ANALYZING THE USAGE OF OPEN SOURCE PRODUCTS FOR SOA

By

Sajid Ali

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

Master of Software Engineering of Distributed Systems

School of Information and Communication Technology (ICT)
KTH University Sweden

2011

Supervisor at KTH ICT: Mihhail Matskin
Supervisors at Vattenfall Business Services Nordic AB: Alaa Karam, Arash Rassoulpour
(Left Blank Intentionally)
Abstract
Service Oriented Architecture (SOA) is rapidly becoming the new standard for today’s enterprise. SOA can enable great agility, productivity and cost savings, but not if it is implemented with complex, closed and expensive enterprise platforms. The best way is to implement it in Simpler, Open and Affordable manner. The main purpose of this thesis is to identify some of the core characteristics of SOA and explain how they are manifested in actual technologies that can be used in a cost effective way in an enterprise. Collectively these technologies will combine to form what we are calling the Open SOA Platform. Once these technologies are identified then our main goal will be to find open source products that can be used to satisfy these technology requirements. We are going to analyse and decide the solution areas for example; Integration, e-Business that we can use Open Source products in and also the functions blocks for example ESB, Registry, and SOA Governance management, within the decided Solution areas that could be implemented by Open Source products.

We will also define the Technical Reference Architecture (TRA) for the SOA (EAI and ESB) and Portal. The implementation of the TRA done by the Microsoft and Red Hat will be compared and the best option will be selected on the basis of different criterion. If we can prove that the usage of Open Source products and SOA within any particularly organization is possible which means there are big savings, and if the savings are much more than today’s IT costs, migration cost and other costs related to IT, then we maybe have decision basis for next generation of standard product(s) in some solution areas in the IT landscape. The last part of the thesis is to analyse the impact of using Open Source products in Vattenfall Business Services Nordic AB, and what kind of competence we need. Other IT-suppliers in Vattenfall can use the Master Thesis as a template to calculate the implementation of the decided functions blocks in their IT landscape.
(Left Blank Intentionally)
ACKNOWLEDGMENTS

First of all I deeply thank my supervisor Mr. Arash Rassoulpour at Vattenfall Nordic Business Services for his continuous support and encouragement. He showed me a different way to approach research problems and taught me how to think critically. I am also thankful to him helping me to complete the writing of my thesis and technical problems lie behind it. A special thanks to my co-supervisor Mr. Alaa Karam for his support for helping and identifying some of the core technical aspects of my thesis. I am also very thankful to my thesis partner Mr. Adnan Gohar for valuable discussions and suggestions. I feel lucky for having the privilege and the opportunity to work with real experts in a beautiful working environment.

I am thankful to my supervisor at KTH Dr. Mihhail Matskin for arranging such a nice master program and providing his support in academic issues during past 2 years. Thanks to my program coordinator Jenny Lundin for helping me in administrative issues during past 2 years. Thanks to all my professors, lecturers and the other staff at KTH.

Last but not least, I would like to thank my family and friends for their love and presence. I pay my gratitude to my parents for the motivation they gave to me and their prayers and wishes for me.
# TABLE OF CONTENTS

1. Introduction ..........................  
1.1 Problem Background ........................................... 9  
1.2 Purpose of the study ........................................... 9  
1.3 Delimitation .................................................. 10  
1.4 Motivation .................................................... 10  
1.5 Detailed problem statement ................................... 10  
1.6 Comparison with Vattenfall TRA (Technical Reference Architecture) ........................................... 11  
1.7 Expected results ................................................ 12  
1.8 Related work .................................................... 12  
1.9 Existing solutions .............................................. 12  

2. Background ............................. 12  
2.1 Current systems in Vattenfall ................................. 12  
2.2 Problems in Current system .................................... 13  
2.3 Why SOA for Vattenfall ......................................... 15  

3. Service Oriented Architecture (SOA) ............ 16  
3.1 What is SOA? ................................................... 16  
3.2 SOA Platform, Tools and Technologies ........................ 19  
3.2.1 SOA Platform .............................................. 19  
3.2.2 Services .................................................... 20  
3.2.3 Enterprise Service Bus (ESB) ............................. 21  
3.2.4 Business Process Management (BPM) ....................... 22  
3.2.5 Presentation (Portals) ...................................... 23  
3.2.6 Registry ..................................................... 24  

4. Vattenfall Needs .......................... 26  
4.1 Technical Reference Architecture of Vattenfall ............... 26  
4.2 Meter reading Case Study ..................................... 27  
4.3 SOA Guidelines for Vattenfall ................................ 33  

5. SOA Vendors ............................ 36  
5.1 Redhat JBoss .................................................. 36  
5.1.1 JBoss Enterprise SOA platform ............................ 36  
5.1.2 JBoss ESB .................................................. 37  
5.1.3 Services in JBoss SOA ...................................... 39  
5.1.4 JBoss Enterprise Business Rule Management System (BRMS) ........................................... 39
1. Introduction

Vattenfall is a Swedish power company and one of the leading energy producers in Northern Europe. The name *Vattenfall* is Swedish for waterfall, and is an abbreviation of its original name, Royal Waterfall Board (*Kungliga Vattenfallstyrelsen*). Vattenfall is wholly owned by the Swedish government. The Vattenfall is growing continuously every day. Other than Sweden, Vattenfall is also working in Finland, Germany, Netherland, Denmark and Norway. As Vattenfall is growing, the need for integration is also increasing. Vattenfall is a large, energy company with a heterogeneous systems environment where systems integration is a vital competence. SAP and BizTalk is largely used within Vattenfall.

In addition, the number of instances is increasing which has created a need for SOA centric integration strategies. The current integration landscape is quite complex and the company uses different methods and technologies. Vattenfall Nordic defined an integration strategy five years ago and selected Microsoft BizTalk as their main integration platform. They have also many new SAP installations. Currently, Vattenfall has EAI (Enterprise Application Integration). Vattenfall is looking forward to adopt Service Oriented Architecture (SOA) in near future. They are already working on it. Vattenfall also have its own Technical Reference Architecture (TRA) for SOA.

The term SOA has been around since 2000, and over past few years it has emerged one of the best approaches for system integration. SOA expresses the perspective of software architecture and defines the use of services to support the requirements of software users. Generally in SOA environment resources are made available on a network as independent services and the purpose is to access these services without having the knowledge of platform implementation. According to the W3C a SOA is a set of components which can be invoked, and whose interface description can be published and discovered. SOA is based on standards due to which there are many vendors who are working on SOA technologies. But SOA is not bound to some technology or platform. SOA can be implemented with the help of open source products. For example JBOSS/Redhat offers open source tools and technologies which can be used to build SOA platform. We will discuss SOA in more detail in chapter 3.

1.1 Problem Background

As mentioned earlier the Vattenfall is using EAI. The current systems which are running within Vattenfall are Microsoft BizTalk and SAP. Vattenfall is spending large amount of money on these technologies. This is the main reason that Vattenfall wants to analyze and understand the Open source products and especially for SOA to take a strategic decision if they are going to use it as their main integration platform or as a complement to the Microsoft BizTalk at Vattenfall Nordic.

1.2 Purpose of the study

For the purpose of this thesis if we can prove that the usage of Open Source products (e.g. JBoss and Microsoft) for SOA implementation within Vattenfall Nordic is possible which means big cost savings. This is because SOA can be implemented using open source products, and for this reason a large amount of licence fees can be saved at first place. And if the savings are much more than today's IT-costs, migration cost and other costs (e.g. new hardware, Support), then the Vattenfall can have decision basis for next generation of standard product(s) in some solution areas in the IT landscape. To find out whether Redhat JBoss and Microsoft provide the components for SOA implementation and their comparison with TRA is also our basic purpose. Then on the basis of TRA comparison we will decide which of the vendor is more suitable for Vattenfall for SOA implementation.
1.3 Delimitation
The investigation does not aim to analyze all the possible integration methods or technologies available in market. The main investigation area is how to find out the needs of Vattenfall, and how these needs can be fulfilled with SOA. In IT market a lot of vendors are available which provides SOA platform but we are not interested in all of them, we are only considering Redhat JBoss and Microsoft for SOA platform. In the start we were also interested in Oracle for SOA solution but due to time limit we eliminate that.

1.4 Motivation
This is a research based thesis. The main objective of the thesis is to analyze the open source products for implementing the SOA. Open source software has proven its value in the organization of every size. There are large number of open source products which many companies are using to enhance their businesses. SOA is one of the options which an organization can adapt to increase the agility in IT systems. There are large numbers of vendors (Open source and proprietary) who are offering their services for implementing SOA. The actual problem is that which of them is reliable for an enterprise?

What we are going to do is to analyze three of the main vendors who provide services to implement SOA to increase flexibility and reusability of the business infrastructure. Following are the vendors

- Red Hat JBoss
- Microsoft SOA

There are some specific parameters for SOA which we are going to analyze for selecting the affordable, simpler and suitable option for implementing SOA architecture.

Some of the very important parameters which we will consider are Business Process Management (BPM), Enterprise Service Bus (ESB), Presentation, Registry and Services. There are some other parameters which are important for SOA which have been explained in detail in next chapters.

1.5 Detailed problem statement
We are going to analyse the open source software for SOA. It’s assumed that Vattenfall Nordic needs for SOA. The Solution areas for integration are SOA for the open source products that can be used (e.g. Redhat JBoss, Microsoft) in. Also decide the functions blocks like SOA governance, SOA registry within the decided Solution Areas that could be implemented by Open Source products (e.g. Redhat JBoss and Microsoft). Figure 1.0 shows the problem area for our
Research.

Figure 1.0 shows that we need to gather the Vattenfall Nordic requirements based on TRA and other needs. Then we will compare Microsoft solution with Open Source Software solution which in our case is Redhat JBoss. Evaluation criteria are the function blocks which Vattenfall Nordic needs to implement SOA. It includes cost, technical functionality, vendor’s roadmap and some other parameters. On the basis of these parameters we will decide which solution is best for Vattenfall Nordic and why.

SOA is not something which an organization can buy. It is nothing like a box which you can buy and use it within an organization. If an organization wants to use SOA environment then they have to implement it with one of the many options available. Vattenfall has its own Technical reference architecture (TRA) for SOA. While we will make comparison between different vendors, we will also make sure that the SOA vendors provide all those functionalities which are requirements for TRA in Vattenfall.

1.6 Comparison with Vattenfall TRA (Technical Reference Architecture)
We will compare these three technologies with the TRA of the Vattenfall. SOA Technical reference architecture (TRA) of the Vattenfall has some parameters which we need to consider while comparing these technologies and then to also compare with TRA. I will explain Vattenfall TRA in detail in next chapters.
For the purpose of this thesis if we can prove that the usage of Open Source products (e.g. JBoss, Oracle and Microsoft) within Vattenfall Nordic is possible which means big cost savings. This is because SOA is open source, and for this reason a large amount of licence fees can be saved at first place, and then we will compare different parameters mentioned previously for the choosing of right vendor. And if the savings are much more than today’s IT-costs, migration cost and other costs (e.g. new hardware, Support), then maybe we have decision basis for next generation of standard product(s) in some solution areas in the IT landscape in Vattenfall.

1.7 Expected results

As I have explained above that the main purpose of our thesis is to decide which of the vendor among Redhat/JBoss and Microsoft, is better for organization. The decision is based on the different criterion which is cost, complexity, true open source, viability, architecture, monitoring and management and extensibility. We are expecting to find a better solution for Vattenfall Nordic, so that they can use one of the decided vendors in future for their SOA implementation. Each of the single parameter mentioned above will be compared with each other to find the best available option for us.

1.8 Related work

A lot of work has been done on SOA and still it’s growing every day. Large organizations are adopting SOA. SOA is not like a product which you can buy. If you want to have SOA in your organization you have designed it. As our work is mainly based on the research and to study the open source and some other vendors for SOA architecture, so we need some related work to have better idea for these vendors.

1.9 Existing solutions

The two vendors which we have chosen Microsoft and JBoss are one of the leading vendors for SOA in the current information technology market. A large number of systems are actually working in market by these vendors. JBoss provides many middleware products for SOA most of them are Open source. Microsoft also provides SOA platforms. The important thing for this research is to study these vendors and compare each of them with each other than with TRA of Vattenfall. We will analyze the existing solutions and products offered by these vendors for SOA. We are following some of the books which are very helpful to study SOA architecture offered by JBoss and Microsoft. We did not find any previously comparison between these three vendors.

2. Background

2.1 Current systems in Vattenfall

Currently Vattenfall is using SAP and BizTalk for Enterprise Application Integration (EAI). Basic idea behind application integration is to share data between business application within and across the enterprise business units. Like any large organization Vattenfall is also using EAI to integrate a set of computer applications. An EAI is the use of computer software, hardware, and architectural principles to integrate application within or across the organization.

Vattenfall is mainly using SAP and BizTalk for its integration solutions. Vattenfall is mainly SAP centric and their data is stored in SAP systems. Vattenfall is heterogeneous with different systems currently working inside company.
2.1.1 Microsoft BizTalk Server

Microsoft BizTalk server is referred as “BizTalk”, is an Enterprise Service Bus. With the help of adapters the BizTalk makes is used to automate the business process. Main function of BizTalk is providing the Business process automation, business to business communication and enterprise application integration.

2.1.2 SAP ERP

According to [2] “The SAP ERP application is an integrated enterprise resource planning (ERP) software manufactured by SAP AG that targets business software requirements of midsize and large organizations in all industries and sectors. It allows for open communication within and between all company functions”. Companies need to integrate systems for many reasons. For example a company like Vattenfall Nordic has many external parties is involved in its different business areas. And Vattenfall needs to share the information between these parties in real time. This sharing could be inside or outside the company. That’s why The EAI is very important. With the help EAI the main advantages are for those applications which are by nature not designed to work together with other applications. But with EAI different applications can be combined to increase the reusability and sharing of resources in real time.

2.2 Problems in Current system

As the current system of Vattenfall is based on EAI, I will try to explain some of the problems and flaws in systems. As explained by the [3], the integration of systems involves connecting the systems together in order to share the information. In integration the business entities and related data are often represented in some form across multiple application data stores, resulting in data inconsistencies across the systems. Integration techniques such as batch data transfer and direct access to cross applications data stores provided a means to share data and to achieve data consistency. On a small scale the Point-to-Point approach is doing a great job. But with the increase in number of interfaces and integration approaches, this approach adds more and more complexity in the organization.
This point to point nature of integration is tightly coupled and also may not be best practice for an architectural or design perspective. Some of the drawbacks or limitations explained by [3] are, it has a limited management infrastructure that provides governance, monitoring, support, change control and other management capabilities. There is a lack of standardization within the enterprise around the tools, technologies and approaches applied.

The next step to these point to point integrating systems is EAI. EAI was designed to overcome the limitations caused by P2P. EAI includes the management, standardization and governance. EAI approach is more centralized with control on integration including competency center development that is responsible for the application integration solution life cycle. As [3] explains that the backbone to this EAI approach is MOM (Message Oriented Middleware). With the help of adapters the data can be exposed and consumed in the form of messages. MOM can physically and operationally decouple source and target systems via message queuing system. But it cannot remove the application coupling at the message level format.

For EAI a common message model requires that all source and target applications to produce or consume standardized formats and the message broker should handle the transformation of messages between different source and receivers. The Hub and Spoke model as shown in the diagram below, integrates the end points with the central mediation hub to message and direct traffic across the network.
The EAI was a great success and improvement where enterprises were prepared to adapt to an integration centric approach. But still there is some limitation in EAI that constrained its effectiveness. According to [3], an EAI is a centralized competency center model means that all the integration skills are based in a single group. The competency center either undertook work within the group to complete on behalf of projects or placed skilled staff on the projects as integration requirements were identified. In both cases the competency center struggles to keep up with demand where many projects were running concurrently. EAI are still point to point integration and there is very little opportunity for reuse. According to [4] EAI system requires high cost for development initially, particularly for small and medium size business. Most of the EAI projects which starts as point-to-point becomes unmanageable as the number of applications increase. For Vattenfall it's important to save the huge amount of money on EAI approaches. Loose coupling and data reusability is another issue. As the number of applications increases EAI becomes more and more complex.

2.3 Why SOA for Vattenfall
In previous section I explained the working of EAI and its problems. Why SOA for Vattenfall? This is an important question. Vattenfall is spending large amount of money for the buying of SAP and BizTalk software’s. They have also large number of License fee. In first place SOA can be implemented without proprietary software. But what else SOA can provide to Vattenfall’s current system. According to [5] in architectural terms, a modern architectural design should be Service Oriented, loosely coupled, driven by events, able to support both integration and assembly, aligned with valuable life cycle support processes, and able to leverage existing infrastructure and applications.

When it comes to Service Oriented Architecture, it can provide different advantages within Vattenfall over different EAI methods. According to [6] in general SOA provides business services offered at different platforms, location independence, authentication as well as authorization support.

Other services provided are loose coupling, dynamic search and connectivity to other services. As explained by the [6] the reliability, reduce in hardware costs, an approach towards standards based servers and application consolidation, the leveraging of existing development skills and the providing of a data bridge are also some short term benefits of Service Oriented Architecture. A typical structure of SOA looks like, as shown in the figure below
As Vattenfall has large number of existing systems based on EAI so SOA can be used to integrate all of the existing systems and new systems as well. But it’s not only SOA which is important for Vattenfall, but to reduce the current IT cost and migration cost is the major cost of Vattenfall. This goal can be achieved using SOA. This is the reason we are interested to analyze the Microsoft and Redhat JBoss in order to see does they fulfill Vattenfall needs.

3. Service Oriented Architecture (SOA)

3.1 What is SOA?

The definition of SOA according to [7] is ”Service-oriented architecture (SOA) is a flexible set of design principles used during the phases of systems development and integration in computing. A deployed SOA-based architecture will provide a loosely-integrated suite of services that can be used within multiple business domains”.

Thomas Erl[8] defines SOA as “Contemporary SOA represents an open, agile extensible, federated, composable architecture comprised of autonomous, QoS-capable, vendor diverse, interoperable, discoverable, and potentially reusable services, implemented as Web services”.

SOA is not something concrete or complete architecture, but it leads towards a concrete architecture. SOA is an architectural style, or way of thinking, or paradigm. SOA is not something which a large or small organization can buy, but it is a way of thinking or designing. Web Services are basic concept in SOA. As stated by [9], for the past few years SOA has emerged as one of the preferred approach for system design, implementation and integration. According to [10] Because of
the large availability of internet SOA is based on the understanding and re-usability for consuming services. And each service is a representation of work unit.

“SOA is dead; Long Live Services”

In one of the very famous blog “SOA is dead, long live services” by Anne Thomas Manes [11], she mentioned that SOA comes to an end in 2009 because of the heavy economic crisis. She also said the” SOA is survived by its offspring: mashups, BPM (Business Process Management), SaaS (Software as a Service), Cloud Computing, and all other architectural approaches that depend on “services”.” According to her, SOA was complete disaster and SOA didn’t fulfill the expectations of organizations, she says that although the word “SOA” is dead, the requirement for service-oriented architecture is stronger than ever.

She emphasizes on the importance of architecture and services instead of just SOA. A successful SOA as she says” Successful SOA (i.e., application re-architecture) requires disruption to the status quo. SOA is not simply a matter of deploying new technology and building service interfaces to existing applications; it requires redesign of the application portfolio. ” Here we have also keep this in our mind and rest of the documentation contains enough information why we need SOA and how SOA can be helpful for organizations even in worse economic crisis. It may be possible that an enterprise never used 100% SOA. SOA is a part of Service-oriented computing and it offers different services. So an enterprise may not be interested in all of the services provided by SOA. It depends on the needs of that enterprise.

SOA is a part of distributed system and in order to understand SOA we must understand the characteristics of large distributed systems. The large system like SOA is designed for legacies. They must deal with existing systems. In practical there might be no organization that wants to implement SOA from scratch and forgot about the legacies. We can say that SOA is an approach for maintenance of large systems. Large distributed are heterogeneous in nature. These systems have different purpose of implementations. These systems are implemented in different programming languages and different platforms. Large distributed systems are very complex and change in one place can affect many places. SOA is an approach to handle these characteristics.

So according to the [11] SOA is a paradigm for realization and maintenance of business processes that span large distributed systems. SOA in technically consist of three concepts, services, loose coupling and Enterprise service bus which is used for interoperability. A service here represents the functionality of a business. Services concentrate on the business value of an interface and that’s the reason SOA is a bridge between IT and Businesses. And ESB (enterprise service bus) is middleware architecture for implementation of SOA. It enables interoperability between systems for services. An ESB can distribute the message over different systems regardless of platforms and different technologies. Loose coupling means to minimize the effects of modification and failure in systems. Systems dependencies are reduced in loose coupling.

According to Gartner Hype of emerging technologies in 2009 SOA is at Slope of enlightenment.
Above figure shows that SOA is a mature enough to adopt by any organization.

A typical architecture of SOA consists of Portals/user interface, Orchestration, ESB, service layer, business layer and application database. See figure 3.2
3.2 SOA Platform, Tools and Technologies

3.2.1 SOA Platform: According to [12] An SOA platform is a development platform for SOA implementation which provides the tools, technologies and software infrastructure that makes a business possible to use the SOA.
SOA platform is a composition of different technologies. SOA platform vendors like Microsoft and JBoss provides different components for their platform. But usually they are same; it may possible that they are with different names.

3.2.2 Services: “A service is a software component that is described by meta-data, which can be understood by a program” [13]. There are different ways of defining services as different services consist of different attributes. In a more business context, services can differ a lot depending on the different business entities. Services can be used to read data, write data or as a composition of services to perform some workflow. Therefore, these services cannot be looked at and treated in the same manner, it is better to categorize them according to their attributes and functionalities.

As explained by [14], technically services can be categorized in the following three categories:

1. Basic Services
2. Composed Services
3. Process Services

Basic Services

Basic Service is the simplest service and it only provides basic business functionality only. In other words, they only provide a service of reading or writing to one specific backend. Their lifespan is short and are usually are stateless services. Basic Services are further divided into two types:

i) Basic data service: This service is associated a specific backend and they perform reading and writing operations to that backend only. Moreover, as these services perform reading and writing operations they must incorporate ACID properties.

ii) Basic logic service: These are similar to basic data service the difference is that these services incorporate basic business rules and logic.

Composed Services

Composed services are composition of either basic services or other composed services. Composed services in the context of SOA are also referred as orchestration services. These can be considered at a higher level than basic service as their lifetime is short and usually are stateless. These run usually within one business process.

Process Services

Process services usually represent workflows or business process. These are activities/services which run for a long term within one workflow or business process. As these are long term running services they need to maintain a state (e.g. shopping cart). These states might need to be maintained over multiple sessions, and they also help in failover activity when the process state of a process needs to be recorded to start over again from the same state after recovery.

Working of Services in SOA: In SOA there are usually three roles which come into play: requester, provider, and broker.
• **Service Provider:** The service provider is the entity which has access to other services, and is also responsible for creating services and publishing them to the service broker.

• **Service Requestor:** The service requestor is the entity which basically requires some operational work performed by another service and works by search through the list of service descriptions provided by the service broker. It is also responsible for binding the services after their discovery.

• **Service Broker:** Service broker has the information about all the services which are registered within the ESB and are responsible for redirecting requests to the corresponding service requestor and provider.

![Figure 3.3](https://example.com/figure3.3.png)

*Figure 3.3
Services in SOA

Source: [15]*

### 3.2.3 Enterprise Service Bus (ESB)

According to [16] “an ESB is a software architecture construct which provides fundamental services for complex architectures via an event-driven and standards-based messaging engine (the bus).” An enterprise service bus (ESB) allows using the services in a productive landscape. ESB is very important part of SOA. In a SOA environment we need a way to call services; ESB is there for this purpose. An ESB is responsible for enabling the consumers to call the services providers supply. According to [11] ESB is responsible for the following tasks

1. Providing connectivity to the systems.
2. Data transformation between different systems.
3. Routing or intelligent routing. This can be content base routing which is to route the message from one system to another on the basis of its contents.
4. Security and reliability
5. Service management monitoring and logging
Although ESB provide all these functionalities but the main purpose of an ESB is provide interoperability. An ESB is used to integrate different platforms and programming languages. Routing and data transformations are two main properties of an ESB. As ESB is designed to integrate the different platforms and programming languages so when a systems sends a message to another system using ESB it is possible that source and destinations are based on different programming languages and with different data format. Here an ESB is responsible to transform the message which is acceptable for the destination system. Routing is used to send a call from consumer of services to the provider of services and also sending an answer back to consumer. Below is the typical structure of an ESB in SOA environment.

![Figure 3.4](http://www.btplc.com/innovation/journal/BTTJ/current/HTML_Articles/Volume26/04Orchestrating/Default.aspx)

3.2.4 Business Process Management (BPM)

BPM is the heart of SOA. Services in SOA represent one or more business processes. Services are designed for one or more business processes, but how to identify the services that which business entity they are representing. According to [11] when we want to identify the services we need business process management (BPM). Basic purpose of BPM is to manage and improve the business processes. According to the [17] Business process management attempts to improve processes continuously. There for it can be described as a "process optimization process." It is believed that BPM enables organizations to be more efficient, more effective and more capable of change than a functionally focused, traditional hierarchical management approach. As services are representing business processes we need to bring services into play. According to [11] the business process management includes the analysis of business which includes needs and opportunities, implementing...
and integrating business strategies, optimizing and monitoring business processes and aligning the IT with business. There are some issues which one should keep in mind before the System design. First of all you should know that how can we break the business unit into small parts and implement them. And how can we design these services in a generic way to reuse them in different places. As [11] explains that there are two approaches to deal with these kind of issues one is top down approach in which we can compose a system, problem or process into smaller pieces until we reach the basic level services. Other approach is bottom up in which we can build business process by composing services into more general pieces. So which approach we should follow? In practice mixture of both the options is appropriate. For example we can decompose our processes in top down approach and bottom up approach for exposing existing systems as services.

Using business process management we can identify the services that can be part of our SOA implementation for business processes. For designing and executing the business process there are multiple standards, which allows using different tools and standards. BPEL (Business process execution language) is one the most important engine for executing business processes. BPEL can define the processes that can serve as composed or process services using web services technologies.

**Orchestration and Choreography**

Orchestration and choreography are the part of BPM. An orchestration is designing services and processes by composing existing services. In a typical orchestration process there is a central control which coordinates the activities in the process. There is only one party which is responsible for the service. When we have more than parties collaboration which can responsible for one or more steps in a process and none of them understand the whole process. This is called Choreography.

**3.2.5 Presentation (Portals)**

According to [18] it is very important for any organizations to understand the basic steps and available technologies which can be involved for implementation of SOA architecture. For this reason portals can be a first step in the implementation of SOA. This is one of the problems for any organization to identify the first step to start SOA. A portal can be a good option to start with. According to [19] “The portal can be first logical step for an organization to think for SOA implementation because its fundamental nature lends itself to SOA approaches”.

In simple portals are the user interfaces, they are evolving from the owner of the services to the consumer of the eservices.
3.2.6 Registry

We need registry to register the web services. As purpose of the registry is to provide the services address so that we can access the services.

According to [20], the implementations artifacts that drive from a SOA should be registered within a repository to maximize reuse and provide for management of enterprise assets.

According to the [21] the definition of SOA registry is, “an SOA registry is a resource that provides controlled access to data necessary for governance of SOA (service-oriented architecture) projects. In effect, it is a constantly evolving catalog of information about the available services in an SOA implementation. An SOA registry allows businesses to efficiently discover and communicate with each other using Web services”.

Figure 3.5
The goal of the SOA registry is to have reliable and fast communication among applications and to also minimize human involvement. Figure 3.6 shows the working of a registry.

*Figure 3.6*

4. Vattenfall Needs
This chapter is all about Vattenfall Needs and its Technical Reference Architecture. Explanation of
Vattenfall needs with respect to TRA. We will also consider a Case study depicting how we can map
out SOA components in Vattenfall. All the information regarding TRA and Vattenfall Nordic is has
been taken from the internal Vattenfall Nordics documents [1].

4.1 Technical Reference Architecture of Vattenfall
A set of generic building blocks which together delivers the capabilities necessary to create business functionality based
on service orientation. [1]

Purpose of Vattenfall TRA: Purpose of the TRA is the template for Service Oriented solutions
within Vattenfall. This is the standard for SOA implementation within Vattenfall. If Vattenfall is
going to implement SOA then they have to follow this standard. Enabling mapping of the current
and target architecture. Also to enable the mapping for legacy applications and future applications
based completely on SOA. To discuss the technical architecture, it creates a common language for
example white spots or overlaps.

Characteristics in Vattenfall: The reference architecture should support the strive to move from
application integration towards service orientation and be able to visualize the similarities and the
differences between these. Then the core part of TRA is a number of different building blocks
which can physically deploy as hardware or software. Some of the building blocks such as security
policy, design directives, service identification method and model are intangible. Their importance is
equal with other building blocks in order to achieve the goal and objectives of SOA. Another very
important thing is that all components might not require at all or at the start. SOA building blocks
should be well defined and described. Then the importance of the connecting of the building blocks
is as equal as the building blocks themselves. TRA can also be defined in different scope like
federation level might be group wide as well as locally. TRA is Vattenfall should not be industry
specific which means it should not be specific to utilities. Then it is also possible that TRA could be
based on other technology other than web services like Java messaging service (JMS), MSMQ and so
on. TRA should also support all the aspects of SOA for example Governance, design, construction,
test and runtime.

Then one of the most important features of TRA is that it can be used to compare the different
solutions from different vendors or service providers. This is what we will do, comparing the SOA
platform provided by JBoss, Microsoft and Oracle.

Service Orientation is a new concept for IT solutions, where all the building blocks of the logical
architecture are necessary to support the whole.
Figure 4.1 shows the Technical Reference Architecture within Vattenfall

4.2 Meter reading Case Study
This is the use case within Vattenfall. The purpose of the use case is to study the Service Oriented Architecture (SOA) for implementation of this use case.

In figure 4.2 we can see the diagram of the use case.

Use Case Diagram
As you can see in the figure 4.2,

1. A market actor/a customer/a seller require a meter reading for a specified date and time.
2. The request is going to an enterprise service that is located on central Enterprise Service Bus (ESB).
3. The Enterprise service gets data from the Key database that has information about the Point of delivery (POD) address.

4. The Enterprise Service Access the right country service that provide information about local PoDs and their Meter Readings.

**Sequence Diagram for the use case**

Figure 4.3 shows the Sequence diagram for the use case.
As we can see that a user can be a market actor, seller, or customer who is interested to get the meter reading for a specified date and time. First user will specify the date and time for which he needs the meter reading and then there will be a service which will be directed to ESB, the ESB is the very important part in our use case. ESB will first get the address of the Point of delivery (POD) from the Key database.

Requirements for use case

The main parts of SOA which we need for this use case are

1. Portals (Presentation)
2. Enterprise service bus (ESB).
4. Registry

Further we will explain each of the specified part of SOA and how we can use to solve this use case.

Portals in our use case

Portals will be the first thing which will be available to the end user. User can interact with the portal. User will specify the parameters to the available portal for which he want to see the meter reading. Then portals will be interacting with the ESB.

Enterprise Service Bus

An enterprise service bus is considered as a middleware which is laid between business applications and transforms and route messages. And ESB acts as a service bus. An ESB eliminates the need of point to point system connectivity. In general when a system wants to send a message to another system it sends message to the ESB and then ESB is responsible for sending that message to its appropriate destination.

ESB for the use case

Main purpose of the use case is to study the use case and to solve it with the help of SOA. In our case ESB is the main part which will be used for Transformation and routing purpose.

Using ESB the messages exchanges between different services is possible. For example a customer’s request if received by the ESB and then transformation is applied if its needed, in next step this request will be routed to the desired KeyDataBase which contains the address of the PoDs, Then Country Service will take this POD as input and will get the information from Country data base. As you can see in next section that we have three kind of services, KeyDataBase Service, to get the information of POD address from KeyData base. The NordicData Service and EUData Service are country Services, for Nordic region and Europe respectively. ESB is responsible for receiving requests from customers, transforming messages, routing them to right place and to give the meter reading to customer with the help of portals.

Services for the use case
For this particular use case we would mostly use basic services and a few composed services. The services which are identified are as follows:

**Basic Services:**

1) KeyDataBase Service
2) NordicData Service
3) EUDataService

**Composed Services**

![Diagram of service flow](image)

**Figure 4.4**

**Basic Services Attributes and Parameters**

**KeyDataBase Service:**

This service would be responsible to retrieve the POD Id from the KeyDatabase which contains POD ID’s for both EU and Nordic region.

Inputs: Customer Social Security Number/ Personal Number

Outputs: POD ID against that Social Security Number/Personal Number
**NordicData Service:** This service would act as a basic service and as provided with the POD ID would return whether this customer exists in the Nordic region or not.

Inputs: POD ID to check whether customer exists or not

Outputs: Some customer information can be retrieved or only a Boolean values of yes or no.

**EUData Service:** This service is same as NordicData service and it would provide information about the existence of customer in EU region or not.

Inputs: POD ID to check whether customer exists or not

Outputs: Some customer information can be retrieved or only a Boolean values of yes or no.

**Composed Service Attributes and Parameters**

**GetCustomerInformation Service:** This service would act as a composed service as it would be responsible to make a call to both Nordic and EU Data service simultaneously. The communication would be asynchronous and the responses would be recorded accordingly.

Inputs: Customer POD ID which is returned by KeyDatabase service.

Outputs: Customer residing in which region Nordic or EU + any other required customer information.

**Mapping Use case with SOA.**

The figure 4.5 shows the SOA representation of the use case.
We have successfully mapped the use case with the SOA tools and technologies. It means that for meter read case we need Portals, ESB, Registry and services. This aim of this case study is to just show that how SOA can be mapped into Vattenfall needs. These needs we will see in next section, which are consist of TRA and some other parameters.

4.3 SOA Guidelines for Vattenfall
Following are the Vattenfall needs which we have identified with the help of Vattenfall Nordic. These needs are based on TRA and other needs which are not part of TRA.

Vattenfall TRA specific “Needs”

1. **Service Creation and Abstraction:** Enabling applications to participate in a SOA. Encapsulate existing applications and data sources into standardized services, which utilize a standardized
interface technology (e.g. web services or JMS). SOA connection, Transformations and Routing are three basic sub parts of Service creation and abstraction.

2. **Run-time Service Management**: Execute discrete Services in a controlled way according to defined policies. Service throttling, runtime statistics and governance, Dynamic service ramp-up and security policy fulfilment is part of run time service management.

3. **Service Orchestration**: Combines basic Services into more complex and business aligned Services. The composition Services can make on several levels; from short-term technical processes (e.g. managing asynchronous behaviour or aggregation of data) to execution of complete long-running business processes, including compensation mechanisms. Process control, workflows and business rules execution is the part of service orchestration.

4. **Process Control and Optimization**: Create feedback to business stakeholders on how executed processes perform. Business activity monitoring (BAM) and reporting tools are parts of process control and optimization.

5. **Presentation**: Provide a user interface to the business logic created in the other layers. Rich graphical interfaces, personalization and syndication are parts of presentation or portals.

6. **Identity Management**: Provides a structure and mean to safeguard the resources in the Service Oriented Architecture from unauthorized usage. Identity management has authorization, authentication, role management and auditing as its sub parameters.

7. **Design-time Service Management**: Control that design-time activities are performed in accordance with governance rules. It includes service repository/service registry and life cycle management versioning of services.

Then there are some “Needs” which are not part of TRA but are very important for Vattenfall. We have analyzed and find out the following function blocks

**Vattenfall “Needs” not part of TRA**

8. **Rapid application development (RAD)**
   RAD is the software development planning which require minimum amount of planning for developing the software. We are interested to see whether RAD is provided by the Redhat and JBoss or not?

9. **Documentation**
   If Vattenfall decided to go with one of the vendors for SOA implementation the documentation will play very important role. In our case we are interested to see whether documentation for all the tools and technologies provided by our selected vendors is enough to understand the concept about them.

10. **Support**
    Again support is very important for Vattenfall in a sense that Vattenfall is very large organization with huge amount of existing systems. And they need quick support with minimal effort. 24x7 support is needed for Vattenfall. So that’s why we considered this parameter to be analyzed.

11. **Development platform**
    Which development platforms are supported by the JBoss and Microsoft for implementation of SOA platform? Are they Java based or some other programming language? This will help us to see the competence level within Vattenfall Nordic.

12. **Cost and licensing**
    First time cost and licensing cost is very important for Vattenfall. We will find out the actual amount which Vattenfall has to pay if they decided to go with JBoss or Microsoft.

13. **Adapters**
Adapters are used to integrate the different systems and platform. For example currently Vattenfall is using SAP systems, so it is important to see whether Microsoft and JBoss provided adapters for SAP and other applications.

14. **Integration with existing systems**
Basic purpose of using SOA for an organization like Vattenfall is to integrate the existing systems as well as support for new systems. So it’s important to see how we can integrate with existing systems using JBoss and Microsoft.

15. **Open source?**
One of the most important parameter is to find out that is the selected tools and technologies are true open source? In a true open source you can have actually the code of the products. There is no vendor lock in. You can actually modify the source code according to your needs.

16. **B2B transaction**
Business to business (B2B) transactions is commerce transaction between business and enterprises. These transactions are different than the transactions between businesses and customers. B2B transaction is important for an enterprise because they have to make transaction with other businesses and partners. So are they provided by JBoss and Microsoft?

17. **Deploy and un-deploy efforts**
This is also very important that how much is deploy and UN-deploy efforts for the selected technologies.

18. **Scalability**
Is it possible to handle the fast growing of business with the help of either JBoss or Microsoft? This is called scalability. Also in case of failover, do they handle this?

19. **Standards**
SOA is standard based. Do JBoss and Microsoft follow those standards? We will investigate to find out this.

20. **Governance platform**
Advantage of SOA over EAI systems is that SOA gives an overall governance of the whole system. We will try to find out that whether this governance is provided by the JBoss and Microsoft.

21. **End to End business process monitoring**
It is very important to align all the aspects of an organization with the needs of customers. BPM provides this functionality. We are also interested to see the end to end monitoring of business and workflows.

These are the Vattenfall needs for which we are interested to know that which vendors among Microsoft and Redhat fulfill these needs. Our main research is based on these parameters.
5. SOA Vendors

5.1 Redhat JBoss

Before going into detail of JBoss Redhat SOA platform it is important to know that this information has been taken from the official JBoss website and Guides for programmers at JBoss website [22]. JBoss is a division of Redhat Inc. JBoss specializes in open source enterprise middleware software. JBoss uses the service based model for company’s profit. For our purpose of thesis research we are interested in JBoss because it provides the platforms and middleware for implementing SOA. Here I have collected the technical information of JBoss SOA platform. Technical information of the JBoss contains the information about SOA architecture and all the technical features provided or supported by JBoss. Below is the figure of JBoss Enterprise Middleware.

![Figure 5.1 Source: www.redhat.com](image)

According to JBoss official website [23] “JBoss enterprise middleware platform enables organizations to do more. Start and finish projects. Deploy more application with cost effective way”. There are three basic parts in it, one it design and develop, second is to run and execute and third is the management, administration and monitoring. For our basic purpose we are more interested in the JBoss Enterprise SOA platform which is a part of JBoss Enterprise Middleware platform.

5.1.1 JBoss Enterprise SOA platform

It is an open source, flexible, standards-based platform to integrate applications, SOA services, and business events. It also allows automating business processes. Below is the figure of JBoss
JBoss Enterprise SOA platform is consisting of JBoss business process management (jBPM), JBoss rules, JBoss ESB and JBoss Enterprise Application Platform. I have explained each part in more detail in rest of the chapter.

5.1.2 JBoss ESB

JBoss ESB is a next generation EAI, without vendor lock-in. An ESB is like not SOA. An ESB does not provide a Service Oriented Architecture, but it does provide the tools than can be used to build one—especially loose-coupling and asynchronous message passing. SOA is a series of principles, patterns, and best practices [24]. Everything is a message inside JBoss ESB.

Figure 5.3 shows the typical structure of JBoss ESB. JBoss ESB provides Orchestration, protocol translation, adapters, repositories, management, quality of service, transformation and content based routing.

The information available in the ESB programmer’s guide [25] explains that we need to understand whether or not our application clients and services are “ESB-aware” or, in other words, whether they can understand the message formats and transport protocols used by the ESB. If you (as a programmer) are ESB-aware, then you can simply code any new clients and services that you write to also be ESB-aware. ESB-aware clients and services communicate with each other using messages. ESB-aware services are identified using Endpoint References (EPRs).
Interoperability with gateway adapters
supporting interoperability between different clients and services is one of the goals of JBoss ESB. One of the ways that JBoss ESB makes this interoperability possible is though gateway adapters.

![Diagram](http://docs.jboss.org/jbossesb/presentations/JBossESBoverview.pdf)

Figure 5.3
Source: [http://docs.jboss.org/jbossesb/presentations/JBossESBoverview.pdf](http://docs.jboss.org/jbossesb/presentations/JBossESBoverview.pdf)

According to JBoss Programmers guide [25] everything is either a message or a service and a gateway is a service which acts as a bridge between an ESB-aware and an ESB-unaware client and service.

In JBoss ESB Gateways translates the information between ESB-message and non-ESB message formats and EPRs. According to [25] following Gateways are supported by JBoss ESB

- file gateways: local filesystem, ftp, sftp and https
- JMS
- HTTP/HTTPS
- email (POP3)
- SQL table
- Hibernate

**JBoss ESB deployment**
JBoss ESB can be deployed on the standard JBoss application server or JBoss ESB server and is bundled with the JBoss SOA Platform. We need to check that the system should meet the following minimum requirements:
1. JDK 5
2. Ant
3. JBoss Application Server or JBoss ESB Server

There are three ways to run JBoss ESB. You can deploy it to JBoss AS / JBoss ESB Server, run standalone, or deploy it to Tomcat. But it is recommended to use JBoss ESB server. According to [26] The JBoss ESB Server is a lightweight server to deploy JBoss ESB. JBoss ESB server is only used to deploy the JBoss ESB for faster processing, because of lightweight. On the other hand JBoss application server contains other information as well which is not required by the JBoss ESB. Quicker boot time of JBoss ESB Server has advantage over JBoss Application Server, which is helpful during development.

JBoss ESB supports a bottom-up approach to building a service, so we did not need to make the schema for these services.

Apache ant:

There is no automatic way to build the JBoss ESB at the moment. The only way we can Build the JBoss ESB is with the help of Apache ant. Apache ant is based on Java library and command line tool to execute the commands. So the reason I mentioned about Apache ant is because we are using this tool to build and deploy the JBoss ESB.

5.1.3 Services in JBoss SOA

According to [25] Services encapsulate either the business logic or the points of integration with legacy systems.

What is Service?

In the JBoss Enterprise Service Bus [25], a service is defined as "a list of action classes that process a Message in a sequential manner". Action pipeline represents the list of action classes on which definition is based. A Service can also define a list of listeners. Listeners work as a router for the service to route messages to the action Pipeline.

5.1.4 JBoss Enterprise Business Rule Management System (BRMS)

According to [27] an open source business rules management system provided by the JBoss Enterprise BRMS. This BRMS makes it easy to make business policies and rules for the management. A highly fast and efficient rule engine is used by the JBoss BRMS. Ultimate goal of JBoss BRMS is to make it easy for the business analyst or auditor to view and manage business rules as implemented or documented in current IT infrastructure. According to [28] the environment for the rule engine execution can be on certified JSE and JEE platforms. It can also be executed at the JBoss Enterprise Application Platform (JBoss EAP), as well as the JBoss Enterprise SOA Platform.

We can have JBoss without BRMS, but usually we get a bundle which includes BRMS. We can use jBPM library to call rules, sometimes it uses Hibernate to map business entities so there is a lot of stuff which falls under this I am still exploring this domain in detail.
5.1.5 JBoss Business Process Management (jBMP)

According to [29], The JBoss jBPM is a workflow and Business Process Management (BPM) engine. jBPM makes it possible to create the business processes that can be used to coordinate with or between people, applications and services. JBoss jBPM combines development of workflow applications with a process engine. The JBoss jBPM represents processes graphically. This is the reason that the gap between a business analyst and technical developer is reduced. As shown in figure 5.4

![Diagram of a business process](http://community.jboss.org/blogs/tom.baeyens)

**Figure 5.4**  
*Source: http://community.jboss.org/blogs/tom.baeyens*

JBoss jBPM includes the following components: Runtime engine as a POJO library, a graphical designer as an Eclipse plug-in, persistence based on hibernate, a web console in JSF, a BPEL extension for orchestration, scheduler for timers, business calendar, and more.

It offers BPM and workflow features in a way that both business users and developers like it. So it aligns the Business people with IT.

5.1.6 Supported Database

According to [30] RDBMS’s supported by the JBoss Enterprise SOA platform are

- Oracle
5.1.7 Monitoring
There are two different monitoring and governance parts in JBoss Enterprise SOA platform. In case of JBoss application Server applications Manager Helps for JBoss Server’s monitoring performance, availability, and usage statistics. As described in [31], it provides detailed information on every component or service deployed on JBoss. Monitoring JBoss involves automatic diagnoses, notification, and correction of performance and availability problems in the JBoss Servers and the services deployed in them. Application manager is a full-blown web based product. It is not free and does not fall under Redhat; it’s a product by Manage Engine.

For monitoring of JBoss Enterprise Middleware suite, JBoss Operations Network (JON) is included in the JBoss enterprise SOA platform. JON simplifies the discovery, deployment, testing, development, and monitoring of applications life cycle. It is also used for end-to-end applications management.

5.1.8 Supporting operating system
As described in the [32], all the leading operating systems are supported by the JBoss Enterprise SOA platform. Red Hat Enterprise Linux, SUSE Linux Enterprise, Microsoft Windows, Solaris, HP-UX, AIX and z/OS are fully supported in both physical & virtual environments.

Window version and Linux kernels
- Red Hat Enterprise Linux 5 (latest update)
- Microsoft Windows 2003 SP2
- Microsoft Windows 2008 R2

According to [33], the system requirements for JBoss SOA platform are as follows

- JDK 5 or higher (recommended and required when using EJB 3.0)
- 512 MB RAM
- 100 MB hard disk space
- 400 MHz CPU

5.1.9 Registry
Registry is built in inside the JBoss ESB. According to [34], a central role is played by the registry in the JBoss ESB. End point references are stored in the registry within the ESB for service deploy. When services first start-up, it can be updated dynamically, or statically by an external administrator.

5.1.10 JBoss Portal platform
JBoss portal platform offers the rich web interfaces, GUI for the users. JBoss portal is a part of JBoss Enterprise Middleware suite. It’s not included in the JBoss Enterprise SOA platform. But it comes with JEMS. According to [35], JBoss Portal provides an open source platform for hosting
and serving a portal's Web interface, publishing and managing its content, and customizing its experience.

5.1.11 JBoss SOA governance

According to [36] SOA Software’s products offer a comprehensive Integrated SOA Governance Automation solution for JBoss

<table>
<thead>
<tr>
<th>Redhat Product</th>
<th>SOA Software Added Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>JBoss Enterprise Application Platform</td>
<td>Governance federation Metadata management Mediation for interoperability Policy enforcement, implementation, and monitoring Security policy implementation and enforcement Automated PKI key distribution</td>
</tr>
<tr>
<td>JBoss Enterprise SOA Platform (including ESB)</td>
<td>Governance federation Metadata management Mediation for interoperability Policy enforcement, implementation, and monitoring Security policy implementation and enforcement Automated PKI key distribution</td>
</tr>
<tr>
<td>JBoss jBPM</td>
<td>Governance Federation Dynamic policy implementation Abstraction of applications from complex policy operations</td>
</tr>
</tbody>
</table>


JBoss doesn’t offer any special or customized product for SOA governance.

5.2 Microsoft and SOA

This section which is about “Microsoft and SOA” is done by “Adnan Gohar” [37]. It’s important to put this section into my report because of analysis purpose. I have contributed partially in this section.

Microsoft has long been associated with providing enterprise application integration (EAI) tools and platforms. It has concentrated on the “middle out” approach for integration of different systems, as for SOA they do not provide any dedicated platform or product. SOA can be achieved in Microsoft by a combination of different products. Microsoft has been working with the concept of service orientation since 1999, when the Web services model was released, which had great impact on the way application architectures were designed. Later on, with release of .NET Framework and SOA tools, Microsoft products became SOA compliant. Since then, Microsoft has concentrated on providing enterprises of all sizes with solutions which are SOA based and which aim to optimize business processes and strive for greater business agility and quicker time to value.

Microsoft has been concentrating on web services for the service part of SOA. For this it has been working with vendors such as IBM and BEA to formulate a collective and standardized WS-*
architecture. Web Services Interoperability Organization (WS-I) [38] was formulized with the collaboration of Microsoft and IBM to support interoperability across platforms, operating systems and programming languages.

Microsoft has been using service orientation across its entire technology stack, ranging from developers tools integrated with .NET framework for the creation of Web Services, to server products such as BizTalk Server, Windows Server Appfabric and Microsoft Share Point which use these web services by connecting and orchestrating them to process business flows. Moreover, these web services can be consumed by different composite applications running on Intranet, Internet or by rich client applications.

**Microsoft Application Platform SOA**

Microsoft has two main products which facilitate the implementation of SOA. The main products involved in implementing SOA are:

**Microsoft BizTalk Server and Windows Server Appfabric**

![Figure 5.6: Microsoft BizTalk Server](image)

**5.2.1 Microsoft BizTalk Server**

According to [39], BizTalk Server is Microsoft’s main integration platform which supports capabilities of connectivity, native messaging, broker, service orchestrations and business process executions. BizTalk was originally designed to cater enterprise application integration issue, but has
evolved from being an integrator to a product which now supports comprehensive set of technologies that facilitate in creation of SOA based systems. It provides support for connectivity to platforms, applications, and devices to consume, expose and compose new services. In addition, is also provides service inventory architectures which are built using orchestration and enterprise service bus style platforms. Moreover, it includes tools which enable enterprises to expose their business processes as services, and gives the capability to connect to both proprietary and standard based systems. This makes BizTalk a central part of SOA strategy provided by Microsoft.

**Features of BizTalk Supporting SOA**

BizTalk Server provides the following features either independently or by simple addition of components which help in implementation of SOA:

**Enterprise Service Bus Toolkit:** Microsoft ESB Toolkit is an add-on component for BizTalk Server, which helps in achieving functionality of an enterprise service bus (ESB) and allows BizTalk to support a loosely coupled messaging architecture for service-oriented architecture (SOA) applications.

**Business Rules Framework:** Microsoft BizTalk Server includes the Business Rules Framework as a stand-alone .NET compliant library, which has different modules and components.

**BizTalk Orchestration Engine:** The BizTalk Orchestration Engine is responsible for coordinating and scheduling message processing and performs complex logic on the message as it is passed through an orchestrated business process.

**Business Activity Monitoring:** Business Activity Monitoring (BAM) is a fundamental part of Microsoft BizTalk Server, giving it the capability to track and monitor information generated by different business processes.

**BizTalk Messaging Engine:** BizTalk Messaging Engine is responsible for receiving inbound messages, parsing and identifying message formats, evaluate routing criteria, delivery to destination and status tracking.

**Business to Business Integration:** BizTalk provides different capabilities which simplify the integration of business processes with external enterprises. BizTalk Server natively provides support for Electronic Data Interchange (EDI) data protocols and Applicability Statement 2 (AS2) protocol.

**5.2.1.1 Microsoft AND Enterprise Service Bus**

As explained by the [24][40] For some time enterprise service bus was an architectural pattern for Microsoft and not a separate product. In Microsoft’s perspective the function provided by an enterprise service bus could be achieve by different Microsoft products including BizTalk Server and WCF. As the concept of ESB got popular Microsoft released its first ESB Guideline in 2006. This ESB Guideline provided an architectural guidance that described how to achieve ESB functionality using Microsoft Platform. This guideline shaped up into a component for BizTalk Server known as the ESB Toolkit.
Microsoft doesn’t offer a standalone enterprise service bus; it comes as an add-on for BizTalk Server and is known as the BizTalk ESB ToolKit.

**Microsoft BizTalk ESB Toolkit 2.0**

As described by [39], [41] Microsoft ESB Toolkit is a set of components, services and architectural guidance which extends the capability of BizTalk Server to start acting as an Enterprise Service Bus. These new capabilities allow creation of service-oriented applications that incorporate itinerary-based service invocation, dynamic resolution of endpoints, WS-* integration, exception management and auditing, and SOA governance using third party tools.

![BizTalk ESB ToolKit integration and components](image)

*Figure 5.7: BizTalk ESB ToolKit integration and components [25]*

**Services of Microsoft ESB ToolKit**

According to [39], [41], [42] The Microsoft ESB ToolKit provides the following core services:

**Resolver service:** The resolver service is responsible for providing capability of endpoint resolution at runtime and dynamic routing. It acts as a service registry making services available in a heterogeneous environment. ESB Toolkit provides endpoint resolvers for the following: XPath, UDDI 3.0, UDDI 2.0, Static, WS-MEX and BRE. In addition, it also provides resolvers for other artifacts such as, Static (for itineraries), BRE (for itineraries), Composite, BRE (for maps) and SQL Server.

**On-ramp service:** This service promotes loose coupling by providing dynamic message transformation and translation of messages. The web service consumer only sends messages (SOAP, WCF, JMS, WMQ, FTP, HTTP, Flat File, XML, or any other custom formats) to the ESB and they automatically get transformed to BizTalk compliant messages.

**Itinerary services:** These include orchestration-based and messaging-based services for performing itinerary-based routing. It also allows creation of custom services for itinerary-based routing.
Transformation service: The transformation service helps in protocol transformation i.e. the services hosted on different platforms can interact with each other using different protocols including WS-* standards for Web Services. This transformation also helps non-BizTalk applications to access and utilize BizTalk capabilities.

Exception Management service: ESB Toolkit provides an exception management service allowing non BizTalk applications to publish faults and errors using the default BizTalk exception management mechanism.

BizTalk Operations service: This service provides information about the current state of BizTalk artifacts.

5.2.1.2 Microsoft BizTalk Business Rules Framework
According to [43], [44] Business rules are conditions on which decision is made in business processes, they can be considered as conditions that determine which business action needs to be performed on the basis of values in factual variables. In conventional application design, business rules are implemented using methods. This makes it difficult to use them in other applications and business processes.

Service Oriented Architecture (SOA) has changed the way applications are designed and created. SOA changes the way applications interact by introducing messaging as the way of communication and every decision is made on the basis of message content. Applications that are prone to constant change in their business really benefit from SOA. The separation of business rules from code makes them easily accessible by other applications and processes. Moreover, it increases business agility and business processes become more flexible to change and adaptive to new requirements.

Microsoft provides the Business Rules Framework as a stand-alone .NET compliant library, which is also by default included into BizTalk Server. Business rules can also be integrated in orchestrations to extend their functionality e.g. changing business logic can be encapsulated into rules, rules can be used to determine variable delays and they can also be used to determine execution path of a business process.

Modules of BizTalk Business rules framework
As explained in [43], [44] Microsoft BizTalk Rules Framework consists of the following modules

Business Rules Composer: The Business Rule Composer is used to create different rule vocabularies. These vocabularies are user-defined names for the facts that are used in rule conditions and actions. These vocabularies can be bound to different data sources e.g. .NET class members or XML schema or database elements. Moreover, it allows to version vocabularies which makes it easy to audit and manage. Composer can be used by developers, administrators and information workers to publish vocabularies.

Run-Time Rule Engine: The Run-Time Rule Engine is responsible for processing rules created as declarative statements, grouping rules together that combine to form a policy without rearranging
them or redefining them and resolving conflicts among rules and allowing support for forward
chaining of rules.

**Business Rule Policy:** Business rule policy is a logical grouping of different business rules. Every
rule in this group corresponds to a condition, which can be evaluated to true or false and if it results
in true the corresponding rule actions are performed. This allows the redeployment of the rule
policy only if a change occurs in any of the business rules used within it and no need to redeploy the
whole orchestration.

**Rules Engine Deployment Wizard:** The Rule Engine Deployment Wizard facilitates in
deploying/un-deploying rules/policies. It allows importing/exporting of a policy or vocabulary from
SQL rule store to an XML file and vice versa.

**5.2.1.3 Microsoft BizTalk Business Activity Monitoring**

According to [45], [46] Monitoring of business activities and generating information regarding
business processes can prove to be vital for the success of a business. The demand for right
information at the right time has increased and businesses require quick and accurate information
from their underlying integration solution. This information enables enterprises to make timely
decisions for the improvement of their business processes and services provided to customers.

Microsoft caters this business need by providing a business activity monitoring tool. Business
Activity Monitoring (BAM) is part of Microsoft BizTalk Server, giving it the capability to track and
monitor information generated by different business processes. BAM provides information about
business processes state, usage trends and critical conditions within the system of process. In
addition to this, BAM also provides an application programming interface (API) that enables
developers to monitor information which is not in the scope of BizTalk processes

**Features of BizTalk Business Activity Monitoring**

As explained by [45], [46] BizTalk Business Activity Monitoring gives the following features

**BAM Activities:** BAM activities are units of work which are used to gather business data. These
activities can also be used to see the overall progress of a business process. As these activities are for
monitoring purpose they do not affect the BizTalk solutions. In addition, activities can be created
for multiple systems and also be related with each other. BAM facilitates to generate one view of
such an orchestrated business process.

**BAM Views:** BAM views helps in organizing and displaying information collected through BAM
activities. BAM views can differ depending on context, as one business process can be initiated by
multiple users and there would be only one BAM activity against it, to distinguish among them but
there could be multiple BAM views displaying information according to context defined.

**Aggregating and Filtering Data:** BAM provides different interceptors for collection of
information from pipelines, messages and business processes orchestrations. Using its aggregation
capability it can predict usage trends within a business. These trends can be viewed using BAM
views, which can help enterprises in making critical decisions. It also provides the capability of filtering data i.e. users can request for data according to their requirements.

**BizTalk Business Activity Monitoring AND SOA**

According to [46] SOA is all about distributed processes and services running on disparate systems, full process execution visibility become a vital component in managing this infrastructure. Microsoft BAM facilitates developers to create a single activity for monitoring huge business processes. This helps in implementing a single view for information about the process execution rather than using different tools to generate a view

**5.2.1.4 Microsoft BizTalk Orchestration Engine**

According to [14], [47], [48], [49] A business process is a combination of rules and actions which function together to meet some specific business goals. Business processes coordinated with other business processes which can be on other systems such as Enterprise Resource Planning (ERP), Customer Management System (CRM), or exposed as web services. This processes execution can span over hours, days, weeks or even months to complete.

Microsoft BizTalk includes such a feature to support orchestration of different processes. It also provides different tools and services which make it easy to design, automate and manage business processes. BizTalk orchestration supports two types of transactional programming models; the first is Atomic transactions, these transactions would be rolled back to previous state if they do not successfully complete, the other model is Long running transactions, these can stay running for days or longer and they require custom exception handling to recover from an exception.

**Features of BizTalk Orchestration Engine**

As explained by [48], [49] BizTalk Orchestration Engine provides the following features

**Orchestration Designer:** This designer gives a drag and drop graphical user interface to construct orchestrated business processes. The different shapes just need to be configured accordingly for the orchestration to work properly.

**BizTalk Orchestration Engine:** The BizTalk orchestration engine executes the orchestrated processes created using BizTalk Orchestration Designer and provides monitoring services for these processes. In addition, it maintains states of orchestrated instances, performs optimization of resources, provides reliable shutdown-and-recovery system and supports routing pattern called correlation.

**Calling External Assemblies:** BizTalk Orchestration also provides support to call external assemblies, i.e. if functionality is available in another .NET application, it can pass parameters to these applications through method calls, making it possible to integrate external applications into BizTalk process
Service Integration Scenarios: BizTalk Server fully supports calling Web services and WCF services even from within an orchestration. This enables developers to create WCF services which can be used within disparate business processes and also publish locally created business processes as services on the internet. BizTalk provides a Services Publishing Wizard to publish WCF services that run in Internet Information Server (IIS).

BizTalk Service Integration Capabilities: In addition, to the WCF adapters, BizTalk Server also supports consuming WCF services, publishing orchestration as WCF services, publishing schema as a WCF service, publishing receive location metadata as WSDL and integration with ASMX Web services.

BizTalk Orchestration in SOA Designs

SOA has a number of principles and business process automation is one of the important ones. Microsoft BizTalk Server supports business process automation in an extensive way. Service aggregation SOA pattern is the most used SOA pattern for business process automation and usually it is designed and implemented using BizTalk Server.

5.2.1.5 Microsoft BizTalk Messaging Engine

According to [50], [51] At the heart of BizTalk Server are the Orchestration Engine and the Messaging Engine, which are responsible for integrating and exchanging messages among services based on different protocols and standards. BizTalk Messaging Engine is responsible for receiving inbound messages, parsing and identifying message formats, evaluate routing criteria, delivery to destination and status tracking.

Features of BizTalk Messaging Engine

As explained by the [50],[51] BizTalk Messaging Engine provides the following features.

Publish/Subscribe Model: BizTalk Server uses the publish/subscribe model to route messages. The publish/subscribe model allows services to be published to a central store (in BizTalk it’s the MessageBox database) and the consumer/subscribers, subscribe to the published services, when a message is received the central store forwards it to all the subscribers. This publish/subscribe model allows services to be added/modified without having any impact on the application design.

BizTalk Server routes messages on the basis of matching message context properties of messages with expressions. This message context is persisted across the message life cycle and if needed also stored in the MessageBox database.

Building Schemas: BizTalk Server supports a number of schemas that it can process. A schema can be either predefined or might require to be created using a XML, flat file document structure or standard EDI document. BizTalk by default supports schema types for XML, Flat-File and more
than 8000 schemas for EDIFACT and X12 message standards, which are used for integration with EDI applications.

**Mapping Data:** BizTalk transforms inbound messages from one schema format to another using BizTalk map convert capability. These transformations can be simple or complex. Maps are usually defined at design time as they link fields within the two schemas which are being mapped.

**Connecting Through Adapters:** BizTalk needs adaptors to interoperate with systems which are not direct BizTalk message format compliant. Adapters are add-ons to BizTalk Server which enable BizTalk to communicate with different types of systems through standard-based protocols and with applications that correspond to proprietary communication standards. Most adapters support send and receive operations, other support only one directional communication. BizTalk supports native BizTalk Line-of-Business (LOB), WCF Adapters, BizTalk partner adapters and Microsoft WCF Line of Business Adapter SDK.

**Processing Messages through a Pipeline:** Pipelines can be thought of message buffers with extended functionality incorporated within them. The purpose of pipelines is to prepare a message (by verifying it against schema, encoding and decoding, encrypting and decrypting and other processing that might be required) for processing by the server when it is received and similarly prepare it for sending when it has been processed. These can also be called within an orchestration.

### 5.2.1.6 Microsoft BizTalk Business to Business Integration

According to [52], [53] BizTalk Server supports integration of external companies or department within a business process. BizTalk provides different capabilities which simplify the integration of business processes with the external companies. BizTalk Server natively provides support for Electronic Data Interchange (EDI) data protocols and Applicability Statement 2 (AS2) protocol. It also provides storage facility for management of external partner’s information and BizTalk Server Accelerators.

**Features of Business to Business Integration**

According to [52], [53] BizTalk Business To Business integration provides the following features:

**Electronic Data Interchange (EDI) and Applicability Statement 2 (AS2):** Electronic Data Interchange (EDI) is the most used format for data exchange within businesses. EDI uses standard message format based on ANSI X12, UN/EDIFACT and messaging protocols. BizTalk EDI and AS2 receive processing capabilities are, parsing EDI interchange, performing HIPAA document splitting, validating messages, serializing EDI interchange, EDI document exchange and AS2 document transport etc.

**EDI Parties:** BizTalk can incorporate external partners by having separate configuration for that external partner with its own unique communication parameters. Both should agree on the message format which would be exchanged among BizTalk and the external partner. BizTalk Server includes several services and tools for management of external partner relationships.
**BizTalk Server Accelerators:** BizTalk Server accelerators are used to support and add functionality within BizTalk to support business integration from sectors ranging from high tech to healthcare. These help in reducing time, cost and effort in development and management of application from these sectors.

### 5.2.2 Microsoft Windows Server Appfabric

According to [54], [55] Services and service-oriented architectures is a concept and exist in many types of applications. One type is of having transactional components which are data driven in a highly distributed business logic and which get managed on a middle tier. As the demand of highly responsive and scalable applications (web applications or applications which provide services) increases, a solid architecture is needed to cater limitation associated with application performance and scalability. Microsoft’s Appfabric technology is developed to cater this middle tier application scaling and performance limitation problem.

Windows Server Appfabric is a set of integrated technologies that is used to manage, host and scale web services giving it a touch of distributed environment or service orientated architecture style. Windows Server Appfabric adds capabilities to Internet Information Server (IIS) by adding functionality like enhanced hosting, management, caching of web applications and middle-tier services. Appfabric hosting features add service management extensions to Internet Information Services (IIS), Windows Process Activation Service (WAS), and the .NET Framework 4.0. Hosting services included Hosting Administration tools which make it easier to manage, configure and deploy Windows Workflow Foundation (WF) based workflow services and Windows Communication Foundation based services (WCF), which in turn help improve performance and scalability of .NET applications especially ASP.NET applications.

![AppFabric Architecture Diagram](image)
5.2.2.1 Windows Server Appfabric Components

As it’s explained in detail by [55], [56], [57], [58] Windows Server AppFabric provides the following components (Shown in figure 5.8)

**Service Hosting:** Windows Server AppFabric was designed specifically to host WCF and WF services. AppFabric incorporates hosting capabilities of Windows Process Activation service (WAS), which enables it to host .NET framework services. Management of workflows developed using Windows Workflow Foundation is done by the Workflow Management Service, which has features of command queue functionality, durable timers and auto-start. In addition, it allows for the monitoring of applications and management of configuration files, security, performance and service endpoints.

**AppFabric Windows Services:** There are two services which Windows Server AppFabric installs for event tracing and collection, and workflow management. The Event Tracing and Collection service is responsible for collecting all the events raised by WCF and WF services hosted on AppFabric and store them into the monitoring store database of AppFabric. The Workflow management service, manages workflows by activating workflow instances, restarting abandoned workflows etc.

**Data Storage:** Windows Server AppFabric uses SQL Server express edition to store its data entities. SQL Server is the recommended store unit; otherwise they can be stored in memory or flat files also. It has two storage entities, the persistence store and the monitoring store. The persistence store is responsible for storing states or workflows and any metadata information associated with the workflows. The monitoring store keeps track of all the .NET framework events associated with WCF and WF, for monitoring purpose.

**Security Model:** Windows Server AppFabric provides a security model by integrating with Microsoft security technologies to manage .NET 4.0 applications and services. It can integrated and use existing Windows Active Directory, .NET 4.0 security, IIS and SQL Server security model. While designing AppFabric security model, three roles can be created i.e. Application Server Observer, Application Server Administrator and Application Server Users. These roles can be assigned different permissions and can also correspond to Windows NT group accounts, IIS application pools and SQL server logins and database roles.
6. Vendor Products Roadmap

For an enterprise Roadmap is very important to make some important decision based on the roadmap provided by vendors. It is important for long term planning and support by vendors.

Microsoft and Red Hat JBoss both are very aggressive in making their product as much SOA compliant as possible. Both the vendors have started supporting cloud platform which is the next generation for service oriented architecture. Microsoft launched its Windows Azure cloud platform in start of 2010 and Red Hat announced its PaaS (Platform as a Service) in August 2010. The table below shows some of the functionality which can be expected from both vendor products in future.

<table>
<thead>
<tr>
<th>Area</th>
<th>Microsoft</th>
<th>Red Hat</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Enterprise Connectivity</strong></td>
<td><strong>Microsoft AppFabric integration with Business to Business applications using LOB adapters.</strong></td>
<td><strong>Apache CXF and Apache Camel Integration for connectivity.</strong></td>
</tr>
<tr>
<td><strong>Application Platform Alignment</strong></td>
<td><strong>Enhanced development, scalability, deployment and management capabilities to be integrated into AppFabric.</strong></td>
<td><strong>Enhanced UDDI v3 registry</strong></td>
</tr>
<tr>
<td></td>
<td><strong>AppFabric and BizTalk integration</strong></td>
<td><strong>BPEL Support</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>HornetQ support</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>SOA Repository</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>ESB 5</strong></td>
</tr>
<tr>
<td><strong>Cloud Support</strong></td>
<td><strong>Direct application deployment from development platform to cloud</strong></td>
<td><strong>Support for deployment across private and public clouds.</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Integration of private clouds with Windows Azure Platform</strong></td>
<td><strong>Extended support for programming models, languages, frameworks and APIs.</strong></td>
</tr>
</tbody>
</table>
7. Analysis
For analysis purpose, the function blocks identified in Chapter 4 would be used as the evaluation and comparison criteria for the vendor products. The functional blocks are the parameters from TRA and other Vattenfall needs. The main focus would be on the technical capabilities which are provided by each vendor product in accordance to the function blocks.

The comparison has been done for the following three areas:
1) SOA technical capabilities provided by vendors
2) Cost Effectiveness
3) SOA Competence

7.1 Weighting Criteria
As these identified function blocks are generic, to compare and evaluate which vendor is better the importance of each individual function block is calculated in accordance with the need within Vattenfall Nordic. The following weighting criteria were devised to assign a weight to each function block, which would keep things simple and easily understandable.

The following would be our weighting criteria: There are in total 3 points and the importance level increases with each point as follows:
- Highly Important = 3 points
- Partially Important = 2 points
- Least Important = 1 point

7.1.1 Parameter Weightings According to Vattenfall Nordic Importance
We asked the solution architects to assign a weight to each individual parameter in accordance to its importance within Vattenfall. The following table shows the importance of each individual parameter and also the importance percentage for that parameter:

<table>
<thead>
<tr>
<th>#</th>
<th>Parameters</th>
<th>Importance Weight</th>
<th>Importance Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Service Creation and Abstraction</td>
<td>2</td>
<td>67%</td>
</tr>
<tr>
<td>2.</td>
<td>Runtime Service Management</td>
<td>2</td>
<td>67%</td>
</tr>
<tr>
<td>3.</td>
<td>Service Orchestration</td>
<td>2</td>
<td>67%</td>
</tr>
<tr>
<td>4.</td>
<td>Process Control and Optimization</td>
<td>2</td>
<td>67%</td>
</tr>
<tr>
<td>5.</td>
<td>Presentation</td>
<td>2</td>
<td>67%</td>
</tr>
<tr>
<td>6.</td>
<td>Identity Management</td>
<td>2</td>
<td>67%</td>
</tr>
<tr>
<td>7.</td>
<td>Design time Service Management</td>
<td>1</td>
<td>33%</td>
</tr>
<tr>
<td>8.</td>
<td>RAD (Rapid Application Development)</td>
<td>3</td>
<td>100%</td>
</tr>
<tr>
<td>9.</td>
<td>Documentation</td>
<td>3</td>
<td>100%</td>
</tr>
<tr>
<td>10.</td>
<td>Support</td>
<td>3</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>Function Block</td>
<td>Weight</td>
<td>Importance Level</td>
</tr>
<tr>
<td>---</td>
<td>-------------------------------------------------</td>
<td>--------</td>
<td>------------------</td>
</tr>
<tr>
<td>11</td>
<td>Development Platform</td>
<td>3</td>
<td>100%</td>
</tr>
<tr>
<td>12</td>
<td>Cost and licensing</td>
<td>3</td>
<td>100%</td>
</tr>
<tr>
<td>13</td>
<td>Adaptors</td>
<td>3</td>
<td>100%</td>
</tr>
<tr>
<td>14</td>
<td>Integration with Existing Applications</td>
<td>3</td>
<td>100%</td>
</tr>
<tr>
<td>15</td>
<td>Open Source</td>
<td>1</td>
<td>100%</td>
</tr>
<tr>
<td>16</td>
<td>B2B transactions</td>
<td>3</td>
<td>100%</td>
</tr>
<tr>
<td>17</td>
<td>Deploy/Un-deploy Efforts</td>
<td>3</td>
<td>100%</td>
</tr>
<tr>
<td>18</td>
<td>Scalability</td>
<td>3</td>
<td>100%</td>
</tr>
<tr>
<td>19</td>
<td>Standards</td>
<td>2</td>
<td>67%</td>
</tr>
<tr>
<td>20</td>
<td>Governance Platform</td>
<td>1</td>
<td>33%</td>
</tr>
<tr>
<td>21</td>
<td>End to End Business Process Monitoring</td>
<td>2</td>
<td>67%</td>
</tr>
</tbody>
</table>

Table 1: Function Block Importance Weightings

**Graphical Representation of Vattenfall Importance Level**

For a more visual representation of these weightings we charted them out with the help of the following graph:

![Vattenfall Importance Level](image)

*Figure 8.1: Vattenfall Nordic importance level*

Figure 8.1 shows the importance level for the specified function blocks at Vattenfall Nordic.
7.2 SOA Technical Capabilities Provided by Vendors
This section describes the technical capabilities of the two vendors under consideration in accordance with identified function blocks. Furthermore, these vendors would be assigned a weight on basis of the capabilities they provide.

7.2.1 Requirements Fulfillment by Microsoft and Redhat JBoss
Following table illustrates the details of capability against each function block for both JBoss and Microsoft. Microsoft provides two solutions for SOA; it can be implemented either using BizTalk Server or Windows Server Appfabric, this analysis would consider both these products for comparison.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>SOA Vendors</th>
<th>Microsoft</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Red Hat JBoss</td>
<td>BizTalk Server</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows Server Appfabric</td>
</tr>
<tr>
<td>1. Service Creation and Abstraction</td>
<td>SOA Connection</td>
<td>Service Throttling</td>
</tr>
<tr>
<td></td>
<td>JBoss ESB and Web Services</td>
<td>Intel SOA expresses way, Service gateway (Third Party component).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BizTalk Server</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Message Buffering and Large Message Threshold property</td>
</tr>
<tr>
<td></td>
<td>Transformations</td>
<td>WCF Services</td>
</tr>
<tr>
<td></td>
<td>JBoss Smooks Transformer Adaptor</td>
<td>Configured through AppFabric Dashboard in IIS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BizTalk Server Adapter pack and WCF Custom Adaptor Pack</td>
</tr>
<tr>
<td></td>
<td>Routing</td>
<td>WCF Custom Adapter Pack</td>
</tr>
<tr>
<td></td>
<td>Achieved by JBoss ESB</td>
<td>BizTalk Server</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Content Based Routing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WCF Content Based Routing Service .NET 4.0</td>
</tr>
<tr>
<td>2. Runtime Service Management</td>
<td>Service Throttling</td>
<td>Dynamic Service Ramp-up</td>
</tr>
<tr>
<td></td>
<td>Intel SOA expresses way, Service gateway (Third Party component).</td>
<td>Not Supported</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Not supported</td>
</tr>
<tr>
<td></td>
<td>*JBoss SOA Governance (jBPM)</td>
<td>Security Policy Fulfillment</td>
</tr>
<tr>
<td></td>
<td>BizTalk Server Business Activity Monitoring service</td>
<td>Windows Security (Active Directory ) for BizTalk</td>
</tr>
<tr>
<td></td>
<td>*JBoss SOA Governance (JBoss EAP)</td>
<td>Windows Security (Active Directory ) for AppFabric</td>
</tr>
</tbody>
</table>
### 3. Service Orchestration

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Workflows</td>
<td>JBoss jBPM workflow and BPM (Business Process Management) engine</td>
<td>BizTalk Orchestration Engine.</td>
<td>Windows Workflows hosted as services.</td>
</tr>
</tbody>
</table>

### 4. Process Control and Optimization

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Reporting Tools</td>
<td>JBoss Developer Studio included Business Intelligence Report Tool (BIRT)</td>
<td>BizTalk Server built-in reporting and alerting portal, Integration with Office Business Intelligence tools, SharePoint and SQL Server Reporting Services</td>
<td>Not Supported</td>
</tr>
</tbody>
</table>

### 5. Presentation

<table>
<thead>
<tr>
<th>SOA Portal Platform</th>
<th>Windows Presentation Foundation and ASP.NET</th>
<th>Windows Presentation Foundation and ASP.NET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personalization</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Syndication</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Rich graphical control</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

### 6. Identity Management

<table>
<thead>
<tr>
<th>JBoss seam, Identity Management - an API for managing a Seam application's users and roles at runtime.</th>
<th>Windows Active Directory</th>
<th>Windows Active Directory</th>
</tr>
</thead>
</table>

**57**
<p>| Authentication | an extensible, JAAS-based authentication layer that allows users to authenticate against any security provider | Windows Active Directory Authentication (BizTalk provides no direct security feature for Authentication) | Windows Active Directory Authentication (Appfabric provides no direct security feature for Authorization) |
| Role Management &amp; Auditing | Permission Management - a set of built-in Seam components to allow easy management of an application's security policy. | Windows Active Directory Role Management (BizTalk provides no direct security feature for Role Management) | Windows Active Directory Role Management (Appfabric provides no direct security feature for Role Management) |
| 7. Design time Service Management | | | |
| Service Repository/Registry | jUDDI registry and repository service which is integrated with JBoss ESB | Microsoft UDDI 3.0 Service registry (Can be integrated with BizTalk) | Microsoft UDDI 3.0 Service registry (Can be integrated with BizTalk) |
| Life Cycle Management and Versioning | JBoss ESB, Change management (hot deployment, versioning, lifecycle management) | Microsoft Team Foundation | Microsoft Team Foundation |
| 8. RAD (Rapid Application Development) | JBoss Developer Studio + Seam-gen is the RAD tool packaged within JBoss Seam. | Achieved using Microsoft Visual Studio Development IDE | Achieved using Microsoft Visual Studio Development IDE |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>10. Support</strong></td>
<td>JBoss portfolio enterprise subscription</td>
<td>Microsoft Software Assurance subscription</td>
</tr>
<tr>
<td><strong>12. Cost and licensing</strong></td>
<td>Detailed comparison in next section</td>
<td>Detailed comparison in next section</td>
</tr>
<tr>
<td><strong>13. Adaptors</strong></td>
<td>JBoss ESB gateways adapters</td>
<td>Microsoft BizTalk LOB adapters, WCF Custom Adapters Pack</td>
</tr>
<tr>
<td><strong>14. Integration with Existing Applications</strong></td>
<td>JBoss ESB Adapters (Exposing application as Web Service)</td>
<td>WCF Custom Adapters (Exposing application as Web Service)</td>
</tr>
<tr>
<td><strong>15. Open source</strong></td>
<td>YES</td>
<td>NO</td>
</tr>
<tr>
<td><strong>17. Deploy/Undeploy Efforts</strong></td>
<td>Apache ant, Eclipse, XML configuration (Manually installations)</td>
<td>Deploy BizTalk Server as a routing and transformation hub for Web Services.</td>
</tr>
<tr>
<td><strong>18. Scalability</strong></td>
<td>JBoss Application Server</td>
<td>BizTalk Provides Horizontal and Vertical Scalability options</td>
</tr>
<tr>
<td><strong>19. Standards</strong></td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td><strong>20. Governance Platform</strong></td>
<td>*JBoss SOA Governance</td>
<td>Not Directly supported Achieved using Third Party Tools</td>
</tr>
<tr>
<td><strong>21. End to End BPM</strong></td>
<td>JBoss jBPM provides visibility into the current end-to-end state of processes in which users and applications are interacting.</td>
<td>BizTalk Server Business Activity Monitoring service</td>
</tr>
</tbody>
</table>
Table 2: Function Blocks and Vendor Product Capabilities

*SOA Governance is mentioned in chapter 5 where vendor product details are listed

7.2.2 Vendor Weighting on the Basis of Provided Technical Capabilities

After mapping vendor capabilities with the identified function blocks, weights are assigned to each vendor product according to the capabilities being fulfilled under each function block. This weighting is important as it would be used to compare the functional capabilities or each vendor product.

The comparison would be done on after calculating percentage of fulfillment of each capability for each vendor.

Weighting criteria is based on the following rules

- The function blocks with no sub function blocks would be evaluated as a one full function block and the percentage would be either provided i.e. 100% or not provided i.e. 0%.
- If there is one or more than one sub function blocks then the percentage would be calculated on basis of the number of sub function blocks being satisfied. For example: as Service creation and abstraction has 3 sub function blocks, percentage would be calculated by looking at the satisfied sub function blocks and dividing them by the total sub function blocks i.e. (satisfiedFB/TotalFB) * 100.

<table>
<thead>
<tr>
<th>#</th>
<th>Criteria</th>
<th>Red Hat</th>
<th>Microsoft</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>JBoss</td>
<td>BizTalk Server</td>
</tr>
<tr>
<td></td>
<td>Parameters</td>
<td>Status</td>
<td>Fulfillment %</td>
</tr>
<tr>
<td>1.</td>
<td>Service Creation and Abstraction</td>
<td>3 / 3 100%</td>
<td>3/3 100%</td>
</tr>
<tr>
<td></td>
<td>SOA Connection</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>Transformations</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Section</td>
<td>Completed</td>
<td>Percentage</td>
<td>Completed</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>-----------</td>
<td>------------</td>
<td>-----------</td>
</tr>
<tr>
<td>Routing</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>2. Runtime Service Management</td>
<td>3 / 4</td>
<td>75%</td>
<td>3 / 4</td>
</tr>
<tr>
<td>Service throttling</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Runtime statistics and governance</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Dynamic service ramp-up</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>Security policy fulfillment</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>3. Service Orchestration</td>
<td>3 / 3</td>
<td>100%</td>
<td>3 / 3</td>
</tr>
<tr>
<td>Process control</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Workflows</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Business rules execution</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>4. Process Control and Optimization</td>
<td>2 / 2</td>
<td>100%</td>
<td>2 / 2</td>
</tr>
<tr>
<td>Business Activity Monitoring</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Reporting Tools</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>5. Presentation</td>
<td>3 / 3</td>
<td>100%</td>
<td>3 / 3</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>------------------------</td>
<td>----------------------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>Personalization</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Syndication</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Rich graphical control</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Authorization</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Authentication</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Role Management &amp; Auditing</td>
<td>✓</td>
<td>❌</td>
<td>❌</td>
</tr>
<tr>
<td>Service repository/Registry</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Life cycle Management Versioning</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Versioning</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Identity Management</td>
<td>3/3 100%</td>
<td>2/3 75%</td>
<td>2/3 75%</td>
</tr>
<tr>
<td>Authorization</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Authentication</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Role Management &amp; Auditing</td>
<td>✓</td>
<td>❌</td>
<td>❌</td>
</tr>
<tr>
<td>Service repository/Registry</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Life cycle Management Versioning</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Versioning</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Designtime Service Management</td>
<td>2 / 2 100%</td>
<td>2/2 100%</td>
<td>2/2 100%</td>
</tr>
<tr>
<td>Service repository/Registry</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Life cycle Management Versioning</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rapid Application Development</td>
<td>✓ 1 / 1 100%</td>
<td>✓ 1/1 100%</td>
<td>✓ 1/1 100%</td>
</tr>
<tr>
<td>Documentation</td>
<td>✓ 1 / 1 100%</td>
<td>✓ 1/1 100%</td>
<td>✓ 1/1 100%</td>
</tr>
<tr>
<td>Support</td>
<td>✓ 1 / 1 100%</td>
<td>✓ 1/1 100%</td>
<td>✓ 1/1 100%</td>
</tr>
<tr>
<td></td>
<td>Development Platform</td>
<td>✓</td>
<td>1 / 1</td>
</tr>
<tr>
<td>---</td>
<td>----------------------</td>
<td>---</td>
<td>-------</td>
</tr>
<tr>
<td>12.</td>
<td>Adaptors</td>
<td>✓</td>
<td>1 / 1</td>
</tr>
<tr>
<td>13.</td>
<td>Integration with Legacy Applications</td>
<td>✓</td>
<td>1 / 1</td>
</tr>
<tr>
<td>14.</td>
<td>Open Source</td>
<td>✓</td>
<td>1 / 1</td>
</tr>
<tr>
<td>15.</td>
<td>B2B Transactions</td>
<td>✓</td>
<td>1 / 1</td>
</tr>
<tr>
<td>16.</td>
<td>Scalability</td>
<td>✓</td>
<td>1 / 1</td>
</tr>
<tr>
<td>17.</td>
<td>Deploy/Undeploy Effort</td>
<td>✓</td>
<td>1 / 1</td>
</tr>
<tr>
<td>18.</td>
<td>Standards</td>
<td>✓</td>
<td>1 / 1</td>
</tr>
<tr>
<td>19.</td>
<td>Governance Platform</td>
<td>✓</td>
<td>1 / 1</td>
</tr>
<tr>
<td>20.</td>
<td>End to End BPM</td>
<td>✓</td>
<td>1 / 1</td>
</tr>
</tbody>
</table>

Table 2: Vendor evaluation of basis of capabilities provided

7.3 Comparison of Vendor Products According to Technical Reference Architecture Capabilities

**Redhat JBoss**

Figure 8.2 shows the percentage of fulfillment of the evaluation criteria for Red Hat JBoss. We can see from the figure 8.2 that JBoss is fulfilling all the function blocks 100% except Runtime service management. There is no information which I could found regarding dynamic service ramp-up inside runtime service management.
Microsoft BizTalk Server

Figure 8.3 shows the function blocks covered by Microsoft BizTalk Server. Open Source, Governance platform is not fulfilled by BizTalk Server and that’s why it’s showing 0%. Other than these function blocks BizTalk is pretty much same as Function blocks provided by JBoss.
Microsoft windows server Appfabric

Figure 8.4: Microsoft Windows Server AppFabric Capabilities and Function Blocks

Figure 8.4 show the function blocks fulfillment but Microsoft Windows Server Appfabric. As we know that Appfabric is not for SOA, it’s mainly for Cloud computing which is out of our thesis scope. But still Appfabric some powerful features and we can use them for SOA implementation. It’s important to understand that Appfabric doesn’t have any support at the moment, it’s not Open Source and it does not support B2B transactions. It also has no SOA governance platform.

Requirements fulfilled by JBoss, BizTalk and Appfabric all together
**Average percentage fulfilled by JBoss:** If we take the percentage of provided by JBoss against the TRA and Vattenfall Nordic requirements, the average of the percentage is given below

\[
\text{JBoss} \% = (\text{Fulfilled by JBoss}/ \text{Total}) \times 100, \text{Total} \%= 2000, \text{fulfilled by JBoss} = 1975
\]

\[
\text{JBoss} \% = (1975/2000) \times 100, \text{JBoss} \% = 99\%
\]

**Average percentage fulfilled by BizTalk:**

\[
\text{BizTalk} \% = (\text{Fulfilled by BizTalk}/ \text{Total}) \times 100, \text{Total} \%= 2000, \text{fulfilled by BizTalk} = 1750
\]

\[
\text{BizTalk} \% = (1750/2000) \times 100, \text{BizTalk} \% = 88\%
\]

**Average percentage fulfilled by Appfabric:**

\[
\text{Appfabric} \% = (\text{Fulfilled by Appfabric}/ \text{Total}) \times 100, \text{Total} \%= 2000, \text{fulfilled by Appfabric} = 1700
\]

\[
\text{Appfabric} \% = (1600/2000) \times 100, \text{Appfabric} \% = 80\%
\]

Here JBoss is on the leading seat with fulfilling approximately 99% of the TRA and Vattenfall needs.

![Figure 8.5: Capabilities Comparison of JBoss, BizTalk Server and Windows Server AppFabric](image)

As we can see that JBoss is leading with 99% in fulfilling the Technical Reference Architecture (TRA) and Other Vattenfall needs. BizTalk is at second place with fulfilling 87% and Appfabric is at third place with 85%. You can also see the functions blocks which are not fulfilled by JBoss, BizTalk and Appfabric.

![Figure 8.6 % fulfilled by each Vendor Solution's](image)
7.4 Hardware and Software Specifications Used For Calculating Cost
These specifications were chosen to have an accurate approximation of each vendor product price.

<table>
<thead>
<tr>
<th>#</th>
<th>Products</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Operating System</td>
<td>Price is calculated for one copy of Operating System + Support per year.</td>
</tr>
<tr>
<td>2</td>
<td>SOA Enterprise Platform</td>
<td>Application Servers + Support are sold by per processors (unlimited client access) based or by the number of client access license (CAL’s). The comparison looks at the number of processors which is 4 processors for each server for this comparison per year.</td>
</tr>
<tr>
<td>3</td>
<td>Support</td>
<td>Support price is calculated on yearly basis per year.</td>
</tr>
<tr>
<td>4</td>
<td>Data Base</td>
<td>Database Servers + Support are also sold by per processors (unlimited client access) based or by the number of client access license (CAL’s). The comparison looks at the number of processors which is 1 processor for each server for this comparison per year.</td>
</tr>
<tr>
<td>5</td>
<td>Development Platform</td>
<td>Price is calculated for one copy of Development Platform + Support per year.</td>
</tr>
<tr>
<td>6</td>
<td>Enterprise Middleware</td>
<td>It is included in the application server cost.</td>
</tr>
</tbody>
</table>

Table 3: H/W & S/W specification for price calculation

7.5 SOA Vendors Solutions Technical Comparison for Cost
For the technical comparison of JBoss, BizTalk and Appfabric for cost we have three different cases.

7.5.1 Case 1: Redhat JBoss vs. Microsoft BizTalk Server
In this case we are considering that if an organization wants to go for SOA implementation from scratch, so for which option they should go? As we have Redhat JBoss and Microsoft. Then we have two solutions from Microsoft, one is BizTalk and other is Appfabric. JBoss is best available option for an organization if they are going to implement SOA from scratch. It means they don’t have any existing platforms like BizTalk.

Implementation cost for building a new SOA system
The following table shows a cost comparison for implementing service oriented architectural system from scratch using either one of the vendors. This comparison is done using List prices available from each vendor website. The specification for each product category is also mentioned:
<table>
<thead>
<tr>
<th>#</th>
<th>Products</th>
<th>Red Hat</th>
<th>Microsoft</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>JBoss Price</td>
<td>BizTalk Price</td>
</tr>
<tr>
<td>1</td>
<td>Operating System</td>
<td>$2499</td>
<td>$4101</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>SOA Enterprise Platform</td>
<td>JBoss SOA Enterprise Platform+JBoss Operation Network</td>
<td>$27000</td>
</tr>
<tr>
<td>3</td>
<td>Support</td>
<td>JBoss Developers Subscription,(20 Suites of JDeveloper Studio)</td>
<td>$16000</td>
</tr>
<tr>
<td>4</td>
<td>Data Base</td>
<td>My SQL Enterprise platinum Subscription</td>
<td>$4999</td>
</tr>
<tr>
<td>5</td>
<td>Development Platform</td>
<td>JDeveloper Studio (Eclipse Based).</td>
<td>$0</td>
</tr>
<tr>
<td>6</td>
<td>Enterprise Middleware</td>
<td>JBoss Developer Subscription covers this.</td>
<td>$0</td>
</tr>
<tr>
<td></td>
<td>TOTAL</td>
<td>$50,498</td>
<td>$309,050</td>
</tr>
</tbody>
</table>

Table 4: Implementation cost for new SOA system

Cost comparison is done for building a new system for SOA for both vendors. There would be two comparisons for Microsoft as it provides two products for the implementation of SOA: The comparison is as follows:

Cost Comparison for Red Hat JBoss and Microsoft BizTalk Server
We can see the huge difference between BizTalk and JBoss regarding cost. BizTalk is not a best solution for SOA implementation if you are going to build your platform from scratch. BizTalk is costs almost double as compare to JBoss. And also the technical comparison which we have done JBoss is better than BizTalk. So JBoss is the winner in this case.

### 7.5.2 Case 2: Redhat JBoss vs. Microsoft windows server Appfabric

In this case we are comparing Redhat JBoss with Microsoft Appfabric. Appfabric is a next big thing for cloud computing, and also it can be used to implement SOA. If you don’t want to go for an open source, and you want proprietary software with cheap price to just implement SOA, you can go for Appfabric. The other reason you should go for Appfabric is that if you are using some existing platforms from Microsoft like Operating systems, data bases and integration platforms Appfabric is better. Case 2 is also states when you want to implement SOA from scratch. Appfabric is good if you consider only price. Although technical capabilities provided by Appfabric are less than Redhat JBoss, but you can sacrifice your needs over price.

### Cost Comparison for Red Hat JBoss and Microsoft Windows Server Appfabric

Figure 8.8 shows the cost comparison of Redhat JBoss with Microsoft windows server Appfabric. You can see that Appfabric is much cheaper solution for SOA implementation as compare to Redhat JBoss. SO in this case Appfabric is the winner when you can sacrifice your needs over price.
Figure 8.8: Cost Comparison for Red Hat JBoss and Microsoft Windows Server AppFabric for new SOA system

Winner: AppFabric

7.5.3 Case 3: Redhat JBoss vs. Vattenfall Nordic’s existing Systems
This Case is very much enterprise specific as in large enterprises an system for integration already exists and they have a contract of at least 3-5 years with the vendor of the integration platform. This thesis is exploring options for Vattenfall and they are currently Microsoft products for integration purpose. The approximated cost which they are paying for their integration solution is shown in table 6.

The comparison in this case would be done without considering the existing integration platform cost. The only cost considered in this scenario would be any addition cost required from the current cost for implementation of SOA.

Vattenfall Approximation cost for existing integration platform
Vattenfall is running all its systems on Microsoft Products and their main integration product is BizTalk Server. The following table shows Vattenfall’s approximation cost per year:

<table>
<thead>
<tr>
<th>#</th>
<th>Microsoft</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Operating System</td>
<td>$ 3500</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2008 R2 + Windows Server AppFabric + Software Assurance</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>SOA Enterprise Platform</td>
<td>$ 80,000</td>
</tr>
<tr>
<td></td>
<td>BizTalk Server (4 processors) + ESB Toolkit + Software Assurance</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Support</td>
<td>$ 0</td>
</tr>
<tr>
<td></td>
<td>Software Assurance is included in product price</td>
<td></td>
</tr>
</tbody>
</table>
Table 5: VattenFall Current Integration Platform Cost

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 Data Base</td>
<td>MS SQL Server 2005</td>
<td>$3500</td>
</tr>
<tr>
<td>5 Development</td>
<td>Visual Studio 2010 Ultimate + MSDN Subscription</td>
<td>$4000</td>
</tr>
<tr>
<td>6 Enterprise</td>
<td>N/A</td>
<td>$0</td>
</tr>
<tr>
<td>Middleware</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td>$91,000</td>
</tr>
</tbody>
</table>

Vattenfall Nordic is running its current system on Microsoft technologies and the graph shows the approximate cost being paid for integration platform.

Figure 8.9: Vattenfall Existing System Cost

Vattenfall Nordic is paying approximately 91000 USD every year for BizTalk Server. Taking this into consideration that huge of amount of money is being paid to Microsoft for using BizTalk they can use BizTalk and Appfabric (which is free with windows server 2008) , Vattenfall Nordic can implement SOA for free. If Vattenfall Nordic is not going to scrap the BizTalk for next 5 to 10 years, they can use this platform with the help of Appfabric to implement SOA. And this would be free of cost. Figure 8.10 shows that Vattenfall Nordic is currently paying more than the JBoss if they want to have JBoss for implementing SOA. But as we know that Vattenfall is using BizTalk and this amount of money they have to pay every year whether they implement SOA or not, so what they can do is that they can save additional overhead money for JBoss and use existing system to Implement SOA.
This figure 8.11 shows SOA implementation costs using the current existing system and if we move towards Red Hat JBoss. There would be no additional cost associated apart from the current cost being paid if the company moves towards a SOA implementation from current integration platform, whereas if the company opts for JBoss SOA Solution there would be an overhead cost as shown in the figure 8.11.

So as a conclusion Redhat JBoss is best in Case 1, Microsoft Windows server Appfabric is best in Case2, and BizTalk plus Appfabric are best in Case 3.
7.6 SOA Competence

SOA is a concept which needs to be understood to level to fully implement it. Therefore, SOA competence within the enterprise becomes one of deciding factors for moving towards SOA implementation. This also in turn effects the vendor selection, as people would be much more competent in the currently running integration platform vendor products.

Vattenfall Nordic is running mostly Microsoft products and employees are much more acclimatized with Microsoft products rather than Red Hat’s Enterprise Middleware suite. The following table shows the competence level of employees within Vattenfall with respect to different Vendors:

<table>
<thead>
<tr>
<th>#</th>
<th>Tools</th>
<th>JBoss</th>
<th>VF Nordic competence</th>
<th>Microsoft</th>
<th>VF Nordic competence</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Development Platform</td>
<td>Eclipse Based JDeveloper</td>
<td>3</td>
<td>Visual Studio, Team Studio Foundation</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>Data Base</td>
<td>My SQL</td>
<td>0</td>
<td>SQL Server (Dev &amp; Management)</td>
<td>11</td>
</tr>
<tr>
<td>3</td>
<td>SOA Tools</td>
<td>JBoss SOA Platform</td>
<td>0</td>
<td>BizTalk Server</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td>Operating System</td>
<td>Red Hat Linux.</td>
<td>0</td>
<td>Windows 2008 R2 + VS</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 6: SOA competence comparison within Vattenfall Nordic

![Red Hat and Microsoft](image)

*Figure 8.12: SOA Competence of employees within Vattenfall Nordic*
8. Conclusion
SOA has evolved at a tremendous rate in the last couple of years. It was originally a technical concept but because of its business integration nature made its way towards enterprises. Businesses gain from SOA implementation, as services in SOA not only help in resolving technical interoperability, but also help in solving specialized business functionality. Services can also be combined to act as a business process for the fulfillment of business activities. From an implementation perspective, Web service technology is becoming the standard implementation technique for SOA solutions and most of the enterprises are building their SOA computing environments using web services.

SOA implementation in enterprises makes it easy to integrate applications and enables building of sophisticated applications which fulfill business and enterprise needs. To make a choice for the right vendor to implement SOA is a daunting task. Every enterprise has its own specific requirement and constraints. These could be financial constraints or business constraints. It is difficult to pin point a specific criteria for the selection of SOA implementation within an enterprise. This thesis tries to generalize a list of technical and non-technical functional blocks that influence implementation of SOA and Vendor selection.

There are a number of tools that provide specialized functionalities for construction of SOA based systems. Both proprietary and open source vendors are providing product suites which include integration platforms, management and monitoring suites, collection of service design techniques and creation and modeling tools for SOA.

SOA is the next big thing for future integration of enterprises. It is constantly evolving as it gets adopted more and more by enterprises and organizations. In future, SOA would have a significant impact on the development of enterprise application integration infrastructure as more and more technologies and vendors start providing solutions. [Done by me and Adnan Gohar [37] ]
References

[1] Vattenfall internal documents, ”Vattenfall Nordic Technical Reference Architecture(TRA) ”, Taken from Vattenfall Business Services Nordic AB.
[14] Nicolai M.Josuttis SOA in Practice
[19] JBoss Portal 2.7.1 User Guide
[21] Website: http://searchsoa.techtarget.com/sDefinition/0,sid26_gci1270465,00.html
[22] Website: www.jboss.com
[26] Website: http://docs.jboss.org/jbossesb/docs/4.2.1GA/manuals/html/GettingStarted.html
[29] Website: http://www.jboss.org/jbpm
[34] Website:  http://docs.jboss.org/jbossesb/docs/4.2.1GA/manuals/html/services/Registry.html
[35] Website:  http://www.jboss.org/jbossportal


[47] Oracle Enterprise Service Bus Developer's Guide 10g (10.1.3.3.0), http://download.oracle.com/docs/cd/E10291_01/doc.1013/e10295/cesb_intro.htm#BABFGJFE