined. Before XML standard, EDI was the only widely used standard for e-business.

There are major differences between EDI and XML based e-business frameworks. EDI based e-business frameworks provide document formats specification and how to represent the business documents. EDI does not consider or specify any business process specification. On the other hand, XML based e-business frameworks deal with three basic properties: business documents, business processes and messaging in business. Moreover, it can define the business document specifications, define the business processes using XML schemas and define header information with messaging using XML schemas.

Every e-business framework has a certain scope to provide interoperability support, such as, cross-industry and industry-specific. Cross-industry e-business frameworks provide the advantage to define public business processes, which are those between companies. They can define private business processes which are processes within a company. On the other hand, an industry-specific e-business framework does not provide to define any new meaning for business processes. (Kotinurmi et al. 2006; Nurmilaakso et al. 2008)
2.7 Cloud Computing

According to Grossman (2009), cloud computing does not have a standard definition, but a good working description of it is to say clouds or clusters of distributed computers provide on-demand resources and services over a network, usually the internet, with the scale and reliability of a data center.

The cloud computing concept is based on the internet technology and it provides different levels of services, such as, huge data center and online Operating System (OS) support eg. Amazon’s EC2 (Elastic Compute Cloud) services, provides software as a service eg. Google web based office tools, development APIs as online services for software developers eg. Google map APIs and so on. Amazon was the first company that started cloud computing services widely and commercially; beside that Google also has their internal and external cloud services. It is predicted that next generation computing is based on cloud computing: “just move all processing power to the cloud and walk around with an ultra-light input device with a screen” (Weiss 2007.) The pricing of using cloud computing services are flexible, and it is called usage-based pricing. Therefore, organizations can use the services in the cloud and pay in different ways, as an example, an on demand service where the user pays based on the resources used, which could include the volume of network traffic or the number of active users. So usually business companies prefer the first option because if there is idle state no need to pay for that time. (Weiss 2007; Grossman 2009)

There are two kinds of clouds, one provides computing instances on-demand and another one provides computing capacity on-demand. However, both services belong to the same machine, such that, the first one is scaled out by providing additional instances and the second one supports data or compute intensive applications via scaling capacity.

Companies can develop their own clouds as private and use internally only themselves or they can rent cloud services from third party hosting companies. For example, Google uses GFS (Google File System), MapReduce and BigTable itself as their private clouds; third party hosted clouds are for example Amazon’s EC2, S3 and
SimpleDB which are open to purchase services anytime by any business organization using online credit card payment services. (Grossman 2009)

Figure 4: Combine both public and private cloud models as hybrid clouds (Sun Microsystems 2009)

There are basic two categories of clouds: public and private cloud. Therefore, Public and private cloud in the same space called hybrid clouds which is represented in figure 4. Public clouds are offered by third party where different customer applications run in the same shared server located away from the customers. Beside shared server customer can also use the storage, infrastructure and network from the cloud. Public clouds also reduce customer cost, risk and increase flexibility, scalability, security using enterprise level infrastructure. However, private clouds are owned by the company itself and have control over the data, security and quality of service. Usually
company has own infrastructure where private cloud deployed either in its enterprise data center or at co-location facility. Hybrid clouds are helpful when there are routine tasks and private cloud needs external services provided by public cloud. It works effectively when amount of data is small. However, hybrid clouds are a complex way of determining how applications are distribute across the both public and private cloud. (Sun Microsystems 2009)
2.8 Summary

Electronic Business (E-Business) is introduced to leverage information and communication technology (ICT) and to enhance business activities. Using eBusiness approaches companies are used electronic media to define their business documents and processes and improved the efficiency and accuracy in business data exchange. Therefore, electronic definitions provide ways to integrate businesses leveraging their existing information systems. However, eBusiness concept was introduced over two decades though the adoption in industry is not widely enough and mostly adopted company size is multinational companies (MNCs).

The study clearly determined that eBusiness is the next generation approach to do businesses. However, there are so many ways to implement eBusiness infrastructure for the companies and also it has been determined that most of them are well suited for MNCs and very less options for SMEs. Though, SMEs have a very important role in business space which is reflected in the EU report. (European Commission 2003) Moreover, most of the SMEs have lack in resources and technical know-how to establish such eBusiness infrastructures which are available in present solutions. On the other hand SMEs business goal, strategy and processes are different than the MNCs to continue business and make profit, which means most of the SMEs are still using manual business activities and business artifacts produced by them have a high possibility to create error prone information. This is the present challenge in eBusiness adoption and interoperability between MNCs and their SME partners.

Different literature review also shows that several eBusiness frameworks are exists and among them few are providing solutions to promote SMEs. Therefore, it ensures that SME-level companies are able to adopt eBusiness frameworks with their existing strength in information systems and technical ability.
3. E-BUSINESS INTEROPERABILITY BUILDING BLOCKS

To build an e-business infrastructure and enable them interoperable is required a set of standards, techniques and technologies as its building blocks. The keyword interoperability is raised when the end-to-end (B2Bi) integration is done. For a loosely coupled and easy to scalable business-to-business integration, one of the most feasible and practically proved approaches is Service Oriented Architecture (SOA) as a middleware to apply integration techniques within and between companies. On the other hand, when the integration is done then interoperability should be present there. Interoperability can be achieved using e-business frameworks and it has also several building blocks which are also kind of standards, such as, CCTS, CCL etc. Finally, when the interoperability works well then the security of exchanged messages should be there, for example, XML Encryption, XML Digital Signature, WS-Security etc.

3.1 Business to Business Integration

Integration is one of the most important keyword in current business concept; integration is possible both within the business and between businesses. Therefore, on the one hand, integration within the business usually means Enterprise Application Integration (EAI), on the other hand, integration between businesses means Business-to-Business Integration (B2Bi). Most of the big organizations are now paying attention on the integration with their partners and to enable all the business transactions are electronically. There are several B2Bi techniques, such as, EDIFACT, SOA etc. (Alonso et al. 2004)

3.1.1 B2Bi Approaches

Business-to-Business integrations are achieved through different approaches. There are two general approaches are found, which are as following: global workflow and point-to-point. Therefore, both of these approaches have different ways of integration and lacking in integrated operations.
When the middleware application support is provided by a third party company and other partner companies are integrated through the third party system is called global workflow. According to figure 5, to join in the global workflow customer, supplier and warehouse all have to agree to use the same middleware platform (e.g. a specific message broker, specific workflow system etc.). In industry practice this kind of integration and automation is very rare due to the lack of trust, autonomy and confidentiality reason.

Furthermore, to avoid the global workflow problems an alternative approach can be adopted by each of the partners who maintain the negotiation separately is called point-to-point fashion. So, when two partners decide to integrate, they agree to use certain middleware protocol and infrastructure, as shown in figure 6. Problems arise when number of business partners are increased and require implementing different kind of middleware platform to be integrated. This kind of solution is complicated, less simplified and hard to maintain the system.

**Figure 5:** customer-supplier-warehouse systems integration using global workflow (adapted from Alonso et al. 2004)
Finally, the contemporary B2B integration approach using service oriented architecture (SOA) provides the way to provide functionalities as a service in the web and possible to invoke the services from both inside and outside the company. As a result, it gives the opportunity to implement global workflow, point-to-point and both approaches in B2Bi. Detail discussion about web services in chapter 3.1.3. (Alonso et al. 2004)

### 3.1.2 CCTS & CCL

According to UN/CEFACT the definition of Core Component (CC) is: “*A building block for the creation of a semantically correct and meaningful information exchange package. It contains only the information pieces necessary to describe a specific concept*” (UN/CEFACT 2003.) The Core Component Technical Specification (CCTS) is a set of meta-models and rules to represent the unambiguous definition of business information. The definitions are also syntax-independent. This way UN/CEFACT stack provides guidelines for the users for correctly naming and combining Core Components, therefore user can apply business-specific restrictions based on the generic data templates. This approach introduces the adaptation of generic data templates for context-specific use and restricting them according to the business-specific requirements. So, though the data templates are customized, they are still globally interoperable due to the standardization and uniquely naming convention of attributes.

The Core Component Library (CCL) is a kind of repository which contains the generic business data components or Core Components. Basically CCL repository does not contain static or context-specific data definition, rather then it provides a huge set of
valid data templates which are commonly used in modern business processes (e.g. postal address, personal information etc.). Advantages to use these generic library components are reducing effort to define same kind of data definition redundantly and the common definitions ensure the higher scale interoperability between enterprises. Moreover, companies can define new or modify existing component, if existing components are not sufficient to meet all kind of business-specific requirements and thus the CCL can be scalable and change over time. (Lampathaki et al. 2008)

An example of Core Components according to the Figure 7: Personal .Details and Address .Details are Aggregate Core Components (ACC) that consists of many root level components, such as Name, Street etc. These root level components are property of ACC and these properties are the Basic Core Components (BCC) and their set of allowed values is defined by a Data Type i.e. Text for Name, Date for Birth Date etc. The Association Core Components are Residence and Official Address and their structures are described by Address .Details. Now any user specific address can be defined from the generic template, for example in case of Finland the user can define
Street, Postal Code and Town according to their national postal services. This specified information is then called Business Information Entity (BIE). (UN/CEFACT 2003; Lampathaki et al. 2008)

3.1.3 UBL

The Universal Business Language (UBL) has been approved as W3C recommendation in 1998. UBL is developed by the Organization for Advancement of Structured Information Standards (OASIS) as a royalty free library of standard XML business documents. UBL is designed to provide a universally understood and recognized commercial syntax for legally binding business documents and to operate within a standard business framework such as ISO 15000 (ebXML), to provide a complete, standards-based infrastructure that can extend the benefits of existing EDI systems to businesses of all sizes.

Before UBL, many companies had started adopting XML as the means for defining messages exchanged for electronic commerce, though the wide use of XML resulted in redundant business-specific XML version for basic documents, such as purchase orders, shipping notices etc. Though an industry specific data format is optimized (only to server for a specific type of business) use for the business context but reproducing same document for deferent companies is wasting of time, effort and money. UBL is intended to help and solve these problems by defining a generic XML interchange format for business documents that can be extended to meet the requirements of particular industries. UBL’s purpose is to provide:

- Reusable data components as an XML schema library, such as Address, Item and Payment usually common data items for different business documents.
- A set of business processes and rules attached with the business documents to define a context of their use.

UBL could be defined as a standard for data modeling, though still it has limitations and covers only the basic business documents related to few common B2B processes. UBL v1.0 had eight business documents supported and the latest UBL v2.0 provides
schemas for twenty nine business documents. However UBL still does not have support for transactions between businesses, government or banking institutions. On the other hand, UBL’s customization is relatively rigid. (OASIS 2006; Lampathaki et al. 2008)

### 3.1.4 Web Services

The word web service is a common term used in internet based software service providing area. The primary use of web services is as business communication within and between businesses. Though businesses share information between each other, however, the individual IT information is not allowed to access which is protected behind the firewall or security configurations. Using of web services is becoming widely everyday in world wide web space and this is one of the effective technique for different kind of integration, interoperability and scalability within and between application-application, business-business and so on. These services provide standard XML based messaging communication among the heterogeneous business applications involved in presenting dynamic context-specific user information. The more specific definition for web services according to the W3C is: “A Web service is a software system identified by a URI, whose public interfaces and bindings are defined and described using XML. Its definition can be discovered by other software systems. These systems may then interact with the Web service in a manner prescribed by its definition, using XML based messages conveyed by Internet protocols” (W3C 2004.)

Furthermore, more technical definition for Web Services is: The term Web service provides a standardized way to integrate heterogeneous applications using the XML, SOAP, WSDL and UDDI Open Standards over the web based Internet protocols. Here XML is used to describe the format of data, SOAP is used to transfer data as client-server communication technique, WSDL is used to describe web services which are available and UDDI is used for listing which services are available.

Web Services have changed the traditional web based client-server application concept where web server and web client as Graphical User Interface (GUI) is present; instead,
Web Services provide programmable interfaces through network to access business logics and data. So, basically Web Services provide functionalities for remote GUI client application or services which can also communicate with each other. It reduces the development cost and time because once a service is developed any other application can use that and avoid redundant coding. On the other hand, due to XML based communication, this technique is Operating System (OS) or Programming Language independent, for example Java application can communicate to Microsoft application and Windows application can communicate to Linux application.

**Figure 8:** Web services provide an entry point for accessing local services (Alonso et al. 2004.)

Web services are introduced as an entry point to access local information system. These entry points are defined as sophisticated interfaces to expose local functionalities through web with a controlled manner. Web service interfaces encapsulate the internal system information from remote client and provide functionality and data as a client specific request. This concept is realized in figure 8, where “Company-A” provides several services as web service to access the business functionalities and the core system is encapsulated in behind. “Company-B” can access the services provided by “Company-A” from the client application, though the client do not have any direct access or idea about the underlying internal system of “Company-A”. (Alonso et al. 2004)
3.2 SOA Concepts

Service Oriented Architecture (SOA) definition adopted from IBM is: “SOA is a business-centric IT architectural approach that supports integrating your business as linked, repeatable business tasks, or services” (IBM 2009).

![Figure 9: Service Oriented Architecture, where web services require an internal and external architecture, along with corresponding middleware support (Alonso et al. 2004.)](image)

Figure 9 shows the SOA, how web services address the B2B integration problem and each company exposes its internal operations as (Web) services, which act as the entry point to the local information system. Basically SOA provides an architectural model to enhance the efficiency, agility and productivity of an enterprise with the services and publishing the solution logics (Web services) as a strategic goal on the right place to achieve the service oriented computing support.

From different perspective, SOA is a kind of technology architecture and it can consists different technologies, products, APIs and supporting infrastructure extensions and many other parts of the systems. Therefore, each enterprise which has deployed SOA is always unique than other enterprises. (Alonso et al. 2004; Erl 2008)
3.2.1 SOAP

SOAP provides a standard definition to organize information using XML through a structured way so that information can be exchanged between requester and receiver. More specifically following things are defined:

- Define message format for one way communication standard and describe the information packaging format using XML (into an XML document).
- To implement the Remote Procedure Call (RPC) interaction pattern SOAP provides a set of conventions and defines how a client can invoke remote methods by sending SOAP messages as a request and how services can respond to the request through SOAP messages.
- Entities should follow a set of rules when processing the SOAP message which is defined as XML elements. The entities should take action if they can not recognize contents from the XML document format.
- Describes how SOAP messages should be transported on HTTP and Simple Mail Transfer Protocol (SMTP). Other transport protocols binding are not supported in current version and perhaps will be defined in future.

![Figure 10: A simple implementation of SOAP (Alonso et al. 2004.)](image)
Figure 10 shows an approach of simple implementation of SOAP-based messaging communication between a service provider and service consumer or requestor. This communication is not different than RPC where a client calls a remote procedure like a local call. Practically client invocation is through a proxy procedure call which is implemented in a client side stub and appended with the client in compilation time. Therefore, client stub communicate the request to the SOAP sub-system and the request is transformed to a XML document format according to the SOAP message specification. Though, finally the SOAP message wrap in to HTTP format. When this step is successfully done HTTP send the message to the target remote location. At the remote location a reverse procedure is done: HTTP sends the received message to SOAP sub-system and extracts the XML document (SOAP message) from HTTP format. This extracted message is then sent to the server stub which finally calls the corresponding procedure. The return message again from service provider is as same manner (vice-versa). (Alonso et al. 2004)

3.2.2 WSDL

According to the W3C description of Web Services Description Language (WSDL) is: “WSDL is an XML-based language for describing Web services and how to access them” (W3Schools 2009.) WSDL became a W3C recommendation on 26 June 2007.
Following figure 11 WSDL structure is possible to divide in two major parts: abstract part and concrete part. Abstract part contains four elements which describe as below:

- **Port types**: each port type is a logical collection of related operations which operations are performed by the web service.
- **Operations**: each operation defines a simple exchange of messages.
- **Messages**: message is a logical unit of transmitted data to communicate with the web services to exchange information between peer.
- **Types**: The data types used by the web service into the messages when it is exchanged so that data can be correctly interpreted in both end.

Concrete part contains two pars which describe as below:

- **Interface Bindings**: Binding helps to specify the message encoding and protocol bindings with all the operations and messages that have been defined with a given port type. Operations are categorized, such as, RPC-style and Document-style operations. RPC-style operation defines the input and output parameters of the corresponding procedure call, on the other hand, Document-style operation defines input and output documents which are agreed between participants. Messages belongs to an operation must follow the SOAP protocol to communicate using the HTTP transport bindings. Currently available transport bindings options are only HTTP and SMTP. Interface Bindings also gives the option to specify the message encoding rules which are used in the XML message.
- **Services and Ports**: Ports are basically the end points which combine the Interface Bindings with a network address and specify as URI. Through the specified network address the port can be accessed. Services are addressed as a logical group of ports. In principle, WSDL service can be located in different web address, such as, one in Europe and another in Asia and also combine variety of port types. Though, in practice WSDL group in the same port and available from the same internet address.

A real life example of WSDL is represented in figure 10 as XML document according to the WSDL specification in figure 9.
In this figure 12 example the `<portType>` element defines "glossaryTerms" as the name of a port, and "getTerm" as the name of an operation. The "getTerm" operation has an **input message** called "getTermRequest" and an **output message** is called "getTermResponse". The `<message>` element defines the **parts** of each message and the associated data types. Compared to traditional programming, glossaryTerms is a function library, "getTerm" is a function with "getTermRequest" as the input parameter, and getTermResponse as the return parameter. (Alonso et al. 2004; W3Schools 2009)

### 3.2.3 UDDI

The basic definition of Universal Descriptor Discovery and Integration (UDDI) is: “**UDDI is a platform-independent framework for describing services, discovering businesses, and integrating business services by using the Internet**” (W3Schools 2009.) Moreover, UDDI is a kind of directory where store information about web services and describes web service interfaces using WSDL. Communication with UDDI is using SOAP (also known as XML protocol messaging specifications) which provides cross platform programming features.

UDDI is an open industry initiative enabling businesses to discover each other and define how they interact over the Internet. At first UDDI 1.0 was proposed by Ariba,
Microsoft and IBM in September 2000. After the first specification released UDDI becomes a combined support of about 300 companies including the leading companies, for example Dell, HP, Intel, IBM, Sun, Oracle, Microsoft and so on. Version 3 was released on July 2002 and after that the project was handed over to OASIS. UDDI uses World Wide Web Consortium (W3C) and Internet Engineering Task Force (IETF) standards.

3.3 XML and Web Services Security

Security has become a very important topic for web services because of its wide implication in enterprise solution area. Moreover, XML is the common messaging technique in web services and also XML-based eBusiness frameworks implementation. XML provides the platform independent messaging approach and this is the reason that secured XML messaging can provide a wide range of eBusiness security. There is some traditional security technologies are found, such as, SSL, TLS etc. and these technologies provide only point-to-point security and do not consider end-to-end security. Point-to-Point security provides only messages are secured when transmit from one point to another point, which means transport level security but no message level security. So, when message arrives to the end-user there is no offline message security using traditional security approaches. However, there are some end-to-end web service security approaches to address the issues, such as, access control, authentication, data integrity, privacy etc. and figure 13 shows the idea according to the above explanation.

![Figure 13: Point-to-Point and End-to-End XML message security](image-url)
Figure 14 shows the core specifications for XML and web service security. It is a standard framework for XML based messaging applications. Therefore, Simple Object Access Protocol (SOAP) is used to transport messages. XML-Encryption and XML Digital Signature ensure message confidentiality and integrity. Security Assertion Markup Language (SAML) provides authentication assertion; XML Access Control Markup Language (XACML) focuses on information access control; XML Key Management Specification (XKMS) is to manage public key infrastructure. Finally, Web service security brings those above standards altogether.

**Figure 14:** XML Security Standards (Sun et al. 2008.)

**XML Digital Signature**

The XML Signature specification is a joint effort of World Wide Web Consortium (W3C) and Internet Engineering Task Force (IETF). It has been designed to apply for transaction of XML document and also non-XML data. This signature type supports
data integrity, message authentication and non-repudiation to the data which it has signed.

**XML Encryption**

XML Encryption specification is defined by W3C which ensures the issue of data confidentiality using the encryption techniques. Encryption can be applied in both part of XML document and non-XML data. Other approaches for confidentiality, such as, SSL which supports only transaction time confidentiality; though XML encryption is also possible to apply to the persistent storage. Encrypted data is wrapped inside XML format according to the specification.

**XKMS (XML Key Management Specification)**

XKMS is a W3C recommendation and a joint effort by W3C and IETF. Key management specification is an essential part for other security specifications, such as, XML-encryption and XML digital signature. XKMS defines protocols between the XKMS client and server for seamless Public Key Infrastructure (PKI) operations. Therefore, XKMS provides the XML messaging format to manage the public keys which ensures public key registration, public key validation, public key discovery and public key revocation. XKMS specification is also possible to attach with XML digital signature and XML encryption to manage the public key enabling signature verification and encrypting for recipients.

XKMS comprises following two parts:

XML Key Information Service Specification (X-KISS)
- The X-KISS specification defines a protocol that resolves public key information contained in XML Signature (XML-SIG) as a trusted service.

XML Key Registration Service Specification (X-KRSS)
- The X-KRSS specification defines a protocol for a web service that accepts registration of public key information. Therefore, once the public key is registered it may be used in conjunction with other web services including X-KISS.
**XACML (Extensible Access Control Markup Language)**

XACML is an XML specification using well defined information access rules and policies in any XML document or any other electronic information. The access control in XACML defines in 4-tuples, which are subject, target objects, permitted action, provision. For example, an access request: “Allow the company marketing manager to create files in the Sales folder on the Marketing server”; here the subject is the “Company marketing manager”, the target object is the “Sales folder on the Marketing server” and permitted action is the “create files”. This access control request can be defined using XACML standardized in XML.

**SAML (Secure Assertion Markup Language)**

SAML defines XML-based framework for communicating user authentication and authorization information. SAML provides the mechanism that once the authentication is done; it provides the authentication and authorization information between interconnected entities. Therefore, it supports to share the security information in Single sign-on (SSO) with different systems and environments. For example, if user logged-in (authenticated) to the document management system, the same authentication information can be used in version management system.

**WS-Security (Web Services Security)**

IBM and Microsoft have provided a specification to enhance WS-Security which bring different security specifications altogether in mid of 2002. It increases the security possibilities which mainly focused on the SOAP messaging. The goal is to provide to create web service applications enabling extensive security in SOAP message exchange. The definition can be specifically presented the following way: “WS-Security describes enhancements to SOAP messaging to provide quality of protection through message integrity, message confidentiality, and single message authentication” (Atkinson et al. 2002.) Therefore, WS-Security concept can be used to define variety of security models and encryption techniques extensively.
3.4 Summary

This chapter provided an overall idea about the building blocks of business-to-business integration and their interoperability. The building blocks considered the core of B2Bi approaches and technologies which are used in major eBusiness framework construction.

Business-to-business integration approaches are focused on three traditional ways, as an overview of integration concept and then continued through the contemporary integration ideas. Three traditional integration ways are addressed here, such as, manual, global workflow and point-to-point. Therefore, all of these integration approaches have major drawbacks in industry practice due to the lacking in automation, simplicity and scalability. So, finally SOA is described as a contemporary approach which brings all traditional ways altogether into the same solution technique to solve the integration problems. Moreover, other technologies are also introduced, such as, CCTS and CCL which have an importance to build eBusiness frameworks, like ebXML, which are discussed detail in later chapter.

Finally, detail discussions are covered on Web services building techniques, properties and their security-level. As a widely adopted technique Web services security issues are very important due to the confidential information exchange between business partners and online based transaction. Reviewed techniques are covered several level of Web service message security, such as, digital signature, encryption, authentication and public key management etc.
4. E-BUSINESS INTEROPERABILITY FRAMEWORKS AND TECHNOLOGIES

E-Business interoperability frameworks are divided into two categories, EDI and XML based. Practically using of these frameworks are in following ways, for example, first, a standard developing organization (SDO) develops an e-business framework either based on EDI-standard or XML-standard data format. Then, companies use information systems which support this e-business framework. Different categories of e-business frameworks always have a specific scope, either industry-specific or cross-industry supported, which are discussed more in chapter 2.6.

4.1 EDI-based E-Business Frameworks

In mid of 1960s, railway companies in United States realized to standardization of their business documents and to improve the quality of inter-company information exchange. So, in 1968, they formed an organization named Transportation Data Coordinating Committee (TDCC) to study the problem area and develop a standard which was resulted the Electronic Data Interchange (EDI) standard. EDI-standard first published in 1975 covering air, motor, railway and few banking applications. (EDI-Guide 2009)

Furthermore, as a continuation of TDCC standardization it received American National Standard Institute (ANSI) standard and the name is changed to ANSI X12 committee in 1975. As a result, Accredited Standards Committee (ASC) X12 is published in 1981, which was gradually extended and improved from the initial EDI format defined by TDCC. However, the ASC X12 was concerned to develop national (in U.S.) wide EDI standards. According to Adam et al. (1998): ”The ASC X12 standard has over 90 transaction sets that support the EDI needs of the U.S. Government, education and specific industries, such as, insurance and banking. Over 100 additional standards and guidelines are currently in development”. The ASC X12 is responsible to process and approve of any request to change or new on the EDI format (in U.S.), for example, to add a new field in purchase order message.
United Nations Economic Commission for Europe (UN/ECE) took an initiative, in 1985, to develop world-wide EDI standard. Finally, developed Electronic Data Interchange for Administration, Commerce and Transport (EDIFACT) and was approved by International Organization for Standardization (ISO) in 1987, as an international standard. However, EDIFACT and ASC X12 are very similar rules to define EDI-message. EDIFACT based EDI standards mainly provide the frameworks, how to format an EDI message. (Adam et al. 1998; Korhonen et al. 2003; Nurmilaakso 2007; EDI-Guide 2009)

Furthermore, on the one hand, ASC X12 and EDIFACT are data formats and on the other hand, EDI-based cross-industry e-business frameworks. In 1992, ANSI declared that ASC X12 development would be stopped by 1997, even so, many companies in North America who already invested for ASC X12 were not interested to switch to EDIFACT. Already, it was clear that, for both, the future is uncertain. Finally, most of the current EDI-based e-business frameworks are developed based on the ASC X12 or EDIFACT and that is the cause which motivates in this analysis to mainly focus on these two base e-business frameworks. (Nurmilaakso 2007)

EDI has a significant contribution in business transaction automation, though, there are some drawbacks are identified concerning the less flexibility to change any EDI message format. For the given change, need to send a request to ANSI ASC X12 committee, if it is approved and then, it is mandatory to update the EDI transaction software accordingly.

EDI (ASC X12, EDIFACT) standards have a set of components used to define the message format. The major components are:

1. **Data element:** A data element is a particular attribute in an EDI message, for example, issue date, quantity of products etc. There is a reference number to identify an individual data element. Additional information is specified in the dictionary: these include the title, description, type, number and minimum/maximum length of the data element.
2. **Data Segment**: According to the ASC X12 format, each line in the message is considered as a segment and each of the items in the segment is considered as an element. For example, the purchase order line item segment constructs with the following elements: a part number, a part description, a quantity, a unit of measure and an item cost. Every segment has an identifier, a data element delimiter, element diagrams, data segment terminator and a note.

3. **Transaction set**: A specific business document (EDI message) stands for a transaction set; for example, a purchase order. A transaction set has different areas: header area, detail area and summary area.

4. **Functional group**: A group of similar transaction sets are together considered a functional group. In a functional group, all transaction sets are identified with the same functional identifier.

To exchange EDI-based messages (business information) between partners, four basic steps are performed for a single transaction (sender-to-receiver):

I) Mapping the data elements in the database for individual transaction type, for example, purchase order.

II) Extraction of the predefined mapped data elements from the database for a specific transaction type, such as, purchase order.

III) Translation of the extracted data in EDI-format, which is now ready for the transmission.

IV) Finally, transmit the EDI-formatted data to the receiver network address using the predefined communication protocol. (Adam et al. 1998)
The given basic four steps in EDI-based message transaction are shown in figure 15, where the initial step, mapping process, collects elements from the database and mapped among them which require preparing the EDI formatted message. This process is not performed in EDI-software, rather, it is done only once when a new EDI transaction type is added in the database. In second step, extraction process collects predefined mapped data from the database and generally restructured the data in-to a flat file. This new structure must be followed according to the translation application conventions. Now, in third step, before transmission translation process prepares the exact EDI-formatted message from the flat file following the EDI standards. In final step, transmission is completed by the communication software using available internet or network connection. When the receiver end EDI-software receives the message, similar EDI standard rules and data dictionary are used to convert the standard message format to receiver internal format. (Adam et al. 1998)
4.2 XML-based E-Business Frameworks

XML-based E-Business frameworks were introduced in the 1990s. Here different kinds of e-business frameworks are considered. Of those RosettaNet is industry-specific and ebXML, OAGIS and UBL are cross-industry frameworks. According to Nurmiilaakso et al. (2004) RosettaNet and ebXML receive the most attention now, whereas OAGIS can be considered a pioneer in the field and UBL a newcomer.

4.2.1 RosettaNet

RosettaNet was formed in 1998 by 40 leading high technology industry companies, including Microsoft, Intel and SAP as a non-profit business consortium. After that many other same level companies joined like Nokia. It now has more than 500 members. It has mainly focused on electronic components and consumer electronic industries, but has later extended to semiconductor manufacturing, telecommunications and logistic industries who can use internet to exchange electronic business documents between each other. Though from the beginning it was mostly solution for big industry companies; recently RosettaNet has introduced RosettaNet Automated Enablement (RAE) as a new approach for small and medium enterprises (SMEs) level partners to enable them inter-operable in e-business. (Alonso et al. 2004; Nurmiilaakso et al. 2005; RosettaNet 2009)

RosettaNet has defined different standards which provide to seamless e-business transactions between business partners. These standards main attention lies in three areas: Business Processes, Data Format and Messaging Services.

The most important part of RosettaNet is business process specifications which are called Partner Interface Processes (PIPs). A PIP specifies the business documents and vocabulary to achieve the specific business goal. Now, RosettaNet has over one hundred PIPs and business documents supported. RosettaNet has defined four-steps methodologies to create its PIP (detail in Alonso et al. 2004) and when RosettaNet collects new PIP's they published them in a PIP directory. PIPs are defined with XML
Data Type Definition (DTD). A PIP document exchange sequence is defined in two categories: Business Action and Business Signal. Business action is to send a message which includes a PIP business document (e.g. purchase order) as an attachment. Business signal is then an acknowledgement message which can be positive or negative based on the receipt of a business action document. Figure 17 shows simple PIP interactions with the business roles, messages and their sequence of exchange. (Alonso et al. 2004; RosettaNet 2002)

![Figure 16: PIP Message Sequence Diagram (RosettaNet 2002.)](image)

Companies develop their own terminologies and reference codes in day-to-day business and this is one of the challenging parts in B2B integration to provide common vocabularies for all partners. To solve this, RosettaNet specifies a common dictionary for all trading partners and in addition to dictionaries that set common properties for PIPs as well as product and partner codes. (Alonso et al. 2004)

RosettaNet provides messaging support using RosettaNet Implementation Framework (RNIF) infrastructure to transport PIP messaging. RNIF addresses three main areas: First, it defines the standardized RosettaNet business documents, packaging protocol independent payload documents, header components and digital signatures. Second, it defines protocol stack for a transport-independent messaging service, which supports Hypertext Transport Protocol (HTTP) and Simple Message Transport Protocol.
RNIF defines an “asynchronous single-action PIP activity” where a single business action message is sent from Partner-A to Partner-B and then Partner-A receives an acknowledgement from Partner-B or error message as a business signal. Finally, RNIF ensures the security of PIP action messages using digital-signature and/or encryption which takes into account authentication, authorization, encryption and non-repudiation (a proof that server has received a message from a certain client). (Alonso et al. 2004; Nurmilaakso et al. 2005)

Despite the importance of RosettaNet standards, many trading partners today cannot manage the cost or complexity of implementing RosettaNet-based B2B integration technologies. Among the challenges faced by smaller suppliers are a lack of information technology staff, network infrastructure, the high-cost and complexity of traditional B2B integration technology, unreliable Internet connectivity and the requirement to deal with many different data formats. These challenges make the cost of integrating a small supplier two to five times the cost of integrating a larger, more technically-capable trading partner. The result is that many RosettaNet-based implementations reach only the largest 10-20% of a company’s suppliers, leaving the remaining 80% behind -- usually small and mid-sized suppliers. (Schenecker 2005)

in order for business partners to integrate by Rosettanet PIPs, they will first have to define the PIP's used. This sounds simple enough, but actually involves a considerable amount of manual work and requires certain level of expertise on RosettaNet. Also, the PIP's need to be mapped to the back-end systems. These are some of the factors affecting RosettaNet adoption at the SMEs level (Pesonen 2009).

4.2.2 ebXML

UN/CEFACT and OASIS jointly started Electronic business using eXtensible Markup Language (ebXML) in 1999 to define a set of e-business specifications and released the first version of the specification in 2001. ebXML does not define any certain business processes but provides an XML-based tool called Business Process
Specification Schema (BPSS) to specify business processes. Therefore, ebXML does not specify any business document but provides Core Component Technical Specification (CCTS) (chapter 3.1.2) to specify business documents. OASIS continues ebXML development and develops specifications for messaging, registries for storing and looking at business processes and business documents, as well as automated business partner discovery and agreements. Specifications for messaging, registries and automatic business partner discovery and agreements were approved by ISO in 2004. (Nurmilaakso et al. 2005; ebXML 2009)

The main components of ebXML are: CCTS, which defines business documents and communicates data in common terms; ebXML Registry Services (ebRS) and Registry Information Model (ebRIM), to register and provide e-business artifacts and services; Collaboration Protocol Profile/Agreements (CPP/CPA), to configure technical contracts between business partners; ebXML Messaging Service (ebMS), which provides secure and reliable communication; and, ebXML Business Process Specification Schema (ebBPSS), which enables to define business processes. (OASIS ebXML Joint Committee for OASIS 2006)

![Figure 17: A high level overview of conducting e-business using ebXML between two partners (Grangard et al. 2001.)](image)
Figure 17 depicts a high level view of ebXML basic architectural components and the scenario of two trading partners first configuring and then carrying out simple e-business transactions. It shows the sequences of how a non-ebXML compliant company develops its own ebXML compliant application and then does an agreement with an existing ebXML compliant company through the internet. Here, in steps 1 and 2, “company A” is a non-ebXML compliant company which recently became interested about ebXML and checked the configuration from “ebXML Registry” over the internet. After that in step 3 it decides to develop its own ebXML compliant application, although developing this application is not mandatory because commercial tolls are available in the market (e.g. shrink-wrapped solution). Therefore, “company A” submits the Company Profile in “ebXML Registry” at the following step 4. This company profile includes implementation details, reference links, capabilities, constraints and supported business scenarios (e.g. a sales tax calculation). In steps 5 and 6, “company B” finds the business scenarios of “company A” in “ebXML Registry” and sends a request to “company A” that they would like to do an ebXML business engagement on a particular business scenario. Before the final agreement “company B” submits a proposal to the “company A” ebXML compliant application interface which includes the scenarios, security and specific agreements in step 7. Finally, if “company A” accepts the proposal, they are now ready for e-business transactions using ebXML as the step 8. (Grangard et al. 2001)

The completeness of ebXML compared to traditional Service Oriented Architecture (SOA) (chapter 3.2) as a middleware technology is also significant. If one company wants to publish its services using an ebXML registry, it is similar to an UDDI (chapter 3.2.3) registry but provides more detailed information about the services. So, in present practice, UDDI and ebXML registries are implement jointly so that a service may first be looked up by UDDI, with additional information available in an ebXML registry. Furthermore, the notion of a Business Scenario in ebXML is similar to the notion of WSDL (3.2.2) but the Business Scenario contains more detailed information, for example particularly coordination protocol information. ebXML uses its own messaging service called ebMS, though it is build upon the Hypertext Transport Protocol (HTTP) and the Simple Object Access Protocol (SOAP) (chapter
3.2.1) but it supports more than SOAP e.g. it supports the inclusion of attachments. (Alonso et al. 2004)

4.2.3 OAGIS

OAG (Open Application Group) and its’ partners like PeopleSoft and SAP started to develop OAG Integration Specification (OAGIS) in 1995 to define an own data format. At first the XML-based OAGIS was released in 1998. Currently it has more than 40 members, such as, IBM and webMethods. OAG has also defined instructions for guiding, how to use RosettaNet RNIF (RosettaNet Implementation Framework) (chapter 4.2.1) in messaging and ebXML BPSS (Business Process Specification Schema) (chapter 4.2.2) in business processes with OAGIS.

OAGIS describes several business scenarios for the business organizations who can identify the most closely matches scenario according to the needs to initiate the B2B integration process. Therefore, when integration is possible then business documents are exchanged in business processes and it is also possible to identify the require business documents from the OAGIS predefined business documents. Predefined business documents by OAGIS are called Business Object Document (BOD). They have defined 434 BODs in their latest release 9.0 according to the different business needs.

Furthermore, BOD message architecture (Figure 18) is defined as a communication protocol independent. As a result, different kind of communication protocols can be used to exchange BOD, simple protocols, such as, HTTP and SMTP, also some other complex protocols, such as, SOAP and ebXML. BOD messages are defined using the standard XML schema definition. To develop the BOD stack OAGIS has used UN/CEFACT Core Components (CC) (chapter 3.1.2), it includes CCTS, ACC, BIE/ABIE and NDR (OASIS 2009).

OAGIS BOD architecture shown in figure 18 describes major two parts exist in a BOD are: Application Area and Data Area. Application area contains information about the
sender and its environment where the message is created, which can be used by the infrastructure of receiver-end to communicate the message. Information which is provided by the Application Area are about the Sender, CreationDateTime, Signature, BODId and UserArea. Data Area contains the business specific data to communicate with the receiver. It has two major areas: Verb and Noun. Verb carries the action to perform on the Noun of a BOD, for example, Cancel, Process, Add and Synchronize. Noun is the business specific data which is to be communicated. For example, if the BOD is to process a particular purchase order, then the BOD is “ProcessPurchaseOrder”, Verb is “Process” and Noun is “PurchaseOrder”. Furthermore, Noun contains components which are complex building blocks and extensible. Components can contain other Components, Compounds and Fields. For example, Components include: PurchaseOrder Header, Party and Address. Compounds are low level elements and logical group of fields, for example, it contains Amount, Quantity and DateTime. Finally, Fields are the lowest level elements of a Noun and it is possible either based on OAGIS-defined or user-defined type. OAGIS includes predefined 77 Nouns and 17 Verbs in the release 9.0.

![OAGIS-BOD Architecture](image)

**Figure 18:** A high-level OAGIS BOD Architecture (Adapted from OAGi 2009)

There are massive similarities and overlapping in definitions between OAGIS and UBL, for example, reusable Party components naming in UBL and OAGIS are ID-Id
(very less difference), Address-Location (logically same), Name-Name (exactly same). So, there is a question that if both standardizations (UBL and OAGIS) provide very similar e-business definitions, then why not emerging them in one standard. However, UBL is not transaction oriented and OAGIS is. OAGIS and UBL both are standardization of UN/CEFACT CC. (Abrams et al. 2004; Nurmilaakso et al. 2005; Rowell 2005; OAGi 2009)

4.3 Comparison between EDI-based and XML-based eBusiness Frameworks

The XML-based standards are now a challenge for EDI-based standards in both industry-specific and cross-industry e-business frameworks area. Though, EDI-based e-business frameworks were introduced two decades before than XML-based e-business frameworks and could not make a barrier for XML-based standards. In 2007 an important survey report by Nurmilaakso gives an idea of adoption of these two standards in European market: “12.4% of the European companies used EDI-based and 12.0% XML-based e-business frameworks”. The report shows how fast XML-based frameworks are taking attention and growing in industry adoption.

The number of XML-based industry-specific and cross-industry document frameworks is larger than EDI-based frameworks. As a result, ANSI and UNECE (who took initiatives for ASC X12 and EDIFACT standards) were also interested to develop XML-based standardization. In 2001, ANSI and UNECE were jointly participated in ebXML CCTS development, which provided to define business documents as XML format.

Furthermore, only XML-based standardization provides cross-industry-process e-business frameworks. This approach provides to automate both, of exchanging business documents and to execute the whole business processes accordingly. However, these standards are new in market and accepted by existing frameworks strongly, such as, ebXML Business Process Specification Schema (BPSS) is included by OAGIS and RosettaNet and provide guidelines how to define their business process by using BPSS.
Therefore, in old (after 1980s) economy market EDI-based e-business frameworks are still dominating, though, in new (after 1995) economy market is mostly dominated by XML-based e-business frameworks. So, from mid of 1990s XML-based e-business framework adoption has increased gradually and tremendously compare to EDI-based e-business frameworks. (NurmiLaakso 2007)
4.4 Summary

The basic idea of e-business frameworks are divided into two categories, EDI-based and XML-based e-business frameworks. Basically at the beginning when e-business concept was introduced it was called Electronic Data Interchange (EDI) and based on the EDI concept several e-business standardization and frameworks were defined, such as, ASC X12, EDIFACT etc. Therefore, these frameworks are called EDI-based e-business frameworks. After two decades of EDI, when Extensible Markup Language (XML) was first introduced, a new paradigm was open for exchanging electronic messages which are well defined and platform independent. Then XML-based data formats were used to define e-business frameworks which are called XML-based e-business frameworks, such as, UBL, ebXML etc.

The general concept of e-business frameworks is same but the approaches and strategies are different in different business context. Even available e-business frameworks are mostly fit for some different kind of businesses but there is no single framework which supports all kind of businesses as a generic manner.

The basic statistics of using e-business frameworks between the EDI and XML-based frameworks, in old economy market EDI has strong position but most of the new economy markets have adopted XML-based e-business frameworks. Therefore, few differences between EDI and XML-based frameworks are, some XML-based frameworks provide specification to define business processes as XML-schema called cross-industry-process e-business framework, whereas EDI-based frameworks do not support any business-process definition rather support only business-document specification. The similarities are also found that both of the e-business frameworks require for back-end integration in both end of trading partners and most of the MNCs are adopted this solutions compare to SME-level partners.
5. SMEs LEVEL ADOPTABLE E-BUSINESS CASE SOLUTIONS

The study and analysis of e-business frameworks show that the standardization of existing e-business frameworks are mostly fit for the MNC-level companies. However, the main objective of this thesis is to find such an e-business solution which is also feasible to adopt for SME-level business partners. The consequence is, few other e-business solutions have been analyzed which are developed as an extension of the existing e-business frameworks and mainly focused on SME-level adoptable issues, such as, RosettaNet/RAE (chapter 5.1). Furthermore, other e-business solutions which are known as e-business architecture and developed for SME-level users are also considered in this thesis study scope, such as, GENESIS and eYellowpages. Nevertheless many other similar projects are going on to develop e-business architecture, for example, ATHENA, ABILITIES etc. Though in this review two of those e-business architectures have been selected based on the availablability of e-business architecture project related articles and supports.

5.1 Frameworks

In chapter 4, generic e-business frameworks are reviewed which are covered both EDI-based and XML-based e-business standardizations. EDI-based e-business frameworks review covers: ASC X12 and EDIFACT; XML-based e-business frameworks review covers: RosettaNet, ebXML, OAGIS and UBL. All of these reviewed frameworks have different importance from different context. As particular, different type of frameworks have different approaches to define business processes, business documents and messaging (communication protocols) as their feature. Therefore, specifying business processes and to provide cross-industry-process is an important feature in XML-based e-business frameworks. On the other hand, EDI-based e-business frameworks have strong position to provide cross-industry-document support. Additionally, one of the XML-based frameworks is reviewed here which does not aim for a cross industry support (applicable in all industries), rather support one or few specific industries (i.e. single/multi-industry in table 2).
All of these given e-business frameworks are also possible to categorize according to their developers. E-business frameworks are developed by committees and markets as drivers, which act as alternatives to defining the standards. These drivers can be involved in different perspectives to develop or define the e-business standards, like, driver only defines the standard for target companies and involved as a neutral (developer is not the user itself), driver is the end-user of the e-business standard and involved as a user-intensive, driver is the vendor and consultant as well as involved as a vendor-intensive. Though, in some cases driver can be both end-user and vendor of their own developed e-business framework. Committee or institute works to define open standards, such as, American National Standard Institute (ANSI) defines ASC X12 and its involvement is as neutral. Often companies do standardization work independently is called market or business-driven approach, it is not considered in this e-business framework review. Finally, the hybrid approach is usually one more companies form a consortium to define e-business standard, such as, Microsoft, Intel and SAP started as a consortium to develop the e-business framework RosettaNet. In this case developer’s involvement can be as user-intensive or vendor-intensive or both. Table 2 collects these differences between frameworks. (Kotinurmi et al. 2003; Nurmilaakso 2007)

Table 2: Categorization of e-business frameworks is adapted from Kotinurmi et al. (2003) and Nurmilaakso (2007)

<table>
<thead>
<tr>
<th>Framework</th>
<th>Based on Standardization</th>
<th>Developer Type</th>
<th>Developer Involvement</th>
<th>Content</th>
<th>Compatibility</th>
</tr>
</thead>
<tbody>
<tr>
<td>ebXML</td>
<td>XML-based</td>
<td>Institution</td>
<td>Neutral</td>
<td>No predefined Processes or Documents.</td>
<td>Cross-industry</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Predefined Messaging System.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OAGIS</td>
<td>XML-based</td>
<td>Consortium</td>
<td>Vendor-intensive</td>
<td>Predefined Documents.</td>
<td>Cross-industry</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>No predefined Processes or Messaging System.</td>
<td></td>
</tr>
<tr>
<td>RosettaNet</td>
<td>XML-based</td>
<td>Consortium</td>
<td>User-intensive</td>
<td>Predefined Processes,</td>
<td>Single-industry</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
5.2 RosettaNet/RAE

RosettaNet Implementation Framework (RNIF) (discussed in chapter 4.2.1) requires complex implementation skill which is the main barrier for SME-level business partners. This is due to the limited use of Information and Communication Technology (ICT) and lack of Information System (IS) applications in SMEs, making them unable to support the RNIF for complete business-to-business automation. In 2003, RosettaNet Automated Enablement (RAE) program was formed to address these challenges. The main objectives of the program were how to reduce time and cost to produce a new Partner Interface Process (PIP) (chapter 4.2.1) by automating the communication and implementation of the Trading Partner Implementation Requirement (TPIR). RAE provides basic-level and standard-based e-business enablement, so that SME partners can start e-business enablement with the RAE. Therefore, SMEs can gradually improve their IT infrastructure based on RAE for more or full B2B process automation if they need. (Cox 2009)
In practice, RNIF works such that if business partners do not have RosettaNet backend systems, then MNCs often create web portals for individual partners as human readable partners' interfaces and SMEs are forced to input data from the web interfaces. The consequences of this process are time consuming and increase implementation costs for MNCs. To overcome these barriers, RAE provides TPIR Partner Interface Process (TPIR-PIP) and TPIR Presentation Format (TPIR-PF) which eliminates these implementation costs and effort significantly. RAE defines the PIP according to the individual TPIR which is machine readable and trading partners can view the defined PIPs as human readable formats using TPIR-PF approach as PDF (the core PDF format is also free of intellectual property constraints and royalty-free) format. So, it also eliminates the tool costs for PDF-based document formats to read or write the document. (Schenecker 2005)

**TPIR-PIP (Trading Partner Implementation Requirements-Partner Interface Process)**

Provides partners to use commercial XML editing tools to define customized PIPs by means of define TPIR-PIPs more concisely and specifically in a machine readable XML-schema (XSD) form. It replaces today's almost all manually generated process definition activities by Data Type Definition (DTD). The TPIR-PIP creation process is shown in figure 19. (Schenecker 2005)

*Figure 19: TPIR-PIP creation process (RosettaNet 2009.)*
**TPIR-PF (Trading Partner Implementation Requirements-Presentation Format)**

It is a standard forms-based machine-to-human readable presentation format which is based on Adobe’s Portable Document Format (PDF) and XML Data Package (XDP) formats. TPIR-PF automates the exchange of PIPs between MNCs and their SME partners. So, the MNCs proprietary web-based RosettaNet portals are replaced by standard-based alternative of TPIR-PF which reduces the implementation costs and time compare to RNIF. The details of the TPIR-PF creation process are shown in figure 20. (Schenecker 2005)

![TPIR-PF creation process](image)

**Figure 20:** TPIR-PF creation process (RosettaNet 2009.)
Figure 21: RAE architecture flow where PIP (XML-schema) turns to human-readable form (as PDF) in Partner end and form data filled and transformed back in to machine-readable PIP (Schenecker 2005.)

Figure 21 illustrates the straightforward flow of RAE activities: at first MNCs integration hub creates a new PIP and then merges it with the TPIR-PF template form. The template is forwarded to the SME-partner and then SME-partner fills up the form and sends it back to the MNC. Therefore, the process extracts data from the form and generates a machine-readable PIP. Finally the MNC's back-end system receives it and reads the data from the PIP. As discussed earlier, one of the key points of RAE architecture is that it ensures very lower technical burden on the SME-level trading partners. SME end-users can send back the filled up electronic form instead of having to manage any back-end system integration. In addition, rather than deploying a full RosettaNet solution, the SME-level trading partners can integrate into an RAE-enabled process by deploying a much simpler and lower cost Web services-based gateway. Furthermore, the gateway provides the asynchronous event management and system alerts necessary to exchange the TPIR-PF forms. The infrastructure to implement this architecture established as a centralized manner at the MNC or solution provider (Application Service Provider) location. This centralized based architecture then provides different services, such as, directory services, security, store and forward messaging, workflow management, protocol binding, test management and data transformation capabilities necessary to create and consume RAE-based PIPs and guarantee the quality of service levels required for robust B2B integration. (Schenecker 2005)

According to Cartwright (2006) the benefits to implement RAE are, in short:

1. Easy to use process integration alternatives.
2. Electronic forms are not dependent on back-end system integration.
3. Cost and effort affordable for SMEs.
4. Best utilization of current B2B investments by MNCs, as they can extend to SMEs level partners.

5.3 GENESIS

GENESIS is one of the EU funded research project, the exact title of the project is:” GENESIS: Enterprise Application Interoperability via Internet-Integration for SMEs, Governmental Organizations, and Intermediaries in the New European Union” (Kuehn 2009.) Research and development (R&D) partners are BOC-Austria, SAP-Germany, Insiel-Italy, Singular-Greece and Logo-Turkey. The initiative was to develop an e-business architecture for various integrations (e.g. B2B, B2G, B2I etc.) and seamless information interchange between different type and size of business partners. Basically this R&D project is to develop a pilot application which includes necessary methodologies, infrastructure and components that allows specially European SMEs to conduct business transactions over the internet. (Kuehn 2009)

Furthermore, the aim is to design and deploy an e-business platform which provides to overcome the entry barriers for the SMEs level companies in Eastern Europe, such as, Romania, Bulgaria and Lithuania. GENESIS architecture is developed based on different existing e-business standards, such as, Business Process Modeling Notation (BPMN) (BPMI 2004), Core Component Technical Specification (CCTS) (chapter 3.1.2), Universal Business Language (UBL) (chapter 3.1.3) and Business Process Execution Language (BPEL) (Andrews et al. 2003). GENESIS was aimed to ensure cross-organizational interoperability where varieties of both business processes and documents handling are considered according to individual business requirements. Though, the end-users, such as, SMEs level companies can access the system easily where technical complexities are hidden behind the end-users interface. (Hoyer 2008; Janner et al. 2007; Kuehn 2009)
The above cube shape (figure 22) model shows the GENESIS project building blocks. It has different kind of integrations and three levels of approaches to ensure the seamless transaction between the integrated business partners. Integration and transaction are defined in four categories are following:

1. Business to Business (B2B) transactions, for example, orders, purchases, invoicing etc.
2. Government to Business (G2B) transactions and information exchange, for example, VAT declaration and payment, tax declaration etc.
3. Business to Intermediary (B2I) transactions, for example, salary payment to bank account etc.
4. Other supportive transactions and information exchange, for example, SMEs company information exchange with financial auditing companies etc.

To achieve above four (B2B, B2G, B2I and Supportive) integrations and seamless transaction, GENESIS defines following approaches in three levels:

1. Process Level: where processes are analyzed and mapped according to the individual country and business area.
2. Information Level: different exchanged information and business documents are analyzed, outputs are ontologies, models and finally, XML schemas according to the followed standardization.
3. Systems Level: technical specifications are produced and finally, infrastructure is developed to provide System-to-System integrations. (Kuehn 2009)

**Figure 23**: GENESIS overall architecture (Janner et al. 2007.)

GENESIS implementation architecture is shown in Figure 23. According to the architecture it has a central server (GENESIS-server) which provides essential functionalities for its users to connect to the system and also to manage communication between the users. It also provides the environment called GENESIS Process and Data Buildtime for business process and document modeling as well as their execution. Therefore, for modeling and execution GENESIS uses ADONIS which is a commercial & automated tool for GENESIS Business Process Management.
The business information modeling environment provides a way to facilitate cross-organizational interoperability and re-use of information entities. It does not require its users (SMEs) to be connected to internet continuously. Rather, it follows a store-and-forward approach which takes into account that the SMEs user may not be continuously connected to the internet. The server can store incoming messages and as soon as the respective user is connected the message is forwarded to the receiver. The server provides its functionalities through web services (chapter 3.1.4), the system design uses SOA (chapter 3.2) and essentials of the ebXML approach. The server also has two other parts, registry and repository which contain GENESIS Common Library (CL). Registry and repository store templates of the supported business processes and business documents for all registered SMEs. It also supports to define conditional business negotiations.

When a user connects to the system, it sends the business context (formalized according to a CCTS (chapter 3.1.2)) information and name of the supported business processes (eg. B2B, B2G and B2I) to the server. The server responds the process definitions and XML schemas (business document templates) to the client which are appropriate for a specific business context. There are two ways are defined to access GENESIS-system. First, an adapter is implemented on the user side and it connects to the legacy IT system of the core GENESIS-system and provides a machine-to-machine interface for efficient automated transactions. Second, user can also use a web based thin client as a machine-to-human interface to access the system over internet. (Janner et al. 2007)

5.4 eYellowpages

This project is a part of a Finnish national-level eLive initiative whose goal is to promote research in ICT sector and to find solution for SMEs for e-business involvement and for more efficient means to conduct business. This project is being managed by Lappeenranta Innovation Ltd., in Finland. It is a non-profit company owned by the city of Lappeenranta. This chapter is based on a study by Pesonen (2009)
on the project and an interview with the project manager in October 2009 on the current status of eYellowpages, which started in the fall of 2009.

The project goal is to develop an information system as a service to make B2Bi more feasible in several areas found challenging in e-business practices. The service has three main objectives: to make it easy for businesses to discover potential trading partners and to learn of partner’s B2Bi capabilities; to make setting up integrated business processes easier between partners at various stages of B2Bi readiness; and building on these, to make SME-to-MNC and SME-to-government B2Bi more feasible (Pesonen 2009).

eYellowpages is designed to provide its services through web services where open APIs (Application Programming Interfaces) are defined for client (any other web-service or application) access to get or put information (from or to the eYellowpages information registry). It has all information storage in its information registry, designed to provide services as per request from client applications. To discover information about an organization, services provide an internet-address for the organization and when invoke the service to get detail information about the organization is possible through the internet-address. Provided information is not only simple traditional data, additionally its products, strategy, services and organization's current B2Bi capabilities altogether. Other organization can establish e-business processes with the given organization based on the provided information, moreover, the information can be interface description (e.g. WSDL), messaging format (e.g. XML schemas) etc. Though, to use the web-services it is not mandatory to have e-business capabilities of the given organization.

Furthermore, open APIs are possible to use in client-side Information System (IS) that need the data from services as an input of client-side IS as its business processes. On the other hand, in the initial plan it has been considered that if client unable to use the APIs means client-side has lacking of IS, a web-based additional user interface will be provided to access of the services. Beside this, services will have integration with KATSO authentication system, already using in several eGovernment initiatives for identity management, authentication and authorization. Importance of KATSO
authentication system is, already it is integrated with the national business information system and organizations are able to use centralized authentication system to access eYellowpages services.

The data stored in the eYellowpages service registry are planned to map as an ontology based, interlinked network information so that service can interpret the information the way to make decision based on the client request. For example, service can find potential business partners according to the requester organization's requirements, capabilities and business area analyzing other organizations' stored information in the registry. Even, it is possible, if a buyer organization request for a bid and the information are available to all similar type of suppliers, even though, buyer does not have previous contact with them. For this purpose, technology for the rational information registry would be either UDDI (chapter 3.2.3) or ebXML (4.2.2), though it is assumed that ebXML is well suited here.

![Figure 24: RosettaNet business process view to eLive projects. (Korpela 2009)](image)

Overall eYellowpages system is shown in figure 24, which contains several services (as web-service) on its top layer (eCatalogue, Quotation, Order, Delivery and Invoicing) to complete the eProcurement. In the bottom layer processes are defined how to process a client (e.g. buyer, seller etc.) request and finally, respond accordingly.
The RosettaNet section in bottom layer is to define the business documents using PIPs (chapter 4.2.1).

5.5 Comparison between Case Solutions

E-business solutions are evaluated in this study can be divided in three different categories, EDI-based e-business frameworks, XML-based e-business frameworks and SME-specific e-business architectures. The features deemed important for the evaluation of the e-business solutions are: Cross-industry-document support, cross-industry-process support, industry-specific support, specifying business-documents, specifying business-processes and messaging (communication protocol) (chapter 4.3). Considering the research questions, it is important to assess all of the reviewed e-business solutions and how they consider implementation complexity, cost and existing Information System requirement for back-end integration. Moreover, a comparison table (like table 3) can be provided to show their adoptability in SME-level.

Table 3: Comparison between e-business solutions from SMEs context

<table>
<thead>
<tr>
<th>Solution</th>
<th>Solution Type</th>
<th>Implementation Complexity (high or medium or low)</th>
<th>Cost (high or medium or low)</th>
<th>Require Back-end Integration (yes/no)</th>
<th>SMEs Adoptable (yes/no)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASC X12</td>
<td>EDI-based</td>
<td>High</td>
<td>High</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>EDIFACT</td>
<td>EDI-based</td>
<td>High</td>
<td>High</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>RosettaNet</td>
<td>XML-based</td>
<td>High</td>
<td>High</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>ebXML</td>
<td>XML-based</td>
<td>High</td>
<td>High</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>OAGIS</td>
<td>XML-based</td>
<td>High</td>
<td>High</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>RosettaNet/RA E</td>
<td>XML-based</td>
<td>Low</td>
<td>Low</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>GENESIS</td>
<td>E-business architecture</td>
<td>Medium</td>
<td>Medium</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>eYellowpages</td>
<td>E-business architecture</td>
<td>Low</td>
<td>Low</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
ASC X12 and EDIFACT both standardizations (chapter 4.1) need well defined back-end integration for each of the business partner involvement. Every partner-side needs the same stack of EDI-software on the top of the underlying IS. This IS is required for the back-end integration with the EDI-based e-business framework. So, it becomes complex implementation and also, the solutions and its implementation require high expertise in this area. Therefore, the result of adopting this approach is also costly, which is not feasible for the SME-level partners.

RosettaNet has potentiality as an e-business solution for MNCs. However, it also has high complexities in implementation which requires proper know-how on RosettaNet Partner Interface Processes (PIPs) and RosettaNet Implementation Framework (RNIF) as well as back-end integration for all partners. As a result, cost becomes very high which means a strong barrier for SME-level adoption.

ebXML itself has provided different tools, such as, CCTS, ebRS, ebRIM, CPP/CPA, ebMS and ebBPSS to define business documents, register and provide e-business artifacts and services. Therefore, configure technical contract between business partners, provides secure and reliable communication, and to enable business processes for the trading partners. However its implementation specification and requirements are also complex and costly for trading partners. It requires a third party, such as, Application Service Provider (ASP) which provides the ebXML registry for partners to find each other through this registry. Finally, the partners need to implement an ebXML compliant information system for seamless e-business transaction. So, to adopt this framework requires proper know-how and back-end integrations.

UBL (chapter 3.1.3) is not included in this comparison table, because it provides such an idea to ensure re-usability of the attributes of business documents and defining business processes accordingly. However, UBL itself does not provide completeness of an e-business framework and usually provides for the completeness of other e-business frameworks, such as, ebXML.

OAGIS provides the possible scenarios for B2Bi and finally, need to use other Integration Architecture for the implementation, for example, ebXML. For the
comparison, implementation complexity is considered to address the cost and in OAGIS case complexity can be varied depending on the implementation environment. However, to adopt this framework require third party messaging architecture and back-end integrations to enable the B2B interoperability.

XML-based e-business frameworks which are covered in this study, all of those except RAE need back-end integration approach for their implementation in B2Bi. RosettaNet Automated Enablement (RAE) is the only XML-based e-business framework which is intended to promote SME-level partners in the e-business area. RAE ensures that SME-level trading partners do not need to implement any enterprise system or back-end integration. Rather, MNCs define the business documents according to XML-schemas and provide them to the SMEs in a human readable format, such as, Adobe’s Portable Document Format (PDF). SMEs read the document using royalty-free software, such as, Adobe Acrobat Reader and fill it up. Finally send the document back to the MNC gateway, like, email or any other communication system.

GENESIS does not require client-to-client compatibility because of its central-server (GENESIS-server) approach, which ensure the messaging between clients are compatible. Even SME partners do not need any back-end integration. Though, to establish integration with a business partner, SME-users should define the business processes and documents as GENESIS business process management. A commercial automated tool ADONIS is used for GENESIS business process management and it require some level of technical understanding. However, GENESIS is developed targeting Northern European countries and not practically tested enough for other European countries or any other part of the world. GENESIS e-business architecture development was started mainly for SME business partners, though, based on this reason can be a risk to adopt GENESIS as a generic SME e-business solution.

eYellowpages is a future extension of e-business concepts. It is intended to provide an infrastructure for e-business functionalities as a cloud computing approach. eYellowpages defines its overall architecture to provide all kinds of eProcurement services through web services as a Finnish national wide service. Companies can upload their business information and then find each other's profile, such as, products, services, B2Bi strength etc. to do business. Even to access the system, client side
information system (IS) is not mandatory. Therefore, it has as its additional feature, a web-based interface for the SME-level users or who can not afford to develop an IS using the open APIs provided by eYellowpages services. Only risk factor to adopt this e-business solution is still this project is under development and only this concept can ensure its possibilities but the final product is not in industry level practice yet.
5.6 Summary

This chapter has been provided an overview of important eBusiness frameworks which are discussed detail in chapter 4 and then some new eBusiness solution architectures, such as, eYellowpages and GENESIS are discussed detail. Technologies are selected which provide several ways to build an eBusiness infrastructure and also consider the challenging areas for this thesis work.

eBusiness solutions are considered in this study are possible to divide in two categories, standardized eBusiness frameworks and eBusiness solution architectures. Though the main discussion of eBusiness frameworks is covered in chapter 4, the continuation is also in this chapter due to get a comparison result between all the reviewed eBusiness solutions. Moreover, after reviewing all above standards, frameworks and architectures, an agile analysis is done on those which are reflected in a comparison table. The main goal was to address the feasible solutions which are cost effective, easy implementation and finally, fit for the SME-level adoption.

The analysis is done successfully in this chapter and expected results also found from the analysis artifacts. This result helps to continue the next step means recommendations for SMEs from the present solutions and also future research proposals, which are discussed elaborately in chapter 6.
6. RECOMMENDATIONS FOR SMEs

Research questions for this thesis work were to study and analyze current e-business frameworks and novel e-business architectures. Therefore, to find what kind of e-business solutions are fit for SME-level business partners considering costs and implementation complexities. With these results, the study also attempted to locate future research aspects in this problem area. Given questions answers are found and described below, which are as the achievement of this thesis work.

6.1 Adoptable Solutions

The main idea was to determine the e-business solutions which are possible to recommend for SME-level adoption considering the cost and implementation complexities. Several e-business standards, frameworks and solutions are already available in the market and few of them were considered to analyze based on current industry practices. In chapter 5.5, final analysis has been done which compares between the existing e-business solutions which were enlisted particularly for this research. Furthermore, different solutions were evaluated based on their common properties and which helps to make decisions for the recommendation and to realize the needs of future research. Evaluated solutions were selected based on different aspects of their use in the business context. Three solutions were found to be recommendable for the SMEs, these are RosettaNet Automated Enablement (RAE), GENESIS and eYellowpages. Though, these recommendations are not fully capable to provide as a generic e-business solution for SME-level partners and several pros & cons are exist.

The first recommendation, RAE, is recommended when an MNC has existing RosettNet back-end integration. The RAE approach leverages the MNC RosettaNet infrastructure and does not require any Information System (IS) at the SME-level business partners. Nevertheless, there are limitations and barriers if the MNC has a different e-business solution instead of RosettaNet and there is not possible to introduce the RAE approach. Even if the e-business approach is newly introduced for
both the MNC-level and SME-level partners, RosettaNet is not feasible for all kinds of organizations. RosettaNet focuses on industry-specific e-business framework and the end-user organization should fit for the adoption. Therefore, to start from the scratch, the MNC has options to find such an e-business solution which can provide cross-industry e-business infrastructure, support generic business documents and process specifications and there should be no implementation complexity and cost overhead for the SME-level partners. RosettaNet/RAE does not offer such a concept as a generic e-business solution. So, in short, only existing RosettaNet users are feasible and effective candidate for RAE approach.

Second Recommendation, GENESIS is novel e-business architecture and it has been initiated to promote the SMEs e-business adoption, to do business more efficiently and less error prone. The most important parts of the GENESIS e-business solution are it hides the whole implementation complexity from the business partners and the services are provided from a central server. So, once the GENESIS-server is deployed, business partners carry out e-business activities through the server and no need to have business partner back-end integration or any existing IS. However, MNC-level business partners can implement proprietary enterprise system for their own business needs, though SME-level partners can avoid that. There are also drawbacks in adopting GENESIS. First, GENESIS was developed for the target business organizations in Eastern Europe and there is no proof that it will support generic cases among heterogeneous business partners over the world; Second, GENESIS solution has not yet been tested enough in industry practice which can be a risk to adopt for the SME-level partners.

The third recommendation, eYellowpages, is included in this list from a different point of view compared to the above recommendations. eYellowpages could be a partial solution for the SME-level partners, as it is an architecture which provides an e-business infrastructure for all of its connected user organizations. Furthermore, the end-user can publish their all business information including B2Bi capability in the eYellowpages registry and interested users can find it to join as a business partner according to the published business information in the eYellowpages registry. It also has a web-interface for users who do not have information system integration
capability. The main barrier for eYellowpages adoption is this solution is still under development and the initial analysis is considering only for a national level service. There is hence a potential risk to adopt this solution worldwide as a generic eBusiness-architecture.

6.2 Cost and Implementation Complexity Issues

To adopt eBusiness frameworks at SME-level, mainly two barriers are addressed as a primary issue: cost and implementation complexity. It is very important for the SME business partners that the recommended solutions (6.1) and future research (6.3) both are considering the cost issues primarily. Therefore, SME business entities have less strength to effort high cost for their business automation and this is the cause which has also motivated to consider the cost issue in eBusiness frameworks assessment table (5.5).

Moreover, cost and implementation complexities are interrelated to each other. Implementation complexities can be considered as a driver to ensure the cost is low or high. If implementation complexity is high, then resourcing, know-how and time scale also need high scale which means cost is also high. So, the opposite is less implementation complexity means low cost.

In this thesis work, eBusiness frameworks analysis, comparison, recommendations and future research scope all of these areas are considered both cost and implementation complexity issues as a primary and main barriers for the SME-level partners.

Finally, all recommendations are come out based on their infrastructure requirements which is feasible for SMEs point of view. Therefore, it has been well mentioned that recommended solutions are less complex and cost effective for small and medium businesses. On the other hand, future research scope fully considered that how the e-business architecture can be a generic solution and end-user does not need to implement any back-end integration which ensure less e-business adoption cost.
6.3 Future Research Scope

Analysis part in this thesis work was conducted to determine such e-business frameworks and novel e-business architectures which are easy to adopt for SMEs context. Though the result of recommendation shows that, there are several e-business solutions are exist to serve for the SMEs but not yet sufficient enough. Therefore, the approach of e-business introduced an efficient way to do business and to decide the suitable e-business solution is a dilemma. Considering the present inefficiencies in e-business solutions and its future prospects, possibility of continuation of future research in this area is very high. In this study, the scope of the future research was also considered as research question and lacking areas are also determined to propose for the future research scope.

Furthermore, the main area to focus for the future research is: Software as a Service (SaaS) in the Cloud; this research scope is strived from this overall thesis study.

**Software as a Service (SaaS) in the Cloud**

In the early stage of this study (in chapter 1) it was realized that cloud computing is becoming more cost effective, reliable and popular nowadays. The consequences are many service providers are providing different kind of services in the cloud; users only subscribe and start to use the services. SaaS is one of the services which have a wide range of potential in e-business area to promote SME-level business partners. The idea is, everything (i.e. Information Systems, B2Bi infrastructure, interoperability technology) is ready in the cloud and SMEs need to be subscribed to use and pay for as much use.

To propose this solution, initial requirements were determined to ensure whether the solution able to serve what is needed. Requirements are as following:

- SME-level business partners do not need to implement any infrastructure.
- Once the solution is ready then SMEs subscribe to the system and should be able to use over the internet. SME-partner does not require any technical know-how.
• There is no prerequisite to have an existing information system to adopt the e-business solution. SMEs users should be able to use the system through Web interface provided by the service provider over the internet.
• If subscriber has an existing system and wants to access the solution through the Web services is also possible.
• All possible e-business standards and frameworks which are mostly adopted among the companies are also present in the proposed solution. As a result, different kind of MNCs can be integrated with different SME partners.
• For example, if different MNCs have different e-business frameworks and their SME partners are able integrate to both of those MNCs through this same solution architecture.

A clear overview of this idea is provided in figure 26, there are three layers and among them middle layer is the core part in this solution. The middle layer contains all possible e-business framework standards including the back-end information systems. There are two types of access points: middleware technologies and web based user interface. Therefore, both options are possible for SME users, for example, if SME user does not have any existing information system then they can access the system in the cloud through web based user interface over the internet. Otherwise, if SME user has existing legacy system then they can access the system using both web based application and middleware technology. On the other hand, MNCs communicate with the service through middleware technology over the internet. Finally, the important issue is e-business solution stack, which provides the flexibility and scalability to integrate business partners according to the particular e-business framework requirement. So, different SME users can choose different e-business frameworks to do businesses with different MNCs.
Finally, the main research area is addressed “e-business solution stack”, which need to be a generalize stack and possible to put a new e-business framework inside the stack or remove an existing framework and then configure a configuration schema. Therefore, there should be another configuration schema for subscribers. So, the system decide which e-business framework is to plug-and-play based on the request from user and the corresponding business partner who is the receiver. Additionally, this idea works efficiently when the same SME has different MNC partners then they can use different e-business frameworks from the same SaaS and configure the e-business requirements according to different MNC partner needs. The challenges are realized to provide proper intelligence for the e-business solution stack and also, the integrations between all components in the cloud, such as, between middleware technologies, frameworks, back-end systems and web clients.

Figure 25: SaaS for SME-level adoptable eBusiness solution
6.4 Summary

After the background study and analysis this chapter is concluded with recommendations. However, there were different challenges to finalize the recommendation. Main part for this chapter was assessment of feasible e-business solutions for SME users among the existing e-business solutions that has been taken in account for this thesis scope. Challenging issues come out for this feasibility study because of the lacking of existing e-business solutions and their approaches to adopt. Though, finally there is few e-business solutions are recommended as a conditional basis means failed to recommend any existing solution as a generic and generalize approach. As a result, to achieve the thesis goal a future research scope and idea is recommended which is able to provide as a generic e-business solution for SME users.

E-business adoptions for SME users have different kind of barriers and among those two main barriers are addressed here to consider for SMEs. Moreover, Implementation complexity and cost issues drive SME users in back-foot to adopt any e-business solution. Though, all three recommended existing solutions were selected based on their feasibility to overcome those SME barriers. Though, recommended e-business solutions are still can not full fill varieties of business processes and their needs. So, future research in this area is very important and this needs motivated to propose the future research scope and idea in this thesis conclusion part.

Future research concept and goal are outcome of this thesis research question and their possibilities to find answer from background study. Moreover, it is completely realized that e-business is the future business concept and interoperability is very important between business partners to do e-business effectively. Proposed future research idea is only the way has been found at this moment to meet this thesis final goal and it is conceived from the existing e-business solutions that how they are providing now and what are their drawbacks.
7. SUMMARY

In this thesis mainly focused on literature review related to e-business interoperability technologies and B2Bi possibilities in different sizes business partners. There are two major reasons for the special concern of SMEs e-business adoption, because SMEs have a big contribution in economy and employment. As a business partner SMEs represent over ninety percent of enterprises. So, e-business adoption and interoperability enablement only in MNCs will not be effective until SME-level partners can also join in the same integration successfully. Therefore, the study intended to recommend e-business solution for SME-level business partners. Two problems, cost and implementation complexity, were addressed. Among the available e-business solutions, well known e-business frameworks were studied. The selected e-business frameworks were EDI-based or XML-based and Industry-specific or cross-industry. They included RosettaNet, ebXML and OAGIS. In addition, SME-oriented e-business architectures were also considered. These were the GENESIS and eYellowpages platforms, which are mainly initiated to facilitate SME-level e-business adoption.

The analysis on the several e-business solutions showed that the existing solutions are still not capable enough to overcome the barriers for the SMEs to adopt them. Mostly, the existing e-business standardizations and frameworks are best suited for the large companies and in practice e-business adoption in MNCs is greater scale than the SMEs. On the other hand, though e-business architecture development was initiated for the SME specific support, but existing solutions in this area are not tested properly in industry yet. However, few of the existing e-business solutions were found which has some potential to recommend for the SME-level business partners, such as, RosettaNet/RAE, GENESIS and eYellowpages. Justifications were shown for the recommendations and the primary concerns were implementation complexity and cost issues which are the main barrier for SMEs for e-business adoption.

Finally, reflection of the overall analysis and understanding are completed through the recommendation of future research scope. For future research topic, the
recommendation is Software as a Service (SaaS) in the cloud. The idea provides a high level of e-business architecture which can ensure an easy way of implementation for SME-level partners and integration is possible between heterogeneous businesses.
REFERENCES

European Commission (EC) SME Definition:

International Journal of Interoperability in Business Information Systems (IBIS):


Kotinurmi Paavo, Laesvuori Hannu, Nurmiakso Juha-Miikka; XML-based e-business frameworks and standardization; Computer Standards & Interfaces; Volume 28, Issue 5, June 2006; Pages 585-599

Nurmiakso Juha-Miikka; Adoption of e-business functions and migration from EDI-based to XML-based e-business frameworks in supply chain integration; International Journal of Production Economics; Volume 113, Issue 2, June 2008; Pages 721-733

Nurmiakso Juha-Miikka; EDI, XML and e-business frameworks: A survey; Computers in Industry; Volume 59, Issue 4, April 2008; Pages 370-379

Askounis Dimitris, Charalabidis Yannis, Gionis George, Lampathaki Fenareti, Mouzakitis Spiros; Business to business interoperability: A current review of XML data integration standards; Computer Standards & Interfaces; Volume 31, Issue 6, November 2009, Pages 1045-1055

Weiss Aaron; Computing in the Clouds; ACM; Volume 11, Issue 4, December 2007, Pages 16-25

Hoyer Volker, Janner Till, Mayer Peter, Schroth Christoph; Small and Medium Enterprise’s Benefits of Next Generation e-Business Platforms; The American Academy of Business and ABI/ Proquest; Volume 10, Issue 2, December 2006; Pages 285-291

Introduction to XML: http://www.w3schools.com/xml/xml_whatis.asp; 15th June 2009

OASIS Universal Business Language (UBL) v2.0 standard: http://docs.oasis-open.org/ubl/os-UBL-2.0/UBL-2.0.html; 22nd June 2009


Guglielmina Claudia, Kiauleikis Mindaugas, Kiauleikis Valentin, Morkevicius Nerijus, Tolle Karsten; Performance and Architecture Modeling of Interoperability System for SME’s; Springer Berlin Heidelberg; Vol. 7, May 2008, Pages 345-356

Sun Microsystems; Introduction to Cloud Computing Architecture; white paper, 1st edition, June 2009

W3C recommended Web Services Architecture Requirements: http://www.w3.org/TR/wsa-reqs/; 13th June 2009

Alonso Gustavo, Casati Fabio, Kuno Harumi, Machiraju Vijay; Web Services (Concepts, Architectures and Applications); Springer-Verlag Berlin Heidelberg, 2004; Pages 123-195

W3Schools, WSDL tutorial: http://www.w3schools.com/wsdl/wsdl_intro.asp; 5th July 2009

W3Schools, WSDL and UDDI: http://www.w3schools.com/WSDL/wsdl_uddi.asp; 6th July 2009

Bao-jian Zhang, Xu Wu, Yue-sheng Gu; Research and Realization of Web Services Security Based on XML Signature; International Conference on Networking and Digital Society, 2009; Vol. 2, Pages 116-118

Sun Lili, Li Yan; XML and Web Services Security; IEEE 2008; Pages 765-770


RosettaNet; RosettaNet Implementation Framework: Core Specification, v02.00.01; 6th March 2002


Grangard et al.; ebXML Technical Architecture Specification v1.0.4; 16th February 2001; Pages 1-39

Abrams Charles, Kenney L. Frank, Rita E. Knox; UBL Standard Needs Wider Use to Succeed; Gartner Inc.; 15th November 2004

Jones Ian, Mackenzie Matthew; OASIS ebXML Messaging Services v3.x Feature Review; 12th January 2001, Pages 1-15

By the OASIS ebXML Joint Committee for OASIS; The Framework for eBusiness, An OASIS White Paper, 24th April 2006
Pattyn Dan, Webber David RR; The New ebXML for SOA; 

OAGi: http://www.oagi.org; 7th August 2009

OASIS; Naming and Design Rules (NDR): http://xml.coverpages.org/ndr.html; 20th November 2009

Rowell Michael; OAGIS 9.0 Introduction; OAGi; 9th August 2005

Hoyer Volker; Modeling Collaborative e-Business Processes in SME environments; Journal of Information Science and Technology, 2008; SAP Research CEC St. Gallen and University of St. Gallen, Institute for Media and Communications Management (MCM), Blumenbergplatz St. Gallen Switzerland; Pages 47-59

Janner T., Pemptroad G., Schroth C.; CCTS-based Business Information Modeling for Increasing Cross-Organizational Interoperability; Springer London; 29th August 2007; Pages 467-478

Kuehn Harald; European Union's GENESIS Project; OMG Winner-best use of standards, Case Study, April 2009; http://www.bptrends.com; 19th June 2009

Cartwright John; RosettaNet Automated Enablement (RAE); Material Composition Workshop, 30th August 2004

RosettaNet; RAE Process Scenario; http://www.rosettanet.org/; 19th July 2009

Schenecker Mark; Inside RosettaNet's Automated Enablement; http://www.ebizq.net/topics/b2b/features/5487.html?&pp=1; 4th August 2009

Schenecker Mark; Case Study in using Forms Integration for SMEs in the High Tech Industry; CONNECT CONFRENCE, 2005

Cartwright John; RosettaNet Automated Enablement; RosettaNet Global Community Summit; 2006

Pesonen Tero; Quality Standards in E-Business Software Development; Lappeenranta University of Technology; Department of Information Technology; Master of Science Thesis; 2009; Pages 76-80

Korpela et al.; Elive osprojektin kuvaus: eYellowpages; 10th March 2009


BPMI (Business Process Management Initiative); Business Process Modeling Notation (BPMN), Version 1.0: http://www.bpmn.org/; 27th July 2009


Korhonen Reija, Salminen Airi; Visualization of EDI messages: Facing the problems in the of XML; International Center for Electronic Commerce (ICEC), 2003; Pages 465-472

Adam Nabil R., Adiwijaya Igg, Atluri Vijay, Yesha Yelena; EDI Through a Distributed Information Systems Approach; IEEE, volume 7, January 1998; Pages 354-363
W3C; XML Key Management Specification (XKMS 2.0):  
http://www.w3.org/TR/xkms2/; 16th September 2009


Erl Thomas; SOA: Principle of Service Design; Prentice Hall; Pages 37-38; 18th July 2007