LEEDS BECKETT UNIVERSITY

Citation:
Ramachandran, M (2017) Service-oriented architecture for big data and business intelligence analytics in the cloud. In: Computational Intelligence Applications in Business and Big Data Analytics. Auerbach Publications, pp. 237-258. ISBN 9781351720250 DOI: https://doi.org/10.1201/9781315180748

Document Version:
Book Section (Accepted Version)

This is an Accepted Manuscript of a book chapter published by Routledge in Computational Intelligence Applications in Business and Big Data Analytics on 06 June 2017, available online: http://www.routledge.com/9781498761017

Service-Oriented Architecture for Big Data and Business Intelligence Analytics in the Cloud

Muthu Ramachandran

School of Computing, Creative Technologies and Engineering

Faculty of Arts, Environment, and Technology

Leeds Beckett University, Leeds, UK

Email: m.ramachandran@leedsbeckett.ac.uk

Abstract

SOA has emerged supporting scalability and service reuse. At the same time Big Data Analytics has impacted on business services and business process management. However, there is a lack of a systematic engineering approach to big data analytics. This chapter will provide a systematic approach to SOA design strategies and business process for Big Data Analytics. Our approach is based on SOA reference Architecture and Service Component model for big data applications known as SoftBD, secondly, a large scale real-world case study demonstrating our approach to SOA for Big Data Analytics. SOA big data architecture is a scalable, generic and customisable for a variety of data applications. The main contribution of this chapter includes a unique and innovative and generic softBD framework, service component model, and a generic SOA architecture for large scale big data

applications. This chapter has also contributed to Big Data metrics which allows to measure and evaluate when analyzing the data.

1. Introduction

Distributed systems have been developed and deployed in a traditional software architecture model based on layered architecture. However, this has not able to provide a sustainable IT system which is cost-effective. Therefore, SOA has emerged to address this issue and has emerged with key design principles such as loose coupling, service reusability, service composability, and service discoverability. The SOA deployment model is based on service provider to publish their services through a registry, and a service requester to be able access the published services, compose new services, and request new services. The major challenge of this work is to integrate SOA for Big Data Applications. Big Data has emerged to address the challenges faced by volume, velocity, and veracity of data being received and analyzed in real-time. Therefore, we need an SOA model which tackles required speed and accuracy of data. The model proposed in this chapter will aim to achieve these two characteristics. This way this paper aims to achieve merging two major issues (SOA and Big Data). Zimmermann et al. (2013) emphasizes the need for an enterprise SOA architecture for Big Data Applications and have proposed Enterprise Reference Architecture Cube (ESARC) for such large scale application.

Big Data has become a key business improvement indicator for large businesses and as the key indicator of success in the Cloud and IoT Computing Technologies. Big Data can be defined as the management of data received from different sources on the use and behavior of a system in real-time at the scale of terabytes, peta-bytes, etc. The size of data depends on the nature of systems such as mobile phone usage, web usage, social media usage, real-time internet and sensor data received, streaming media data received and sent. In formal term, Big Data has been defined as the 5Vs model (volume, velocity, variety, value and veracity). Value and veracity are two essential characteristics that specify the need of valuable and truthfulness data (Neves and Bernardino 2016).

Therefore, it is important for businesses and organizations to develop a long term strategy for managing, monitoring, analyzing, and predicting data. We have identified a measure of big data value:

Value(of data/information) ∞ √Number of Business Users (BU)×Number of Business Areas

Equation 1 Measuring the value of Big Data

As equation 1 suggest, the value of a big data is directly proportional (increases in value) to the square root of a number of business users which is multiplied by the business areas that they work. The need for integrating business intelligence and business process modelling for big data, with service-oriented architecture is the key to achieve business value as suggested by Curko, Bach, and Radonic (2007) SOA concept as a key technology for integration of BI, BPMS, transaction, big data, and other IT systems.

However, there is a lack of business intelligence analytics applied to large scale big data that has been received from multiple sources and also lack of applying intelligence and enterprise architecture for large scale big data that are emerging from multiple business and data sources. In addition, existing approaches (Zimmermann et al. 2013) in this area doesn't consider applying intelligence analytics for prediction by applying soft computing methods such as Bayesian theory, Fuzzy logic, and Neuro-Fuzzy.

This chapter divides into two major sections: SOA approach for Big Data with SOA reference Architecture and Service Component model for big data applications, secondly, a large scale real-world case study demonstrating our approach to SOA for Big Data Analytics. Our approach is a scalable big data architecture model which is generic and customisable for a variety of data applications. The main contribution of this chapter includes a unique and innovative and generic softBD framework, service component model, and a generic SOA architecture for large scale big data applications. This chapter has also contributed to Big Data metrics which allows to measure and evaluate when analyzing the data.

2. SOA Based Soft computing Framework for Big Data

One of the key reasons for chossing soft compute approach is to apply predictions to large amount of data being generated by IoT, IoE, the cloud system, and other sources such user generated data. Existing approaches in this area have considered architecting data, but not to have considered predictive analysis based on the currently collected and previously collect data for the similar situation. Service Oriented Architecture (SOA) has emerged supporting scalability and service reuse. At the same time Big Data Analytics has impacted on business services. This chapter will provide a systematic approach for SOA design strategies and business process for Big Data Analytics. Distributed systems have been developed and deployed in a traditional software architecture model based on layered architecture. However, this has not able to provide a sustainable IT system which is cost-effective. Therefore, SOA has emerged to address this issue and has emerged with key design principles such as loose coupling, service reusability, service composability, and service discoverability as shown in Figure 1. Thus, this chapter has proposed a reference architecture which is based on SOA and it has potential to solve classical problem of customisation, composbility interoperability etc. The major focus of this chapter to integrate SOA for Big Data Applications. Consequently, big data have emerged to address the challenges faced by volume, velocity, and veracity of data being received and analyzed in real-time. Therefore, we need an SOA model which tackles required speed and accuracy of data. The model proposed in this chapter will aim to achieve these two characteristics. The earlier studies emphasize the need for an enterprise SOA architecture for Big Data Applications and have proposed Enterprise Reference Architecture Cube (ESARC) for such large scale application. Consistent with earlier studies, this research aims to achieve merging two major issues (SOA and Big Data).

Big Data is now a reality for businesses. Examples include Google's Gmail is an Exabyte of data, Amazon web services, streaming media globally every second, and other real-time life sensitive data such as media, weatherforcasts, earth monitoring, space application data, etc. However, we need a structured model to select, process, and monitor highly relevant data. In this context, Gorton (2013-14) describes a lightweight with a risk reduction approach to big data, known as Lightweight Evaluation and Architecture Prototyping for Big Data (LEAP4BD). This provides a good starting point to organizations to use a semantic knowledge based tool for data selection and acquisition. However, it is yet another tool for data extraction. Similarly, Li et al (2015) has proposed a framework for Geoscience big data which is a multidimensional data where Hadoop (Hbase0 has been adopted for storing and managing multi-dimensional geoscientific data and MapReduce-based parallel algorithm has been used for processing such data. Although, this framework does propose a large scale data extraction, but lacks in generalization and application of SOA architecture.

Therefore, we have proposed an SOA based soft compute framework for big data (SoftBD) with soft compute intelligence algorithms for decision making when collecting, selecting and extracting, validating and evaluating the data. In addition, soft compute algorithms such as neuro-fuzzy and Bayesian theorem allows data prediction based on historical data collected. SoftBD framework is shown in Figure 1.1

Figure 1.1 SOA Based Soft Compute Framework for Big Data (SoftBD)

The SoftBD starts receiving data continuously with cloud platforms such as Amazon EC2, Windows Azure, google, Saleforce, etc. The data sources can be from multiple sources including Geoscience, Entertainment monitoring, selection, and prediction (Home, Cinema, Theoter, etc.), Aircraft monitoring, IoT devices, healthcare, Mobile, Wireless Sensor Networks, and natural Disaster areas such as fire, flood, and earthquake, and epidemics. Cloud is the best suitable platform as it provides elastic services to store and maintain data globally. The next layer is data clustering layer using the Hadoop platform which can be part of the chosen cloud platform. This layer does the data processing, categorizing the data into entertainment, Geoscience, wsn, etc. The next layer is the data analytics layer which could include Hadoop / MapReduce algorithms to speed up analyzing the data parallelisation. The big challenge in Hadoop is resource allocation and scheduling for parallel high performance computing. The final layer on the vertical services is our new soft compute intelligence layer which includes algorithms and models on fuzzy-logic, nuero-fuzzy, and Bayesian theorem. The main purpose of this service layer is to offer prediction models on the categorized and analyzed data. This also covers the data evaluation, assessment, and validation. All of the vertical layers are based on the service-oriented architecture providing loosely coupled services.

3. SOA Design Characteristics

SOA has emerged to tackle legacy systems design for distributed information systems problems and its limitations by providing loosely coupled systems where new business services can be composed and reused. The enterprise of Data requirements can be explained by the RDBMS (Relation Database Management System) term. An enterprise which is based on the RDBMS would be willing to adopt either dedicated database servers and present query services to SOA components or a database application. These two designs are accepted for more than five years. This is very much successful as because of the three dimensions that are Query as a Service, or QaaS. The QaaS is not directly linked to the Data storage but is mapped to the data storage by a single RDBMS architecture. The major problem of maintaining the big data is the double entry or the duplication; this is easily solved by the single architecture system.

Most Big Data are not maintained in a frequent manner as it will be massive, the data can be non-structure, non-relational, or even non-updated. It is very difficult to abstract it to a query service is it is not in good form. The big data is not easy to store in an order as it will be in different places where it is connected by networks or the data's can be in different formats. These types of problems are handled easily by the two different broad choices in SOA that is Horizontal and Vertical integrated data model.

The Horizontal integrated data module uses multiple interfaces to the applications and provides a data management facility with all the integrity the actual data is collected behind an abstract set of data service. The data are not accessed by the components directly, but the process is done in the form of service as the same way that is done in the single RDBMS architecture. The data management an the application components are kept separate and maintained not keep in contact. While this approach can't create the simple query model of RDBMS for the reasons already given, it atleast replicates the simple model of RDBMS that we presented above.

The vertically integrated data model links application data services to resources in a more application-specific way, where the customer relationship management or enterprise resource planning or dynamic data authentication application data is largely separated first at the as-a-Service level and that separation is maintained down to the data infrastructure. In this model the application in most of the time might have the Service Oriented Architecture components which access the data storage by itself. To supply a well structured, maintained data integrity there is a hazard of sacrificing the management service as SOA components which delivers the ability to do on various data system doing the common task like filtering the duplication and matching the integrity in database specific ways. This access is easier to conform to the data structure and the application. SOA reference architecture discusses, in detail, in the later section of this chapter.

Therefore, it is essential to understand the foundations of SOA design that can be tailored for big data applications. Figure 1.2 shows the desired design characteristics for solving big data processing requirements. As shown in the figure, the characteristics was evolved from the large systematic literature as well as the requirements for big data analytics which is shown in Figure 1.3.

Figure 1.2 SOA Design Characteristics

However, there are some of the fundamental characteristics of SOA presented by Erl (2008) as follow:

- business-driven
- vendor-neutral
- enterprise-centric
- composition-centric
- Loosely coupled
- Every service has a contract
- Service can be discovered
- Services are abstract
- Services are autonomous
- Services can be composed
- Services are stateless with respect to complete transactions.
- Services are reusable

These are essential characteristics of SOA and SOA based services are business driven and should be designed with principles of loose coupling and message driven services which are autonomous. Figure 1.3 shows the different approaches to Big Data which has provided us an insight into developing an SOA reference architecture for big data analytics that supports various demands for analytics. We have extended this work to include financial data predictions based on SOA (Chang and Ramachandran 2015).

Figure 1.3 Big Data Research Directions

4. SOA Reference Architecture and Infrastructure for Big Data Applications

Component-based software engineering has been successful in leveraging large scale reuse and productivity (Ramachandran 2008; 2011 & 2013). Therefore, components based service development has been a natural choice as it supports service design principles as well as big data design principles of security, privacy, and large scale, real-time processing, and customisation. Figure 1.4 provides a service component model for Big Data Applications. It is customisable and provides services with two types of interfaces such as providers and requires. The requires interface is semi-arc with interfaces on IRealTimeData and ISensorData. The provider interfaces include IDataPreProcessing, to IDataCustomisation&Presentation.

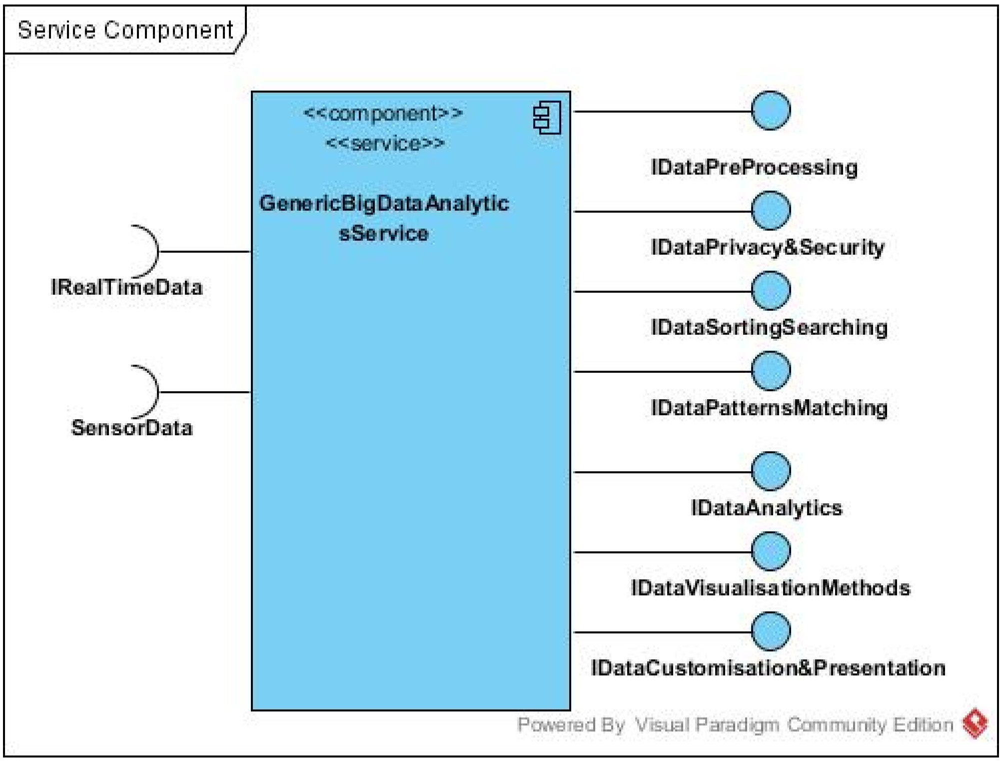

Figure 1.4 Service component model for Big Data

Designing relevant architecture is the key for processing and analyzing data at the required scale. Oracle (2013) has proposed a reference architecture for big data analytics and discusses that any reference architecture for big data should consider including *Any Data Any Source*, full range of applications, and integrated analytic applications. The Oracle's proposed reference architecture is based on layered model and focuses three main layers of the architecture: universal information management, real-time analytics, and intelligent processes. However, since it is based on a traditional layered architecture model which will have the similar problems as the traditional IT systems. Therefore, in this paper has proposed a reference architecture which is based on SOA which is to solve classical problem of customisation, composability Secure Big Data Service Bus, interoperability, etc. Figure 1.5 shows the SOA reference architecture model for big data processing and analytics. As shown in the figure, it divided into four layers, first layer consists of Business and Orchestration where new business services in big data monitoring, analyzing, organizing, and prediction takes place. The second layer is the most important layer that integrates and scales big data applications and data sources, is known, as. This is the backbone of SOA concept that all communication and data flows through the service bus which is central to enterprise integration.

Figure 1.5 SOA Reference Architecture for Big Data Processing & Analytics

The third layer shows Data Analytics and other Big Data Services such as guided analytics, integration, policies, event handling, business rules, business activity monitoring, etc. This also includes horizontal and vertical data integration.

5. SOSE Iterative Methodology to Soft Compute Approach for Big Data Analytics

Service computing has emerged to address IT systems development and maintenance life cycle by reducing cost and improving efficiency and reuse. However, existing approaches to service computing has been adhoc and opportunistic. This chapter proposes a systematic approach which is based on established software engineering best practices. Figure 1.6 shows SOSE Approach to Big Data Analytics. This consists of a number of steps that are interactive, starting from our SoftBD which provides data from multiple sources. The next step is to use soft computing models to analyze data and conduct some prediction patterns, knowledge discovery. The next step is to use BPMN models for business intelligence, this process include developing the UI interface, simulation to validate business process to make sure they are viable and efficient. The next phase is to develop service requirements using traditional requirements engineering process such as use cases, story cards, etc., The next step is to start developing service design where task modelling approaches can be used, and finally test and deployment.

Figure 1.6 SOSE Iterative Methodology to Soft Compute Big Data Analytics

6. Case Study: British Energy Power & Energy Trading Ltd (BEPT)

The concepts of service orientation (Erl 2005, p. 291) would be applied to analyze, design and implement a new system for the British Energy Power & Energy Trading Ltd i.e. BEPET. First, the business process would be modelled using business process model notations i.e. BPMN models (K Pant and M Juric 2008). The BPMN modelling would be carried out using Bonitasoft (Bonitasoft 2015). Using advanced features of the tool, user interfaces would be generated and a process simulation would be carried out for particular workflows. Services would then be identified to realize these workflows. Next use case models would be generated using Enterprise Architect (Enterprise Architect 2015). All identified services would be classified (Sommerville 2010 and Erl 2005, p.392) and any non-functional requirements would be identified. Next, as a part of the Service Design, the system architecture comprising of the orchestration, business & application layer services would be identified (Erl 2005, p.336). Then the component models would be generated using Enterprise Architect (Enterprise Architect 2015) elaborating how these services would interact with each other.

And finally these services would be implemented in J2EE (Java EE 2015) using SOAP (SOAP Specifications 2015) and WSDL (Web Service Definition Language 2015) technologies.

British Energy Power & Energy Trading Ltd. (BEPET) totally rethought both the way it designs software and its relationships with suppliers as a result of shifting to a service-led model for delivering IT. The energy company moved to a process-driven architecture, deliberately shunning the SOA label to disassociate itself from the hype. The move enabled BEPET to prioritize processes and make IT better serve the business. "Processes are the DNA of our organization. We had to focus on higher-value activities, rather than factory-type programming, to be in good shape for future business. There are no prizes for second place," says Jeremy Lock, IT manager at BEPET.

Having an SOA would enable the IT department to support value added processes, rather than support functionality in a more piecemeal way. In order to facilitate this shift, BEPET divided its business applications into three categories of services, defining a service as a self-contained and independent unit of work. The three categories were: a task requiring a human decision, an information service and a functional service. Technology services support all of these three categories.

Converting these activities into services has exposed the energy company to some new ways of thinking about intellectual property rights and the execution of design. "Traditionally, we bought packages and did little bespoke development. We would lob our requirements into the marketplace, get bids back, build them, and then accept or reject," Lock says.

However, using an SOA entails a shift in thinking about intellectual property rights. Within the new regime, BEPET looks at the best of breed packages on the market, but does not customize. Where packages are lacking in functionality, the team writes a service to supplement it. "Whether the service is inside or outside a package we have to connect to it. And the intellectual property rights of every service have to be captured within our model."

This means that BEPET may own the intellectual property rights of a service within a package, an unusual concept for some software and system integrators. "The big five consultancies all have their own method for implementing packages. And we are now saying to them, 'we want you to do it our way'," says Lock.

Harvesting reuse, another major objective of the SOA investment, has also called for a radical rethink of the design, says Lock. "You need to design business services at the right atomic level. And an upfront investment in design is crucial if you are going to get reuse later down the line. It really challenges all the normal paradigms of software design and support." Even companies that are compliant with the IT Infrastructure Library (ITIL), breaking everything into a more granular units makes everything more complex by default.

"We have a team of seven supporting 60-odd applications, and when I tell them we are breaking these into services, they are rightly concerned about the risks," says Lock. Governance perhaps represents the biggest expenditure of effort in moving to SOA. "Only 30% of SOA is about development, the rest is about governance and managing services. Working with partners accelerates the adoption rate, but it is important to internalize the lessons and to assume control."

Assumptions

- BEPET sells two types of energy resources i.e. Gas and electricity. These are sold to direct customers (B2C) or to other companies (B2B).
- Direct customers of BEPET can be of two types Regular Customers or Online-Only Customer
- The regular customer receives their bills based on meter readings taken by BEPET while the online customer to submit their own meter readings using the customer user interface and thereby receive an online customer discount.
- Direct customers can pay their bills either by direct debit or after the bill has been generated for each billing period either monthly or quarterly.

- Business Customers have a dedicated Business Sales Team assigned to them.
- Business Customers can buy energy i.e. Gas or Electricity from BEPET only after they have entered into a legal contract with BEPET.
- The Business Sales Team comprising of the Sales Representative and Sales Manager is responsible for creating and maintaining these contracts.
- Business Customers need to enter into a separate contract for each energy type i.e. Gas & electricity that they are interested in buying from BEPET.
- Business Customer contracts are end-dated only i.e. They are not deleted from the system are retained for audit purposes.
- After entering into a contract with BEPET, Business Customers can buy their energy in bulk over a period of time or in exceptional cases request an ad hoc energy top up at the rates agreed in their business contracts.
- Business Customers can only pay BEPET using CHAPS transfer mechanisms since they deal with significantly large amounts.
- Direct Customer can contact the BEPET Call Centre is they have any issues.
- The BEPET Call Centre only deals with the direct customers and not the business customers.
- When customers call the call center, the Call Centre staff would usually take their customer details and use them to search the customer accounts on their internal system. And they can update this system as appropriate.
- The Call Centre staff can also raise a refund request for the customer in exceptional cases. These refund requests need to be approved by their Call Centre Managers before they are sent to the Accounts team for processing of the refund.
- Only the use cases relevant to the BEPET core energy B2B and B2C businesses have been modelled. For instance the Call Centre Manager would also have additional responsibilities like Manage the Call Centre Team, Reporting to the Management, Appraisals, etc. As these

are no more relevant to the BEPET core business they have not been modelled in the Use case diagrams.

- Similarly, only the departments within BEPET that directly deal with the core energy business have been considered. For instance BEPET would have additional departments like Marketing, Sales, HR, etc. As these are not directly involved in the energy business they have not been considered for this activity.
- Meter Reader travel to the local end user premises with handheld devices which can directly synchronize with the BEPET servers.
- It is assumed that Inventory Management System is an existing process written in a low level language and it works as expected. Since it very rarely requires updates and due to the prohibitively high cost & potential business impact of migrating it to a new system, BEPET has taken a business decision to not migrate it to the new system for now.

Business Process Modelling

To understand the various business workflows, the business processes for BEPET would be modelled using BonitaSoft (Bonitasoft 2015). Before the BPMN models are generated, the various actors involved in the business process are identified based on the role that they perform within the business process & the workflows that these actors would be involved in are shortlisted (Pant and Juric 2008). These actors can either be external actors like the Direct Customer or the Business Customer or they can be internal actors like the Call Centre Representative, Business Sales Representative, etc. There are also some automated 'system' actors like the Payment Gateway, Bank Interface, etc.

List of workflows and business processes

The following list of workflows has been identified for external actors as shown in Table 1.

Table 1.1 External Actors

Actors / Roles	Workflows	Workflow Business Processes
Direct Customer	Direct Customer UI Interaction Workflow	Login
		Record Readings
		View Bill
		Pay Bill
		Add/Update Direct Debit Details
		Check Energy Consumption
		View/Modify Personal Details
Business Customer	Business Customer Interaction Workflow	Buy Energy in bulk
		Ad Hoc Energy Top Up
		Make Payment

BPMN Process Models for BEPT

Once all the workflows have been identified, the business process models have been generated using the Bonitasoft Community Edition BPM Studio software. (Bonitasoft 2015). As seen from the business process model, the interactions between the various business processes have been captured well. Additionally, the trigger for each business process as well as the exchange of any business messages / notifications have been captured in the BPMN model.

User Interface screens using Bonitasoft

The workflow for Business Sales was used to generate the sample User Interface screens for BEPET (Create and Run Your First Process 2015). The next few screenshots indicate how this process was carried out. To generate these UI screens the following workflow process was utilized and the BPMN model for adding a new contract is shown in Figure 1.7 and its UI interface is shown in Figure 1.8.

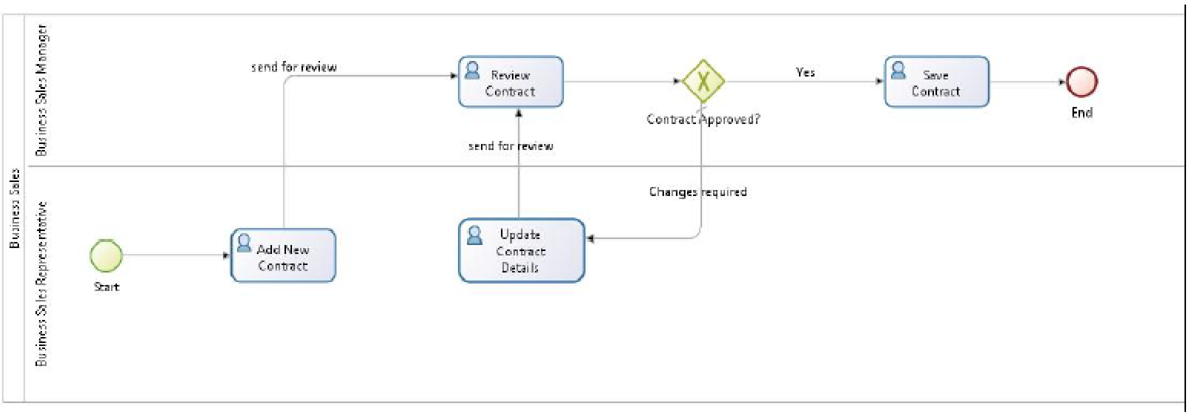

Figure 1.7 New Contract Business Process

The groups, roles and users in the BEPET organization were created for Billing, Call Centre, Customer Direct, meter reading, Overall manager, Sales manager, and sales rep.

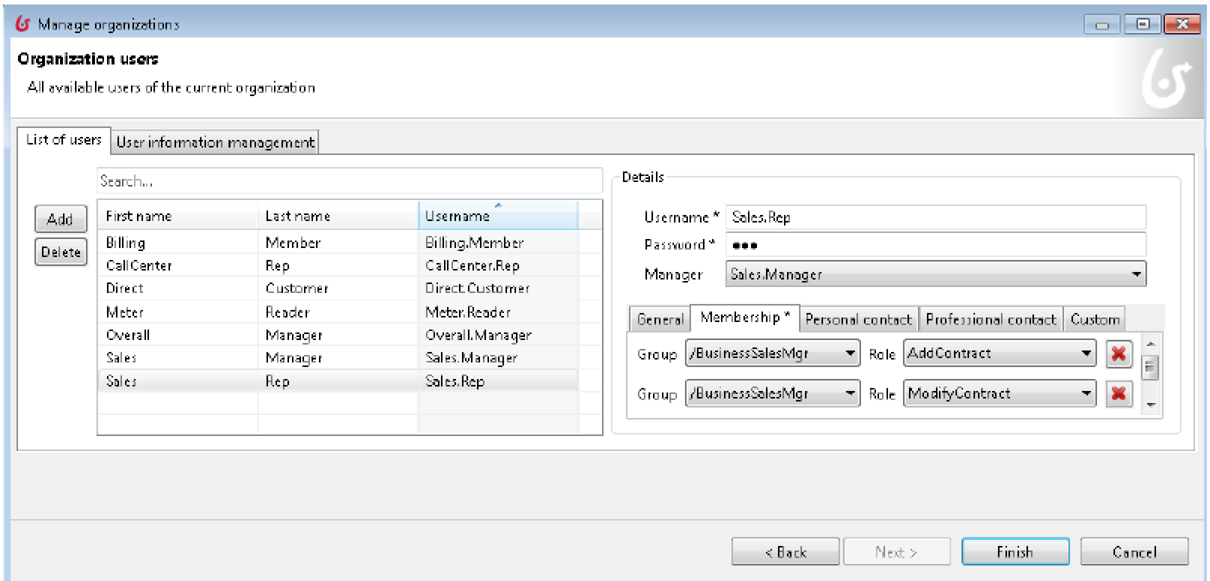

Figure 1.8 UI Interface for New Business Contract

Process Simulation using Bonitasoft

The workflow for 'Take Meter Reading' was used to perform a process simulation activity in Bonita BPM Studio (Simulate Processes for Better Optimization 2015). This process is used to identify opportunities for process optimization. The next few screenshots indicate how the simulation process was carried out.

To generate these reports using Bonitasoft the following workflow process was utilized as shown in Figure 1.9.

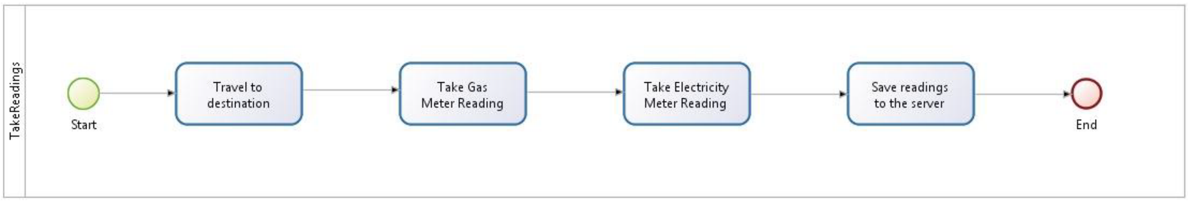

Figure 1.9 Meter Reading Business Process

For each task in the workflow the required time was configured as shown in Figure 1.10.

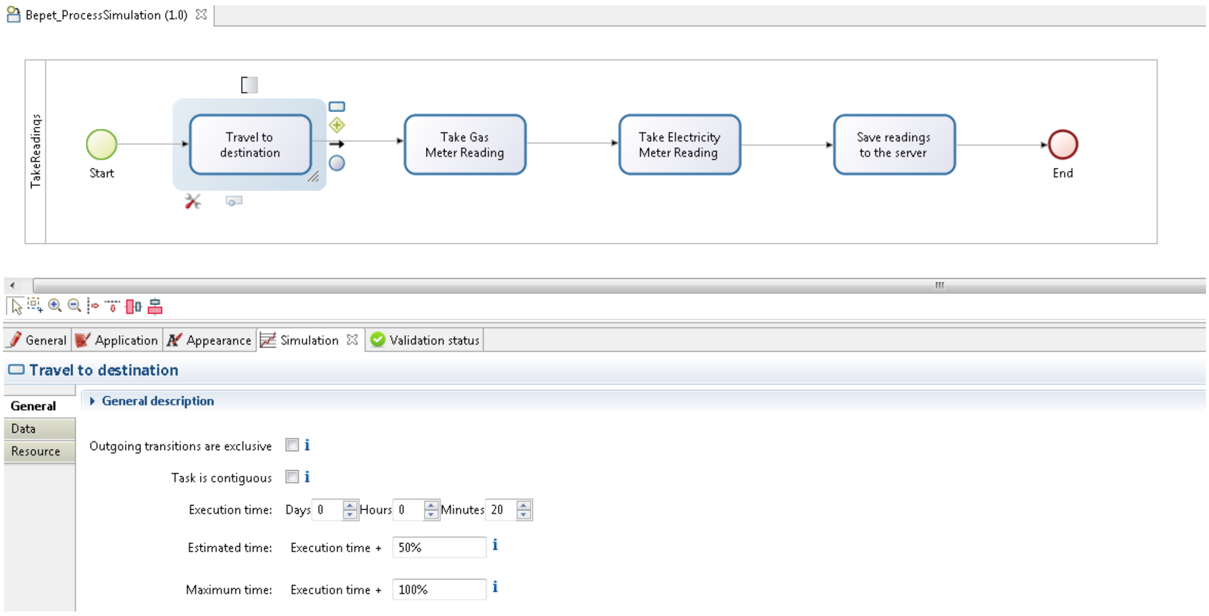

Figure 1.10 Simulation Parameters for Meter Reading

And the resources needed to perform the task were configured as shown in Figure 1.11.

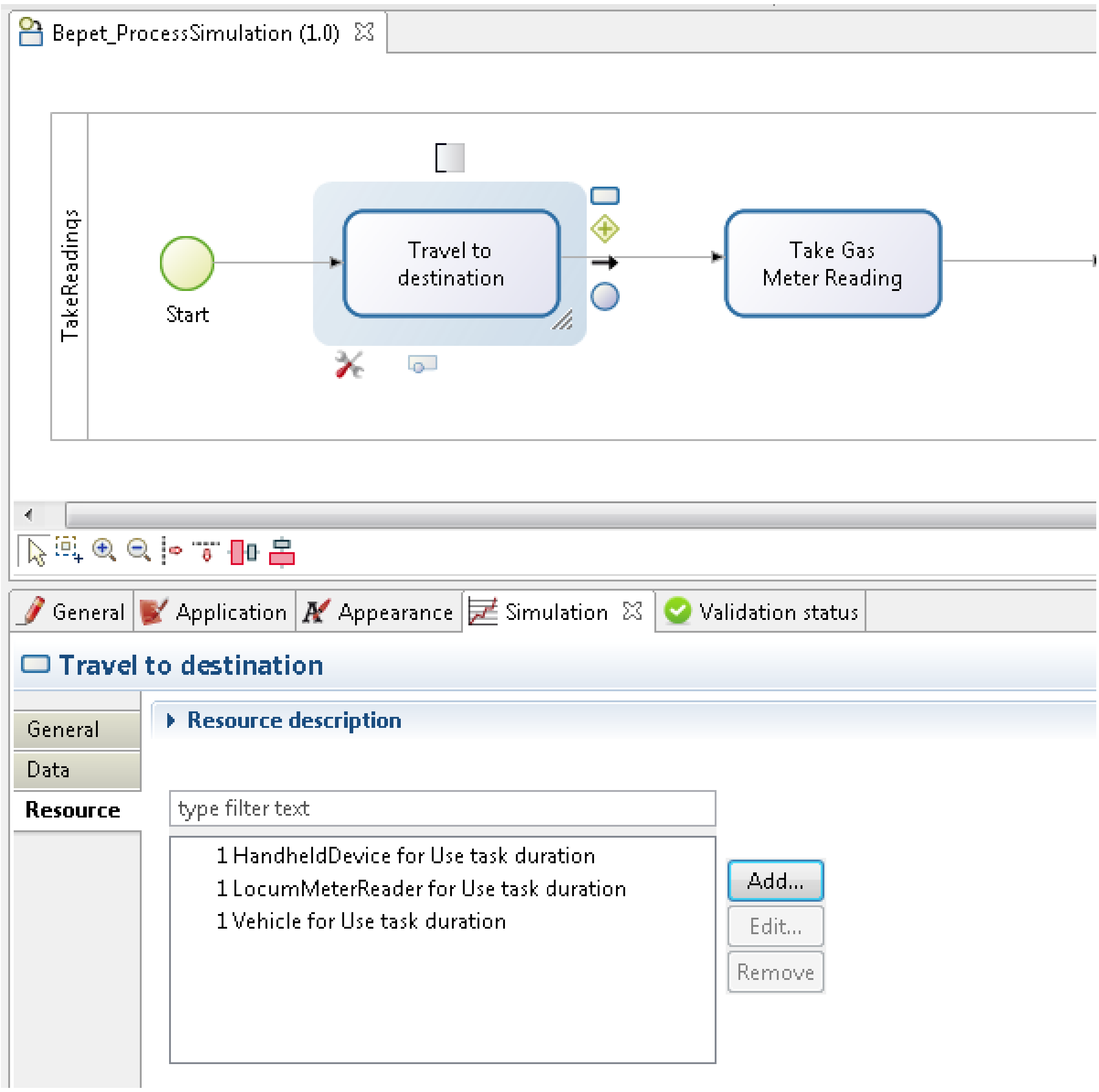

Figure 1.10 Resource configuration simulation parameters

Next required load profiles were added as shown in Figure 1.11.

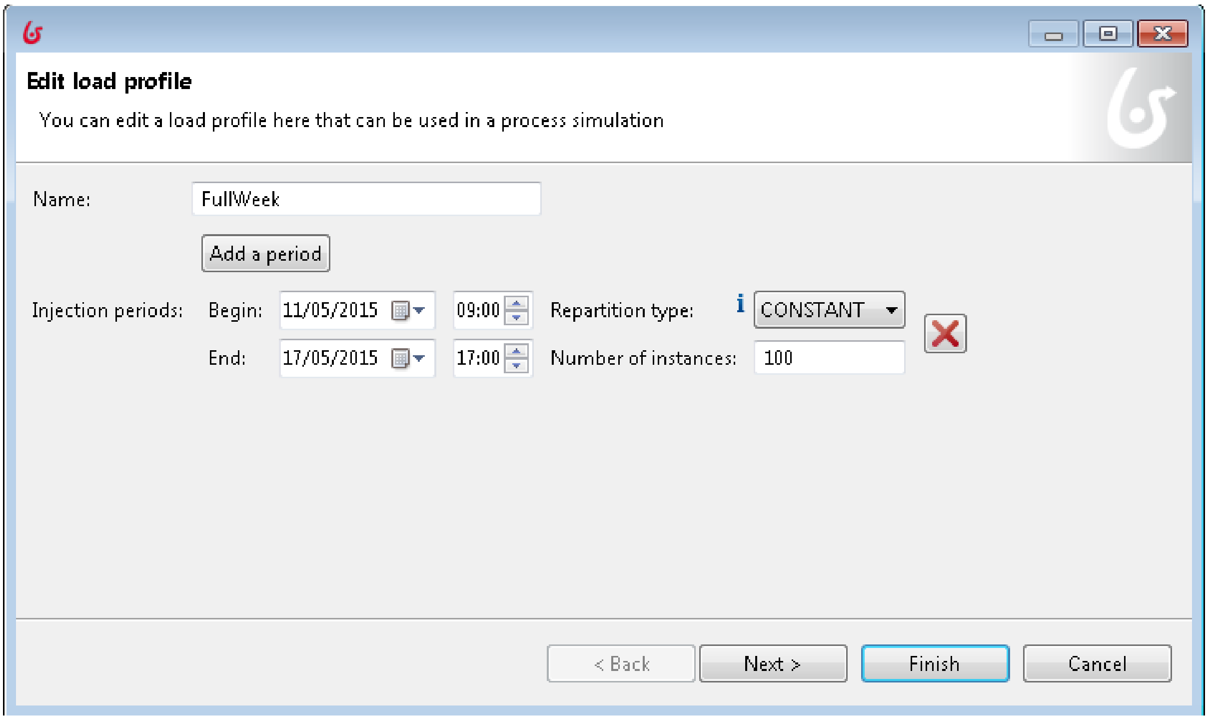

Figure 1.11 Load Profile for Simulation of Meter Reading BPMN

And finally the simulation process was started as shown in Figure 1.12.

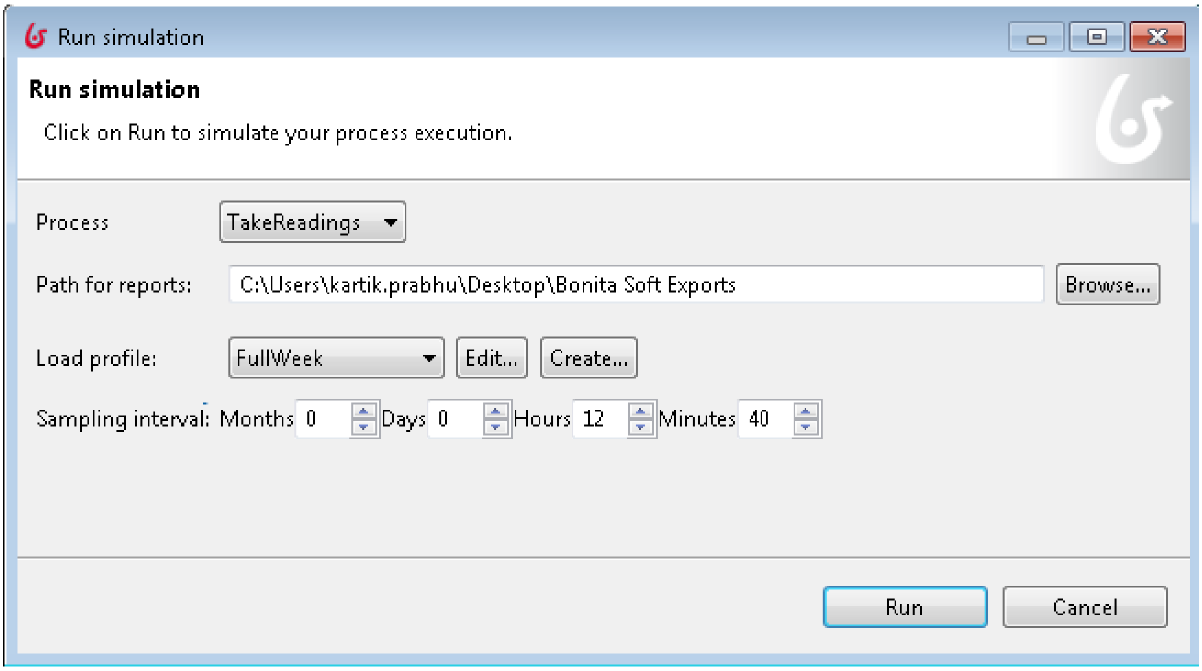

Figure 1.12 Starting the business process simulation

This produced the report as shown below as shown in Figure 1.13.

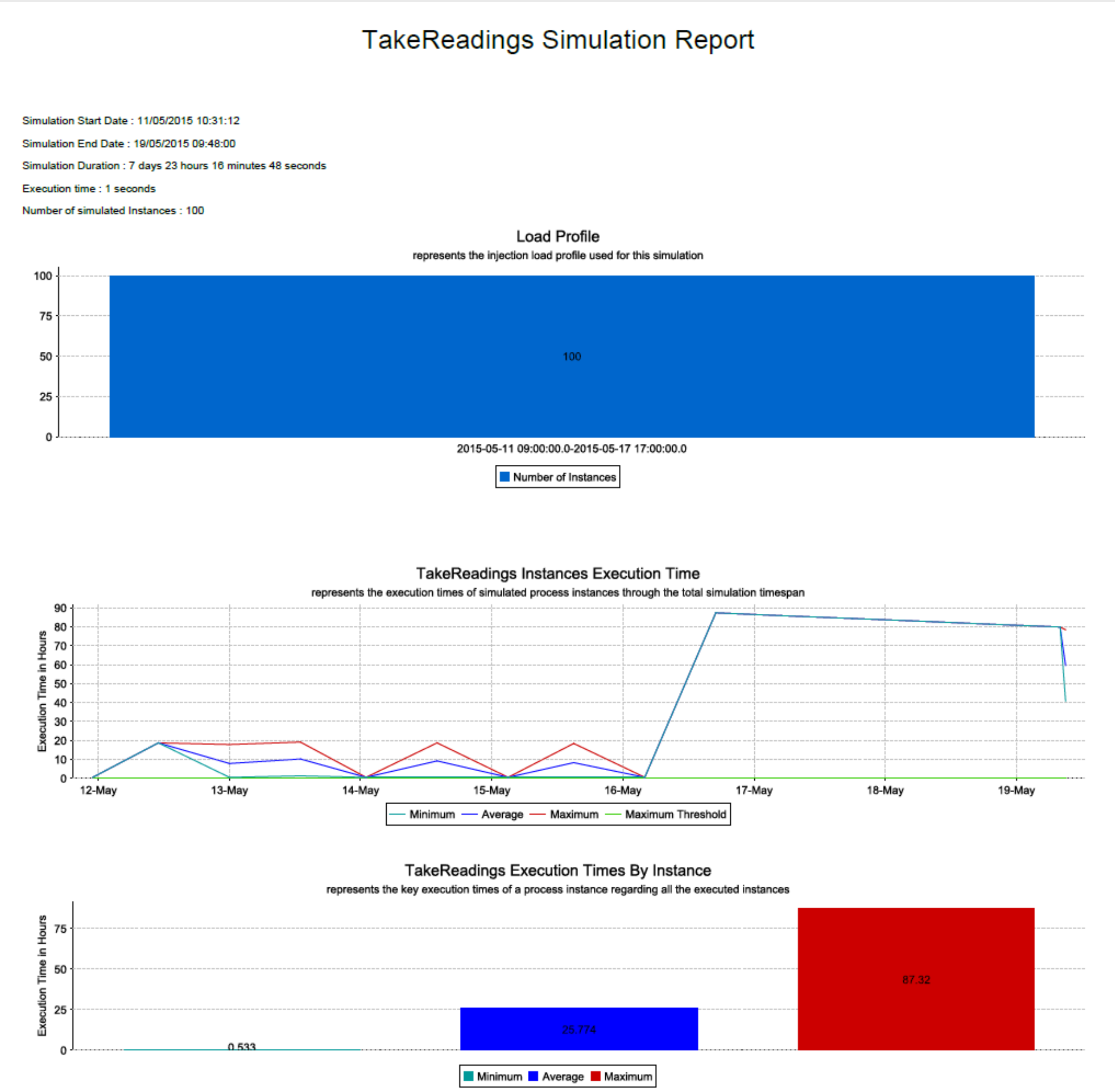

Figure 1.13 Simulation Output Graphs

Cost Optimization

The cost of providing the 'Meter Reading service' to 'Regular Customers' was identified using this process simulation. This also allowed BEPET to determine the extent of discount that it can offer its

‘Online-Only Customers’. This allowed BEPET to differentiate itself from other similar energy providers by providing additional cost savings. These are the manpower cost estimates generated as shown in Figure 1.14.

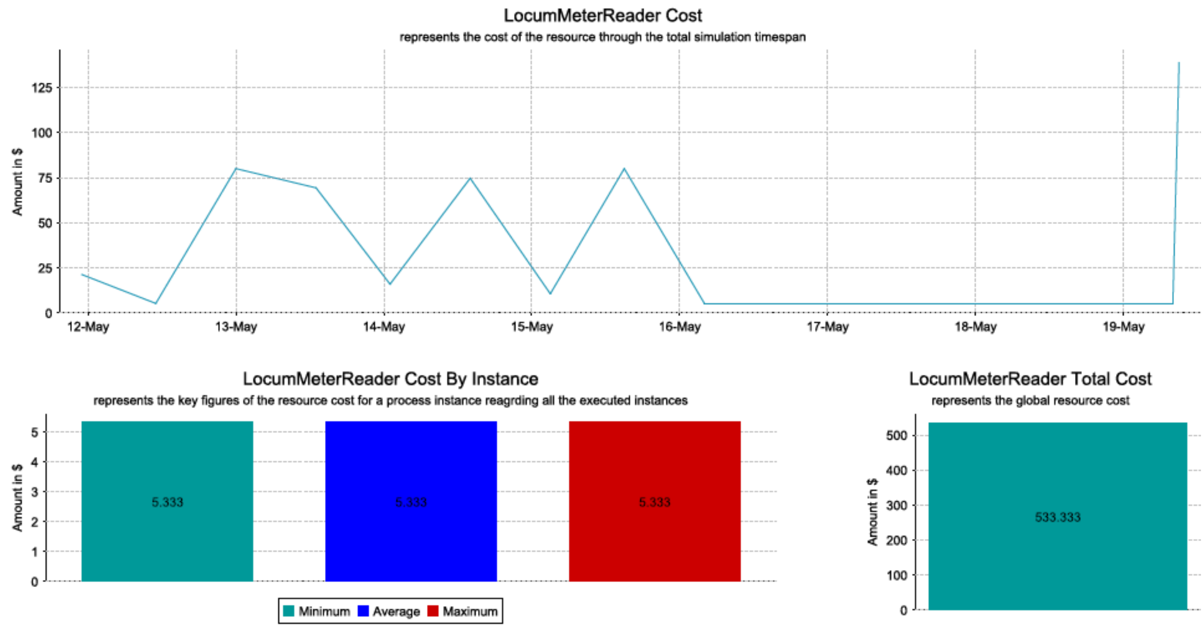

Figure 1.14 Cost optimization for meter reading business process

Resource Profiling

Process Simulation also enabled BEPET to identify an optimal resource profile for

Its Meter Reading Operations. It could identify using various load profiles, varying on the

basis of number of customers, to determine how many Meter Readers it would need to

hire and whether it would be advisable to hire locum Meter Readers or simply hire them

as permanent staff. BEPET could also perform a comparative assessment of providing the

meter reading services only on weekdays v/s throughout the week. As seen from the graph below the wait time increases over the weekend due to lack of human resources as shown in Figure 1.15.

Figure 1.15 Resource Profile Simulation

7. Conclusion

This chapter has contributed a unique and innovative and generic SoftBD framework, service component model, and a generic SOA architecture for large scale big data applications. This chapter has also contributed to Big Data metrics which allows to measure and evaluate when analyzing the data. Developing a new system for BEPET has made it possible to apply the concepts of service orientation. Business process modelling has allowed a better understanding of BEPET's businesses. The application of the various techniques for service analysis and design has enabled a better understanding of the decision-making involved in developing a real world system. The execution has made it possible to explore the subjects confronted in developing complex system. The proposed methodologies SoftBD and SOSE have demonstrated a systematic engineering approach to full life-cycle service engineering, which has resulted in high quality services and assurance.

8. References

Curko, K. Bach, P. M., Rodanic, G (2007) Business Intelligence and Business Process Management in Banking Operations, Proceedings of the ITI 2007 29th Int. Conf. on Information Technology Interfaces, June 25-28, 2007, Cavtat, Croatia

Erl, T (2005). Service-oriented architecture: concepts, technology, and design, prentice hall, 2005.

Gorton, I (2013) SEI Blogs, Addressing the Software Engineering Challenges of Big Data, Ian Gortan's report, https://insights.sei.cmu.edu/sei_blog/2013/10/addressing-the-software-engineering-challenges-of-big-data.html

Gorton, I (2014) The Importance of Software Architecture in Big Data Systems, https://insights.sei.cmu.edu/sei_blog/2014/01/the-importance-of-software-architecture-in-big-data-systems.html

Li, Z, et al. (2015), Enabling Big Geoscience Data Analytics with a Cloud-Based, MapReduce-Enabled and Service-Oriented Workflow Framework, PLOS ONE, DOI:10.1371/journal.pone.0116781 March 5, 2015

Neves, C. P, and Bernardino, J (2016) Big Data in the Cloud: A Survey, Open Journal of Big Data (OJBD) Volume 1, Issue 2, 2015

Oracle (2013) Big Data & Analytics Reference Architecture, An Oracle White Paper

Ramachandran, M (2008) Software components: guidelines and applications, Nova Science, NY.

Ramachandran, M (2011) Software components for cloud computing architectures and applications, Springer, Mahmood, Z and Hill, R (eds.). www.springer.com/computer/communication+networks/book/978-1-4471-2235-7

Ramachandran, M (2013) Business Requirements Engineering for Developing Cloud Computing Services, Springer, Software Engineering Frameworks for Cloud Computing Paradigm, Mahmood, Z and Saeed, S (eds.), http://www.springer.com/computer/communication+networks/book/978-1-4471-5030-5

Chang, V and Ramachandran, M (2015) Quality of Service for Financial Software as a Service, ESaaSa 2015-CLOSER 2015

Zimmermann, A., et al. (2013) Towards Service-oriented Enterprise Architectures for Big Data Applications in the Cloud, 17th IEEE International Enterprise Distributed Object Computing Conference Workshops, 2013.

Apache Ant (2015) - Welcome. 2015. Apache Ant - Welcome. [ONLINE] Available at: http://ant.apache.org/. [Accessed 11 May 2015

Apache Axis2 (2015) - Apache Axis2/Java - Next Generation Web Services. 2015. Apache Axis2 - Apache Axis2/Java - Next Generation Web Services. [ONLINE] Available at: http://axis.apache.org/axis2/java/core/. [Accessed 11 May 2015].

Bonitasoft (2015) Bonitasoft - Open Source Workflow & BPM Software. [ONLINE] Available at: http://www.bonitasoft.com/. [Accessed 05 May 15].

Chef (2015) Chef IT automation for speed and awesomeness | Chef. 2015. Chef | IT automation for speed and awesomeness | Chef. [ONLINE] Available at: https://www.chef.io/chef/. [Accessed 11 May 2015

Bonitasoft (2015) Create and Run Your First Process | Bonitasoft | Open Source Workflow & BPM software. 2015. Create and Run Your First Process | Bonitasoft | Open Source Workflow & BPM software. [ONLINE] Available at: http://www.bonitasoft.com/for-you-to-read/videos/create-and-run-your-first-process. [Accessed 03 May 2015].

Enterprise Architect (2015) Enterprise Architect - UML Design Tools and UML CASE tools for software development. [ONLINE] Available at: http://www.sparxsystems.com/products/ea/. [Accessed 06 May 15].

Sommerville, I (2010) Software Engineering, 9th Edition. Pearson Education

Java Platform (2015), Enterprise Edition (Java EE) | Oracle Technology Network | Oracle. 2015. Java Platform, Enterprise Edition (Java EE) | Oracle Technology Network | Oracle. [ONLINE] Available at: http://www.oracle.com/technetwork/java/javaee/overview/index.html. [Accessed 10 May 2015].

Pant, K and Matjaz, J (2008). Business Process Driven SOA using BPMN and BPEL. Online: Packt Publishing.

Bittner, K and Spence, I (2002). Use Case Modelling. Online: Addison-Wesley Professional.

Maven (2015) – Welcome to Apache Maven. 2015. Maven – Welcome to Apache Maven. [ONLINE] Available at: https://maven.apache.org/. [Accessed 11 May 2015].

Puppet Labs (2015): IT Automation Software for System Administrators. 2015. Puppet Labs: IT Automation Software for System Administrators. [ONLINE] Available at: https://puppetlabs.com/. [Accessed 11 May 2015].

Bonitasoft (2015) Simulate Processes for Better Optimization | Bonitasoft | Open Source Workflow & BPM software. 2015. Simulate Processes for Better Optimization | Bonitasoft | Open Source Workflow & BPM software. [ONLINE] Available at: http://www.bonitasoft.com/for-you-to-read/videos/simulate-processes-better-optimization. [Accessed 05 May 2015].

SOAP Specifications (2015). SOAP Specifications. [ONLINE] Available at: http://www.w3.org/TR/soap/. [Accessed 09 May 2015].

Erl, T (2005). Service-Oriented Architecture: Concepts, Technology and Design. Online: Prentice Hall.

Erl, T; Pethuru Chelliah; Clive Gee; Jürgen Kress; Berthold Maier; Hajo Normann; Leo Shuster; Bernd Trops; Clemens Utschig (2014). Next Generation SOA: A Concise Introduction to Service Technology & Service Orientation. Online: Prentice Hall.

WSDL (2015) Web Service Definition Language (WSDL) [ONLINE] Available at: http://www.w3.org/TR/wsdl.html. [Accessed 09 May 2015].

5Vs model, 2

Big Data Analytics, 1

Business and Orchestration, 9

business workflows, 15

guided analytics, 10

key business improvement indicator, 2

Secure Big Data Service Bus, 9

Service Component model for big data applications, 1

SOA, 1

SOA based soft compute framework for big
data (SoftBD), 4
SOA Based Soft computing Framework for Big
Data, 3
SOA reference Architecture, 1
soft compute intelligence algorithms, 4
SoftBD, 1
SOSE Approach, 10