A Distributed Simulation Approach for Contractor Company Performance Management in the Construction Industry

by

Ronald Ekyalimpa

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

Doctor of Philosophy
in
Construction Engineering and Management

Department of Civil and Environmental Engineering
University of Alberta

ABSTRACT

Acquiring and sustaining an advantage over competitors in an era that is characterized by more complex and massive projects, scarce resources, more stringent client requirements, and higher expectations from board members is quite challenging for construction companies. It is believed that implementing sound performance management systems is an effective way of overcoming this challenge. A number of traditional performance management techniques such as the Balanced Score Card (BSC), Key Performance Indicators (KPIs), Data Envelopment Analysis (DEA), etc. have been in use, but have shown a number of problems. Examples of these include: they are self centered, data driven, tedious to use and lagging.

In an attempt to overcome the problems with traditional performance management systems, a number of simulation-based systems were created. These also had their shortfalls. For example they did not model the project arrival process and competitive work acquisition process explicitly. They also did not model the work execution process and performance measure generation process in detail.

This thesis study set out to advance the state-of-the-art of simulation-based performance management systems. Real world constructs that relate to the business operations of a typical construction company were abstracted and represented using different simulation paradigms. For example, the competitive work acquisition process was modelled using an agent-based approach because of the interaction that exists between autonomous or semi-autonomous and concurrently self-executing constructs. On the other hand, the execution of awarded projects at the companies was emulated using a Discrete Event Simulation (DES) modeling approach.

The agent-based model was developed using the AnyLogic simulation system while Simphony and Visual Studio were used for developing the DES model. Subsequently, these two components were configured into High Level Architecture (HLA) federates and integrated to form a distributed simulation system using a distributed simulation framework known as COnstruction SYnthetic Environment (COSYE).

A simulation-based performance management application was developed in this study. A number of insights were gained in the course of developing the application. For example, robust design patterns and system architecture were used that could be applied in solving other similar complex problems. Also, a number of approaches were devised for effectively modeling different ill-structured phenomena (such as safety and quality) that exist within the construction domain.

Verification and validation work done on the developed application proved that the application was reliable and realistic.

PREFACE

This thesis is an original work by Ronald Ekyalimpa. The research project, of which this thesis is a part, received research ethics approval from the University of Alberta Research Ethics Board, Project Name "An Investigation of Performance Management Best Practices at Select Contractors in the Alberta Heavy Civil and Industrial Construction Industry.", No. Pro00043578, 24thOctober 2013.

Some of the research conducted for this thesis has been published or will be published, and represents collaborative work done with Dr. Simaan AbouRizk, and Ronald Ekyalimpa at the University of Alberta.

A summarized version of Chapter 7 was submitted to the EMSS 2014, as Ekyalimpa, R., & AbouRizk, S. (2014). A Multi-Agent Model that Emulates the Typical Competitive Bidding Process in the Construction Industry. *The 13th International Conference on Modeling and Applied Simulation.* Bordeaux, France.

ACKNOWLEDGEMENTS

Priscilla, it's been long since we left home in pursuit of this PhD. You always inquired about the tasks I worked on each week. Initially, I thought this would end along the way but to my surprise, you never stopped. You have been extremely supportive all the way and I am truly grateful for that. You always joked saying that I needed to complete my thesis before Elisha makes his first baby steps. I am happy to have achieved that.

Nathanael, Ethan and Elisha; you guys are the best gift that God has given Priscilla and myself. The best part of my typical work day is seeing you guys run towards me in excitement when I get home. You guys and mum make me smile regardless of how my day had been.

A big thank you to my siblings Denis Owoyesigyre, Agnes Rugumire Nuwagaba, Rosette Baroraho Arinaitwe, and Fabian Mpoora and their families for being so supportive and staying touch with my family. Thanks for the love you have shown us. I also like to convey my appreciation to my mum and late dad (Mr. & Mrs. Nyangababo) for taking good care of me and educating me. All the good in me is a result of your parenting. You are the best parents that one could have ever wished for. I love you so much. I am also grateful to my father and mother in-law (Mr. & Mrs. Tuunde) for always being there for us.

There are family friends that have been extremely close and supportive during our stay in Edmonton. Gerald Karegeya and family, Edgar Twine and family, Joseph and family, Gertrude, Emmanuel and family, Dan Keragi, Robert and Catherine Thornberry. Thanks for being true friends.

Dr. AbouRizk, you have been many things to me; my instructor in class, my academic supervisor, a mentor, a boss, and a father figure to me. The majority of the tasks you assigned me were extremely challenging – especially this thesis study but also quite interesting and enabled me learn & grow academically and professionally. You never constrained me in approach to problems and allowed me sufficient time and space to get things done; something that has paid off. The training and mentorship you offered me has been rigorous and top notch and I would not trade it for anything else.

Dr.Yasser, whenever I met you in the corridor or walked past your office, you always took a moment to say hello and inquire how everything were with me, the family and my research. It was things like that which made a lot of difference especially when the going got tough. Thanks a lot for your genuine concern and kindness.

I would also like to thank other professors in the Construction Engineering and Management research group that have taught me a course or two during my PhD program. These include Dr. Robinson Aminah, Dr. Al-Hussein Mohamed, and Dr. Ming Lu.

I am truly grateful to Stephen Hague for the skill and knowledge that he divulged to me. You were also very swift at extending the functionality of the simulation tools to avail services that I needed to implement developments in this thesis. I thank you for that. As a side note, your thought process and approaches to solving analytical problems still fascinates me to this day. You and Dr. AbouRizk made my learning experience through the PhD journey worthwhile.

I would also like to thank the co-op computer science students (Han, John, James, Steven, Aaron, Christian, Jeanine, Terence, and Timotei) that worked under Steve’s supervision during my time in the program. There were times that we were assigned to work on some things together and I enjoyed those moments.

I would like to convey my appreciation to Brenda Penner for all she has done. You always gave me guidance on what to do when I was in doubt about administrative processes and systems. You also took to time to inquire whether everything was well with me and my family and did what was in your means to ensure that I had a comfortable work environment at the Department. Thanks very much.

Maria, I enjoyed the time we spent working on the projects with the industry partners. I got to learn a number of things from our interaction. I am grateful for that.

Amy thanks for reviewing this thesis and making sure that it was of good quality. Thanks too for your work in getting technical papers reviewed and above all for understanding the few times that I was not able to submit work to you in time.

Hexu, Mona, Hamid, Di, ND, Abraham, Moataz, Farhad, Mao, Shawn, Morley, Lance, Nasir, Francis, Sherry, Veronica, Mubarak, Shade, William, Naimeh, and Muaz; thanks so much for being good friends within and outside school. You guys were also quite supportive throughout the course of my studies.

I would also like to thank my colleagues (i.e., Fayyad Sabha, Kimoya Tucker, Masha Ghaznavi, and Jangmi Hong) who I started the program with and shared an office with in the earlier years of my program. Thanks for your friendship and for staying in touch.

Last but not least, I am truly thankful to God for all the blessings, gifts and his will to see this through. There was nothing I could have achieved without you. All credit and glory for all that has been achieved in this thesis and my life in general goes to you and your son, Jesus.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE—INTRODUCTION

1.0 BACKGROUND

The construction industry is viewed as a *goods-producing* industry within most economies around the world. It usually ranks amongst the top 20 contributors to the Gross Domestic Product (GDP) for the majority of economies. For example, in the past 5 years, the construction industries in the US and Canada have contributed significantly to their economies, with the industry in the USA averaging a contribution of 3.7%, and that in Canada averaging 7.0%. These figures are based on data accessed from the U.S. Bureau of Economic Analysis (2014) and Statistics Canada (2014), respectively. A graph showing the contributions of the construction industry to the GDP of USA and Canada respectively from 2009 to 2013 is presented in Figure 1.0.

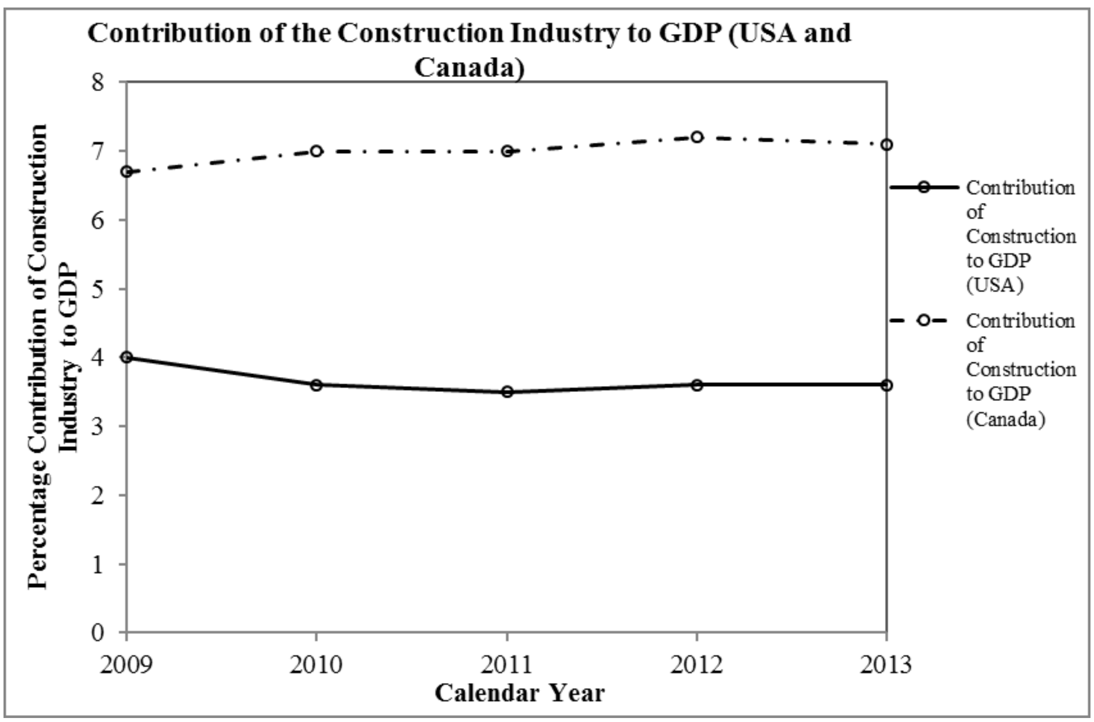


Figure 1.0: Contribution of the Construction Industry to National GDP in USA and Canada

According to GPS USA Division (2014), the construction industry plays a dominant role in sustaining a nation's growth because the infrastructure it develops facilitates other economic activities, i.e., the transportation of goods, services and commuters, provide office space and shelter.

Typically, a significant portion of the effort to enhance the productivity and competitiveness of economies have been dedicated to creating more robust and effective construction industries because they have been known to traditionally underperform relative to other industries. (Lee, Cooper, & Aouad, 2000; Smith, 2001; Kagioglou et al., 2001; Yang et al., 2010).

This begs the question: *What is the most effective way to improve the competitiveness (efficiency) of a given construction industry?* Some researchers have proposed an option that involves the adoption of performance measurement methods to improve the state of the construction industry (Latham, 1994; Egan, 1998). There have been studies that followed these recommendations by attempting to ratify performance management systems within the industry. The majority of these have not served the industry as expected because of its uniqueness among other challenges. Consequently, there has been a recent drive towards developing more robust performance management systems that address the unique issues that exist within the construction industry. Examples of these can be found in Orozco, Serpell, Molenaar, & Forcael (2014) and Ekyalimpa & AbouRizk (2014).

There have also been mixed signals about where to focus efforts meant to enhance performance in construction. Some have argued that it may be better to start at an industry level and concentrate most of the resources and effort there in an attempt to enhance the competitiveness of an entire industry. Another school of thought suggested directing these efforts at the company level (bottom-top approach), rather than at an industry level. Others, however, proposed addressing competitiveness issues at both levels, either concurrently or sequentially.

This thesis proposes a holistic approach in dealing with competitiveness problems in any construction industry. At an industry level, most competitiveness issues arise when local jobs are shipped out of the country to be performed by foreign competitors. This typically occurs when the indigenous companies over-price their services relative to their foreign counterparts. In most cases, higher pricing will arise from inefficiencies in the production and business operations of the construction companies. Another reason for jobs being shipped abroad could be a lack of sufficient production capacity among indigenous construction companies. The second competitiveness issue that construction industries face is the entry of foreign companies into the local market; foreign companies pose a serious threat of acquiring a significant portion of the market share.

Most of the problems that arise with respect to competitiveness at a company-level will vary from company to company. For example, the challenges faced by small companies are different from those faced by large companies. Small companies will typically be faced with the challenge of acquiring resources needed to support their survival and growth for example access to good lines of credit (financing), access to efficient workers (human resources) and the acquisition of work (projects). On the other hand, the large companies typically face a challenge of managing their resources, e.g., finances, workers and jobs. Both company categories could face efficiency problems in their technical operations.

To a large extent, the competitiveness challenges faced at an industry level are different from those faced at a company-level however, there is an overlap, to a limited extent, with regards to operational inefficiencies at companies as well as over pricing for construction services, across the board for the construction industry within which those companies operate.

1.1 PROBLEM STATEMENT

A number of performance management techniques have been proposed for enhancing company performance. The early 1920's saw the advent of metrics used to track financial performance. A popular example is the Return on Investment (ROI) proposed by DuPont. The late 80's saw the introduction of performance measures that relied on cost and non-cost measures with a typical example of the performance pyramid. In 1992, the balanced score card was introduced which was later followed by the introduction of the European Foundation for Quality Management (EFQM) excellence model in Europe. There are also other performance models that have been proposed by other researchers that have not yet been widely used. These performance models helped a great deal in tracking company performance for several years but have limitations. Examples of those limitations include:

- *Most performance management systems are self-centered:* Some performance management systems rely on measuring the performance of a company based on its past data only. No comparisons are made to other companies in similar sectors or industries.

- *Past performance management systems are limited in scope:* The performance management systems used in the past focused mainly on financial aspects of the company. Some recent systems attempted to address this problem by including other measures, but they are still somewhat limited and not customized specifically for companies in the construction industry.
- *Nearly all existing performance management systems are not automated:* Most performance management systems heavily rely on performance data which is usually not centralized, but rather, archived within the different units from which it is generated. This makes the process of appraising a company's performance slow, tedious and prone to error.
- *Performance management systems are static and lagging in nature*: All the performance management systems cited don't support making performance projects based on artificial or realistic scenarios formulated by the analyst. Moreover, they don't take into consideration the prevailing conditions at the company which lead to the generated data and performance. This poses a significant constraint in attempts to develop strategic plans that could enhance a company's performance in the future.

Given the limitations mentioned above, over time it was believed that utilizing simulation-based approaches would address most them and consequently a trail of research was started that made use of this modeling approach. Examples include Orozco, Serpell, Molenaar & Forcael (2014), Du & El-Gafy (2012), Al-Qirem & Yaseen (2010), and Ogunlana, Li, & Sukhera (2003). However, these systems also had a number of limitations, namely:

- *Failure to model project arrival dynamics in a robust fashion:* The community of owners within any given construction industry will announce its intention to invest in projects at different times. This typically depends on the state of the economy (interest rates), the demand for the facility that they plan to construct, etc. This process of the entry of projects into the market is crucial in assessing the performance of a company over a period of time, because the amount of work that a company undertakes at any given moment affects how it performs. Furthermore, the portion of the new projects that get awarded to the company of interest provides a measure of its level of competitiveness, and therefore, this should to be tracked, quantified and aggregated along with other performance measures. Nearly all existing applications that simulate company performance don't explicitly model the arrival of projects into the market; hence, they fail to provide the entire picture of a company's performance to the analyst.

- *Failure to model the acquisition of work (projects) by the company of interest in a realistic way:* In most cases, companies in the construction industry have to participate in a competitive process to acquire work. Representing this process on computer in a realistic fashion requires that the attributes (e.g., production capacity) and behaviors (e.g., bid/no bid decision making and bid price generation) of the company of interest and its competitors be modelled explicitly.
- *Failure to model the project (work) execution process in detail*: The execution of work at a company requires resources. The pace at which this work is performed is therefore based on the availability of these resources and their efficiency. Also, components of this work typically follow a logical sequence that affects the rate at which the work is done. Existing simulation applications modeling company performance don't explicitly model the execution of work awarded to the company. However, the few applications that attempt to do so, do not constrain the processing of this work to resource availabilities, hence, results are not realistic.

The pitfalls that were present in traditional performance management systems, along with those in existing computer applications that simulate company performance (competitiveness), served as a justification for the creation of the computer simulation-based system for managing company performance in this thesis.

1.2 THE PURPOSE OF THE STUDY

The main objective of this thesis study was to propose and develop a holistic simulation-based application that could be used for contractor performance management within the construction industry. It was expected that such a tool could be used for enhancing company performance and competitiveness.

In order to achieve this objective, a number of other sub-objectives had to be achieved first and these included:

I. Identification of performance measures that represent the competitiveness of a typical construction company.
II. Formulation of an efficient approach to represent the different issues related to construction company performance i.e., develop concept models.
III. Develop an application for contractor company performance based on the concept models from objective II.
IV. Verify and validate the reliability and accuracy of the developed simulation application presented in objective III.

1.3 RESEARCH APPROACH

This section discusses the steps that were taken in this study to achieve each of the enumerated objectives.

I. ***Achieving objective I:*** The relevant performance measures were identified after a thorough literature review. This literature review was complemented by a questionnaire survey of contractor companies in the heavy civil and industrial construction sectors in Alberta, Canada that sought to establish the main performance measures that give a good indication of their competitiveness.

II. ***Achieving objective II:*** Prior to the development of a simulation model, the real world system or operation needs to be abstracted. The abstraction process generates concept models that detail the design specifications of the model to be built. To achieve the objective II, a number of design aides were utilized for representing concept models. These included: state charts, sequence diagrams, block diagrams, activity diagrams, and flow charts. The development of these design specifications was based on publications, information acquired from experts and self-ingenuity.

III. ***Achieving objective III:*** Easy-to-use existing simulation systems that provide advanced features for modeling the selected simulation modeling paradigms in objective II were used. Simphony simulation system and Visual Studio were used to model the discrete event aspects of the system (company level aspects for the *company of interest*), while the AnyLogic simulation system was used to model the agent-based aspects (at an industry level – project arrival, and the competitive acquisition of projects). The discrete event component and the agent-based component were integrated in a distributed simulation environment using a frame work known as COnstruction SYnthetic Environment (COSYE).

IV. ***Achieving objective IV:*** The simulation application was verified and validated component by component. Verification was used to prove that the technology and software applications that were used were reliable. This was achieved through tracing of events that occurred in the system during execution. A test federate (a component within the COSYE framework), was used to verify the accuracy in implementation of the distributed simulation concepts. Validation was treated as a continuous process that started at the concept development phase and was carried through till the end of development. Concept models and designs were validated when discussing or presenting them to colleagues, professors in the research group and other technical support staff within formal or informal meetings/discussions. Sensitivity analysis was also performed to confirm the validity of the simulation model.

1.4 BENEFITS OF THE STUDY

In the course of implementing this study, a number of contributions were made. Some of these are academic contributions while others are industry related. The end product of this thesis, i.e., the developed application, was also a contribution.

1.4.1 Academic Contributions

- *System architecture and design patterns:* This thesis showcased efficient ways for developing large-scale simulation models using different simulation paradigms. It also demonstrated efficient ways to design and specify architecture that encapsulates different simulation modeling paradigms and software in a distributed simulation system. For example, the thesis demonstrated how a JAVA-based, agent-based simulation federate can be integrated with a DOTNET discrete event simulation federate using High Level Architecture (HLA).

- The verification and validation of the simulation application developed within this thesis highlighted the challenges likely to be encountered in the verification and validation of similar, large scale complex simulation applications and demonstrated how to overcome those challenges. Approaches used to overcome verification and validation challenges encountered the developed large scale model were documented and were useful. Lessons learned could be applied by researchers to develop similar large-scale simulation models for complex systems.

- *Modeling different construction phenomena:* A number of phenomena needed to be abstracted and represented in a computer-based simulation model. Some of these phenomena are not new and have been extensively researched in construction. However, no attempt has been made to simulate them in a realistic fashion. Examples of these phenomena include: the bid/no-bid decision, bid price generation in a competitive bidding process, and modeling safety incident occurrence and related issues. This thesis presented novel approaches (with the exception of the bid price generation algorithm) for modeling these phenomena. The bid/no-bid decision process was modelled using information on the internal attributes of the company, prevailing conditions of the company (workload) and the anticipated competition. The bid price computing algorithm was adopted from a book by Winston (2000) on modeling uncertainty using @Risk and extended to suit the needs of this thesis. A unique approach was adopted for modeling safety incidents, which was based on scheduling safety events based on inter-arrival times that were sampled from statistical distributions. This concept was adopted from approaches used to model equipment failures/breakdowns. Human beings were idealized to replace equipment and the same concept used to model safety incidents, which are synonymous to equipment breakdowns.

1.4.2 Contributions to the Industry

This thesis resulted in the creation of a simulation-based application for modeling company performance. The application can be used within the construction industry in a number of ways. The two main possible applications include:

- *Strategic planning tool:* The developed simulation application could be used as a tool for managing a company's performance. The simulation-based application provides an easy-to-use cheap test-bed that facilitates the experimentation with different bidding strategies to enhance company performance. The application could also be used as a work load planning tool at construction companies.

- *As an employee training tool:* Components of the application could be used in isolation for training company employees that work towards specific business operations that companies engage in. For example, the tendering module can be used as a standalone application in training estimators on how to generate competitive bids by either learning then adopting the algorithm for the bid decision and markup estimation into their day-to-day operations, or by learning the art of effective bidding through experimentation with the application. The entire application can also be used to train novices working for the company on how to manage performance related issues.

A number of contributions were made in the course of this thesis that are practical in nature and can be directly applied in the construction industry to improve their business and technical processes, consequently enhancing the competitiveness of the companies that apply them and that of their industry at large. Some of the academic contributions can also be made use of in industry to enhance company competitiveness.

1.5 THE ORGANIZATION OF THE THESIS

This thesis comprises of a total of nine chapters. The first chapter has been successfully presented with this section marking the end of this chapter. The next chapter is a literature review. It summarizes the state of the art in performance management within the construction industry, bidding strategies and algorithms, and simulation modeling technologies. The third chapter outlines the methodology in this thesis. It discusses the techniques used to identify the performance measures to track and represent in the simulation model, the simulation modeling paradigms used to develop the performance management application and the techniques used to check the reliability and accuracy of the model. The fourth chapter explains the distributed simulation federation. It highlights the details of the performance management federation and these include: federates that exist in the simulation federation, the communication protocols between them, and the object model (Federate Object Model—FOM) used in the federation and the simulation framework, the strategy used in implementing time, and systems used to develop the federation. The next two chapters, five and six, summarize the details of the design and development of each respective federate. The process adopted to model performance details is presented and discussed in Chapter five. Chapter seven explains how the developed simulation application was verified and validated. The last chapter highlights the main achievements of this study. These were discussed in-line with the objectives that were set out to be achieved at the commencement of the study.

CHAPTER TWO – LITERATURE REVIEW

2.0 INTRODUCTION TO CHAPTER TWO

This chapter presents information on literature related to this thesis that was reviewed. The majority of it provides information on topics that relate to the actual problem dealt with, i.e., competitiveness and performance management, and the methodology used in addressing these problems, i.e., simulation-based approaches.

The discussion on competitiveness is commenced by providing definitions for the term. This is then followed with different ideologies of competitiveness and a presentation of a model that summarizes the key ingredients of competitiveness. Performance management is then highlighted as a key component of competitiveness and is then discussed. The discussion on performance management covers a number of issues such as: popular definitions, its history and evolution, traditional (KPIs, Balanced Score Card, and European Quality Excellence Model) and other performance management techniques such as Data Envelopment Method (DEA), Structural Equation Modeling (SEM), and simulation-based performance management systems (VOICE, AROUSAL, etc.). A section is also dedicated to presenting research work done on performance management within the construction domain.

Literature on the methodology used in this thesis is covered through the use of a top down approach. A high-level discussion of different techniques for analyzing systems is presented, i.e., analytical and numeric approaches. The discussion is then narrowed down to simulation-based methods and further refined to computer simulation. This discussion on computer simulation is commenced with an introduction to the different computing technologies—distributed and monolithic. This is followed by a discussion of the different types of simulation, i.e., Monte Carlo Simulation, Discrete Event Simulation (DES), Continuous Simulation (Dynamic Systems—DS, and System Dynamics—SD), Agent-Based Modeling, and Distributed Simulation. In the discussion of the above, the different applications that have been developed and are in use are presented. Also, a brief discussion of the research activities making use of these approaches is presented with a focus on the construction domain.

2.1 COMPETITIVENESS

2.1.1 Competitiveness and the Competitiveness Model

This section of the chapter attempts to cover, at a high-level, the different things that affect or determine the competitiveness of an entity. Scholars investigating problems within or in close proximity to the subject of competitiveness acknowledge that it is a multifaceted abstract concept that is quite difficult to understand and quantify (Momaya, 1996). Consequently, there is no consensus on the definition of competitiveness or how it can best be measured (Flanagan, Shen, & Jewell, 2007; Liyin & Yam, 2006; Lu, 2006).

Despite the complicated nature of competitiveness, there have been attempts by researchers to present the phenomenon of competitiveness in an easy to understand form. For example, some work was done by the World Competitiveness Report (WCR) and Momaya that involved breaking down the concept of competitiveness into different criteria that can easily be understood and quantified (Momaya, 1998). This work resulted in the advent of a competitiveness model. Prior to discussing and presenting this model, an overview of the work that Momaya did for his PhD at the University of Toronto, is presented as a background to this model. The main contribution of Momaya's work was the introduction of a framework to evaluate the relative competitiveness at an industry level. Momaya studied the construction industry in 3 countries, namely Canada, the United States and Japan. Momaya (1998) pointed out that at the time, effective processes at the macro level had been neglected in the Canadian construction industry. He believed that radical improvements in the industry productivity would come from innovative actions at the industry level, rather than unrelated optimization or automation at the site level.

In the course of his work, Momaya (1998) followed the work done by the World Competitiveness Report (WCR) to formulate the concepts of competitiveness into three facets which facilitated the understanding of the competitiveness phenomenon. They include:

- Assets,
- Processes and
- Performance.

These were summarized in a schematic layout shown in Figure 2.0.

COMPETITIVE ASSETS	COMPETITIVE 'PROCESSES	COMPETITIVE PERFORMANCE
• FACTOR COSTS	STRATEGIC	* PRODUCTIVITY
• HUMAN RE SOURCES	^MANAGEMENT: NORMAL PLAN	* HUMAN RE SOURCES
• INDSTRY IN-FRASTRUCTURE	> IMPLEMENTATION	• QUALITY/ EFFECTIVENESS
		•COST
• TECHNOLOGY		
• DEMAND CON-DITIONS	HUMAN RESOURCE DEVELOPMENT	• FINANCIAL • NTERNATIONAL
	R & d	•TECHNOLOGICAL
• GOVERNMENT	SYNERGIES	

Figure 2.0: Facets and Factors related to Competitiveness (Momaya, 1998)

The competitive assets presented in this model include factors which are often considered key sources of competitiveness. These are dormant unless they are transformed by a competitive process (Momaya, 1998). Both Porter (1990) and Momaya (1998) discussed the processes that ensure long-term competitive performance. They referred to these as *strategic management practices*. Momaya (1998) defined these strategic management practices as harmonious interaction among key stakeholders in creating and upgrading the assets for sustainable performance. He further stated that concepts within strategic management include: firm strategies, firm structure and rivalry.

Momaya (1998) further argued that effective processes were more likely to improve future performance and competitiveness, unlike assets and performance, which are typically based on past statistical data.

This competitiveness model is believed to have removed a layer of complication with regard to understanding the concept of competitiveness. Next, another piece that exists in literature is presented, which adds to the understanding of the phenomenon of competitiveness. This is the hierarchical nature of competitiveness and the interrelatedness of these different levels.

2.1.1.1 Competitiveness Types and Perspectives

Rating the competitiveness of a business entity may vary depending on the appraiser's perspective (See Figure 2.1). Varying outcomes could arise from the assessment of the same business entity by different parties (customer, proprietor or competitor). Such variation could be attributed to the fact each of these parties utilize different criteria for performing their assessments.

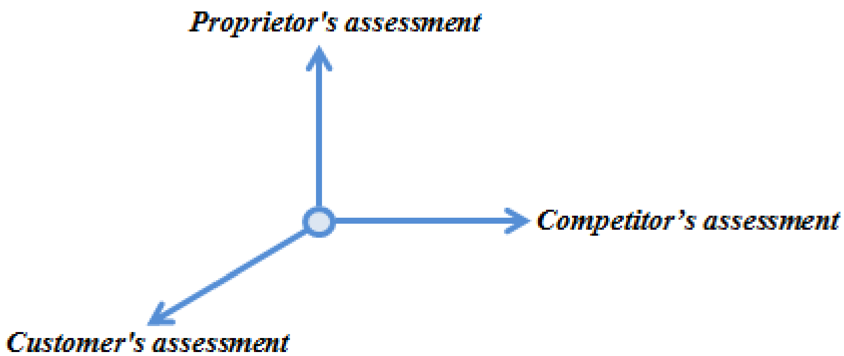


Figure 2.1: Different Perspectives of Competitiveness

It should be noted that appraising competitiveness in one dimension may be done to meet specific requirements but could result in a biased picture of the competitiveness of the enterprise evaluated.

Appraisals to establish the competitiveness of a business entity can be undertaken on different aspects. This gives rise to a term referred to as *"type of competitiveness."* One can opt to assess the competitiveness of a company by evaluating its operations and processes, financial health or the effectiveness of its policies, strategies and management style. Strategic efficiencies in this context refer to performance scores that can be directly attributed to policies adopted by the decision making unit. Operational efficiency on the other hand can be attributed to a number of things, such as the competency of the work methods, frontline workers, their supervisors and management, level of automation, systems and processes in place. Financial efficiency is a higher level aspect that refers to how well inputs are converted to outputs. It goes beyond that and covers aspects of financing (decision on the type and amount of credit to get), taxes, investments and asset management. Figures 2.2 summarize the different types of competitiveness that can be assessed.

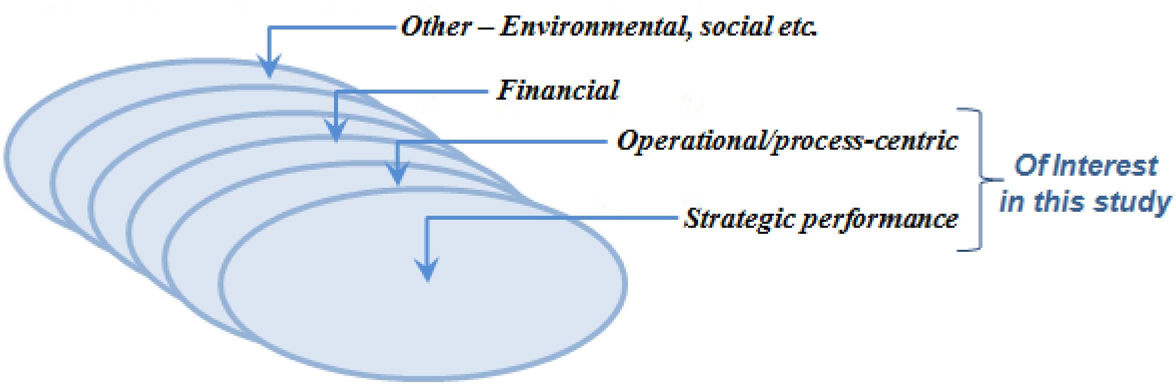


Figure 2.2: Types of Competitiveness

In this thesis, an application was developed that provides for the evaluation of the competitiveness of a company within the construction domain from a strategic (work acquisition, i.e., competitive bidding) perspective and an operations perspective.

2.1.1.2 Competitiveness – A Hierarchical Phenomenon

Competitiveness is a complex phenomenon that can be looked at from different perspectives and different levels (i.e., it is hierarchical) depending on who is doing the evaluation and the purpose for which it is done. In this section, the different hierarchies of competitiveness are presented and discussed. It is at this point that the various definitions for competitiveness will be presented. Competitiveness at any given level emanates from the competencies that exist at the lower levels. Competitiveness may be measured and addressed at three different levels, hence, giving rise to four different types of competitiveness.

- Within firm competitiveness (Level C^0)
- Firm competitiveness (Level C^1)
- Industry competitiveness (Level C^2)
- National competitiveness (Level C^3)

These different levels, along with some of the details that they entail, are summarized in Figure 2.3.

This multifaceted concept of competitiveness needs to be defined clearly at the appropriate level considering the views of important stakeholders. At the lowest level, competitiveness can be measured at the individual worker level, i.e., by assessing their competencies. Competitiveness at the most basic level, company level, is considered the most important by some, while the country level is considered most important by others. Next, a description of competitiveness at each of these levels is presented.

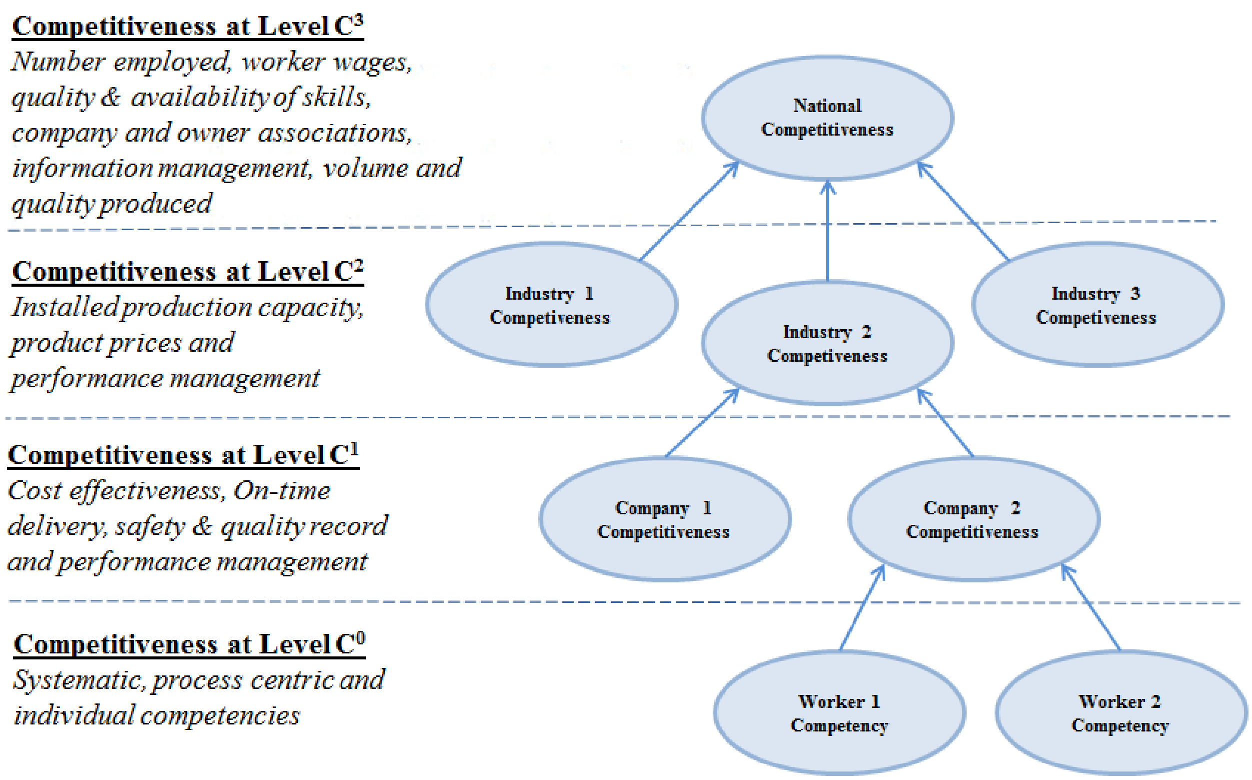


Figure 2.3: A Hierarchical Layout for the Concept of Competitiveness

2.1.1.2.1 Company Competitiveness

In 1988 Buckley, Pass and Prescott found that only a few definitions in the literature were tailored to describe competitiveness at a firm level. Of those that do, the Aldington Report (1985) provided the most complete picture by stating, “a firm is competitive if it can produce products and services of superior quality and lower costs than its domestic and international competitors. Competitiveness is synonymous with a firm’s long-term profit performance and its ability to compensate its employees and provide superior returns to its owners.”

In the same line of thinking, the Department of Trade and Industry (DTI, 1998) states that, “for a firm, competitiveness is the ability to produce the right goods and services, at the right price, at the right time. It means meeting customers’ needs more efficiently and more effectively than other firms.”

Company competitiveness is defined as “the ability to design, produce and/or market products superior to those offered by competitors, considering the price and non-price qualities” (WRC, The World Competitiveness Report 1991, 1991).

Blunck (2006) defines company competitiveness as the ability to provide products and services more effectively and efficiently than the relevant competitors in local and international markets without protection or subsides, leading to sustained success. He further stated that such success can only be realized if the company takes advantage of opportunities and efficiently utilizes its resources to produce a good or service (Blunck, 2006).

In this thesis, competitiveness was evaluated at a company level, but with a bias towards the construction industry. This was because it was believed that the concept of competitiveness is a hierarchical process which is influenced from the bottom up. As a result, this study focuses on developing tools for enhancing competitiveness at the lower level (competitiveness at a company level).

2.1.1.2.2 Industry Competitiveness

There are some scholars that believe in tracking competitiveness at an industry level because it gives a better picture of how well things are going. For example, in 2006, Blunck stated that competitiveness at the industry level is often a better indicator of the economic health of a nation than competitiveness at the firm level, because the success of a single firm might be due to company-specific factors that are difficult or impossible to reproduce (Blunck, 2006). Competitiveness at the sector level is often considered the result of the strategies and actions of firms that operate in that sector. Some formal definitions that exist in the literature on sector or industry competitiveness include:

Industry competeviness is the collective ability of firms in that sector to compete internationally (D'Cruz & Rugman, 1992).

Competitiveness at a sector level is the extent to which a business sector offers potential for growth and attractive return on investment.

Porter extensively studied and provided valuable insights into factors shaping the competitiveness of industries and nations. For details of this, see Porter (1986). These studies were later criticized by Momaya in the course of his PhD studies at the University of Toronto in Canada. As an example, Momaya pointed out several limitations in Porter's theories with regard to their use in evaluating industry level competitiveness on the international scene (Momaya, 1996).

Competitiveness at an industry level was studied in a research theme under which this thesis work was conducted. Two studies were carried out as part of a larger research theme (competitiveness in the construction industry) related to the Fourth term of the IRC industrial chair held by Dr. Simaan AbouRizk. These studies were done for the benefit of companies that are partners to the industrial chair, but also for the benefit of the Canadian construction industry at large. They included:

i. Carrying out an assessment of the installed fabrication production capacity of structural steel products (bridge work, plate work and stick members) for the Canadian construction industry. The second piece of this first study involved performing a price inquiry comparison for structural steel products fabricated in Canada, the U.S., China, Korea and selected countries in Europe.
ii. An investigation into the industrial module fabrication capacity for the province of Alberta in Canada.

The findings of these two studies confirmed the speculations about the competitiveness of an industry being largely dependent on its installed production capacity and its production efficiency, which translates into prices charged.

2.1.1.2.3 National Competitiveness

Although there is a school of thought in competitiveness which argues that international competitiveness has meaning only at the industry level (WRC, 1989), competitiveness at a national level is still presented and shown in the competitiveness hierarchy for the benefit of economists who believe in it.

National competitiveness is high-level (Level C^3) and is the aggregation of the competitiveness of different industries within the nation. This is usually the focus of development economists at a national, regional or global level. Industry competitiveness on the other hand is an aggregation of firm competitiveness. There are a number of definitions for national competitiveness in the literature. Some of these are presented below.

A pioneering definition of competitiveness on a national level was formulated by Scott and Lodge (1985). They were amongst the first scholars to formally define competitiveness at a national level. They defined it as "a country's ability to create, produce, distribute and/or service products in international trade while earning rising returns on its resources" (Scott & Lodge, 1985).

The US commission on Industrial Competitiveness defined competitiveness as "the ability of a country to produce goods and services that meet the test of international markets and simultaneously to maintain and expand the real income" (Tyson, 1992).

The OECD (1997) adopted this definition, and thereby developed the arguably most frequently cited one, but added the criteria that competitiveness is to be proved "under free trade and fair market conditions" and "over the long-term" (OECD, 1997).

2.1.1.2.4 Generic Definition of Competitiveness

From the previous sections, it can be appreciated that there are numerous definitions within literature for the term competitiveness. It is also worth noting that there are a number of definitions that don't clearly fall within the categories or levels described. A good example is that provided by the World Economic Forum. The Global Competitiveness Report of the World Economic Forum defines competitiveness as "the set of institutions, policies, and factors that determine the level of productivity of a country" (World Economic Forum, 2009).

There are a number of shared elements that can be identified as cross-cutting after a careful scrutiny of most definitions of competitiveness that exist in literature (Orozco, Serpell, Molenaar, & Forcael, 2014). Orozco et al. (2014) summarize the main elements in competitiveness, as follows:

- It is a concept more powerful than traditional economic indicators, such as profitability, productivity, or market share;
- It is associated with achieving objectives;

- It is relative to competitors; belongs to the eye of the beholder (it means different things for different people);
- It not only reflects past performance, but also allows the perception of potential;
- It must satisfy the needs of clients and personnel;
- It is related to superior quality;
- It implies continuous improvement; and
- It is associated with high productivity and innovation.

Orozco et al. (2014) added that competitiveness, in brief, is related to having better abilities and capabilities than competitors, and it also involves both the results achieved in the past and the perception of the future potential of a company.

2.1.3 Inadequate Competitiveness and Company Failure

An organization's performance is commonly assessed by different criteria such as profitability, growth in sales, competitiveness, etc. (Arditi, Koksal, & Kale, 2000). According to Arditi et al. (2000), Dun and Bradstreet (1989–93) is one of the most important sources of information on business failures (Dun and Bradstreet, 1989–93); however, its business failure reports do not differentiate between driving factors (i.e., organizational and environmental factors) and an organization's performance criteria (i.e., profitability, competitiveness, sales, growth, etc.) (Arditi, Koksal, & Kale, 2000). Their annual business failure reports are routinely used for exploring factors underlying business failures and, hence, for gaining insights into processes that lead to business failures. However, Arditi et al. (2000) argue that one should be cognizant of the fact that the organizational and environmental factors are the driving forces that directly impact the performance of an organization, which in turn leads to either success or failure.

Besides the above, there are other studies that have covered the topic relating to causes of company failure. Some of these include: Wong & Thomas (2010), Ghaffari & Jain (2013). The reader is advised to review them for further details. There also exists evidence that the competitive edge of a company can be greatly affected by the size of the company. That evidence seems to suggest that smaller companies are constrained by a lack of resources, while the competitiveness of the larger companies is affected by the mismanagement of the abundant resources that they possess.

2.1.4 Competitiveness Issues Addressed in this Study

In this study, competitiveness will be studied for the construction industry at a company level. This is the case because the construction sector is believed to make a sizeable contribution to the national economy. This argument was also supported by AbouRizk (2011), who stated that for the Canadian economy (and that of Alberta) to remain competitive globally, it is of the utmost importance to make the construction sector of the economy competitive both within Canada and in international markets. He added that this is because construction is a key source of jobs and contributes a significant portion to the gross domestic product (GDP) (AbouRizk, 2011). For example, in 2011, the construction industry accounted for 8.6% of Alberta's GDP and was responsible for employing 10.24% of its workforce (Alberta Finance and Enterprise, 2010; Government of Alberta, 2011).

Competitiveness of companies was studied by considering the tail-end of the competitiveness model proposed by the WRC and Momaya. The focus was in the benchmarks for the measures that are indicative of a company's performance and the management of those, i.e., measurement and possibilities of improvement. This is because performance measurement and management are considered a crucial part of competitiveness, especially at a company and industry level. Literature contains a number of performance management systems that have been used and continue to be used by companies within the construction industry. These are discussed in the following sections along with their merits and demerits.

2.2 PERFORMANCE MANAGEMENT – A KEY INGREDIENT FOR COMPETITIVENESS

The term performance management was formulated by Dr. Aubrey Daniels in the late 1970s. His intention at the time was to describe the process of managing behavior and results, two critical elements of what is known as performance (Aubrey, 2004). According to Grünberg (2004), performance measurement has two main aims: to connect company goals and objectives to improvements and to set targets for improvement activity (Grünberg , 2004).

The relevance of performance management to understanding or assessing competitiveness has been presented within the competitiveness model in previous sections. Performance management is used in most companies to accomplish organizational goals that enhance their competitiveness in the market place (Namho, Hyun-Soo, Moonseo, & Seungjun, 2007). Next, literature on performance management will be presented. This includes definitions for performance management, and the different types of performance management systems. Literature on performance management systems within the construction domain is also discussed.

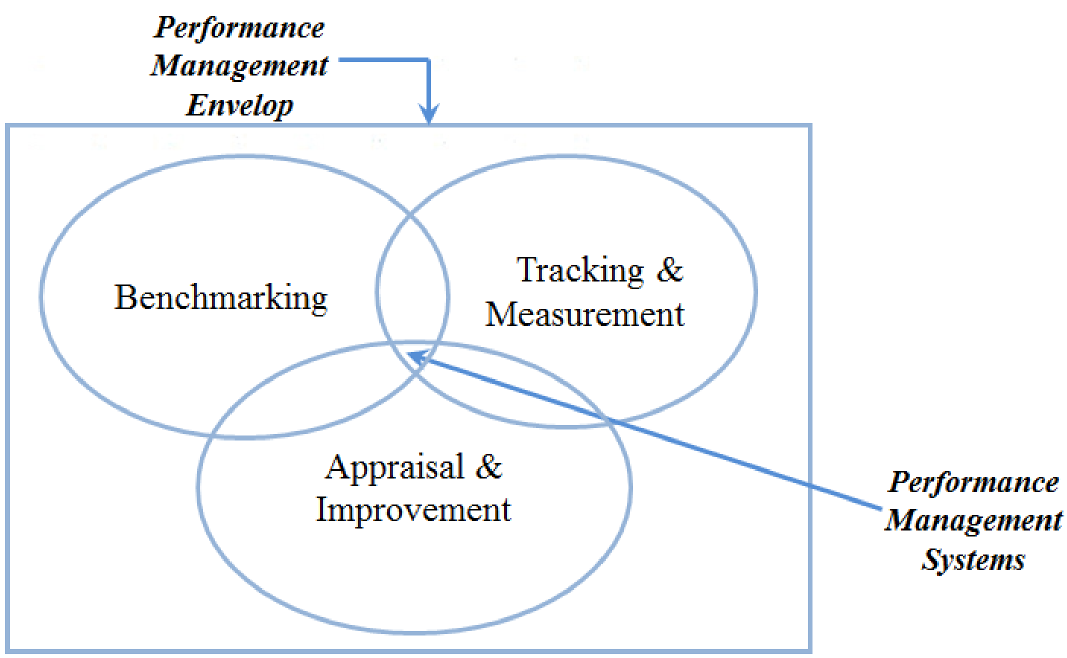


Figure 2.4: The Different Aspects of the Subject of Performance Management

Prior to getting into a detailed discussion of performance management, the aspects that frequently come up when dealing with this topic are presented. These were identified after an extensive review and synthesis of literature on this subject. It can be said that performance management involves three broad aspects that include:

- Benchmarking,
- Tracking and measurement of performance and
- Evaluation and continuous improvement.

All these are summarized in the diagram presented in Figure 2.4. Each of these aspects is a big subject on its own and will be discussed to varying degrees later in this chapter or subsequent chapters. It is worth-noting; however, that these aspects serve as the building blocks for sound performance management systems.

2.2.1 Performance Management – Definitions

A critical review of the literature revealed that a number of definitions for performance management have been put forward by diverse scholars. Most of these seems to bare the same theme with just a few variations amongst them. A few of those that stand out are summarized in this section.

Neely et al. (1995) refer to the performance measurement system as "… a set of metrics used to quantify both efficiency and effectiveness of actions" (Neely, Gregory, & Platts, 1995).

Another definition for performance management is provided by Aguinis (2011). He defines performance management as a continuous process of identifying, measuring and developing performance in organizations by linking each individual's performance and objectives to the organization's overall mission and goals (Aguinis, 2011).

According to the U.S. Office of Personnel Management, performance management is the systematic process by which an agency involves its employees, as individuals and members of a group, in improving organizational effectiveness in the accomplishment of agency mission and goals (U.S. Office of Personnel Management, 2008).

In 2008 Huprich defined performance management as a system designed to identify the ways to achieve organizational goals through constant assessment and feedback leading to improvement of employee performance (Huprich, 2008).

The above definitions of performance management share two main themes – performance management is a continuous process and it is linked to organizational objectives. Each of these is briefly discussed.

1. Performance management as a continuous process. Performance management is ongoing. It involves a never-ending process of setting goals and objectives, observing performance, and giving and receiving ongoing coaching and feedback.

2. Linkage of performance management to organizational mission and goals. Performance management requires that managers ensure that employees' activities and outputs are congruent with the organization's goals and, consequently, help the organization gain a competitive business advantage. Performance management therefore creates a direct link between employee performance and organizational goals, and makes the employees' contribution to the organization explicit.

Next, the genesis of performance management systems is discussed.

2.2.2 The History of Performance Management

It is argued by some authors that the origins of performance management can be traced back to the 1940's and 1950's. This is because it was the time when most work that was used as stepping stones in the formulation of performance management processes was started. Examples include the work on motivation theories done by Maslow (1943) and Herzberg (1959). There were 3 systems that were used at different times for purposes of tracking and monitoring performance. These included:

- Merit system,
- Management by objectives and
- Performance appraisal.

It can be argued that performance management evolved from these systems. However, it is distinguishable and unique in nature, although it incorporates elements of each of its predecessor systems.

According to the Organizational Heart Beats (2012), the merit system or rating system required managers to evaluate worker's technical and soft skills (Organizational Heart Beats, 2012). This system originated in the 1940's and 1950's. A rating scale of 1 to 5 was the normal ranging from 1 (poor) to 5 (outstanding). The criteria used in the evaluation of technical skills included:

- Knowledge of the job,
- Output and
- Accuracy (quality).

On the other hand, the criteria used in the evaluation of soft skills included:

- Confidence,
- Attitude and
- Judgement.

It has been stated in literature that the Merit System was used as a criteria for pay reviews. The system had a number of shortcomings. These included:

- As they were not tied to objectives they tended to generalize, were inherently subjective and attempted to quantify the patently unquantifiable judgements of personality.
- Although simple Merit Systems have been discredited and no hard evidence exists to demonstrate they actually improve performance, they still exist in practice today—in some cases masquerading as performance management systems.

The system that succeeded the merit system was referred to as "Management by Objectives." The system came into existence in the 1960's and 1970's and was given this name in 1955 by Peter Drucker and Douglas McGregor (Evans, 2004) (Murphy & Cleveland, 1995). The system used a hierarchical top-bottom approach to ensure that organizational/corporate objectives trickle down to the different organizational units and individuals. The Management by Objectives system was developed as a feedback system that reviewed unit and individual objectives against those of the organization. Organizational Heart Beats (2012) pointed out a number of pitfalls with this system. These included:

- It adhered to rules and methods of the system as opposed to a process of working.
- It concerned itself primarily with the managers of the organisation with the employees invariably subjected to the pre-existing merit system.
- It was a top down process which did not engage with employees and paid little attention to core values or their communication.

The last system to be used extensively prior to the advent of the performance management system was the "Performance Appraisal System." Performance Appraisal systems emerged in the late 1970's and were used throughout the 1980's. This system was a mix of the merit system and

Management by Objectives system. It was operated using a top-bottom approach. Organizations that made use of the system set up annual appraisal meetings as a means of facilitating the use of this system. This system was also referred to as pay-related–reward or performance-pay system. The reader is advised to read other literature on the history of performance management systems such as Neely (2002), to acquire more details on this.

The organizations that have chosen to use a performance management process have often done so because the annual evaluation process has failed to meet their appraisal needs (Huprich, 2008). The origins of performance management can be traced back to the 1940's. Primarily the process was developed by managers to justify whether the salary that was being paid to the individual was justified. Since then, performance management processes have become much more sophisticated and have evolved to encompass variations on the usual line manager-employee appraisal to encompass areas such as competencies, 360 degree feedback, and development planning.

2.3 PERFORMANCE MANAGEMENT SYSTEMS

This section presents a number of systems/approaches that are popularly used as performance management systems. Some of these have been in existence for decades and have become standard tools for performance management, while others have been used in isolation. Each of these will be discussed in the following sub-sections.

2.3.1 Elements of a Sound Performance Management System

The practice of performance management is treated as a continuous process that is comprised of a number of components. It is for this reason that this process is considered a success by employers, employees and their supervisors. According to Armstrong (2006), the 5 components of a performance management system include:

- Agreement,
- Measurement,
- Feedback,
- Positive reinforcement and
- Dialogue.

Agreement involves clearly identifying and communicating the organizational goals and those of the units and employees. This phase stipulates how employee and unit objectives relate to those of the organization. The purpose of this phase is to ensure that the employees get to know what is expected of them so that when an assessment of their performance is carried out, it is fair. Measurement and feedback are the most critical parts of a performance management system. The last performance management steps involve actions to improve registered performance.

There are various ways of measuring performance. Some of the techniques used in measurement are adopted from other performance appraisal techniques such as performance appraisal. It has been pointed out in literature that traditional performance measurement techniques have a number of shortcomings that make the entire performance management process ineffective. As a result, a lot of the work aimed at improving the performance management process has been focused on this component (i.e., performance measurement). It is this phase of the performance management process that this thesis strives to contribute towards through the development of a simulation-based approach for modeling performance aspects of a contractor construction company. A critical review of literature revealed that a number of simulation-based models have been proposed for tracking the performance of companies. A number of these are discussed in detail in this chapter along with the shortcomings of each. These shortcomings served as a justification for the development of the model in this thesis.

2.3.2 Traditional Performance Management Systems

2.3.2.1 Key Performance Indicators

Key performance indicators (KPIs), also known as key success indicators, are targets that add the most value to a business (MESA International & Cambashi Inc, 2012). KPIs are also defined as a set of quantifiable measures that a company or industry uses to gauge or compare performance in terms of meeting their strategic and operational goals (Public Record Office Victoria, 2010). KPIs vary between companies and industries, depending on their priorities or performance criteria (Investopedia, 2014).

In his paper on KPIs for human resources, Iveta (2012) and Hursman (2010) cited the main attributes that KPIs must possess. These included: a KPI should be specific, measurable, attainable, relevant, and time bound.

The process of setting up and implementing a performance management system that utilizes KPIs in its evaluations requires three key elements, namely; an agreement of project Key Performance Indicator targets, a post evaluation of Key Performance Indicators against agreed targets, and a quarterly Key Performance Indicator 360 degree evaluation.

KPIs have traditionally been summarized into the following sub-categories:

- *Quantitative indicators* that can be presented with a number.
- *Qualitative indicators* that cannot be presented as a number.
- *Leading indicators* that can predict the outcome of a process.
- *Lagging indicators* that present success or failure *post hoc.*
- *Input indicators* that measure the amount of resources consumed during the generation of the outcome.
- *Process indicators* that represent the efficiency or the productivity of the process.
- *Output indicators* that reflect the outcome or results of the process activities.
- *Practical indicators* that interface with existing company processes.
- *Directional indicators* specifying whether or not an organization is improving.
- *Actionable indicators* are under an organization's control to effect change.
- *Financial indicators* used in performance measurement and when looking at an operating index.

Performance using KPIs may sometimes be measured in terms of making progress toward strategic goals, but often success is simply the repeated, periodic achievement of some level of operational goal (e.g., zero defects, 10/10 customer satisfaction, etc.) with respect to a specific KPI.

2.3.2.2 Construction Industry Institute Benchmarking & Metrics

The Construction Industry Institute (CII) was established in 1983 on the recommendation of The Business Roundtable Construction Industry Cost Effectiveness (CICE) Project to address issues on construction research and fragmentation of the industry (Mulva, 2011; Construction Industry Institute, 2003). The CII represented the first owner-contractor-academic research collaboration for the construction industry. CII's head offices are at the University of Texas at Austin. According to Mulva (2011), the CII started off with 29 member companies and had grown to 110 members in 2011. In 1995, CII started a benchmarking and metrics program (BM&M), which has up to 2,039 projects in its database that are worth a total installed cost of $133 Billion. These projects are from the various domains of construction such as building, infrastructure, light and heavy industrial facility construction. The BM&M program has developed numerous performance, practice use, and productivity metrics since its inception in 1995.

2.3.2.3 Balanced Score Card

According to Bassioni et al. (2004), it is believed that the balanced scorecard (BSC) is one of the most influential business ideas to have been proposed, and is being used by many companies (Bassioni, Price, & Hassan, 2004). Many argue that the advent of Robert S. Kaplan and David P. Norton's concept of the balanced scorecard in 1992 revolutionized conventional thinking about performance metrics (Norton & Kaplan, 1992). BSC is believed to have superseded other traditional measures of financial performance by introducing additional perspectives from which a company's performance can be assessed. These nonfinancial metrics are so valuable mainly because they predict future financial performance rather than simply report what's already happened.

Advocates for the balanced score card claim that the introduction of the method resulted in an advancement of the state of the art in performance management on two fronts (Kaplan & Norton, 2007). These include:

- It provided a means of assessing company performance from a more holistic perspective compared to previously utilized methods. Most performance management techniques focused on measuring financial performance and used that to project the performance of the company being appraised. BSC introduced other perspectives that could be used alongside the financial measures, resulting in a more realistic assessment of performance.

- The advent of the BSC also introduced an easy-to-use technique that translates an organization's long-term strategy (mission and vision) into short-term goals. This was not the original goal of the BSC, but came up as companies strived to make use of the technique in strategic management.

BSC provides a means for tracking and evaluation of performance from a number of perspectives namely; a financial perspective, customer perspective, internal business perspective and innovation and learning. BSC has an important principle—the cause effect between perspectives. This technique is one of the ways in which performance at a company can be managed.

2.3.2.4 Excellence Model

The European Foundation for Quality Management (EFQM) was founded in 1989 by CEOs and Presidents of 67 European companies who set up a team of experts from industry and academia to develop a holistic framework that could be applied to managing performance at any organization. The EFQM Foundation was formed to recognise and promote sustainable success and to provide guidance to those seeking to achieve it. This was realised through a set of three integrated components which comprise the EFQM Excellence Model:

- The Fundamental Concepts of Excellence,
- The Model Criteria and
- The RADAR Logic.

The fundamental concepts of the excellence model outline the steps for achieving sustainable excellence in any organization (EFQM, 2012). According to EFQM (2012), the concepts of excellence represent the characteristics of organizations that are considered top performers.

Model criteria summarize a mapping between *"enablers"* and *"results"*. The EFQM (2012) argues that for an organization to achieve success, it requires a number of tangible and intangible ingredients, which are referred to as enablers. These include good leadership, a clear strategic direction, competent people, good partnerships and efficient processes. *"Results"* represent the end products of utilizing the enablers. These could be used for setting organizational targets and assessing how the organization is doing.

The RADAR represents a flexible framework and management tool that provides a structured approach to assessing the performance of an organization (EFQM, 2012). According to the EFQM (2012), the RADAR philosophy stands for the following:

- Determine the **Results** it is aiming to achieve as part of its strategy.
- Plan and develop an integrated set of sound **Approaches** to deliver the required results both now and in the future.
- **Deploy** the approaches in a systematic way to ensure implementation.
- **Assess and Refine** the deployed approaches based on monitoring and analysis of the results achieved and on-going learning activities.

All these components put together allow people to understand the cause and effect relationships between what their organization does and the results it achieves.

2.3.3 Non-Conventional Approaches to Performance Management

2.3.3.1 Data Envelopment Analysis—A Performance Management System

Data Envelopment Analysis (DEA) is a non-parametric technique that was introduced by Charnes, Cooper and Rhodes in 1978. At the basic level, the DEA model is comprised of a non-linear programming formulation which can be translated into a linear programming problem to ease its evaluation.

According to Mostafa (2009), the transformation from non-linear to linear formulation is accomplished by constraining the denominator of the efficiency formulation in the equations presented, to 1.0, hence, making it a constraint. The solution to the linear formulation usually results in one firm (DMU) emerging with an efficiency of 1.0. This means that there is no other DMU that is more efficient than it and confirms that this DMU is located on the optimal frontier (Mostafa, 2009). The efficiency of all other firms will be compared to this firm using the optimal set of weights generated by the solution.

There are variants to this basic DEA model. A good example is the CCR model, which assumes a constant return to scale for both inputs and outputs (Banker, Charnes, & Cooper, 1984). BCC is another version of the DEA model that instead uses a variable return to scale (Banker, Charnes, & Cooper, 1984). Details of these versions of the DEA model can be found in Banker, Charnes and Cooper (1984).

Data Envelopment Analysis has been used for a number of performance measurement purposes. These can be categorized into two, at a high level.

- *Comparison of the performance of decision making units:* This would require a sample containing a number of DMUs each with data on the criteria that they are to be compared to.
- *Prediction of the performance of decision making units:* Developing a DEA that can do this would require a DMU that has both input and output data that is representative of the domain. Once trained, the DEA can then be used to make predictions based on new inputs.

DEA has been applied in different domains for evaluation of the performance or efficiency of DMUs, as enumerated in the first of the two bulleted points. For example, it was used in 1996 by Miller and Noulas to assess the efficiency of large banks in the U.S (Miller & Noulas, 1996). It has also been used in the domain of Agriculture to evaluate the production efficiencies of farms. Other areas that DEA evaluations have been performed include: internet companies (Serrano-Cinca, Fuertes-Callen, & Mar-Molinero, 2005), football teams (Haas, Kocher, & Sutter, 2004), retail stores (Barros & Alves, 2003), insurance companies (Cummins & Rubio-Misas, 2006), seaports (Cullinane, Wang, Song, & Ji, 2006), airports (Sarkis, 200), hotels (Sigala, Jones, Lockwood, & Airey, 2005), universities (Flegg, Allen, Field, & Thurlow, 2004), and advertising agencies (Luo & Donthu, 2005).

DEA has also been applied in research within the construction industry. For example, in 2003, McCabe (2003) published a journal paper that described a DEA model that could be used for contractor financial evaluation within the construction industry. The following financial metrics were used in the evaluation of the performance of these contractors: current ratio, accounts receivable and payable times, debt to equity, fixed assets to equity, gross profits to sales, administrative expenses to net worth, net income to sales, and net income to equity.

A few years later, McCabe (2005) used DEA to assign contractors relative efficiency score in a pre-qualification exercise. She first determined a "practical frontier" of best contractors, against which other contractors were compared (McCabe, Tran, & Ramani, 2005).

There have been attempts to combine Neural Networks (NN) with DEA in the estimation of the efficiency of DMUs (Wang, 2003). An example of this can be found in Athanassopoulos & Curram (1996). The authors used NN and DEA to predict and classify the efficiencies of a set of branches for a specific bank.

From the above, it is evident that the traditional DEA method and hybrid DEA, i.e., DEA-NN have been successfully applied in numerous domains for performance evaluation. These show that they have the potential for being extended to the construction domain for the evaluation of company performance. In the case of traditional DEA, to achieve this, one would need to create several DEAs that would each generate a unique performance measure. These performance measures would then serve as inputs to another DEA that would generate the overall efficiency of the company. However, in order to accomplish this, one would need data on the input variables that affect each performance measure, along with values for the performance measure that correspond to each set of inputs. A similar data set would be required that maps performance measure values to overall company efficiency values. Such an approach would heavily rely on the availability of a sizeable dataset that is of good quality, something that is a huge challenge within the construction domain. It is for this reason that data driven approaches such as the DEA, were not adopted in this thesis for modeling the competitiveness of construction contractor companies. The same limitations apply to DEA-NN techniques because they are also data driven approaches.

Another limitation with the DEA is that when the number of variables being analyzed increases, as would be the case in any comprehensive endeavors to assess company competitiveness, its ability to discriminate between DMUs decreases (Mostafa, 2009). Moreover, there are higher chances of inefficient DMUs dominating as efficient DMUs (Smith, 1997). Raab and Lichty (2002) proposed a diagnostic check that can be performed in the course of any DEA analysis to ensure that this problem is not experienced. These scholars proposed a general condition that needs to be fulfilled – the number of DMUs must be greater than three times the sum of the number of inputs and outputs (Raab & Lichty, 2002).

In the presence of good data, DEA and its variants can be effectively used for the entire process or a part of the process of evaluating company competitiveness. For example, it has been stated that DEA can be effectively used as a tool for benchmarking since it supports the identification of a group of efficient companies from a set comprised of efficient and non-efficient ones. This group of efficient companies would then be used in the definition of the operational goals (benchmarks) for the entire population of companies.

2.3.3.2 Structural Equation Modeling

Structural Equation Modeling (SEM) is a technique used for research that is predominantly social and qualitative in nature. The approach is comprised of a number of constructs (factors and indices) enclosed in circles/ellipses and arrows that represent the relationship between constructs. SEM makes use of sophisticated multivariate statistical tools and methods like the Partial Least Squares (PLS). It is because of these features that the technique (i.e., SEM) has been used in the analysis of highly complex, ill-defined qualitative problems or systems. In 2014, Orozco et al. (2014) applied the SEM approach to study competitiveness of construction companies. The authors considered a total of forty one constructs (factors and indices) in their study, which they zeroed down on after a critical review of related literature and subsequent approval of these by Managers at different construction organizations within Chile. The approval of these factors and indices was accomplished through an extensive questionnaire survey. In the course of this study, the researchers gathered information about the perceptions of top managers regarding how these factors and indices relate with each other and with the performance of their companies. These perceptions were used to filter out the factors that are relevant to competitiveness and also provided data that was used to the weighting (strength) of the relationship between factors and indices. The performance measures that they considered included: financial indices, market share, bidding effectiveness, client satisfaction, productivity, cost, quality, time, health and safety. The broad factors that the authors studied include: strategic management policies, project management practices, human resource management, technology and innovative approaches, financial capacity, institutional and business relations, bidding factors, external factors.

This SEM approach is powerful for analyzing performance and competitiveness because of its ability to represent and model the relations between many factors. However, the technique falls short with regard to capturing the dynamic and stochastic aspects of these factors and the metric(s) that they aim at predicting. The technique can be useful in gathering, understanding, and representing knowledge about huge complex systems that can be described by numerous variables. This can be vital at the front-end of any System Dynamics, and Agent-Based Modeling studies.

2.3.3.3 Simulation-Based Models/Applications for Performance Management

Literature shows that there have been a number of researchers that have used simulation-based methods for dealing with the performance management problem, especially within the construction domain. Simulation is a method that has immense potential for solving lots of problems that have characteristics similar to those observed in the area of performance management. Use of simulation as an approach for analyzing systems will be discussed later. The following sub-sections present some of the simulation-based performance management systems that exist within literature.

2.3.3.3.1 System Dynamics Model for Credit-Worthiness Evaluation

System dynamics has been used to evaluate the performance of companies. For example, in 2005, Moscardini et al. developed a system dynamics application that could be used to evaluate the credit worthiness of retail companies. The idea was to have a tool that generates output (credit worthiness) which could be compared to output (credit worthiness) based only on financial ratios. The researchers developed an application that was comprised of a User Interface at the front end and a system dynamics model behind the scenes (Moscardini, Loutfi, & Al-Qirem, 2005). The application was developed using *Powersim Studio*. The type of business modeled by the application is one that involves the purchase and sale of finished products. Production processes are not modelled at all within the application. The application also tracked and output only financial performance metrics, which are not sufficient for assessing the overall performance of companies especially in the construction domain. This limited scope of the application prevents its extension for use in the performance evaluation of construction companies.

However, the work of these researchers demonstrated how dynamic and inter-related business processes at a company are. Their work also showed that simulation-based approaches are the most appropriate technique to use to represent these processes because they generate more superior results compared to other techniques. Simulation-based methods also produce tools that can be used in experimentation work that would lead to reasonable decisions made.

2.3.3.3.2 System Dynamics Model for Enhancing Construction Company Performance

Based on the suggestions of Ofori (1993b) and Ogunlana et al. (1996), Ogunlana et al. (2003) created a systems dynamics model that could be used to explore possibilities (policies) of enhancing the performance of a construction organization (Ogunlana, Li, & Sukhera, 2003). Ofori and Ogunlana had suggested improving the construction industry using four broad approaches: resource development, enterprise development, documentation & procedures development, and the implementation of appropriate policies (Ofori, 1993; Ogunlana, Promkuntong, & Jearkjirm, 1996). The authors of this model based their work on the operations of a US based construction company that was doing business within Pakistan since the 90's.

The developed system dynamics model was developed in *iThink*©, a well-known system dynamics simulation software. A number of attributes and relationships that exist in the business operations of companies were modeled explicitly. A good example is the complexity of projects. The model explicitly represented the complexity of projects that the company was performing. This attribute was expressed in such a way that it could be affected by the project scope. Project complexity is increased by a large scope, which results from a larger economy along with changes in scope during execution. Project complexity affects the rework rate at the company, which in turn affects the company's productivity. The model also represented, as a relationship, replenishment of the available project execution capacity through the completion of projects that were in process. Completed projects also increase the size of the economy resulting in the creation of other new projects.

The paper indicates that there are two performance measures that this model of a company generates, namely: budget slippage and schedule slippage. The scores on these measures are dependent on the company's productivity, which in turn is dependent on workforce availability and motivation, the amount of rework, and the company's on-going performance. These factors are also dependent on the number of new projects awarded to the company and the available capacity to execute these projects. These factors were tied to the different policies that the modeller could experiment with, i.e., Joint Venturing (JV), the presence and effectiveness of a Management Information System and the development of a Construction Industry Development Board (Institutions). Ogunlana et al. (2003) used their model to experiment with each of these policies individually. They also experimented with a combination of these policies. In their paper, the authors presented only the model boundary and casual-loop diagram. They did not present the actual developed system dynamics model.

Although Ogunlana et al. (2003) found the model appropriate for their purposes; it has a number of limitations, most of which emanate from the simulation modeling paradigm that was adopted, i.e., system dynamics. The abstraction of the business operations of the construction company were done at a very high level ignoring a lot of detail which would be necessary if an analyst were to thoroughly investigate the performance of a construction company with plans of improving its competitiveness. For example, the number of performance measures tracked was limited. Company resources were not modelled explicitly; hence, the effect of their quality was not captured. The model only quantified the effect that the resource number had, not the production process, but not to a granular level.

2.3.3.3.3 VOICE—Virtual Organization Imitation for Construction Enterprises

In 2012, a study was carried out by Du and El-Gafy in which they used an agent-based approach to model the complex interactions that take place within organizations operating in a construction environment (Du & El-Gafy, 2012). This simulation system was given an acronym, "VOICE", which stands for Virtual Organization Imitation for Construction Enterprises. This is one of the few, if not the only existing study in literature that used an agent-based approach for modeling company performance. The focus of the researchers in this study was to investigate the influence that organizational and human factors have on construction performance. Projects and organization employees (workers including administrative staff) were modelled as agents that interact together to emulate the business processes that take place at construction organizations. Projects represented the work at the construction company and were modelled at a task level. The work execution process (quality, efficiency, and time) and the interaction between humans (effects of emergency meetings) were also explicitly modelled. This means that the dynamics of projects, task complexity and employee competencies were explicitly represented and modelled along with their influence on the efficiency of work execution. The application also modelled the influence that workload has on the effectiveness of the company's ability to deliver projects on time and within budget. Other performance metrics that were evaluated by the model include: safety, quality, turnover rate, communication efficiency, work efficiency and effectiveness, work-related growth and qualification growth.

This model had a number of pitfalls. These include:

- *Modeling of constructs that follow the level of abstraction chosen is not sustainable*: The authors chose to abstract projects at a task level. This means that a user of the model would have to possess information on task duration, complexity, resources required, predecessors, etc.
- *Dynamics of work acquisition:* The authors mentioned that they created an array of projects with associated details which they embedded into their system based on surveys they carried out. These would represent the workload of the organization. It can be concluded that these endeavors constituted an attempt to construct a library of different project types typically executed within the various domains in construction and their related details, something that is very challenging to do given the diversity and uncertainty that are characteristic of projects, equipment, and methods used within the industry. Such a list would be very restrictive. Moreover, in reality, companies have to compete for their work and there are no guarantees that they will be awarded all projects that they bid for. This uncertainty in this aspect of the operation of a construction company is not represented in VOICE. Also, the dynamics of project entry into the market are not well represented.
- *Processing of work:* The execution of projects (work) at the organization within the model is not constrained by key resources available at the company, something that is not realistic.
- *Modeling approach for the concepts on quality:* The manner in which quality was modeled is inconsistent with general practice. The authors accumulate mistakes made by different actors and then the manager makes a decision as to whether rework is necessary or not based on some prescribed threshold. This is not realistic given that errors made by individual actors

are looked at in isolation and fixed when they occur. Also, the factors affecting the creation of mistakes by actors usually have a dynamic influence, something not represented in the VOICE ABM system.

- *Modeling approach for safety:* Safety incidents are not modelled at a granular level.
- *Representation of the output (generated KPIs):* The authors of this application did not clearly state in their paper the format in which performance measures are quantified and reported. The aggregation of these measures to generate a single index also seems not to have been covered in their developments.

In a nutshell, many concepts were considered, as expected of ABM approaches, but with interactions that are not dynamic to a comprehensive manner. Also, the detailed approach makes the model less flexible for deployment in different domains.

2.3.3.3.4 AROUSAL—A Real Organization Unit Simulated As Life

AROUSAL was developed as a management training simulation-based system by Peter Lansley in 1984 when he was the Assistant Director of Research at Ashridge Management College in the UK.

Lansley's basic premise was that this could only be achieved if the training is imbued with a high degree of realism. With this in mind he conceived a training course that simulates the operations of a real company, christening it AROUSAL (A Real Organization Unit Simulated As Life). AROUSAL is in effect a highly elaborate business game based on the detailed case study of an actual building company. The package is computerized so that trainees can test the results of their decision-making (Lansley, 1985).

Project arrivals into the market and eventually to the company, within AROUSAL, were modelled using approximate methods that heavily relied on the past experiences of the company. The competitive bidding process for projects amongst companies was not explicitly modelled in AROUSAL. Lansley used probabilities to represent the uncertainty associated with this process. Projects are set up in such a way that they are comprised of a number of tasks which require a specific skillset.

People were explicitly modelled within AROUSAL, with the performance of individuals tracked and their skills translated into project efficiency and progress (Lansley, 1986). The technical and managerial skills and the capacity of individual workers at a company are characterized based on their education, training, experiences, age, personalities, their aptitude and attitudes.

In a game session, participants are assigned to roles such as managing director, contracts manager, chief accountant and marketing manager. In the course of the game, information is passed onto each of these roles, which comprises routine production and progress reports, balance sheets and profit-and-loss statements, staff reports giving individual responsibilities and performance, and invitations to tender and general information about the building market. Each role is then expected to use that information to make decisions such as price tenders, select, hire and fire staff, review salaries, assign staff to projects, redefine job roles and reorganize company structure.

AROUSAL provides for a decision session after a period equivalent to three months has been simulated. The application also allows for a sequence of eight decision-making sessions, representing a total of two years in the life of the model company. AROUSAL demonstrated a novel approach for modeling the performance of an organization using a bottom-up approach. A very good job was done in the development of this application with representing the individual and their influence on the performance of an organization. However, it is not clear from reading the journal papers published on this application by Lansley, which simulation modeling paradigm was used in the development of AROUSAL.

2.3.3.3.5 Studies with Other Techniques – DEA and PNN

In 2009, Mostafa did some work on the evaluation of the performance of companies using parametric and non-parametric methods. He used a hybrid Data Envelopment Analysis-Probabilistic Neural Network (DEA-PNN) approach for developing a tool that could be used in the prediction and classification of the efficiency of a company listed on the Cairo and Alexandria stock market in Egypt. Mostafa (2009) used data on company assets and employees as his inputs and revenues, profits, market capitalization and share price as his outputs when developing his model. The DEA was used to compute the efficiency scores for the companies. The PNN were then used to classify the companies based on their performance scores.

Commercial packages known as Frontier Analyst Professional and Neural Network Tools Professional package were used for the DEA and PNN analysis in this study (Mostafa, 2009).

The challenge with this approach is that it heavily relied on data which is difficult to come across in industry. Second, the analysis was performed at a very high level and focused on one dimension of a company's performance, i.e., financial performance.

2.4 COMPETENCIES, PERFORMNACE AND COMPETITIVENESS

Now that the basics of the different performance management concepts have bene discussed, it is important to clarify some terminology that is often mistaken to mean the same thing and yet it does not. Details of these are discussed in the following paragraphs.

Although related, competency, performance and competitiveness represent different concepts in the domain of performance management. However, it is not uncommon for practitioners in the industry and researchers to tend to use the three metrics interchangeably. In order to aid the distinction between these terms, Figure 2.5 was developed.

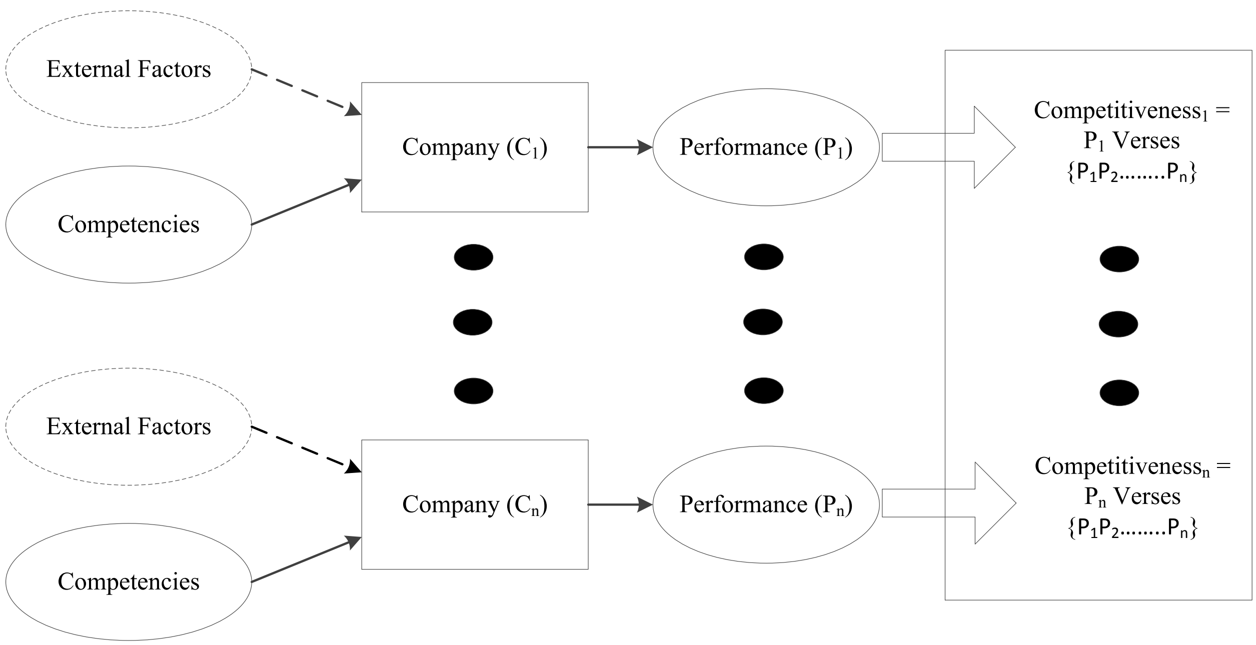


Figure 2.5: Schematic Layout Showing the Relation between Competency, Performance and Competitiveness

It can be seen from Figure 2.5 that competency is a metric that maps onto the input side of a company's business operations. Performance and competitiveness on the other hand map onto the outputs. Details of each of these metrics and their relation are discussed next.

Competencies represent "tangible" qualities or attributes that reflect the potential effectiveness with which a company executes its business operations. Competencies typically emanate from the people employed at the company and work processes or methods utilized there. It is important to note that besides competencies, there are other factors such as market conditions that also affect how well a company performs and its competitiveness.

Performance is a metric that is used to represent an output of a company with respect to its effectiveness. This output indicates how well a company is fairing i.e., how effectively a company translates its inputs into outputs. It is also worth-noting that Performance is local to a specific company.

On the other hand, competitiveness is a metric that gives an indication of how well a company performs relative to other companies. These companies might represent the rest of the industry or just a sub-set of the industry.

2.5 LITERATURE ON PAST PERFORMANCE MEASUREMENT AND MANAGEMENT STUDIES IN CONSTRUCTION

The subject of performance management has been extensively studied in numerous domains. Niven (2002) supported this argument when they stated the same about performance measurement methods, a crucial component of performance management (Niven, 2002). To further support this argument, Neely (1999) reported that between 1994 and 1996, approximately 3,615 articles regarding performance measurement were published. He added that in 1996 a new book on the topic of performance management was published every two weeks within the USA (Neely, 1999).

Despite the activity in line with performance management in most industries and strong emphasis that many researchers have placed on the importance of adopting the performance measurement methods to improve the current state of the construction industry (Egan, 1998; Latham, 1994), the construction industry continues to underperform (Lee, Cooper, & Aouad, 2000; Kagioglou, Cooper, & Aouad, 2001; Smith, 2002). All these issues aside, there have been a number of studies carried out on the subject of performance measurement and management within the construction domain. This section summarizes some of those studies. This is by no means an exhaustive list of studies done in construction on this subject, but they provide insights into how far the state-of-the-art has been advanced on this front within the construction domain.

It is necessary to appreciate the fact that within the construction industry performance can be and has been studied at different levels (Yang, Yeung, Chan, Chiang, & Chan, 2010). Examples include: at the project level (Lin & Shen, 2007; Cooke, 2001; Jaselskis & Ashley, 1991), and at an organizational level (Tan, Shen, Yam, & Lo, 2007). There has been little done at the industry level, though.

Kagioglou et al. (2001) proposed a performance measurement framework for the construction domain and demonstrated how to make use of it to get the desired results. These authors set out to develop a framework for improving the performance of organizations. They argued that it was by understanding how the organization arrives at a particular performance that a company might start to improve and increase its market share (Kagioglou, Cooper, & Aouad, 2001).

Cox et al. (2003) carried out a survey that determined the correct performance measures for the construction industry. They proposed base measures to which others can be added to get a true indication of how well a company is performing (Cox, Issa, & Ahrens, 2003). It was findings from this study that were used as a basis for determining the performance measures abstracted, modelled and tracked in the performance management application produced in this thesis.

It has been stated in literature that a proper assessment of a construction contractor's performance can be helpful to both the client and the contractors themselves (Shen, Lu, Shen, & Li, 2003). Shen et al. (2003) pointed out that this information on contractor performance can be used by the client to select the appropriate contractor that will deliver their project without problems. The authors further stated that the contractor could also use the information of their performance to know their strengths and weaknesses and possibly device improvement strategies. It was with this background that Shen et al. (2003) developed a computer based tool for determining the competitiveness score of a contractor from a set of performance measures.

Most of their work went into accurately aggregating the performance measures into a since competitiveness score. Their Windows-based application was referred to as Contractor's Competitiveness Assessment Scoring System (C-CASS).

Pilateris and McCabe (2003) evaluated financial measures using DEA to rank contractors from the building and heavy civil industries, and other specialities from different parts of Canada, by performance. In 2005, McCabe et al. demonstrated how DEA can be used for contractor pre-qualification considering multiple criteria.

A year prior to McCabe's DEA contractor prequalification study, Bassioni et al. (2004) reviewed the different performance measurement frameworks that are in use in the UK, identified gaps within each and proposed areas of improvement/future research (Bassioni, Price, & Hassan, Performance Measurent in Construction, 2004)

Mostafa (2007) applied a two-stage approach, production frontier analysis (PFA) to measure the relative market efficiency of 62 listed companies in Egypt, and Tobit regression to examine the dependence of efficiency on the specific operating environment of these companies. In that same year, Tan conducted another study in Hong Kong with a number of his colleagues, in which they carried out a survey to determine key competitiveness indicators for companies (Tan Y., Shen, Yam, & Lo, 2007). The authors came up with a total of thirty six indicators for tracking the competitiveness of organizations.

Other techniques such as fuzzy logic have also been used by researchers such as Marsh and Fayek (2009) and Awad and Fayek (2010), to evaluate both qualitative and quantitative factors in order to rate the performance of contractors for bonding and surety purposes (Marsh & Fayek, 2009), (Awad & Fayek, 2010).

In 2011, Tsolas (2011) assessed the performance of listed Greek construction firms in terms of efficiency and effectiveness by integrating DEA and ratio analysis. The author analyzed nineteen companies to obtain their performance with respect to profitability and efficiency in market value generation (Tsola, 2011).

Tan et al. (2012) studied the relationship between competition environment and performance in Hong Kong. They found that an increase in the external competition (amongst their peers) in the industrial construction environment lead to poor company performance. However, an increase in internal competition, for example, amongst suppliers, led to better company performance (Tan, Shen, & Langston, 2012). In the same year, Cristobal (2012) conducted a study in which he used both TOPSIS and VIKOR in the selection of a suitable contractor for a road project based on different criteria such as experience in similar jobs, financial status, safety and management capability (Cristobal, 2012).

Other studies have been done using the simulation methodology. For example, a systems dynamics model was proposed by Elliott et al. (1994) to enhance the performance of a company (Elliot & Moscardini, 1994). In 2003, another systems dynamics model was used by Ogunlana et al. (2003) to explore options for making a company more competitive. The authors tested the model on an oil and gas company working in Pakistan and got results that they were expecting (Ogunlana, Li, & Sukhera, 2003). Al-Qirem and his colleague also did a study in which they modeled a small firm in Jordan using systems dynamics in order to check for its credit worthiness (Al-Qirem & Yaseen, 2010).

2.6 DESIGN AND ANALYSIS OF REAL WORLD SYSTEMS

A comprehensive study of any system requires that the analyst experiments with the actual system or a replica of that system. The object experimented with is usually referred to as a model. The process of constructing a replica of a real or hypothetical system i.e. the model is referred to as modeling.

It was important to discuss the theoretical details on the different options available for abstracting real world systems and how these are represented on computer. This discussion set the stage for the different tools and modeling techniques that were utilized in development work that was done. This discussion was commenced with the presentation of a Figure (See Figure 2.6) that summarizes the various options available for studying systems. This was followed by a detailed discussion of the different options detailed in Figure 2.6.

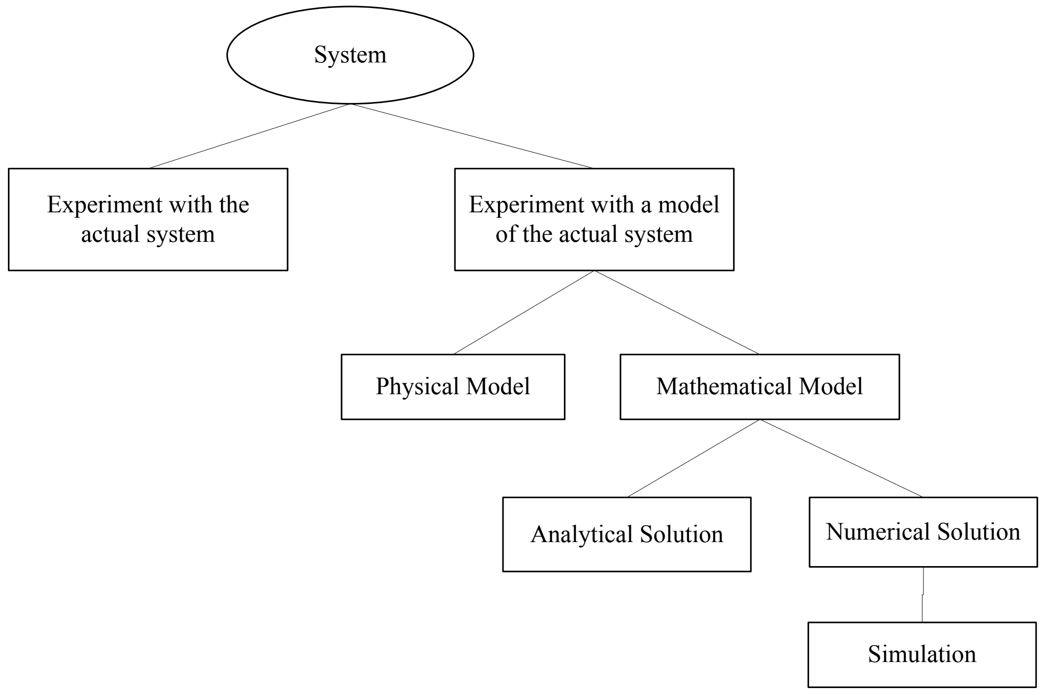


Figure 2.6: A Schematic Layout Showing the Different Techniques for Analyzing Systems

Figure 2.6 summarizes the possible routes that an analyst of a system may opt to make use of. Each route results in a specific type of model, e.g., a physical model, mathematical model or the actual system, which can be analyzed.

In cases that the analyst chooses to construct a mathematical model, they have two possible ways of solving that model – using an analytical approach or a numerical one. Analytical solutions are best applied when the mathematical model has a well-defined equation or formulation that relates the variables that define the state of the system. Numerical solutions on the other hand, serve well when no such defined relation exists, or in cases where it exists but it is too complex to solve analytically, or worse still when that relation is unknown but exists. Other situations in which numerical solutions come in handy include cases that variables exhibit stochastically and/or have dynamic behavior.

Simulation is an example of a numerical solution that is often used in the analysis of systems. Other numerical techniques exist for solving problems that are also useful but those are not discussed in this thesis. Simulation is discussed in detail because it is one of the techniques applied in developing the system required in this study.

2.6.1 Simulation

Simulation is defined as an imitation of the operation of a real-world process or system over time (Banks, Carson, Nelson, & Nicol, 2001). Page and Roger (1998) also define simulation as a process of designing a model of a real or imagined system and conducting experiments with that model.

Simulation starts with the creation of a replica of the system that is to be analyzed. This is a process referred to as abstraction. It entails representing the critical characteristics, behaviors and functions of the system. This replica of the system is often referred to as a model. The process of simulating an operation involves actions being made on the model in a fashion that mimics the operations of the system over time.

Simulations are a viable method of analyzing systems because they are cheaper and less risky in all other aspects to experiment with than the actual object or system that is being studied. Specific reasons as to why simulations experiments may be carried out include:

- Simulation experimentation with models of natural systems or human systems provides insight into their functioning.
- Simulation can be used to show the eventual real effects of alternative conditions and courses of action.
- Simulation is also used when the real system cannot be engaged, because it may not be accessible, or it may be dangerous or unacceptable to engage, or it is being designed but not yet built, or it may simply not exist.

Simulation can be categorized based on the type of model that is made use of during the experimentation process. These are discussed below.

2.6.1.1 Physical Simulation

This type of simulation makes use of physical models that represent the real system. These models are typically identical to the actual system in all aspects with the exception of the geometrical size, which is scaled down. There is a special kind of physical simulation which involves human operators. This sub-category is often referred to as interactive simulation or "human in the loop simulation." This specific type of simulation may make use of computer simulation as a synthetic environment (Page and Roger, 1998). Examples of this type of simulations include a flight simulator, driving simulator, etc.

2.6.1.2 Computer Simulation

Computer models are used in this type of simulation. These models also represent a real-life system or process or may represent a hypothetical phenomenon. Another term used to refer to simulation that makes use of this type of model is referred to as computer simulation experimentation. In computer simulation experimentation, the modeller changes variables in the model so that predictions are made about the behavior of the system (Banks, Carson, Nelson, & Nicol, 2001).

As mentioned, computer simulation is a technique used to develop and execute computer models of real or hypothetical systems. Real life systems tend to vary in the way that they behave. Some are deterministic while others are stochastic. A secondary behavior that is evident in systems is one that makes them exhibit static or dynamic behavior. These types of system behavior are summarized in Figure 2.7.

A deterministic system is one in which the occurrence of all events is known with certainty. The output of a deterministic system can be predicted with a probability of 100%. Most mathematical and scientific models are deterministic in nature. Deterministic systems are sub-categorized into static and dynamic systems.

A stochastic system is one in which the occurrence of events cannot be perfectly predicted. This is because they have an aspect of randomness associated with them. Stochastic systems can also be sub-categorized as either static or dynamic.

A static system is one in which outputs only depend on the value of the system inputs at a given time. The outputs don't depend on previous or future input values. As a result, static systems are also assumed not to be affected by changes in time. Dynamic systems on the other hand may depend upon past (including initial values) and future values of the input variables for the system. It is for this reason that static systems are considered memory-less.

Stochastic static systems would typically be analyzed using Monte Carlo simulation-based approaches. On the other hand, stochastic dynamic systems are typically analyzed using either discrete or continuous simulation or a combination of both types of simulation. This is summarized in Figure 2.7.

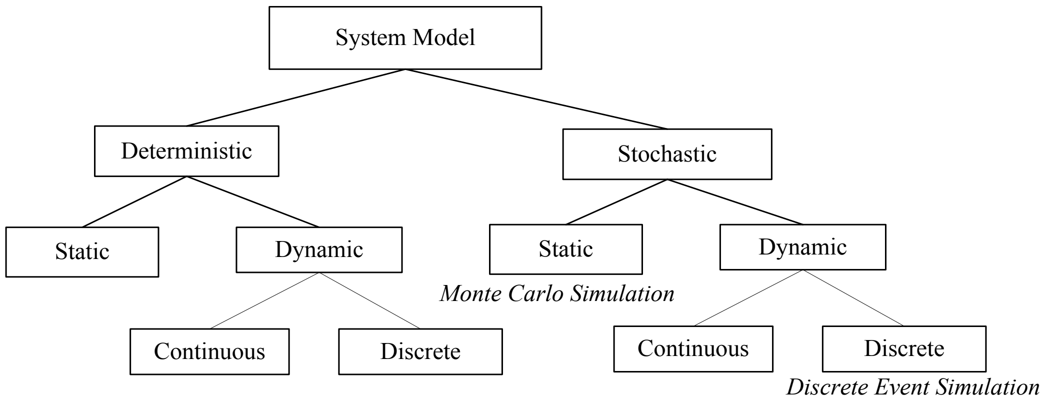


Figure 2.7: A Schematic Layout Showing the Different Types of Systems that exist in Real Life

To cope with these different system behaviors, various types of computer simulation methods have been developed to facilitate the analysis of the broad spectrum of such systems, for example, Monte Carlo simulation, discrete event simulation and continuous simulation.

2.7 COMPUTER SIMULATION MODELING PARADIGMS

Simulation models may be categorized as either monolithic or distributed depending on the thread execution strategies they implement. Common modeling approaches include Discrete Event Simulation, System Dynamics and Agent Based Modeling. These methods can be categorized as summarized in Figure 2.8. HLA based simulation are another type of simulation.

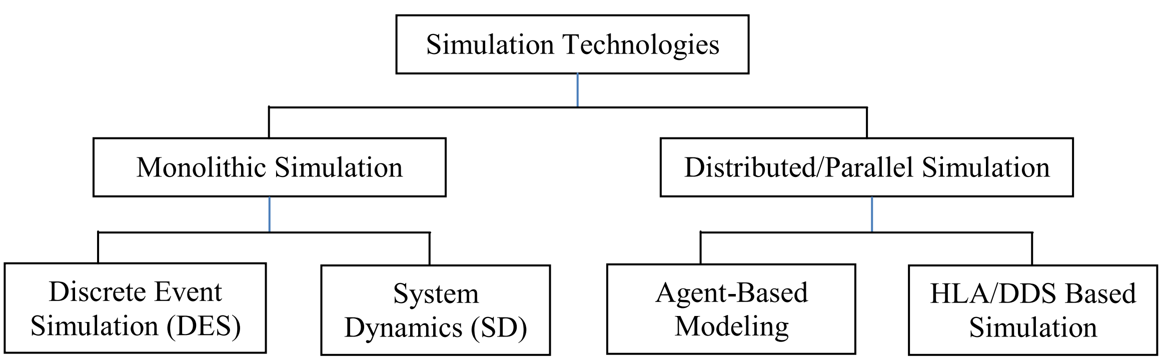


Figure 2.8: A Schematic of Different Computing Technologies and their Mappings to the Different Simulation Modeling Paradigms

DES, ABM and the HLA were the techniques applied in model development within this thesis.

2.7.1 Monte Carlo Simulation

Monte Carlo simulation utilizes models of uncertainty where representation of time is unnecessary. The term was originally attributed to "a situation in which a difficult non-probabilistic problem is solved through the invention of a stochastic process that satisfies the relations of the deterministic problem." A more recent characterization is that Monte Carlo is "the method of repetitive trials." Typical of Monte Carlo simulation is the approximation of a definite integral by circumscribing the region with a known geometric shape, then generating random points to estimate the area of the region through the proportion of points falling within the region boundaries.

As discussed earlier, Monte Carlo simulation is a technique that is appropriate for the analysis of systems or processes that are stochastic and static in nature. Systems of this nature will typically produce different outputs for the same inputs hence making their behavior difficult to predict. In addition, their outputs depend on only the input values at the current time, i.e., are not affected by past or future input values, hence the static nature. The lack of predictability of such systems requires a technique that can easily cope with randomness. Monte Carlo simulation has this capability making it suitable for solving such problems. It is for this reason that Monte Carlo simulation has acquired a reputation for solving problems that are too complicated to solve analytically (Weisstein, 2014).

In 1946, the method was assigned a name by Ulam after a famous town that was an international gaming destination and in honor of a relative that had a propensity to gamble (Hoffman, 1998). The invention of this method is attributed to both Ulam and von Neumann; scientists who used the laws of chance to develop better atomic bombs (Anderson, 1999). Soon after the Second World War it was possible to solve a wide range of notoriously difficult problems using this new technique – Monte Carlo Simulation. The Monte Carlo Simulation technique became increasingly popular in that era and is still popular today (Anderson, 1999). According to Anderson (1991), despite the wide-spread use of this method, it is not easy to find a precise unified definition for the Monte Carlo Simulation Method in the literature, which may be a result of the intuitive nature of the method which spawns many definitions by way of specific examples. Nonetheless, a few definitions that exist in the literature have been summarized in the following paragraphs.

Weisstein (2014) defines Monte Carlo simulation as a method that solves a problem by generating suitable random numbers and observing that fraction of the numbers obeying some property or properties. Anderson (1991) also defines Monte Carlo Simulation as an art of approximating an expectation by the sample mean of a function of simulated random variables.

2.7.2 Discrete Event Simulation

Concepts for Discrete Event Simulations (DES) were developed in the late 1950's. The first DES-specific language was developed at General Electric by Tocher and Owen. The General Simulation Program (GSP) was created to study manufacturing problems at General Electric and was shared with the rest of the world at the Second International Conference on Operations Research.

Discrete event simulation (DES) is the process of emulating the behavior of a complex dynamic/stochastic system by representing it as an ordered sequence of well-defined events. A system is represented by a number of variables, which also describe the state of the system. In systems simulated using a discrete event approach, time does not advance until the next event is due to occur. During the advance period, the state of the system does not change. The simulation engine time is changed just prior to processing the next event. The size of the advance periods are not the same in typical situations. It can be concluded that events drive the simulation process in discrete event simulation systems. An event is defined as an occurrence if it causes a specific change in the system's state at a specific point in time. Discrete event simulation utilizes a mathematical/logical model of a physical system that portrays state changes at precise points in simulated time. Both the nature of the state change and the time at which the change occurs mandate precise description. Customers waiting for service, the management of parts inventory or military combat are typical domains of discrete event simulation.

Next, a classical definition of discrete event simulation is presented. This is one of the many definitions that exist in literature. *"Discrete-event simulation represents modeling, simulating, and analyzing systems utilizing the computational and mathematical techniques, while creating a model construct of a conceptual framework that describes a system"* (Nance, 1993).

2.7.2.1 The Evolution of Discrete Event Simulation Technologies

Technologies supporting the discrete event simulation community have been evolving progressively since the 1960s. This evolution has been spear headed by both industry and academia in the various domains (Babulak & Wang, 2010). In their 1993 conference paper, Wang and Sun (1993), discussed the four generations of simulation software products that have thrived within the world of discrete event simulation. They include:

- *1st Generation (late 1960s)* – The first type of system required that the modeller programs the model logic and simulation engine logic in a program environment. High level programming languages such as FORTRAN were used for this purpose. This approach was predominantly used in the 60's.

- *2nd Generation (late 1970s)* – In the 70's, the discrete event simulation community started developing and using software that took control of processing the simulation events. These tools also provided for statistical distribution generation and reporting features. Examples of these systems include: GPSS (IBM), See Why (AT&T), and AutoMod (ASI).

- *3rd Generation (early 1980s)* – Systems developed in this time include: SIMAN (Systems Modeling), and EXPRESS (AT&T). These systems reduced the model development time and execution time. Modellers mainly used simulation languages in these systems to represent their logic, which would then be converted into executable code.

- *4th Generation (late 1980s)* – This era saw the emergence of simulation systems that provided modeling features that were interactive. They provided constructs, also known as modeling elements that the user could use to put together a process flow diagram that represented the logic that they intended to analyze. The approach made it easier to develop and modify models, hence, attracting larger crowds from both academia and industry into the discrete event modeling community. Examples of systems that spear-headed that era includes: WITNESS (AT&T), and ARENA (Systems Modeling). This generation served as a basis for contemporary simulation systems. Some argue that these contemporary simulation systems belong to yet another generation that descended from this one, while others argue that the evolutions that have taken place in these systems are well within the boundaries of the 4th generation. The biggest improvement in this generation was enhancement of the graphic capabilities of simulation systems.

The foregoing discussion has indicated that discrete event simulation is implemented using a simulation language or languages in some form. This could either be at a low level (as a programming language) or using interactive constructs. These languages have common requirements and characteristics that they all have to meet. Six of the most common features that cut across these languages were enumerated by Nance in 1993 and 1995. He pointed these out as the minimum requirements that any simulation language must meet. They included:

- Generation of random numbers to represent uncertainty,
- Process transformers, to permit other options than uniform random varieties to be used,
- List processing capability, so that objects can be created, manipulated, and deleted,
- Statistical analysis routines, to provide the descriptive summary of model behavior,
- Report generation, to provide the presentation of potentially large reams of data in an effective way for decision making, and
- A time flow mechanism.

2.7.3 Continuous Simulation

Continuous simulation is another type of simulation used to represent and analyze systems whose state changes continuously. The variables that define the state of the system keep changing at every point in time so, to analyze the system, artificial events or pseudo-events are created by the modeller to enable then track the fashion in which the state of the system changes over time. Consequently, it is often referred to as a time-stepped simulation (Kuhl, Weatherly, & Dahmann, 2000). This type of simulation is also often referred to as utilizing a scanning algorithm because at every pseudo-event, it interrogates the state of the system. The time steps in this type of simulation will typically be of equal size.

Continuous simulation uses models based on equations, often of physical systems, which do not portray precise time and state relationships that result in discontinuities. The objective of studies using such models does not require the explicit representation of state and time relationships.

Examples of such systems are found in ecological modeling, ballistic re-entry, or large-scale economic models.

2.7.3.1 Simulation Paradigms that Apply Continuous Simulation Algorithms

Continuous simulation may be extended to take one of two forms, namely, dynamic systems and system dynamics. The concept of dynamic systems will be briefly presented followed by system dynamics.

2.7.3.1.1 Dynamic Systems

Dynamic systems are those that have some or all of their state variables changing continuously. This causes the state of the system to continuously change in a synchronized fashion. The uniqueness of dynamic systems is that their variables are not intensely inter-related, as is the case in other systems, and therefore, they do not have pronounced feed and feedback loops within them.

2.7.3.1.2 System Dynamics

System dynamics is a technique that applies continuous simulation principles to understanding the behavior of complex systems as the systems evolve over time. The method models the relationships between variables in the system and how these relationships influence the behavior of the system over time. System dynamics makes use of stocks, flows, feed loops, feedback loops and time delays that are internal to the system, to model these complex relationships and how they affect the behavior of the entire system (Sterman, 2000; Sterman, 2001; Forrester, 1971).

Jay Forrester proposed system dynamics as a method for analyzing systems (Radzicki & Taylor, 2008). This took place in the mid-1950s while has was working as a professor at MIT, Sloan School of Management. According to Radzicki and Taylor (2008), Forrester was in a position to formally come up with the system dynamics approach subsequent to his involvement in solving an employment instability problem that was being faced at the time at General Electric (GE). He developed solutions based on hand simulation computations. A Computer Scientist (Richard Bennett) then built on Forrester's work and created the first system dynamics computer modeling language called SIMPLE (Simulation of Industrial Management Problems with Lots of

Equations) in 1958 (Radzicki & Taylor, 2008). In the following year, Phyllis and Alexander Pugh developed the first version of DYNAMO (DYNAmic MOdels). DYNAMO provided more enhanced features compared to SIMPLE and has since been used for solving system dynamics problems (Radzicki & Taylor, 2008).

In those early years, system dynamics was predominantly used for modeling and analyzing corporate and managerial problems (Radzicki & Taylor, 2008). This changed as time passed. The technique is currently used in different domains including construction engineering. It is worth noting that most applications in which system dynamics has been successfully used tended to investigate the impact of specific policies or strategic management decisions on the system. This confirms that system dynamics is a technique that is most appropriate for studying systems at a high level.

2.7.4 Agent-Based Modeling

2.7.4.1 Definition(s) of an Agent

According to Nwana (1996), the concept of an agent emerged as far back as the late 1970s. The concept first manifested in the work of Carl Hewitt in 1977. Hewitt (1977) proposed an object that is self-contained, interactive and concurrently-executing, which he referred to as an actor. The concept of actor later evolved into what is known today as an agent. "An 'Actor' is a computational agent which has a mail address and behavior. Actors communicate by message-passing and carry out their actions concurrently" (Hewitt, 1977).

To further support this view, Macal and North (2005), also stated that so far, there has been no single definition put forward for the term "agent." However, the authors note that definitions that already exist tend to agree on more points than they disagree. Some definitions are presented here to give insights into what an "agent" actually is: an "agent" is any type of independent component (software, model, individual, etc.) (Bonabeau, 2001).

Extensive research work has been done over the years that covered the subject of agents. Some of it was academic while some was applied. This work can be categorized into two main streams. Each of these is briefly discussed in the following paragraphs.

The first stream of research focused on efficiently implementing issues relating to agents such as their interaction, coordination, assignment of roles/responsibilities, conflict resolution via negotiation, etc. (Nwana, 1996). Gasser (1991) stated that this work dealt with the 'macro' aspects of agents. The emphasis was on a *society of agents* rather than *individual agents* so that systems could be analyzed, designed and integrated using multiple collaborative agents (Nwana, 1996). Research on macro aspects of agents focused on the society of agents and this constitutes the earlier part of this stream of research. The later part of this stream investigated micro issues related to agents, that is to say, it looked at agents as individual 'actors' rather than in groups. Examples of systems developed in the era that 'macro' issues were studied include the actor model (Hewitt, 1977), MACE (Gasser et al., 1987), DVMT (Lesser & Corkill, 1981), MICE (Durfee & Montgomery, 1989), MCS (Doran et al., 1990) the contract network coordination approach (Smith, 1980; Davis & Smith, 1983), MAS/DAI planning and game theories (Rosenschein, 1985; Zlotkin & Rosenschein, 1989; Rosenschein & Zlotkin, 1994). Research work done on agent 'micro' issues can be found in Chaib-draa et al. (1992), Bond & Gasser (1988) and Gasser & Huhns (1989). Examples of other recent systems developed under the banner of this first stream of research include TÆMS (Decker & Lesser, 1993; Decker, 1995) DRESUN (Carver et al., 1991; Carver & Lesser, 1995), VDT (Levitt et al., 1994), and ARCHON (Wittig, 1992; Jennings et al., 1995).

According to Nwana (1996), the second stream of research is believed to have been initiated in the early 1990s. This stream focused on broadening or diversifying the range of agent types that exist. Nwana (1996) and Wooldridge & Jennings (1995) extensively discussed the progress that has been made in this research stream in their papers.

2.7.4.2 Attributes of Agents

To further clarify what an "agent" is or is not, researchers have attempted to enumerate the items that should exist within a given component for it to qualify as an "agent." There has still not been a universal consensus on this. However, from summarizing all these views, one can deduce that an agent should have a behavior that can range from primitive reactive decision rules to complex adaptive intelligence (Macal & North, 2005). The most basic requirement for an "agent" was put forward by Jennings (2000) when she stated that the "essential characteristic in an agent is its autonomous behavior." She further stated that this would result in agents being active rather than passive and facilitate independent decision making. Mellouli et al. (2003), on the hand, insist that a component's behavior must be adaptive (change their behaviors in response to the environment) in order for it to be considered an agent. Casti (1997) argues that agents should contain both base-level rules for behavior as well as a higher-level set of "rules to change the rules." The base level rules provide responses to the environment, while the "rules to change the rules" provide adaptation.

Certain properties, attributes and methods were enumerated and stated as required ingredients of agents for practical modeling purposes (Macal & North, 2011). These were summarized in a Figure that was presented at a Winter Simulation Conference in 2011 – see Figure 2.9.

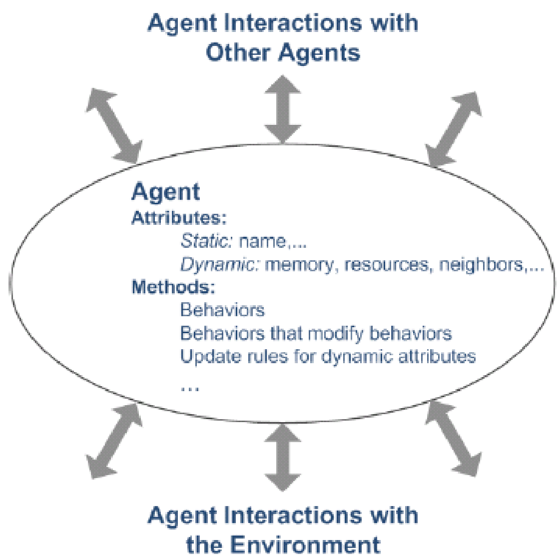


Figure 2.9: Essential Ingredients of an Agent (Macal & North, 2011)

Macal & North (2011) proceeded to present four other specific properties that they believed agents must have for practical modeling purposes. The first three properties are identical to what (Nwana, 1996) presented in his paper. They include:

- *Autonomy:* Autonomy refers to the ability of an agent to make decisions and act independently without the intervention or direction of any humans.

- *Modularity:* This refers to a modular or self-contained nature of an agent. This also means that the boundary of an agent can be clearly drawn – things that don't belong in it can be easily separated from those that do not belong to it. It can also be inferred from the paper written by Nwana (1996) that the term modularity can be assumed to cover the concept of agents being concurrently executing.

- *Sociality:* Sociality refers to the ability of an agent to interact with other agents and its environment. Common agent interaction protocols include contention for space and collision avoidance, agent recognition, communication and information exchange, influence, and other domain- or application-specific mechanisms. This social interaction results in emergent group behavior from simple individual social behavior (Axtell & Epstein, 1994). According to Axtell and Epstein (1994), this is referred to as a "bottom-up effect." When referring to agent modeling, interaction and emergent behaviors, Axtell and Epstein (1994) also stated in their article that this concept would be a means to "let a thousand artificial flowers bloom." They implied that each artificial flower would represent an agent and the blooming flowers, the agent population's emergent behavior. The artificiality aspect implied that agents are just an abstraction of a real life concept.

- *Conditionality:* This refers to the ability to identify and represent the various states that an agent can take on and the transitions between these states. An agent has a *state* that varies over time (Macal & North, 2011). Macal & North (2011) state that just as a system has a state consisting of the collection of its state variables, an agent also has a state that represents its condition, the essential variables associated with its current situation. The authors further stated that an agent's behaviors are conditional on its state. As such, the richer the set of an agent's possible states, the richer the set of behaviors that an agent can have.

Macal & North (2011) believed that all four need to exist for a component to qualify as an agent. Nwana (1996) on the other hand emphasized that only the first three (autonomy, modularity and sociality) are the fundamental requirements within an agent. Nwana (1996) was right on that because although agents have numerous states that they transition through as time evolves, there are ways of defining agent behavior without directly tying it to these states, or without the need to explicitly model these states. Consequently, it can be stated that the need to explicitly represent an agent using its states is an attribute that can be added to the fundamental attributes of autonomy, modularity and sociality.

Other properties that can be embedded within agents to extend their behavior are discussed below. These are based on work presented in papers authored by experts in the agent-based modeling domain.

- *Mobility:* This refers to the ability of agents to move around the environment in which they are situated. Based on this criterion, agents may be categorized as either static or dynamic. Static agents are not mobile, while dynamic agents are mobile. Mobile agents will typically have attributes that define their position or location and speed or velocity when they are in motion.
- *Memory:* This is an extra feature that can be embedded when agents are being designed. When activated, agents can have knowledge of a state that they were previously in before being disrupted and sent off into other states. In addition, agents can also store information about their past experiences. This information can be stored within attributes of the agent.
- *Learning:* Agents that exhibit learning are able to improve their performance based on the time that they spent within an environment. Learning is also enhanced by the experiences of agents within the environment. A learning agent will also have some sort of logical (if … then…) rule base. This philosophy can best be implemented together with memory capabilities.

2.7.4.4 Designing and Developing Agent-Based Models

Agent-based models are usually used for representing complex systems. This is to facilitate a better understanding of the behaviors of such systems. Typical agent-based models will generally be complex as well, and their development will require careful thought and design. Designing agent-based models simply refers to documenting the specifications of what the agents are, their behavior and properties and detailing the nature of communication envisaged to take place between the agents.

The first step in designing agent-based models is deciding what the agents are in the system being analyzed. This is followed by the definition of the exact roles assigned to each agent.

These roles dictate the behavior of the agent and the nature of its interaction with other agents or the environment.

There are a number of tools available from computer science and software engineering that can be used as aides in specifying designs. For agent behavior, process interaction flow models, flowcharts or state charts could be used. The use of state charts requires a good understanding of Universal Modeling Language (UML) or System Modeling Language (SysML), or both. These are ontologies used for documenting agent-based models. In addition to this, the modeller needs to have a solid background in Object Oriented Programming concepts. XJ Technologies (2013) cited guidelines to use for deciding the constructs to make use of when modeling agent behavior. These included:

- Does the agent just react to the external events? Use message handing and function calls.
- Does the agent have a notion of state? Use a state chart.
- Does the agent have internal timing? Use events or timeout transitions.
- Is there any process inside the agent? Draw a process flowchart.
- Are there any continuous-time dynamics? Create a stock and flow diagram inside the agent.

On the other hand, communication between agents can best be designed using sequence diagrams. The designer needs to specify the parties involved in the communication, how it is started and completed. They also need to specify whether messages are asynchronous or not. Since the term "asynchronous " has two definitions, which can both be applied to and affect the design, the designer needs to further clarify which of the two meanings he/she is referring to.

Another aspect of an agent-based model that is good to specify and document is the classes envisaged for use in the model and the relationship between these classes. This is especially important for complex systems that are making use of object oriented concepts extensively. Block diagrams may be used for specifying these classes.

Once the design of the system is completed, the developer can then proceed to translate the design into a model within a simulation system that contains an agent modeling tool box. The design phase in developing agent-based models is the most critical and should always come as the first step in agent modeling. Well done agent system designs provide two advantages:

- It makes the modeling process easy, fast and results in an accurate model.
- It provides proper documentation of the model development and the thought process that the modeller applied so that other interested parties may learn from this process.

Consequently, designing agent systems should be mandatory for beginner modellers using this simulation method (agent-based approach) regardless of the complexity of the system being abstracted. Expert modellers may opt to skip this step for very simple systems that they are analyzing. However, as the complexity of the systems that they are analyzing increases, it becomes mandatory for them to design these systems before they move on to modeling.

2.7.4.5 Components of an Agent-Based Model

According to Macal and North (2010), typical agent-based models comprise three elements. These include:

- Agents – their behaviors and attributes.
- Agent relationships – methods for interaction and rules governing which agents get to interact and which do not. This also encompasses agent topology.
- An agent environment – the space in which agents thrive and interact.

Agents, their behaviors and attributes have been comprehensively discussed in the previous section. In the following section, a discussion will be presented about agent relationships and topology and agent environment.

2.7.4.6 Structure/Topology of Agent-Based Models

The relationship between agents is dictated by the roles that each agent plays within the model. Communication between agents may be direct, i.e., agent-to-agent, or may be indirect, i.e., agent-to-environment-to-agent. Also, interaction may be restricted between certain types of agents in certain systems, while in other systems; there may not be any restriction on the inter-agent communication. Also, agents may be banded into different hierarchical layers. The one at a higher level in a hierarchy could have more authority and autonomy and could decide a significant part of the behavior of those at the bottom of the hierarchy.

The topology shown in Figure 2.10 is one in which there is no *super-agent* controlling the different agent populations. Each agent population is autonomous and communicates or interacts within populations and across populations.

In the topology shown in Figure 2.11 and 2.12, there exists a *super-agent*. This agent, to some extent, controls the actions of the different agent populations that exist within the model. It is usually a singleton (can only be one instance within the model). The requirement for this *super-agent* to control the other agent populations means that there are interactions between themselves, i.e., *super-agent* and agent populations. However, the communication across agent populations is not always a must (see Figure 2.11), but may exist (as shown in Figure 2.12).

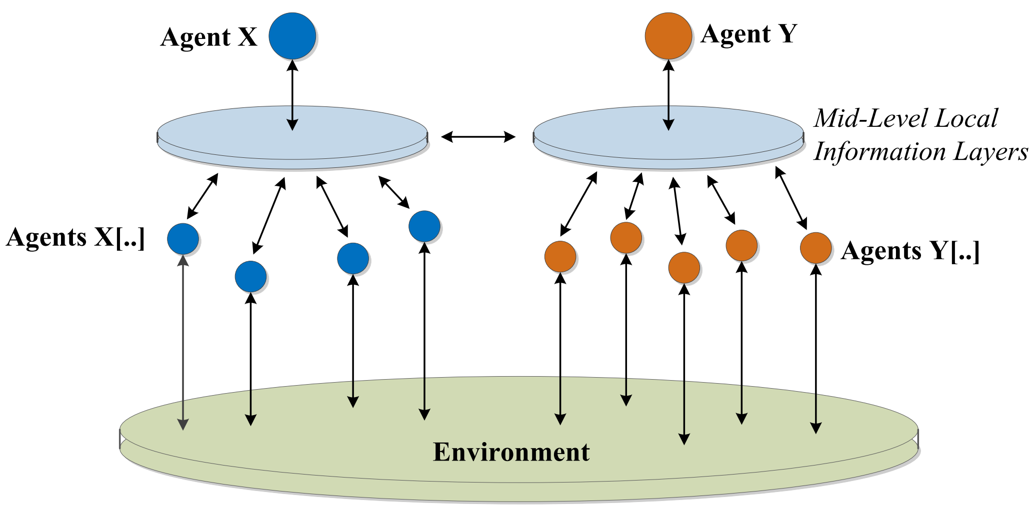


Figure 2.10: Topology for an Agent-Based Model with Autonomous Agents Interacting with Each Other (No Super-Agent)

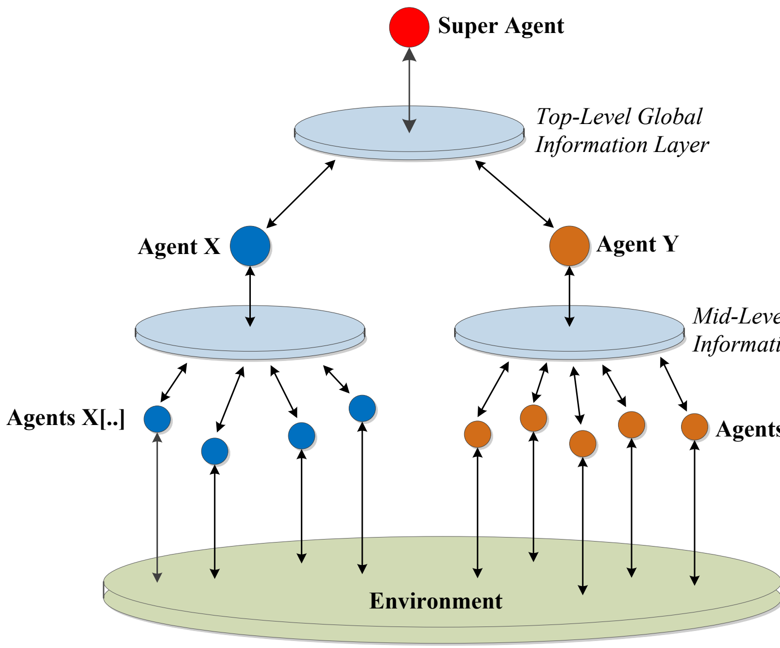


Figure 2.11: Topology for an Agent-Based Model no Agent-Agent Direct Interacting with Each Other (Has a Super-Agent)

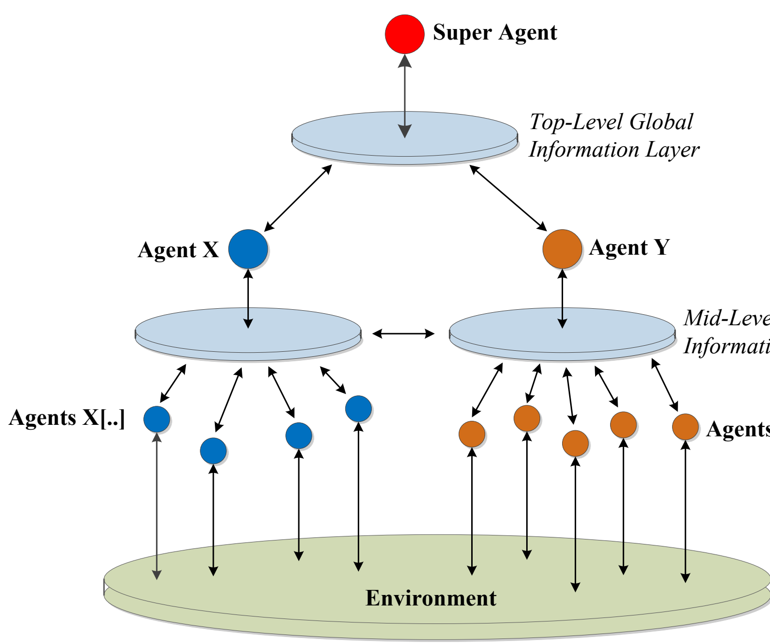


Figure 2.12: Topology for an Agent-Based Model with Direct and Indirect Agent-Agent Interaction (Has a Super-Agent)

In this thesis, the topology shown in Figure 2.11 was found suitable for the problem domain and was hence adopted in the model development process.

2.7.4.6 The Environment in ABMs/MAS

An environment is the place that agents thrive. From a computing science perspective, an environment is a container into which agents are placed. Environments introduce a number of concepts to a model that contains agents. These include:

- A concept of agent position (cells) and arrangement – also referred to agent layouts.
- Possibility of agent movement.
- A concept of neighbors and connectivity – also referred to as agent networks.
- A concept of indirect agent-to-agent communication.
- A concept of spatial information and visualization.

Environments can facilitate the development of more sophisticated models that are also closer to reality than had ever been imagined before. Agent environments vary in type based on the way that space is represented. They include:

- Discrete space type.
- Continuous space type.
- GIS space type.

2.7.4.7 Past Research Studies Making Use of Agents

The flexibility of the agent-based simulation modeling paradigm has made it a popular choice as an approach for analyzing and designing complex systems in different domains. Macal and North (2011) cited the use of the method in human social, physical, and biological systems. Other domains cited include Transportation, Economics, and Logistics. Sample applications are presented in Table 2.0.

Table 2.0: Studies Carried Out Using ABM in Different Domains

Application Area	Model Description
Agriculture	A spatial individual-based model prototype for assessing potential exposure of farmworkers conducting small-scale agricultural production (Leyk, Binder, and Nuckols, 2009).
Air Traffic Control	Agent-based model of air traffic control to analyze control policies and

Application Area	Model Description
	performance of an air traffic management facility (Conway, 2006)
Anthropology	Agent-based model of prehistoric settlement patterns and political consolidation in the Lake Titicaca basin of Peru and Bolivia (Griffin and Stanish 2007)
Biomedical Research	*The Basic Immune Simulator*, an agent-based model to study the interactions between innate and adaptive immunity (Folcik, An, and Orosz, 2007)
Construction Engineering and Management	(Du & El-Gafy, 2012); (Liu & Mohamed, 2012)
Crime Analysis	Agent-based model that uses a realistic virtual urban environment, populated with virtual burglar agents (Malleson, 2010).
Ecology	Agent-based model to investigate the trade-off between road avoidance and salt pool spatial memory in the movement behavior of moose in the Laurentides Wildlife Reserve (Grosman et al., 2011). Agent-based model of predator-prey relationships between transient killer whales and other marine mammals (Mock and Testa, 2007). A risk-based approach for analyzing the intentional introduction of non-native oysters on the US east coast (Opaluch, Anderson, and Schnier, 2005).
Energy Analysis	Agent-based model to identify potential interventions for the uptake of wood-pellet heating in Norway (Sopha et al., 2011). Agent-based model for scenario development of offshore wind energy (Mast et al., 2007).
Epidemiology	Synthetic age-specific contact matrices are computed through simulation of a simple individual-based model (Iozzi et al., 2010).
Evacuation	A simulation of tsunami evacuation using a modified form of Helbing's social-force model applied to agents (Puckett, 2009).
Market Analysis	A large-scale agent-based model for consumer marketing developed in

Application Area	Model Description
	collaboration with a Fortune 50 firm (North et al. 2009). An illustrative agent-based model of a consumer airline market to derive market share for the upcoming year (Kuhn et al., 2010). Agent-based simulation that models
Organizational Decision Making	An agent-based model to allow managers to simulate employee knowledge-sharing behaviors (Wang et al., 2009). An agent-based model to evaluate the dynamic behavior of a global enterprise, considering system-level performance as well as components' behaviors (Behdani et al., 2009). Agent-based modeling approach to allow negotiations in order to achieve a global objective, specifically for planning the location of intermodal freight hubs (van Dam et al., 2007).
Social Networks	An agent-based model of email-based social networks, in which individuals establish, maintain and allow atrophy of links through contact-lists and emails (Menges, Mishra, and Narzisi, 2008).

2.7.4.8 When to Use ABM

Identifying the right method to use in solving a problem is always the first step towards implementing an efficient solution to that problem. This is because it results in an accurate solution and leads to proper use of time and other resources. Macal & North (2011) highlighted criteria that qualify agent-based simulation modeling approaches as the most suitable approach for use in the analysis of systems. These include:

- When a system is comprised of constructs that are autonomous or semi-autonomous and self-executing.
- Systems comprised of constructs that possess unique behaviors that can be well-defined.
- Situations that warrant dynamic interactions between constructs in the system
- When scaling-up to arbitrary levels is important in terms of the number of constructs, their interactions, and states.
- Situations in which constructs exhibit adaptive learning and memory usage.

- When process structural change needs to be an endogenous result of the model, rather than an input to the model.

A combination of these were used as a basis for making the decision to utilize an ABM approach in the representation of the tendering module for the simulation application developed in this thesis.

2.7.4.9 Validation of ABMs

Axtell and Epstein (1994) presented a paper in which they proposed a framework that could be applied in validating ABMs. They stated that there are both pitfalls and powerful diagnostic tools unique to agent-based simulations which need to be appreciated if confidence is to be gained in model results (Axtell & Epstein, 1994).

The authors enumerated 4 labels that represent the levels of confidence that can be practically achieved in ABMs. The labels included: *Level 0*, *Level 1*, *Level 2* and *Level 3*. Each of these levels will be explained in more detail in the following paragraphs. However, prior to getting into this discussion a few definitions will be presented to make understanding of these levels of model confidence building.

An individual agent is a term used to refer to an autonomous, self-executing entity that can react and act on stimulus from an environment. An agent population refers to a group of object instances of a specific type of agent. The term macro-structures is used to refer to agent populations while the term micro-structures is used to refer to individual agents.

A detailed read of Axtell and Epstein's 1994 paper reveals that there are a number of ways of validating ABMs. These are summarized below:

- Identification of model variables and parameters that is measureable in the real world, and measuring those. Statistical tests can then be carried out on the model results using actual data to determine whether the model is valid or not.
- Abstract typical scenarios from a real world system and strive to make the model match those scenarios. The scenarios should include a good representation of the real world system's extreme behavior (good and bad) and its normal behavior. A scenario can be created by setting the model parameters to pre-defined values.

- Introducing stimulus of some sort to the ABM and tracking the behaviors in the model – simple and emergent behaviors.

The qualitative aspects of an ABM refer to the fashion in which agents and agent populations respond to stimulus introduced into the model. These may be studied by observing visualization effects of individual agents or groups of agents. Another approach involves keeping track of the number of agents or agent populations that behave in a specific fashion when stimulus is introduced into the model. Plotting the distributional properties of the agent populations can be extremely helpful at this stage (Axtell & Epstein, 1994). Examples of such plots include pie charts that show proportions of agents that respond to stimulus in various fashions.

Confidence building in the quantitative aspects of ABMs refers to establishing how realistic the values for the quantities of stimulus assimilated by the agents or agent populations are. In the example presented by Axtell and Epstein, the stimulus is the food introduced into the ant model. The quantitative aspects to track are the amount of food that each ant can move/assimilate. At another level, one would look at quantities of food moved by ant populations. Examples in the construction domain would include the amount of dirt moved by a single truck at one level and then at another level, dirt volumes moved by a fleet of trucks. In this study, stimulus could be projects introduced into the industry for tender and the quantitative aspects could include the number of projects awarded to a specific company or to a certain group of companies.

Next, a description will be presented on how the previously discussed concepts and definitions apply to the various levels of model performance presented by Axtell and Epstein. *Level 0*, *Level 1*, *Level 2* and *Level 3* are the four levels presented in their paper.

Level 0 and *Level 1* represent stages at which confidence building in the qualitative aspects of an ABM has been achieved. *Level 0* represents establishing confidence in an ABM's qualitative aspects at a micro level, i.e., at the individual agent. *Level 1* on the other hand represents confidence established in the qualitative aspects of an ABM at a macro level, i.e., for agent populations. Details presented in prior sections indicate that confirming the performance of a model at these levels can best be accomplished using visualization, especially if the agents are mobile. In a nutshell, these levels of model performance track the trends in agent and agent population behavior.

The next two levels, 2 and 3, represent performance of an ABM with respect to a chosen quantitative aspect. *Level 2* represents the quantitative performance of a model at a micro level, while *level 3* represents the quantitative performance of a model at a macro level. The terms micro and macro refer to individual agent and agent population respectively. Confidence in the quantitative aspects of a model can be determined by tracking numeric values that relate to a specific stimulus introduced into the model. The numeric values of parameters that are not directly related to a specific stimulus can also be tracked and used to check the performance of the model. A good example is the average cycle time of trucks in an earth-moving operation or the average production rate at the individual agent (truck) or for the agent population (fleet of trucks). Some or all of the agents in the model may be configured to gather data about the system, hence the term data gathering agents (Axtell & Epstein, 1994). Statistics collected on model parameters can be compared to data collected on the same parameters on real projects. Statistical tests can then be carried out to confirm the validity of the model. Examples of such tests include the student's t-test. In such tests, the analyst would put forward a null hypothesis that states that there is no relationship between the two sets of data. The challenge then is for the analyst to gather sufficient evidence to reject that null hypothesis, hence proving that in fact there is a relationship between the two data sets. Data of real systems can be obtained from actual measurements of parameters or using domain experts that define outputs or trends using their experience.

According to Axtell & Epstein (1994), these levels are progressive. Performance that is established to be satisfactory at level N implies that it is also satisfactory at level N-I, i.e., lower levels (Axtell & Epstein, 1994).

2.7.4.10 Verification of ABMs

Verification of ABMs is similar in many ways to the verification of other types of simulation models. It can be done on two fronts. These would involve:

- Establishing that the simulation modeling system and framework are doing whatever they were designed and developed to do.
- Confirming that the models developed using these simulation environments are behaving the way they were intended.

The first phase of verification (in the first bullet) should be done before the second (the second bulleted point) because there is no way to guarantee the second, even if it has been well done, if the first is not tested and verified. In most cases, verification of a simulation system or framework is done by the developers at the tail-end of the development process.

Confirming the behavior of a model is not a new process. The traditional techniques for achieving this include:

- Visualizations of the operation (Kamat, 2000; Rohani, Fan, & Yu, 2013; Al-Hussein, Niaz, Yu, & Kim, 2006).
- Tracing numeric data and simulation events as the model execution evolves (Ekyalimpa, AbouRizk, & Farrar, 2012).

Most visualization platforms are tied to simulation models behind the scenes. Visualization permits one to observe and assess the logic represented in the simulation model that drives this visualization and decide whether it is consistent with what they intended the model to do. This simulation model may be an ABM or developed using any other simulation paradigm like DES.

Nearly all simulation systems provide a console onto which the modeller can trace data generated in the simulation. This console can be used to print out the simulated events as the simulation progresses. An assessment of the log of both data can give valuable insights into whether the model is reliable.

The validation and verification of very large-scale and complex system/application can prove to be very challenging and in some cases not feasible. The most suitable approach to this is to carry out this task step-wise, in phases. Components of the system can be validated and verified independently, and then inferences of the validity of the entire system, when all components are put together, can be made.

2.7.4.11 Past Studies in Construction Using ABM

Taghaddos (2010) used and ABM approach to analyze and solve a complex resource allocation problem commonly faced in module fabrication yards in the industrial construction sector. He also extended his work to modeling the different operations that feed into industrial projects. He produced a thesis and a number of papers from this work (AbouRizk, Mohamed, Taghaddos, Saba, & Hague, 2010; Taghaddos, Hermann, AbouRizk, & Mohamed, 2010; Taghaddos, 2010).

The study by Du and El-Gafy (2012) has already been presented in this chapter (in the section on simulation-based performance management systems). However, it is worth noting once again that the researchers used an agent-based modeling approach to solve a problem within the construction domain. An agent-based application for managing construction organizations, which they referred to as VOICE, was produced from this work.

Liu and Mohamed (2012) stated that adopting an agent-based modeling approach in the dynamic allocation of resources to different activities under a set of dynamic and diverse constrains would be more easy-to-use and generate more accurate results. They developed an ABM using Repast Simphony for a real case study for assembly operations of industrial construction modules (Liu & Mohamed, 2012). The ABM was used to evaluate the effects of different optimization algorithms and modeling parameters on the generation of a construction schedule. Experimentation with their model showed sensitivity only under large and continuous workloads.

In 2013, Ahn, Lee and Steel (2013), used agent-based modeling approaches to study the absence behaviors of workers within the construction industry. They investigated the effects of social learning and worker's perceptions to social norms on their absence behavior (Ahn, Lee, & Steel, 2013). The authors later used empirical data that they collected through questionnaire surveys to validate their agent-based models (Ahn & Lee, 2014).

The four studies presented are show-cased to demonstrate that the ABM approach can be successfully used in construction to solve academic research and practical problems that are highly complex, ill-defined and distributed in nature. It was with this background that the agent-based approach was adopted as a methodology for developing the front-end component of the simulation-based company performance management system.

2.8 HIGH LEVEL ARCHITECTURE (HLA) AND DISTRIBUTED SIMULATION

HLA is a standard that promotes re-usability and interoperability of distributed simulation systems. Distributed computing offers many advantages for all types of computational applications (Usman, Mueller, Elsheikh, Palensky, & Widl, 2013). Sample domains within which distributed simulation has been extensively used include: defense, space, air traffic management, energy, off-shore, railway and car industry, manufacturing, and health care.

Distributed simulations systems have been developed in each of these domains to support analysis, engineering and training in a number of ways.

2.8.1 Distributed Simulation

Simulation systems can be designed and implemented using different topologies. The architecture of the simulation system may either be monolithic or distributed, just like we have in computing technologies. Monolithic simulation systems are used for abstracting and analyzing simple systems on the same computer platform. Distributed simulation systems are used for analyzing large-scale complex systems. Distributed simulation systems are also popularly known as networked simulation because of the form of their topology. Each component of this large-scale system can be implemented on different platforms that interact (exchange data and execute synchronized actions). Distributed simulation environments make this possible. Distributed simulation promotes portability and interoperability of different simulation components (Luis et al., 2013).

Literature indicates that in the early 90's, developers of distributed simulation systems and models acknowledged that there was lots of activity and work being done on distributed simulation. However, most of this was done in isolation and became a concern. These individuals believed that if there was a means of information exchange between companies and groups, the technology would advance more rapidly. They also believed that when the technology stabilized, there would be a need for standardization, something that would be easy to do with groups working together. This led to the formulation of the Simulation Interoperability Standards Organization (SISO). This organization started up, then started the Simulation Interoperability Workshop (SIW), a semi-annual event held in spring and fall. The work of this organization also led to the advent of distributed simulation standards. For example, The Institute of Electrical and Electronics Engineers (IEEE) came up with the HLA standards while the Object Management Group (OMG) proposed Data Distribution Service (DDS) standards. These standards are widely used and the different aspects of each have been compared by some researchers in the domain, such as Rajive & Gerardo-Pardo (2006).

These guidelines were formulated and packaged into rules and standards, which were to be used to guide distributed simulation framework developers and developers of distributed simulation systems. High Level Architecture (HLA) is a popular standard and is discussed here because the simulation framework used for developments in the thesis work was based off of this standard. Before a discussion of the different aspects of the HLA is presented, terminology commonly used in the distributed simulation domain is presented.

A *federate* is an HLA compliant simulation entity. It is typically a simulation model that can be an integral part of a distributed simulation system.

A *federation* is a system that is comprised of multiple simulation entities connected through a Run-Time-Infrastructure (RTI) using an Object Model Template (OMT) or Federate Object Model (FOM).

An *object* is a collection of data shared by federates (simulators). A federate can register an instance of an object and then change the attributes. Other federates that are subscribed to the object receive attribute value updates.

An *attribute* is a data field of an object.

An *interaction* is a message (an event) sent between federates (simulators). Interactions work in a similar way, except that an interaction is only used once with a specified set of parameter values and then discarded.

A *parameter* is a data field of an interaction.

2.8.2 High Level Architecture (HLA)

The HLA standards are guidelines that were proposed by the Institute of Electrical and Electronic Engineers (IEEE) for the benefit of those in the area of distributed simulation. A more formal definition for the HLA can be found within IEEE Standard 1516. These standards have evolved over the years with the first version having been released in 2000. Subsequent versions were produced in 2003, 2007 and 2010. Prior to publication of IEEE 1516, the US Defense Modeling and Simulation Office was in charge of developing the HLA standards. The first complete version of the standard ever released was published in 1998 and was known as HLA 1.3.

In 2000, four HLA standards were released. Each of these was structured in such a way that they addressed a specific aspect of distributed simulation. These are summarized below.

- *IEEE 1516–2000*: Standard for Modeling and Simulation High Level Architecture (Framework and Rules)
- *IEEE 1516.1–2000*: Standard for Modeling and Simulation High Level Architecture (Federate Interface Specification)
- *IEEE 1516.2-2000:* Standard for Modeling and Simulation High Level Architecture (Object Model Template [OMT] Specification)

Other enhanced versions of the HLA standard were subsequently released three years after the 2000 version. This 2003 version recommended a practice for the process of developing and executing HLA compliant simulation federations. Years after, other improved versions of the standards followed; i.e., the 2007 and 2010 versions, respectively. These versions recommended (1) a practice for verifying and validating distributed simulation models – IEEE 1516.4-2207 (i.e., the 2007 version), (2) a federate interface specifications – IEEE 1516.1-2010, (3) object model template specifications – IEEE 1516.2-2010, and (4) the framework and rules – IEEE 1516-2010 (i.e., the 2010 version).

In the next sections, a brief discussion of each of the components (i.e., the rules, the interface, and the OMT) in the HLA standard will be presented.

2.8.2.1 HLA Rules

The HLA rules describe the responsibilities of federations and federates in any given distributed simulation system (U.S. Defense Modeling and Simulation Office, 2001). There are a total of ten rules. The first five are about the federation, while the next five are about federates. These rules can be viewed from the HLA standards.

2.8.2.2 Federate Interface Specifications

Interface specifications are a part of the HLA standards that provide details of how HLA compliant simulators interact with the Run-Time Infrastructure (RTI). The RTI is a software program that provides a programming library and an application programming interface (API) compliant to the interface specification.

The interface provides the simulation services that make it possible for modellers to develop and execute their distributed simulation systems. These services were first highlighted in the HLA standards, version 1.3. They are briefly discussed below.

- *Federation management:* Defines how federates can connect to the RTI, create, join and manage federations, save and restore federation states and defines a system to synchronize federates to the same time.
- *Declaration management:* Defines how federates declare their intentions with regard to publication and subscription of classes and interactions.
- *Object management*: Defines how federates can utilize objects and interactions once they have ownership of them.
- *Ownership management:* Defines how federates divest and acquire ownership of registered objects.
- *Time management*: Defines how time is used in a federation and how it affects object and interaction updates, federate saves and other services.
- *Data distribution management:* Defines the various ways that object and interaction data is transferred from and to federates through the RTI.
- *Support services:* Defines various services to retrieve information about the current federation, such as classes and interactions.

The specifications for these services are summarized within the HLA standards and are used in the development of distributed simulation framework software referred to as a Run-Time-Infrastructure (RTI). Object oriented concepts are used in the development of the RTI software with the majority of the enumerated services implemented as methods. Although all these services may exist within a given RTI, not all are required in the development and execution of a distributed simulation federation. Literature reviewed to-date indicates that there are a number of RTIs developed at different institutions. Examples of these are enumerated in Table 2.1.

Table 2.1: Examples of Run-Time-Infrastructure Developed for Use in Distributed Simulation

RTI Name	Developer/Vendor	Programming Language	License
CAE RTI	CAE Inc.	C++	Commercial
Chronos RTI	Magnetar Games	C++	
HLA Direct	General Dynamics C4 Systems	C++	
MAK High Performance RTI	MAK Technologies	C, C++, Java	
Mitsubishi ERTI	Mitsubishi Electric Corp. and Mitsubishi Space Software Co. Ltd	C++	
Openskies RTI	Cybernet Systems	C++	
Pitch RTI	Pitch Technologies	C++, Java, Web services	
RTI NG Pro	Raytheon Company	C++, Java	
SimWare RTI	Nextel Aerospace Defence & security S.L.	C++	
BH-RTI	Beijing University of Aeronautics and Astronautics Virtual Reality Laboratory	-	Open Source
CERTI	ONERA	C++, Fortran90, Java, Matlab, Python	
COSYE RTI	Hole School of Construction	C#	

RTI Name	Developer/Vendor	Programming Language	License
	Engineering, University of Alberta		
EODISP HLA	P&P Software	Java	
GERTICO (German RTI based on Corba)	Fraunhofer IOSB	C++	
jaRTI	Littlebluefrog Labs	C++, Java	
MATREX RTI	Dynamic Animation Systems	C++, Java	
Open HLA	-	Java	
Open RTI	Flight Gear Project	C++	
Rendezvous RTI	National University of Sciences and Technology (NUST)	C++, Java	
RTI-S	Naval Warfare Development Command	C++, Java	

2.8.2.3 Object Model Template (OMT)

An object model template (OMT) is a template used for specifying the details used to model objects. This specifications development is usually done at model design time. The OMT serves as a common platform for the communication between distributed HLA simulation components. OMT consists of the following two documents:

- *Federation object model (FOM)*. The FOM describes the shared object(s), attribute(s), interaction(s), and parameter(s) for the whole distributed simulation federation.
- *Simulation object model (SOM)*. A SOM describes the shared object(s), attribute(s) interaction(s), and parameter(s) used within a single federate.

It is mandatory for distributed simulation modellers to develop an FOM because without it, they would not be able to execute the developed federation because the communication/interactions between federates would not be possible. An SOM can also be developed for purposes of ensuring that the development process for federates is well documented. However, the federation can be developed and executed without an SOM developed or documented on paper. For this reason, a brief discussion is presented on the FOM in the following paragraphs.

2.8.2.3.1 Federation Object Model (FOM)

It is important to note that the acronym "FOM" is often used to refer to a federation object model by modellers in the distributed simulation community. A federation object model simply contains information on classes and data types that are to be used in the distributed simulation federation. It contains defaults, but the federation developer can also define their custom types that get added to these defaults. The FOM is saved as a file that is referred to as a Federate Document (FDD). In the 1.3 HLA standard, the FDD was in the form of Lisp-like syntax but it later evolved into an XML file with the advent of the 1516 HLA standard.

The HLA standards prior to 2010 provided for the interface to keep track of and manage all classes and data types specified within the FOM. This was changed in 2010 to enhance efficiency through the interface loading and managing only those that are required by federates in the distributed simulation. A number of government agencies and other institutions that have created RTI software using the HLA standards have been discussed. These same agencies create OMT editors because distributed simulation systems cannot be developed and executed without an FOM. Examples of FOM editors include: COSYE FOM editor, Pitch Visual OMT, and Sim Gen OMT editor etc.

2.8.3 Developments and Research Studies Applying HLA

HLA has been used to create applications in different domains such as defense, space, air traffic management, energy, off-shore, railway and car industries, manufacturing, and health care. Literature shows that most of this work has been done in the US, Canada, Australia, Germany and in Korea. The bulk of this work has been centered on the military, with a decent portion also appearing in academic circles. Most of these military applications were aimed at creating virtual environments within which military personnel would train for combat (Dahmann, Fujimoto, &

Weatherly, 1997; U.S. Department of Defense, 1994). Details of such sample applications (for the United States) can be found in the U.S. Department of Defense (1994), and Dahmann et al. (1997). HLA has also been used for military purposes by the Department of Defense in Australia (Clark et al., 2000). More examples of work have been done in the military within Korea (Cox, 1998; Cho, 2003; Cho, Kim, & Youn, 2005; (Kim, 2002; Kim, Hong, & Kim, 2006).

There are also a number of projects that have been done in academia using distributed simulation approaches. For purposes of limiting this discussion, those done within the construction domain are presented here and briefly discussed for the benefit of the reader.

COSYE is an application programming interface which supports the development of large-scale distributed synthetic simulation environments. It is based on the High Level Architecture (IEEE 1516) standard for developing large-scale models (AbouRizk and Hague, 2009) and facilitates the creation of separate simulation components (also known as federates) and their integration into a single simulation system (known as a federation) during execution.

The reader should quickly note that all the systems discussed were developed in the COnstruction SYnthetic Environment (COSYE) environment within the Construction Engineering and Management program at the University of Alberta. This is because a distributed simulation framework – COSYE, has been developed there based off of the HLA and has extensively been put to use within the various research activities there.

2.8.4 Recent Research Activity that Applied COSYE

The HLA has been extensively used at the University of Alberta, Hole School of Construction Engineering and Management, to create simulation games for educational purposes. In all these cases, the COSYE framework – based off of the HLA, and Simphony were used as the development environments. All these developments were done as part of PhD theses that students undertook at the time. Each of these is briefly summarized in the following paragraphs.

In 2010, Taghaddos was able to successfully implement his generic resource allocation framework for construction using a distributed simulation approach. He made use of COSYE, an HLA distribution simulation framework, in developing his models. Case studies that involved the allocation of resources in module fabrication yards were presented in this work (Taghaddos, 2010).

In another study, Azimi et al. (2011) used the HLA concepts to develop a visualization platform that could be used for project control work in industrial projects (Azimi, Lee, AbouRizk, & Alvanchi, 2011). The COSYE framework was used to develop and integrate federates in a distributed fashion. Tekla, a 3D modeling software for structural steel, was used for visualizing the progress of work on industrial projects. Simphony, a discrete event simulation software, was used to perform the simulation of the construction operation. The Tekla and Simphony applications were each embedded within separate standalone federates. In addition to these two federates, Azimi (2011) also had four other federates in his system. They included: an as-built data federate, an as-planned data federate, a calendar federate, and an intelligent adjuster (artificial neural network) (Azimi, 2011).

Xie (2011) used a distributed simulation based approach to investigate the possibilities of improving project control in tunnel construction. Xie made use of COSYE to integrate Bayesian updating techniques with different simulation components (federates), which simulated the different parts of a tunnel (a shaft excavation, tunnel excavation and dirt removal). Xie was able to generate cost reports and construction schedules from her distributed simulation system.

In 2013, Moghani presented a distributed simulation system that she developed in Simphony and the COSYE environment for performing as-built documentation of tunnels built with tunnel boring machines (Moghani, 2013). Her simulation system was comprised of four autonomous simulation entities (federates). These included a planned process model (developed and executed within the Simphony environment), an as-built simulation controller that was a database of daily site information of the constructed tunnel, i.e., weather conditions, shifts, resource details and progress made. The simulation system would then generate an as-built process model along with outputs and reports. The process models were implemented within the Simphony simulation modeling environment. These models, along with the other components, were integrated into one system using a distributed simulation approach. Moghani used COSYE, a distributed Simulation framework based off of the HLA standards.

These studies demonstrated that with a distributed simulation approach, complex problems can be solved through the development of systems that have diverse features from numerous technologies and applications, all put together in one synthetic environment.

2.8.5 Simulation Games Developed using COSYE

2.8.5.1 Bidding Game

Different versions of the bidding game have been developed since the release of SUPERBID (AbouRizk, 1993). The version discussed here is that developed within the COSYE framework (AbouRizk, Hague, Mohamed, & Robinson, 2010). It comprises six federates, i.e., distributed simulation components. These include: an administrator federate, the player federate, a virtual player federate, the market federate, a simulator federate, and a bank federate. At the beginning of a game session, the administrator federate creates and joins a federation. If a virtual player is needed, it is enabled. Player federates join the federation; each player represents a unique general contractor. A bank account is created for each player with an initial amount of money, randomly sampled from a statistical distribution. The market federate joins, creating an environment in which projects and sub-contractors exist. The player decides which projects to bid on, secures a bond, selects subcontractors, and submits a bid, which includes their profit margin. As the game advances, the project is awarded to the contractor that submitted the lowest bid. The winning contractor is the one who has the most money in their account at the end of the game. The performance of the player in each period is dependent on the quality of the chosen subcontractors, their past experience in building similar projects and the location of these projects relative to the contractor's location. Details of this game can be found in AbouRizk (1993) and AbouRizk et al. (2010).

2.8.5.2 Crane Game

The COSYE framework has been used at the University of Alberta, Hole School of Construction Engineering to teach students about distributed simulation technologies. As part of this training, students are expected to develop an application using this distributed simulation framework. During this course, the author developed a "Mobile crane lift planning game." The objective was to have a virtual environment that could be used to teach students about analyzing and planning heavy lift operations on congested sites using mobile cranes. The game that was developed was comprised of five modules (federates): a scenario-setup federate (Ekyalimpa & Fayyad, 2010), a player federate (Jangmi, Zhang, & Saba, 2010), an operations simulator (Gonzales et al., 2010), visualization federate, and an emissions federate.

The game assumed an industrial construction site in which modules are lifted into place using mobile cranes. Modules arriving from a hypothetical assembly yard are lifted into place, or transferred to storage depending on mobile crane availability at the site at the time of their arrival. The game provided for a finite number of mobile cranes (set by the game administrator) with stipulated lift capacities from which the player could choose from. The game also provided for a possibility of mobile cranes to move from one location to another to complete a lift depending on the prevailing site conditions at the time of that lift. A lift plan would then be generated by player who would be interfacing with the lift federate and then passed onto an operations simulator that executes it within a simulation environment. The game generates vital statistics such as crane utilization, waiting times and overall duration. This is done in cycles (modules arrive, lift plan generated and lift plan executed) until the game session times out.

2.8.5.3 Tunneling Game

The tunneling game was built from an existing tunneling distributed simulation federation, initially developed to support planning and analysis of tunnels. Incorporating gaming features into the federation was possible because the HLA and COSYE facilitate extensibility, while maintaining inter-operability and reusability. In this development, one federate was developed from scratch to host a number of gaming features: the user interface, reporting facilities and scenario generator. The game creates an instance of a tunnel (whose attributes are read from a database), which the students being assessed are expected to construct. Attributes of the tunnel (length, depth, soil conditions, diameter, budget and schedule) are then availed to the player. The database also contains a list of different tunnel scenarios and resources required to execute the project. In this game, resource options are made available to the players (sizes of muck carts, excavation rate and failure rate of TBMs). Each has a different cost associated with it. At the beginning of the game, players plan for the rate to perform work and resources to be assigned (muck carts, TBM and crews). Each play period, the simulator takes this plan and generates results (money spent, actual time taken and liner distance advanced). If the player isn't content with their performance in a previous period, they change their plan. At the end of the game, players are ranked based on performance using earned analysis. Details of this game can be found within a conference paper published by the developers (Ekyalimpa, Al-Jibouri, Mohamed, & AbouRizk, 2011).

To summarize, all these applications or simulation systems were developed in-line with the High Level Architecture guidelines to behave as intelligent federates (agents) within the synthetic simulation environment (COSYE). They also demonstrate that COSYE (and the HLA) can be effectively used as a tool to develop distributed simulation systems for analyzing complex, large-scale problems. Another lesson learned is that COSYE can be used as a means to bridge the gap between different applications in cases where a system to be developed has to run off of multiple applications as a result of working around pooling different required application features, or because it is dictated by the fashion in which the application is to be deployed (e.g., in simulation games).

2.9 SUMMARY OF CHAPTER TWO

A comprehensive review of literature was successfully conducted and presented in this thesis chapter. Topics covered include the fundamental principles that underlie the concept of competitiveness and performance management. The chapter also covered the different performance management systems that are in use at construction organizations and those developed from research activity on the subject, but that may not be in use at most construction companies. Most of the popular performance management systems in use at the organizations are from the category of traditional performance management systems. The other performance management systems reviewed applied more advanced techniques such as DEA, ANN, and simulation, to generate more accurate performance results.

Although the simulation-based methods were meant to address the shortfalls of the traditional performance management systems, some of these were never adequately addressed. This can be attributed to two reasons. These include:

- In some cases, the developers of these systems left out constructs that are relevant for representing the operations of a typical construction company in a realistic fashion. For example, the dynamics and uncertainty that surround the acquisition of work in a competitive environment was never explicitly represented in most of these systems.

- In other cases, monolithic simulation modeling paradigms were used for developing these simulation models. Using any monolithic modeling approach in isolation is not sufficient for this type of problem (representing a significant portion of company operations in a computer simulation model) because of its complexity and large scale.

The above two challenges were overcome in this thesis through the abstraction of a larger scope of the system that realistically represents the operations of a typical construction company. An agent-based model that was coupled with discrete event modeling approaches was adopted in the developments of the model in order to cope with the complexity and scale issues related to this problem.

This chapter also presented an overview of the different scientific methods that exist and could potentially be used in the analysis of systems. The chapter was then narrowed down to methods that are based on computer simulation. This discussion of computer simulation was commenced by an introduction to monolithic and distributed computing technologies. This was then followed by a discussion of the different computer simulation modeling paradigms, i.e., Discrete Event Simulation, System Dynamics, and Agent-Based Modeling.

The Chapter reviewed literature related to the relevant performance measures that can predict performance in the construction domain. These studies were found to report on a wide range of measures, some of which were presented in a hierarchical fashion. Most of these factors were validated through empirical questionnaire studies and informal interviews. These questionnaires and interviews served two purposes: (1) gathering information on the perceptions of management on performance issues and (2) confirming the validity of the factors considered and their influence on performance (Dess & Robinson, 1984; Kale & Arditi, 2002; Kale & Arditi, 2003; Phua, 2007). It was also observed that a significant number of these measures were highlighted in the majority of these studies. See some of the following publications for details of this: Takim & Akintoye (2002), Yang, Yeung, Chan, Chiang, & Chan (2010), Constructing Excellence in the Built Environment (2012), Kagioglou, Cooper, & Aouad (2001), The Construction Users Roundtable (2005), Bassioni, Price, & Hassan (2004). The list of measures utilized in this thesis study were based off of these and were limited in number to ensure that the developments did not experience problems arising from scope creep. Based on this literature review, the seven performance measures adopted in this study included:

- Cost performance,
- Schedule performance,
- Safety performance,
- Quality performance,
- Market share,
- Tendering success and
- Production efficiency.

These measures were strategically selected such that some indicate performance at an operations level (e.g., production efficiency, quality and safety), while others reflect the profitability of the organization (e.g., market share, tendering success, cost and schedule slippage).

CHAPTER THREE—METHODOLOGY

3.0 INTRODUCTION TO CHAPTER THREE

This chapter on methodology discusses details on the performance measures considered in this thesis, and how these were identified and modeled. The chapter also highlights the knowledge and skills needed to implement the required methods in the development of the model. The first section of the chapter is dedicated to discussing performance measures used in the model and how those measures were selected. This is followed by a discussion of an analytical hierarchical process which was used for assigning relative importance weights to performance measures being tracked in the simulation. This method was also used to model the influence that factors have on performance measures. Later on, details of how the simulation-based performance management application was developed and validated are presented.

3.1 IDENTIFICATION OF PERFORMANCE MEASURES

There are numerous ways that companies within the construction industry are currently tracking their performance. In an attempt to narrow this list to a number that could be carried through development phase, a holistic approach was adopted. This involved two methods:

- A *comprehensive review of literature* to establish the most frequently reported performance measures tracked by construction companies. Peer reviewed publications such as journal papers, conference papers were reviewed. Also, other sources such as government reports and reports published by associations within the construction industry were reviewed.

- A *questionnaire survey* of construction companies was conducted to establish the performance measures that are consistently used. The survey was narrowed to companies operating within the heavy civil and industrial sector of the construction industry. The sample set included all companies that participate in the industrial research chair within which this research was conducted. All other Alberta-based companies other than these that were confirmed to belong to the heavy civil and industrial domains were also targeted. All the companies considered in the sample had to meet the minimum requirement of possessing atleast *25 employees* and a minimum turnover of *50 million US dollars*. The intention was to conduct a census type questionnaire survey for that cluster of construction companies. An ethics approval for this questionnaire survey was approved by the Research and Ethics Office at the University of Alberta. Subsequently, questionnaires and consent letters were administered and responses collected. A total of 68 companies were targeted and 22 of these responded. The questionnaire and consent letter used in this study are included in the appendices.

An analysis was conducted that involved ranking the performance measures that were consistently observed using a frequency criteria. This process resulted in a total of seven performance measures namely; tendering success, market share, production efficiency, quality rating, safety rating, cost slippage, and schedule slippage. These were then carried forward and utilized in the development of the simulation application.

3.2 SIMULATION APPLICATION DEVELOPMENT

The development of the simulation application intended to model company competitiveness required certain knowledge, skills and tools. Each of these is discussed in detail within the following section.

3.2.1 Knowledge and Skill set required for Development

A specific knowledge and skillset were required prior to embarking on development work for this thesis. These were required in the following areas:

- Computer Simulation,
- Computer Programming and
- Analysis and design of construction operations and business processes.

A broad, solid knowledge base of theoretical and modeling concepts in simulation proved to be essential. This knowledge was required in two main areas of simulation i.e., Discrete Event Simulation (DES) and Agent-Based Modeling (ABM). Also, knowledge of the High Level Architecture (HLA), and an understanding of the creation and behavior of large-scale distributed simulation systems were required for the developments in this thesis. Knowledge of computer programming especially Object Oriented Programming (OOP), design patterns etc. proved to be vital in the development work. Languages such as CSharp (C#) and JAVA were very useful in the application development. Last but not least, a good understanding of business processes in the construction industry also came in handy especially when abstracting and designing approaches to represent specific phenomena that exist within the construction domain, in a realistic fashion.

All the above were acquired in the course of completing mandatory classes for the PhD program, working on specific projects in collaboration with industry or on in-house projects and during the implementation of specific academic tasks assigned by my supervisor.

3.2.2 Design Aides Used in Simulation Model Development

Creation of concepts, designs and specifications of constructs to be represented were an integral part of the model development process. The process of putting ideas abstracted of a system on paper clarified a lot of issues which would otherwise have resulted in an invalid or unreliable model. A number of design aides exist within the simulation domain, computer science and software engineering which were applied in this process. These include:

- Flow charts,
- Activity diagrams
- State charts,
- Sequence diagrams and
- Block diagrams.

Combinations of these were used throughout the development work. Flow charts were used to represent the flow logic for processes analyzed using discrete event simulation approaches. State charts were used in designing the behavior of agents in development of the agent-based models. Activity diagrams were used to detail concurrent behaviors of agents. Sequence diagrams were used for specifying the communication protocols between agents in agent-based models and between federates in the developed distributed simulation system. Block diagrams showed the objects that exist within a model and the relationship between those objects. Block diagrams were useful in the development of both discrete event and agent-based models.

3.2.3 Software and Frameworks used in the Development Work

The development work in this thesis heavily relied on numeric-based approaches. This was because of the highly complex and dynamic nature of the problem that was being solved. A number of statistical distributions and simulation models were used collectively to produce the model that mimics the operations of a typical construction contractor company for purposes of performance management.

Simulation is a very powerful approach for gaining insights into how systems or processes that are characterized by uncertainty evolve over time. Simulation was adopted as a method for analyzing the company competitiveness problem because it is difficult to know precisely beforehand how many projects will require execution within a specific period in the future. It is also difficult to know the level of competition that any company interested in acquiring these projects will be engaged in and ultimately the volume of work that it will acquire.

The simulation paradigms used included Discrete Event Simulation (DES), and Agent-Based Modeling (ABM). Also, a distributed simulation modeling approach was adopted as a result of the large-scale and complex nature of the problem. Most statistical distributions used were continuous in nature. The different software tools used in the implementation of these simulation paradigms are discussed individually in the following sections.

3.2.3.1 Simphony Simulation System

Simphony is a robust extensible simulation system that currently provides for discrete event and continuous simulation modeling paradigms. Simphony was created in the late 90s by AbouRizk and Hajjar (Hajjar & AbouRizk, 1999; Hajjar & AbouRizk, 2002).The system doubles as both a software/application and a framework because it facilitates the development of models and the development of tools that can be used to build models. In this thesis, a number of services were utilized from the Simphony simulation system. These included:

- The Application Programming Interface (API) – Core services (e.g., simulation, resources, waiting files, statistics), Modeling services, Math library.
- Simphony Template Development Services – for development of a special purpose template.

- The modeling Interface – for development of the model that was embedded in the Windows form application.
- General template.

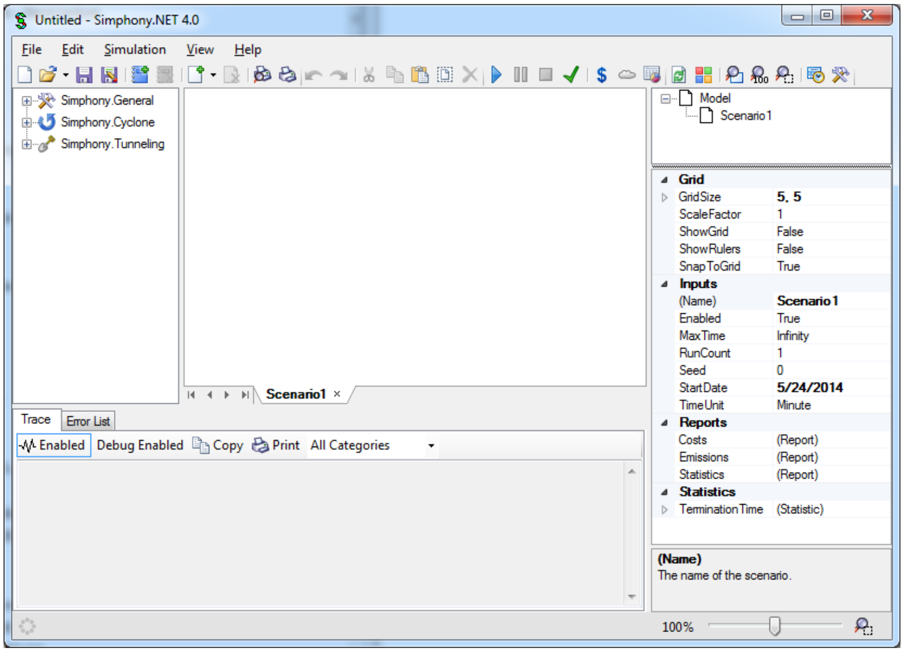


Figure 3.0: A Screen Shot Showing the UI of the Simphony Simulation System

A screen shot showing the Simphony simulation system's interface is shown in Figure 3.0.

3.2.3.2 AnyLogic Simulation System

AnyLogic is an easy-to-use simulation system. It also doubles as software and a framework, given that it also supports both model development and the development of tools that can be used to build models. The AnyLogic simulation system is developed and maintained by xjTechnologies. AnyLogic supports all three simulation modeling paradigms, namely, discrete event simulation, system dynamics and agent-based modeling.

AnyLogic was used as a standalone federate in the distributed simulation. Development work in this thesis greatly relied on its agent-based modeling services to abstract and represent the operations that take place within the construction industry. Details modeled included:

- The entry of projects into the market.
- Competition for and award of projects.

- The execution of projects by competitors.

A screen shot of the interface of the AnyLogic simulation system is shown in Figure 3.1.

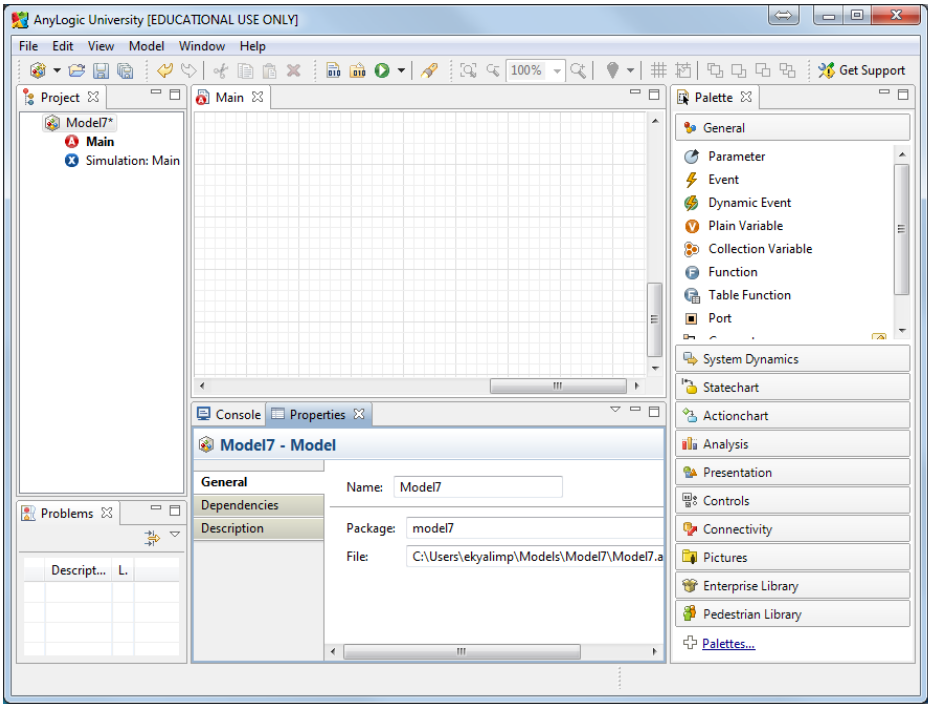


Figure 3.1: A Screen Shot Showing the UI of the AnyLogic Simulation System

3.2.3.3 COSYE

It was envisaged that the company performance management model would be large scale. As a result, it was modularized to simplify its development and implementation. This meant that it was to be developed and implemented as a distributed simulation model containing federates that represent the modules in the system.

For the development of federates and the federation (distributed simulation system), a simulation framework based on the High Level Architecture (HLA), was used. COSYE – COnstruction Synthetic Environment is one such simulation framework. COSYE is a synthetic simulation environment developed by AbouRizk and Hague at the University of Alberta, Hole School of Construction Engineering and Management (2009). COSYE was therefore used for this purpose, given that it is free for academic use and is developed and maintained by a team put together by my supervisor, which provided the necessary development support. In order to create this distributed simulation system, an object model (FOM – Federate Object Model) needed to be created. This was done using an Object Model Template (OMT) editor. Each federate also had to be developed using the appropriate COSYE HLA Application Programming Interfaces (APIs). These are discussed in more detail in the following sub-sections.

3.2.3.3.1 Object Model Template (OMT) Editor

The OMT editor is used to create the Federate Object Model (FOM). The FOM is an xml document that contains metadata on all the objects classes, their attributes, interaction classes, their parameters, and data types to be used to define the attributes and parameters for the distributed simulation (federation). This document allows for the sharing of information among federates. The OMT editor allows the developer to specify the order type to be used for each interaction or object instance update in information exchange. COSYE has an OMT editor that can be used as a plug-in to Visual Studio 2010 (see Figure 3.2). The screen shot below shows this OMT editor being used to create the FOM for this thesis work.

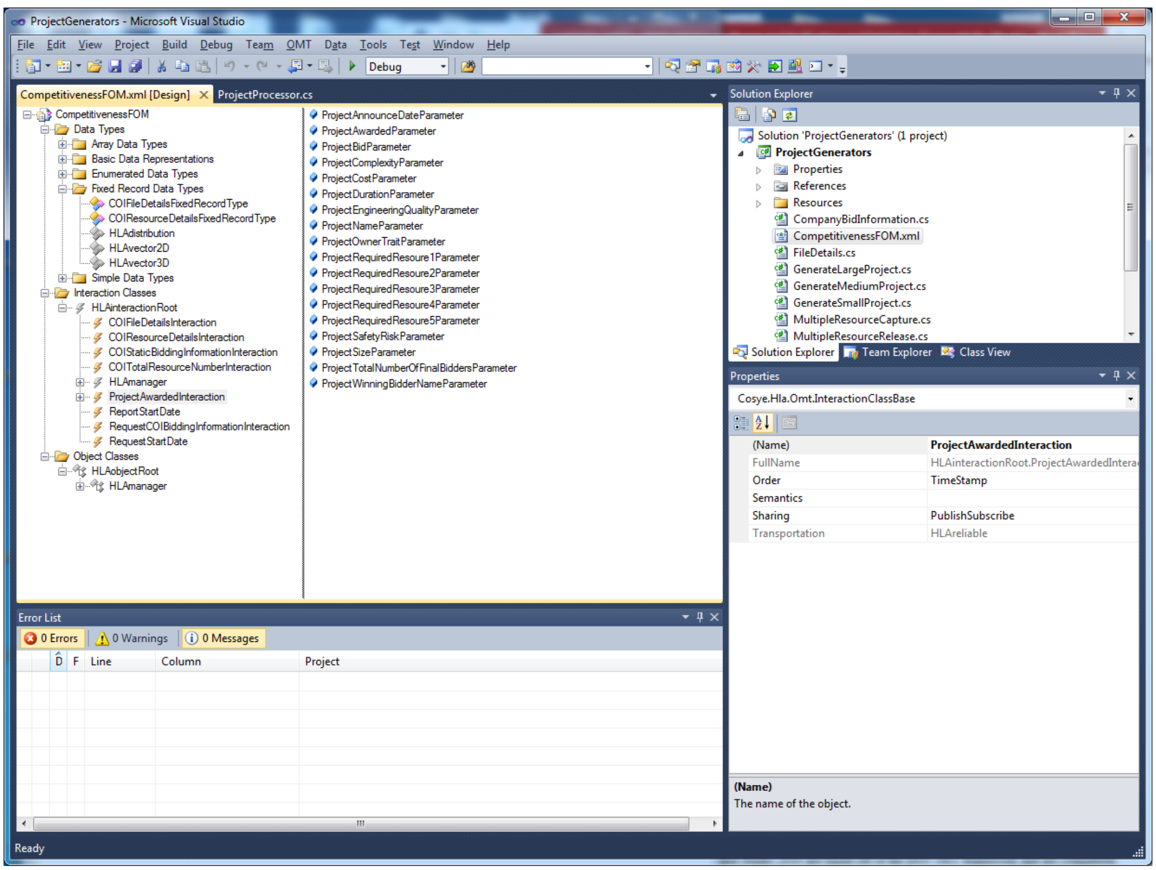


Figure 3.2: A Screen Shot Showing the COSYE OMT Editor Plug-in in Visual Studio

At present, another OMT editor is under development in COSYE that will be independent of Visual Studio. This is because new Visual Studio versions are regularly released, which are not always compatible with the COSYE OMT editor. However, the OMT editor that is compatible with Visual Studio 2010 is still being supported.

3.2.3.3.2 Federate Development

The development of HLA federates requires access to Application Programming Interfaces (APIs) that contain the classes and interfaces that facilitate the development of federate and distributed simulation components. Fortunately, COSYE contains three APIs that facilitate the development of HLA compliant federates. They include:

- A DOT NET HLA API,
- A JAVA HLA API and
- Python HLA API.

Only the first two APIs were used in the development work in this thesis. Each of these is discussed briefly in the following sections.

DOT NET COSYE-HLA API

The DOT NET version of the COSYE-HLA API was mainly applied in developing federates within applications or application development environments that can reference DOT NET APIs or Dynamic Link Libraries (DLLs). This was the case when developing the Simphony federate using Simphony simulation system and Visual Studio 2010. Simphony simulation system and Visual Studio 2010 are based off of the DOT NET framework and are compatible with this API. The DOT NET COSYE-HLA API is in the form of a DLL.

JAVA COSYE-HLA API

AnyLogic is a simulation system that runs off of the JAVA runtime environment. In the development work, there was a need to make use of AnyLogic in the creation of a federate. This was because the AnyLogic simulation system provides an easy to use agent-based modeling paradigm. In order to achieve this, a JAVA COSYE-HLA API was used within AnyLogic and the federate created. The JAVA COSYE-HLA API exists as a .jar file.

3.2.3.4 Visual Studio

Visual Studio was used in the development of the Windows form application. Visual Studio is the DOT NET software development environment often used by professional programmers in their development work. Visual Studio was used to create the user interface in the Windows form application which served as a means for the user to enter model inputs prior to simulation and view outputs after simulation. Visual Studio also facilitated the process of embedding the Simphony discrete event simulation model that represented the operations at the company of interest. It further facilitated the deserialization of this discrete event model and its simulation when the application was run. The version of Visual Studio used in these developments was 2010. CSharp was the programming language used within this development environment.

3.2.4 Dynamic Link Libraries

A dynamic link library (DLL) is a collection of resources that are intended to be shared by multiple programs. The sharing is made possible by each programming referencing (getting linked to) the library. The resources may be one or a combination of the following:

- Icons and images,
- Controls,
- Text files and
- Classes that contain data and methods (functions and sub-routines).

A DLL is a good way to achieve inter-operability amongst multiple programs within the DOT NET framework. In developing the simulation-based system for performance management, DLLs were used for various purposes. These included:

- DLLs are used typically to wrap simulation frameworks that were used in the development work. The COSYE Framework APIs and Simphony APIs are packaged as DLLs, which were referenced and used in development work.
- Special purpose template development in Simphony makes use of the concept of DLLs. This was utilized in the development of the performance management system.

- Custom classes were used for wrapping algorithms that were required for computation within the program. An example is the Eigen Value computation algorithm, which was used in the pair-wise preference calculations.

The development work in this thesis made extensive use of both existing and custom developed DLLs. For example, the Simphony and COSYE APIs were imported and utilized as DLLs within the thesis application. Also, the Eigen Value and Vector computation algorithm was wrapped in a DLL then imported and used in computations.

3.3 METHODS USED IN MODEL VERIFICATION

Verification of a model is the process of confirming that the model does whatever it was designed to do. Verification becomes important at the point of translating concept models into computer models and applications. In this thesis study, there were two main aspects that needed to be verified. These included:

- The software environments and frameworks that were used in the development work.
- The actual developments (program code and models) produced using the above software and frameworks.

Fortunately, extensive unit tests had been performed on some of the simulation software and development frameworks, e.g., Simphony and COSYE, prior to the commencement of this thesis work. NUnit was the software used for the testing. I was fortunate to be involved in some aspects of this testing because the development team supports research activities spear-headed by my supervisor, Dr.Simaan AbouRizk. There were no significant cases involving flaws in the software and frameworks reported during this exercise. The minor issues identified (mainly improvements to the functionality of the software) were fixed and passed all tests. Consequently, Simphony and the COSYE framework are considered to be reliable given that no major flaws have been identified during this testing phase and during their use by students and practitioners in industry.

It is believed that a similar testing process has been applied in the development of the AnyLogic simulation system. This could not be confirmed given that the software is produced and maintained by a commercial enterprise and such information is considered proprietary.

Nonetheless, it can be argued that this assumption holds because there have not been complaints of major flaws in this simulation system published on the Blog for AnyLogic users. Moreover, no flaws have been encountered in the use of this software in this thesis and other simulation projects.

The same argument (in the previous paragraph), can be extended to the Visual Studio software development environment. Visual Studio is a software development environment produced and maintained by Microsoft, a reputable software firm known for creating reliable software. This software is widely used by professional software developers that seem comfortable with it, at least for the features that we made use of in our developments within this thesis.

Verification was also done to confirm that the models and applications developed using these software and frameworks were reliable. The reliability of the distributed simulation framework and related APIs were tested using a combination of these approaches:

- A test federate that exist within COSYE
- Unit tests

Other techniques used to verify developments included the use of the following:

- Message boxes
- Trace logs
- Breakpoints
- Data visualization

3.4 METHODS USED IN MODEL VALIDATION

Validation was extremely crucial given that the model was to be put to meaningful use within academia or industry. The process was therefore handled a systematic way. A considerable amount of research has been published on how to validate simulation models (Sargent, 1998; Martinez, 2009; Phelps & Horman, 2009; Lucko & Rojas, 2009; Leicht, Hunter, Saluja, & Messner, 2009). If time and other resources are in abundance, one may opt to apply all these and other validation techniques in their validation work.

However, it is not always possible to apply all these validation techniques. For example, data driven validation approaches may be hampered by one or a combination of the following:

- Data that closely maps onto model inputs and outputs may not exist.
- The data may be challenging to collect—it may take too long to collect, or may be proprietary.
- There may be no data available.

This was the case in this study. As a result, a number of other validation techniques were explored. These included:

- Validation of model designs and specifications—conceptual models and other design aides (content and construct validation).
- Face validation—using domain expert feedback in an attempt to identify flaws in the model.
- Sensitivity analysis—through experimentation of extreme and typical cases.

3.4.1 Validation of Model Designs and Specifications

According to Brains et al. (2011), validity, in the arts and sciences, is the extent to which a concept, result, conclusion or measurement is well-founded and corresponds accurately to the real world. The authors further noted that validation of model designs is the first step towards ensuring that a model is valid. If done well, it is a sure way of creating a valid system, application, or model and could save a lot of time and frustration at the end of the development process.

3.4.2 Validation of Simulation Models

Simulation studies typically commence with a process that involves the abstraction and representation of specific constructs from a real world system on a computer. The precision with which a modeller carries out this phase of the simulation modeling process determines whether their model is valid or not.

3.4.2.1 Content Validation

Content validity is defined as a non-statistical type of validity that involves *"the systematic examination of a model or experimental test content to determine whether it covers a representative sample of the domain to be measured"* (Anastasi & Urbina, 1997). According to

Anastasi & Urbina (1997), a model or test experiment has content validity built into it by careful selection of which items to include.

The first aspect dealt with when abstracting a specific phenomenon, process or construct of a real world system pertains to the fixation of boundaries within which the abstraction is to be done. This step curves out the constructs of the system that will be considered in the modeling process and those that will be left out. In this thesis, this process was guided by the objectives of the study and the underlying assumptions. It was carried out carefully to ensure that the resulting model was valid with respect to content. Figure 3.3 shows a schematic layout of model boundaries drawn to include specific constructs for an arbitrary real world system.

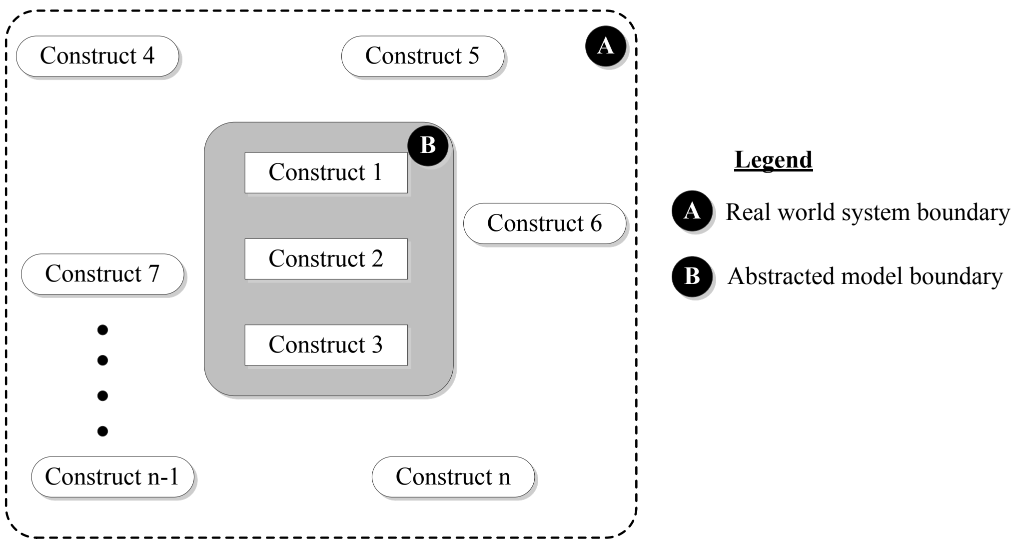


Figure 3.3: A Schematic Showing Fixation of Model Boundaries Relative to Real-World System Boundaries

3.4.2.2 Construct Validation

After the boundaries had been drawn and it had been confirmed that they included all the relevant constructs. The next step involved accurately mapping these real world constructs to an appropriate simulation modeling paradigm. This gives rise to a concept of construct validity.

Construct validity is "the degree to which a test measures what it claims, or purports, to be measuring" (Brown, 1996; Cronbach & Meehl, 1955; Polit & Beck, 2012). Construct validity evaluates the appropriateness of the method used to represent and analyse an abstracted construct. Confirming construct validity requires a domain expert in the analysis and design of systems.

3.4.2.3 Methods for Achieving and Assessing Content and Construct Validity in Simulation Models

A number of methods were adopted to ensure the content and construct validity of the simulation model developed in this thesis. These included:

- Acquisition of the necessary simulation and construction knowledge and skills through courses, readings and projects undertaken.
- Representing abstracted systems on paper as designs. A number of design aides were used for this such as concept schematic layouts, flow charts, activity diagrams, state charts, sequence diagrams and block diagrams.
- Scrutiny of design concepts by academic supervisor, other professors within the construction research group, technical support staff, and colleagues. From time to time, they pointed out possible improvements to the model.

3.4.2.4 Face Validation

Face validity relates to whether a model or test experiment appears to be a good or inaccurate representation of the constructs of interest. This judgment is made on the "face" of the model, thus it can also be judged by the amateur. Face validation was assessed by domain experts within the construction industry in Alberta, Canada. A few of these experts had decent knowledge of computer simulation, but the rest were novices in simulation modeling. Nonetheless, these experts provided feedback on the validity of the model based on its face value. The experts that had knowledge of simulation participated in assessing content constructs and face validity.

3.4.2.5 Sensitivity Analysis

This section discusses the different types of sensitivity analysis. Sensitivity analysis is an experimentation process that involves generation of multiple scenarios that could be used to investigate the behavior of a model (Chinneck, 2000; Taylor, 2009). It permits an analyst to assess the impact that changes in a specific parameter will have on the model's outcome (results). The following sections discuss the different types of sensitivity analysis that can be performed on models and the details provided of those used in this thesis.

3.4.2.5.1 One-Way Sensitivity Analysis

In this type of analysis, only one parameter is changed at a time, hence the name one-way sensitivity analysis. The value of a selected parameter in the model is varied by a given amount and the impact that this change has on the model's results is assessed. By first increasing or decreasing a selected parameter by a pre-defined percentage (e.g., 20%), one can generate results from which they can quantify the impact that these changes have on the model output variables. These can then be summarized on charts such as a tornado diagram. This process can then be repeated for all or select key input variables one after the other resulting in tornado diagrams for each parameter. The tornado diagram can then be used to reveal the parameters that have the greatest influence on the model results.

Another form of one-way sensitivity analysis involves varying a parameter to the highest and lowest possible values. It is not always obvious what the highest and lowest possible values of a parameter might be. However, according to Taylor (2009), there are a number of ways of defining these. They include:

- Obtaining the confidence intervals of the data (if it exists) for that parameter and making use of the boundaries.
- Through readings in the literature, to identify these extreme values if they have been highlighted in the literature.

Once extreme values are identified, the analyst can assess the impact of a range of values that within these boundaries for the parameter, on the output of a model. Then a simple graph plotting the main model results against each possible input value can be generated. This type of analysis can also be used to judge the threshold at which the main conclusions of a model might change, if at all one exists.

3.4.2.5.2 Multi-Way Sensitivity Analysis

This type of sensitivity analysis is carried out when the interest is studying the model behavior resulting from simultaneously changing two or more different parameters. This type of analysis can quickly become very complex to perform especially as the number of parameters to be investigated increases. As a way of overcoming this complexity, Taylor (2009) suggested performing such an analysis for two scenarios. One scenario would involve setting all model parameters being varied to their high values. Another would be to set these parameters to their low values. Another viable scenario that can be investigated would involve setting these parameters to their typical values. In this thesis, this type of analysis was not performed because of the high number of input parameters that exist in the model and the possible complexity that would result.

3.4.2.5.3 Probabilistic Sensitivity Analysis

When a model is built using statistical distributions and other probabilistic parameters, it qualifies as one with which probabilistic sensitivity analysis can be performed. In order to run such an analysis, the seed used for random number generation should not be fixed so that every time a unique random number is sampled from the statistical distributions. The model should then be run multiple times and the results of key output variables recorded. A scatter diagram was then generated using results of appropriately selected output variables. The spread of the scatter points on the graph was then used to make deductions about the level of confidence that should be placed in the developed application. Higher confidence levels will be built in models that have a tighter spread compared to those that have a large spread of results. It is possible to have two models (having identical average results, but different confidence levels or reliability) (Taylor, 2009).

A combined approach that utilizes multi-way sensitivity analysis and probabilistic sensitivity analysis was utilized in the experimentation work done with the model.

3.5 SUMMARY TO THE METHODOLOGY CHAPTER

Chapter three successfully discussed the different tools, approaches, and design aides utilized in this study. Numeric approaches were extensively used, such as statistical distributions, and computer simulation. The computer simulation systems made use of are also presented in the chapter. A background is also presented on validation techniques currently present in the literature, and details presented on the techniques that were applied in this study. Details on verification work that was done are also presented.

CHAPTER FOUR—DISTRIBUTED SIMULATION FEDERATION FOR CONTRACTOR PERFORMANCE MANAGEMENT

4.0 INTRODUCTION TO CHAPTER FOUR

This chapter summarizes the topology of the distributed simulation federation used to model contractor performance in the construction industry. There are a number of reasons as to why simulation was deemed the most appropriate as a methodology for tackling the company performance management problem. They include:

- *It is a pragmatic approach*: In most cases, real world systems are running 24/7 and cannot be disrupted for purposes of experimentation as a means for decision support. In such situations, simulation becomes a viable approach.
- *Simulation provides a risk free environment*: Risks associated with safety, cost, and time loss can be averted by not experimenting with the real system
- *Simulation facilitates superior decision making*: This is because an analyst can experiment with many scenarios, an opportunity that they would never have when dealing with the real system

Details of the distributed simulation federation are presented in this chapter. An introduction to the two federates that exist within the federation is also made. However, discussions on the design and implementation of the behavioral aspects of each federate are deferred to the next two chapters. The simulation modeling paradigm used in their development and the simulation system within which they were developed are also presented. The chapter also discusses concepts of federation management and how they applied to the federation that was developed in this study. Explanations are provided on the data exchange protocols adopted in the developed federation along with the time management schemes. The chapter is finalized with an explanation as to why ownership management was not necessary in the developed federation and the system requirements necessary to run and get a result from the federation. In the course of discussions in this chapter, the reader will encounter the term *company of interest* (COI). This refers to the company that is being tracked and analyzed by the modeller in the simulation. All other companies can be regarded as competitors to this company.

4.1 CONCEPT MODEL OF THE PERFORMANCE MANAGEMENT FEDERATION

Prior to developing any simulation application, it is necessary to create a concept model of the system in a fashion that maps its inputs, process and response. This was done for the performance management application developed in this thesis. Figure 4.1 summarizes the layout of the concept model that was developed.

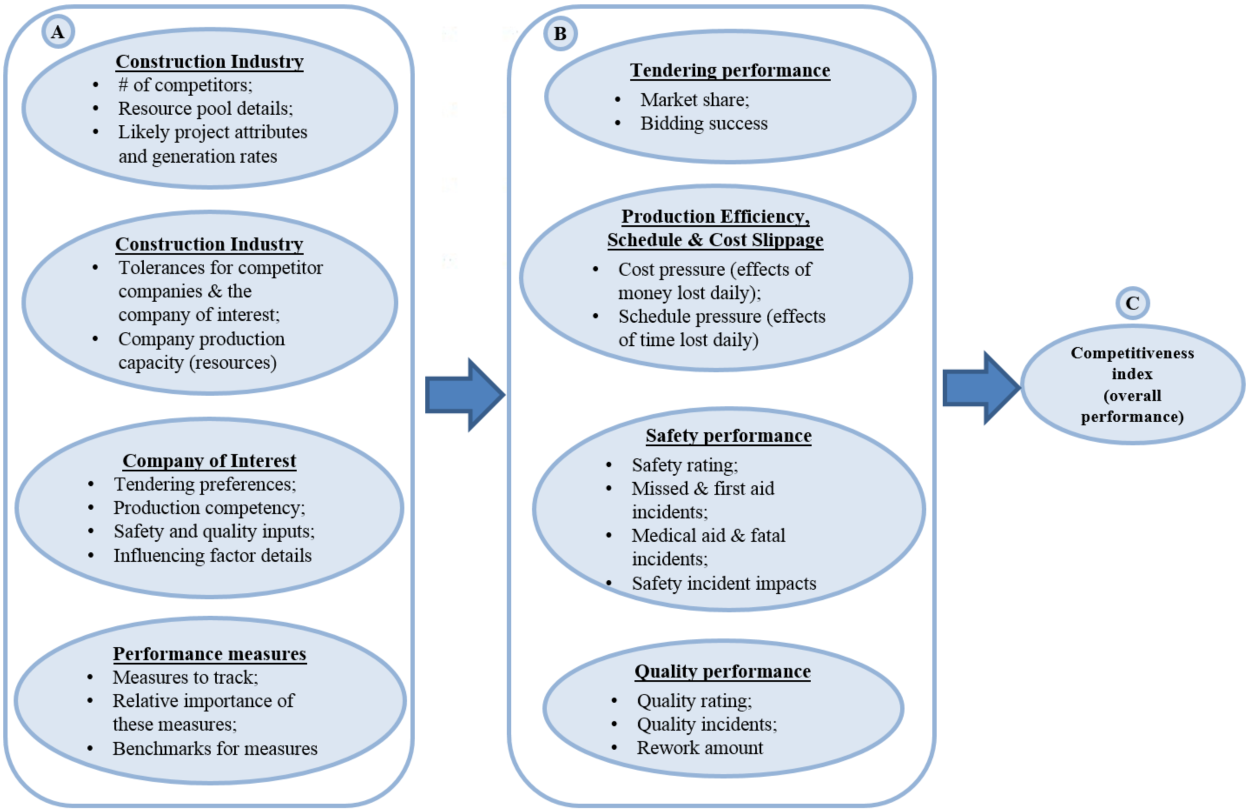


Figure 4.0: Conceptual Model Illustrating the Inputs, Process and Outputs for the Performance Management Simulation Application

Component "A" represents the inputs, "B" represents the process and inter-mediate outputs generated as simulation progresses and "C" represents the output. The inputs define the competition, constraints and projects that are envisaged within the construction industry. The other set of variables relate to the performance measures that would be tracked and used for assessing the competitiveness of the company of interest. The last set defines the competencies that exist at the company of interest.

The inter-mediate variables represent the metrics used to track the various performance measures at an operational level. Parts of these represent how well the company of interest performs in acquiring projects through a competitive process. The others represent the performance of the company of interest as it executes work that it was awarded. The performance at an operational level was setup in such a way that it would be dynamically affected by the competencies that exist at the company of interest and the type of work (projects) that the company executes.

This concept layout was subsequently mapped onto two model components i.e., a Tendering module and a company of interest module. The detailed discussion of these was differed to subsequent sections in this chapter. Component "A" and "B" were utilized in both modules. Result in component "C" was reported in the company of interest module.

Another concept model (shown in Figure 4.1) was created which explicitly illustrates the modules that were curved out of the concept layout presented in Figure 4.0.

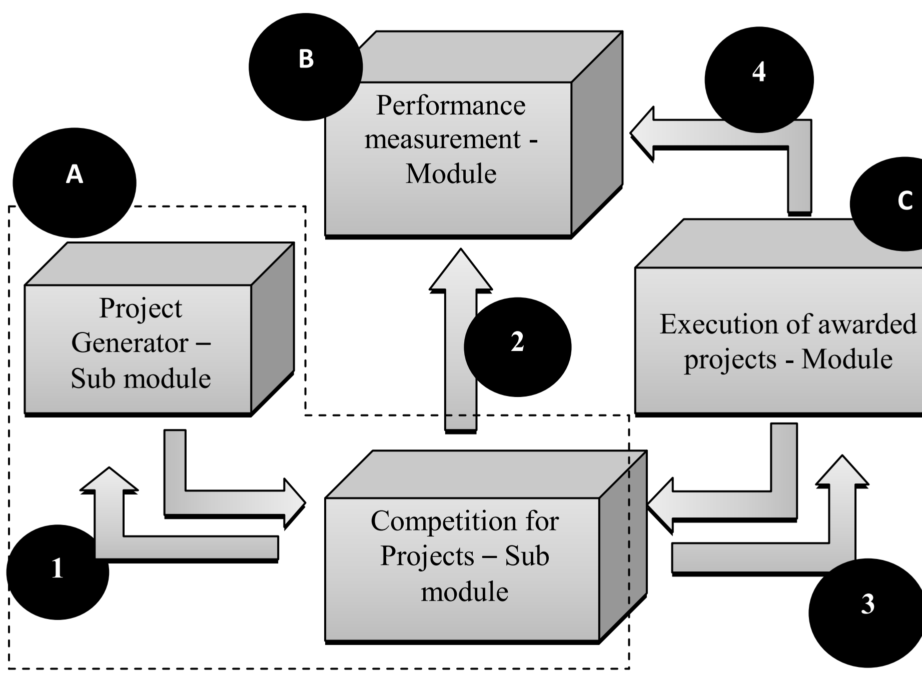


Figure 4.1: Architecture of Simulation-Based Performance System Components/Modules

The modules labelled "A", and "B + C" were setup to model different processes namely:

- The project creation process and competition for work amongst companies within a virtual environment (Module "A").

- The detailed process of work execution that leads to the generation of performance measures (Module "B+C").

In an attempt to simplify the development of such a system, a concept model was first created which maps out these two core business processes that contractor companies engage in. The schematic layout of the concept model is presented in Figure 4.1.

"A" represents a "Tendering" module. "B" and "C" represent the "Performance Measurement" and "Operations" modules, respectively. The "Tendering" module models project arrivals and competition for these projects by companies that are operating within a virtual construction industry. The "Operations" module processes projects awarded to the "company of interest." The "Performance Measurement" module collects observations on performance measures from all other modules and generates an overall performance rating for the company at the end of the simulation. The numbers 1-4 represent the communication that takes place between the modules during the simulation. "1" represents bid submissions and companies being notified of the winning bid. "2" represents the collection of statistics/observations on tendering performance as the simulation progresses. "3" represents communication between the "Operations" module of the "company of interest" to the modeller/user and the "Tendering" module. Information transferred includes data on prevailing conditions in the operations module at the time of a project arrival so that this can aide with the "company of interest's" bid/no-bid decision. Also, it represents notification of the "Operations" module of the projects that have been awarded to the "company of interest" and that need to be processed. "4" represents the collection of performance measures (e.g., quality, production efficiency, safety, cost slippage, schedule slippage, etc.) as the simulation advances.

Another figure is presented (Figure 4.2) that shows more details within each of the components and the type of interaction that exists between them. This figure also shows the simulation method intended to adopt for the implementation of each component and the integration of these components into a distributed simulation environment using a synthetic environment referred to as COSYE (AbouRizk and Hague 2009). The "Tendering" module was implemented using an agent-based approach while the "Operations" and "Performance" modules are implemented using a discrete event simulation approach.

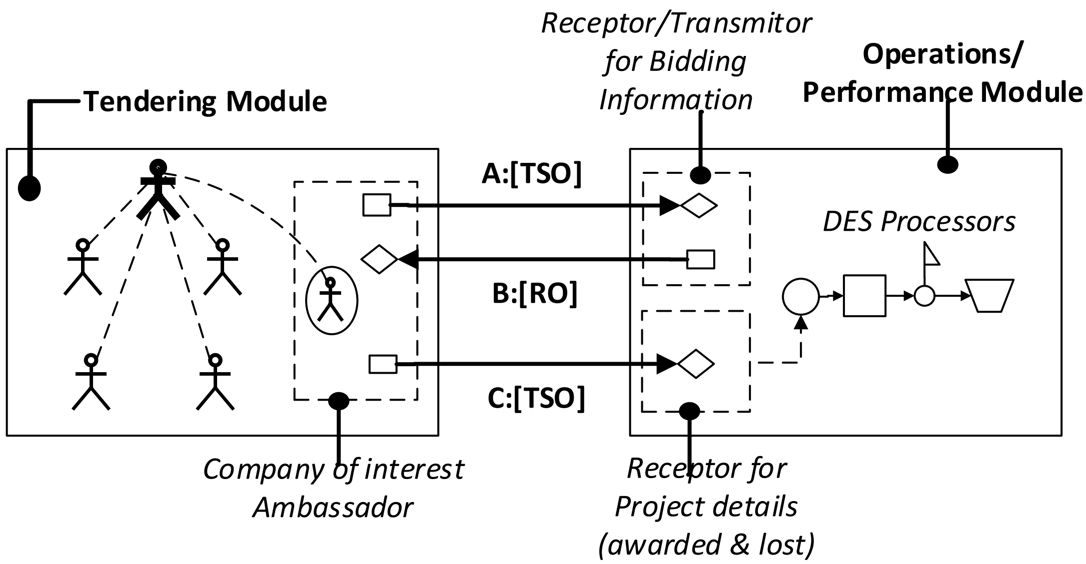


Figure 4.2: Schematic Layout Showing the Modeling Paradigms Used to Implement the Simulation-Based Performance System

4.2 DEVELOPMENT AND BEHAVIOR OF DISTRIBUTED SIMULATION COMPONENTS

The developments of the performance management simulation system were designed and implemented as a distributed simulation federation for two reasons. They include:

- *Acquisition of desired simulation functionality from different software:* The best use of the HLA and distributed simulation is in the integration of software with different functionality within a unified synthetic environment. It is not uncommon not to have functionality and features desired for use in a simulation within single software. Alternatively, the features required may exist in that software, but with some being advanced and robust, while others are not. In such a case, modellers tend to seek and adopt software that provides these features in a manner that suits their needs. They would then need to adopt a distributed simulation architecture if they would like to make use of all of these software tools in a seamless fashion.
- *Modularization of developments for convenience*: The simulation system developed for modeling performance management issues at a company was complex and large in scale. In order to simplify the development of such as system, components had to be modularized and

treated as such in the design and implementation, hence, resulting in a distributed simulation architecture being adopted.

Two components within the system were conveniently modularized such that one of them represents the constructs and dynamics that exist at an industry level and the other models those that exist at a company level (i.e., at the *company of interest*). Each component was then developed as a separate federate resulting in the federation comprising of only two federates. The first federate was referred to as a *tendering* or *AnyLogic federate* while the second was referred to as an *operations and performance management federate*. The second federate was also referred to as a *company of interest* or *Simphony federate*.

The *tendering* or *AnyLogic federate* was responsible for modeling phenomena that take place at the construction industry level. This included the following:

- It is comprised of controls that permit the definition of parameters for the construction industry being modelled.
- It models the entry of projects into the market (i.e., the construction industry).
- This federate models the process that involves the bidding and award of these projects. It models the bid/no-bid decision and bid price generation process of each company operating within the construction industry being analyzed.
- It models the competition to the *company of interest.* The *company of interest* is the company closely being tracked and analyzed in the simulation. It is the company that belongs to the modeller. This federate embeds logic that permits modeling the execution of projects awarded to the *company of interest's* competitors. The federate also models the dynamics surrounding the entry of new companies into the market and the attrition of existing companies in the industry.

The AnyLogic simulation system was used for developing this federate. COSYE was used within the development of this federate to ensure that it was capable of functioning as an HLA compliant simulation federate. The constructs and dynamics that exist at a construction industry level were abstracted and emulated using an agent-based simulation modeling paradigm. The community of owners (along with their representative) were aggregated and represented as a single agent. The competitor companies were represented as large size, medium size and small size agents. Each of these agents could have agent populations with numbers that were dependent on the number of competitors operating within that specific industry. The ambassador (or representative) for the *company of interest* was also represented as a single agent.

The company of interest or *Simphony federate* was responsible for modeling the dynamics that take place at the company level, specifically, at the *company of interest*. The Simphony federate serves the following purposes:

- It is comprised of a user interface for capturing inputs that define the attributes of the *company of interest* and outputs from the simulation i.e., performance results for the *company of interest*.
- It models the execution of projects awarded to the *company of interest*.
- It tracks, collects and reports data on the performance of the *company of interest*.
- It provides feedback to the ambassador of the *company of interest* within the *tendering federate* on prevailing work conditions at the *company of interest* so that it can make the appropriate bid decisions.

The *Simphony federate* was developed as a Windows form application using Visual Studio (2010), Simphony simulation system and COSYE. Simphony was used to develop the special purpose template elements. These elements, along with general purpose template elements were used to create a discrete event simulation model that models the processing of projects awarded to the company of interest. This model included a section that tracks and collects data on company performance. The Simphony model also includes a component from COSYE that enables the model and other components it is associated with to become an integral part of an HLA complaint federate. The Windows form application includes a user interface that captures user inputs and displays performance results. The Simphony model is embedded as a resource in the Windows form application (which also contains the user interface) to complete the development of the *Simphony federate*.

4.3 FEDERATION MANAGEMENT

This section discusses details for the creation of the federation, joining, declaration management, resigning the federation and destruction of the federation. Details on how each of these steps is synchronized during the federation life cycle are also discussed.

4.3.1 Synchronization of the Distributed Simulation Federation

Given that a distributed simulation federation should contain at least two federates, there is a need to ensure that all that needs to be done during the setup of each federate is completed before the execution of the entire federation begins. This is because the speed at which each federate completes its setup varies. Moreover, the scope of things that need to be done at start-up in each federate varies. Also, computers cannot do more than one thing concurrently for the same execution thread; hence, there is a need for synchronization.

There are two concepts within the subject of synchronization of distributed simulation systems that are usually mixed up. That is the *achievement of a synchronization point by the federate* and the *achievement of a synchronization point by a federation*. After a federate achieves a synchronization point, its state does not change unless it is the last joined federate to achieve this point. The entire federation achieves a synchronization point only when the last joined federate announces the achievement of that synchronization point. In that case, the federation can proceed and the state of the different federates can start changing. There are specific points in the life cycle of a federation at which all federates need to be synchronized to guarantee consistency in the distributed simulation system. These include:

- The start of declaration,
- The start of populating the federation, i.e., creation of object instances,
- The start of initializing the attributes of the objects,
- The start of federation execution and
- The commencement of simulation termination.

In the HLA domain, these are technically referred to as synchronization points. These points represent major events in the implementation of each federate and the federation as a whole. However, they need to be registered for the federation to know that they exist. It should be noted that some of these synchronization points may not exist within specific federations as a result of the way that they are implemented. In such as case, all federates don't implement these synchronization points. On the other hand, some synchronization points are mandatory and will exist within any distributed simulation federation for it to function properly. These synchronization points include: *ready to declare*, *ready to execute* and *ready to terminate*. This implies that at minimum, any meaningful distributed simulation federation must implement these three synchronization points. Synchronization of the federation at the *ready to execute* synchronization is a major milestone because it signifies the commencement of simulation execution. On the other hand, the *ready to terminate* synchronization should be achieved last by the federate that is responsible for terminating the simulation execution of the entire distributed simulation federation. All the other federates can announce the achievement of this synchronization point as soon as the federation execution commences. The achievement of this synchronization point marks the end of the distributed simulation execution. The developed system in this thesis implemented these three mandatory synchronization points. A discussion is presented on the federate that was responsible for federation execution termination.

The process of synchronizing federates within a distributed simulation federation can be managed in one of three ways. It may be achieved through *manual synchronization, automated synchronization* or *hybrid manual and automated synchronization*. In the former, the modeller or user of the distributed simulation explicitly has to push a button to communicate to the RTI the achievement of each synchronization point by each federate. In the automated synchronization code is written within each federate to announce the achievement of all synchronization points and to manage the achievement of these synchronization points by the entire federation. There is no human interference. In the last approach, the program within the federate manages some of the events associated with the achievement of the synchronization points, while others are managed by the modeller (through human intervention). The application developed in this thesis implemented a hybrid (*manual and automated*) approach for announcing the achievement of synchronization points.

4.3.1.1 Life Cycle of the Distributed Simulation Federation

All distributed simulation federations implement the four steps that summarize the life cycle of an HLA complaint distributed simulation. These steps include the *creation of the federation execution*, *joining the federation execution*, *resigning the federation execution* and *destroying the federation execution*. The sequence in which these events are implemented in the life cycle of a federation is summarized in Figure 4.3.

A distributed simulation federation is started on its creation. This is usually done by one federate in the federation. The *Company of Interest (Simphony) federate* was designated to create the performance management simulation federation in this thesis study. All federates must join this created federation before execution commences. After all federates have announced the achievement of the *ready-to-terminate synchronization point* (i.e., the federation is synchronized at *ready-to-terminate synchronization point*, all federates resign the federation execution, one at a time. After all federates have resigned the federation execution, the designated federate destroys the federation execution. In the application developed for this thesis, the *Tendering Module (AnyLogic) federate* is designated to destroy the federation execution at the end of the simulation.

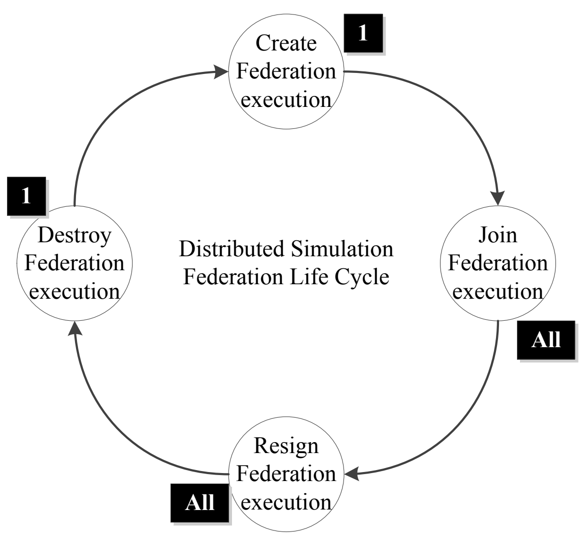


Figure 4.3: Schematic Layout of Federation Management Life Cycle Details for the Created Federation

4.4 DATA MANAGEMENT

Distributed simulation systems are characterized by the exchange of data amongst federates at run-time. This was the case with the federation developed within this thesis. In order to achieve a seamless data exchange experience in any distributed simulation, the developer needs to properly design the data exchange protocols. Development of a useable Federate Object Model (FOM) and proper handling of declaration management details (publish/subscribe to interaction classes or object class attributes) is central to this. An Object Model Template (OMT) editor is a tool that would be required to create a useable FOM. Details of this are discussed later in this section (subsequent sub-sections). Prior to that discussion, details of the configurations that need to be appropriately set up prior to data exchange within a federation are introduced.

4.4.1 Sharing Data in the HLA

This section presents a background on how messages are exchanged in HLA compliant distributed simulation systems. This serves as a basis for appreciating the design patterns that were adopted for implementing the communication between the AnyLogic federate and the Simphony federate.

Components of a distributed simulation system are run concurrently so that they are able to share information/data that they generate in real time as the simulation progresses with each other. The nature of delivery of messages to a federate will depend on two factors, namely:

- *Asynchronous status:* This refers to the state of a federate throughout the federation execution. If a federate has asynchronous delivery enabled, then the federate can receive a certain type of message (RO message) instantaneously, i.e., as soon it is sent. If asynchronous delivery is disabled, these messages are received at the point in time that a time advance request has been issued to the federate by the RTI.
- *Type of message being delivered:* Messages in the HLA can only be one of two types, i.e., receive order messages (RO) and time stamped order messages (TSO). The type of message to be associated with an attribute or a parameter is defined within the federation object model (FOM). The only difference between the two is the fashion in which the RTI delivers the messages to the target federate(s). With the RO messages, they are delivered as soon as they are sent (if the receiving federate is asynchronous delivery

enabled), or just prior to granting a time advance request (for receiving federates that don't have the asynchronous delivery enabled). TSO messages on the other hand are always delivered to the target federate just prior to a time advance request being granted. The TSO messages that get delivered are those of a time stamp that is less than or equal to the time being granted to the receiving federate.

4.4.2 Object Model Template (OMT)

An Object Model Template is one of the three components of the high level architecture (HLA). It summarizes the specifications of the data to be communicated between simulations and the documentation of that data. The object model template consists of the following documents:

- *Federate Object Model (FOM)*: A FOM describes the objects, interactions, attributes and parameters that are shared within a given federation.
- *Simulation Object Model (SOM)*: A SOM on the other hand specifies the objects, interactions, attributes and parameters that are used within a single federate.

In this thesis, attention was paid to the documentation of the FOM. No formal documentations were provided for the SOM, but rather, variations to this are presented because they served as better design aides for development and explanation.

4.4.3 Federate Object Model (FOM)

In order to develop and execute a distributed simulation federation, one needs to create a federate object model that represents all that needs to be shared among the federates in the course of the simulation. This FOM is created using an editor that generates a file that can then be used within the simulation. As per the 2010 *1516 HLA standards,* the FOM is to be represented as an XML file. Prior to simulation execution, this XML file is passed to the Run Time Infrastructure (RTI), which makes use of it in managing data exchange between federates. An *open file dialogue control* was provided within the application developed for this thesis to facilitate the modeller to locate and specify the file path of the FOM for the federation to use.

The synthetic simulation environment that was used (COSYE), has an OMT editor that facilitates the creation of FOMs. Currently, this tool is supported as a plug-in that is loaded into Visual Studio (2010) and used within the Visual Studio environment to create the FOM. The version of the FOM specified within the IEEE standards (IEEE 1516.2-2010) released in 2010 is different from the FOM format specified in the IEEE standards (IEEE 1516.2-2000) released earlier in 2000. To cope with this change, developments of a new OMT editor in COSYE that is independent of Visual Studio (i.e., is a standalone application) were underway at the time this thesis was being compiled. However, at that time, the RTI in COSYE had been modified to expect an FOM that is complaint with the IEEE 1516.2-2010 standards, and yet, the COSYE OMT editor plugged into Visual Studio (2010) was still generating outdated FOMs (in the IEEE 1516.2-2000 format). To cope with this challenge, a few extra steps had to be undertaken in this development that led to the upgrade of the FOM to a format (IEEE 1516.2-2010) that was usable. Figure 4.4 summarizes this process.

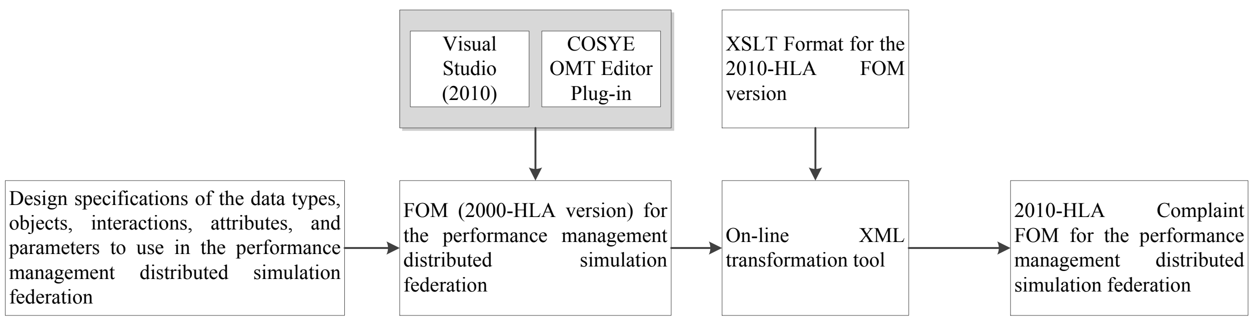


Figure 4.4: A **Schematic Layout of the Process Involved in Creating the FOM for the Federation**

It is worth-noting that there are other software tools in existence that can be used for the creation of an FOM to be used in a distributed simulation. An example of such a tool that is open source is *SimGen*. This tool can also be used for the creation of federates and federations. Another OMT editor that can be used is that created and maintained by Pitch™ called *Pitch Visual OMT 2.0* (Moller, Antelius, Johansson, Lofstrand, & Wihlborg, 2010). A screen shot of the COSYE OMT editor in Visual Studio is shown in Figure 4.5.

In this thesis, the majority of the data was exchanged using interactions and parameters. This choice was made because most constructs that were to be shared and data related to those constructs don't persist in real life; hence, there was no need to make the federation development more complicated that it already was.

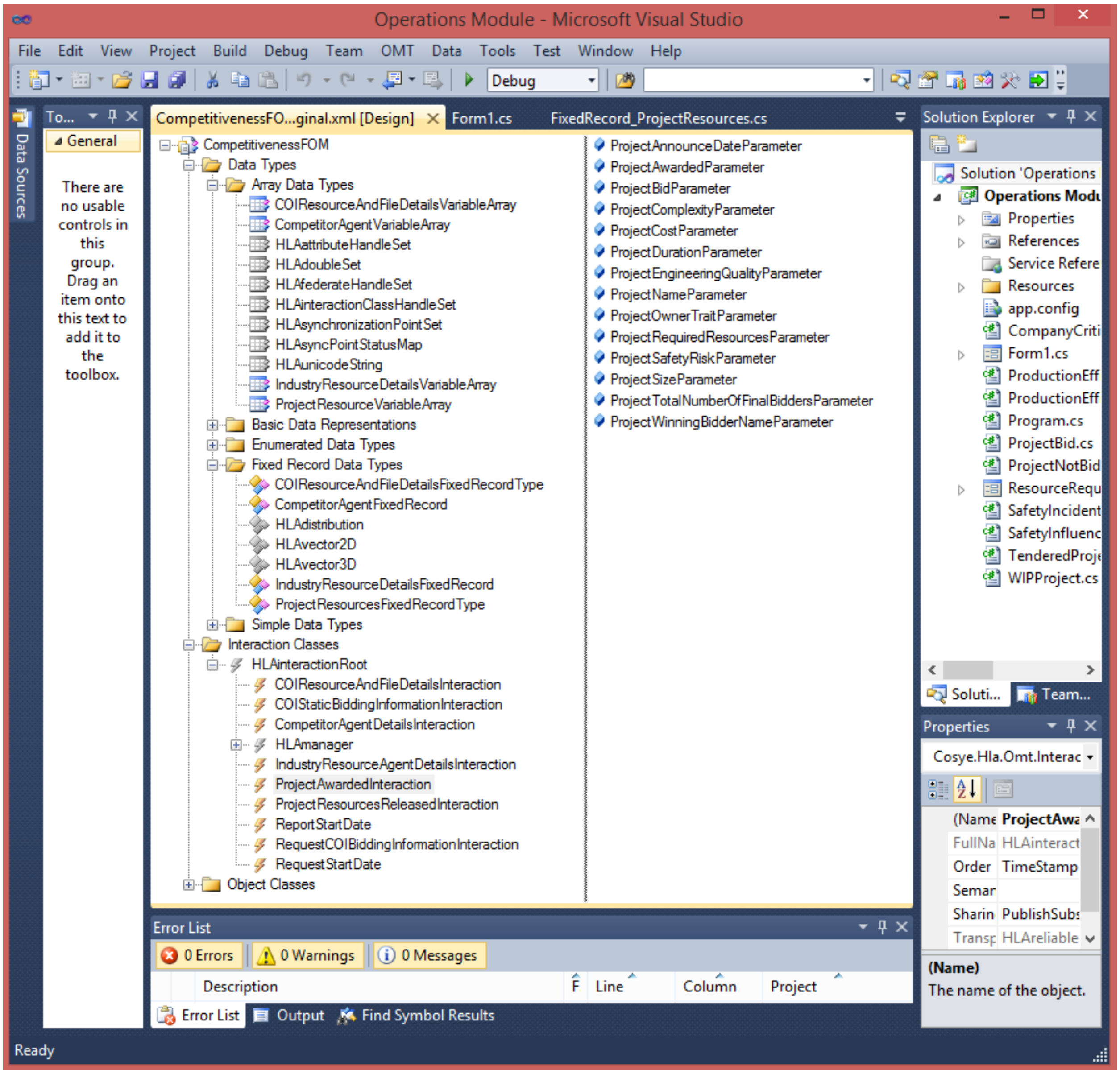


Figure 4.5: COSYE OMT Editor in Visual Studio

A number of constructs were shared between the tendering (AnyLogic) federate and the operations (Simphony) federate. These included:

- Parameters from the *company of interest* that specify the company's bid strategies and other information (such as workload) to guide on a bid/no-bid decision.
- Critical resources at the *company of interest*.
- Projects—new projects entering the market, projects awarded at the end of a bidding cycle.

- Resource agent details sent from AnyLogic to Simphony federate at the end of simulation.
- Company agent details sent from AnyLogic federate to Simphony federate at the end of simulation.

All communication was sent using interactions because there was no need to use persistent objects (i.e., object instances—they would add a layer of complexity to the implementations). Information received by a federate was used to update the federate's state (stored in a buffer to avoid its loss), after which it would be utilized.

4.4.3.1 Interaction Classes, Parameters and Data Types Used

Table 4.0 summarizes the basic data types defined in the FOM and subsequently used within the distributed simulation federation.

Table 4.0: Simple Data Types Defined in the FOM of the Performance Management Federation

Interaction	Parameter	Data Type
ProjectAwardedInteraction	ProjectAnnounceDateParameter	HLAdouble
	ProjectAwardedParameter	HLAunicodeString
	ProjectBidParameter	HLAunicodeString
	ProjectComplexityParameter	HLAdouble
	ProjectCostParameter	HLAdouble
	ProjectDurationParameter	HLAdouble
	ProjectEngineeringQualityParameter	HLAdouble
	ProjectNameParameter	HLAunicodeString
	ProjectOwnerTraitParameter	HLAdouble
	ProjectRequiredResouresParameter	ProjectResourceVariableArray
	ProjectSafetyRiskParameter	HLAdouble
	ProjectSizeParameter	HLAunicodeString
	ProjectTotalNumberOfFinalBidde	HLAinteger

Interaction	Parameter	Data Type
	rsParameter	
	ProjectWinningBidderNameParameter	HLAunicodeString
ProjectResourcesReleasedInteraction	ProjectResourcesReleasedParameter	ProjectResourcesFixedRecordType
COIResourceAndFileDetailsInteraction	COIResourceAndFileDetailsParameter	COIResourceAndFileDetailsVariableArray
COIStaticBiddingInformationInteraction	COIBiddingStrategyParameter	HLAunicodeString
	COIMaximumNumberOfCompetitorParameter	HLAinteger
	COIPreferenceRatingForLargeProjectsParameter	HLAdouble
	COIPreferenceRatingForMediumProjectsParameter	HLAdouble
	COIPreferenceRatingForSmallProjectsParameter	HLAdouble
	COIProjectComplexityParameter	HLAdouble
	COIProjectEngineeringQualityParameter	HLAdouble
	COIProjectOwnerTraitParameter	HLAdouble
	COIProjectSafetyRiskParameter	HLAdouble
RequestCOIBiddingInformationInteraction	ProjectNameForRequestingCOIInfoParameter	HLAunicodeString
IndustryResourceAgentDetailsInteraction	IndustryResourceAgentDetailsParameter	IndustryResourceDetailsVariableArray
CompetitorAgentDetailsInteraction	CompetitorAgentDetailsParameter	CompetitorAgentVariableArray
RequestStartDate	StartDate	HLAdateTime
ReportStartDate	StartDate	HLAdateTime

Complex Data Types

The complex data types included variable arrays which wrapped fixed record types. Each fixed record type had a number of fields defined within it. Details of all of these are presented in Table 4.1.

Table 4.1: Complex Data Types Defined in the FOM of the Performance Management Federation

Array Data Type	Corresponding Fixed Record Data Type	Fields in the Fixed Record
ProjectResourceVariableArray	ProjectResourcesFixedRecordType	ResourceName
		ResourceQuantity
		ResourceManHoursRequired
		ResourceManHoursCompleted
COIResourceAndFileDetailsVariableArray	COIResourceAndFileDetailsFixedRecordType	ResourceName
		ResourceTotalServers
		ResourceServersAvailable
		ResourceMeanUtilization
		FileName
		FileCurrentLength
		FileMeanLength
		FileMeanWaitingTime
IndustryResourceDetailsVariableArray	IndustryResourceDetailsFixedRecord	ResourceName
		ResourceTotalServers
		ResourceMeanUtilization
CompetitorAgentVariableArray	CompetitorAgentFixedRecord	CompetitorName
		CompetitorProjectsAwarded
		CompetitorProjectsBidAndLost
		CompetitorBiddingStrategy

Array Data Type	Corresponding Fixed Record Data Type	Fields in the Fixed Record
		CompetitorLPProductionCapacity
		CompetitorSMPProductionCapacity
		CompetitorNumberThreshold
		CompetitorOwnerTraitThreshold
		CompetitorProjectSafetyRiskThreshold
		CompetitorProjectEngineeringQualityThreshold
		CompetitorProjectComplexityThreshold

The last two variable arrays are used for transferring information about company and resource agents from the AnyLogic federate to the Simphony federate at the end of simulation. The interactions containing this information are sent at the end of the simulation run from within the AnyLogic simulation experiment object's method, named *"After Simulation Run()."* On the Simphony federate end, the performance measurement modeling element receives the interaction and temporarily stores the data in buffers internally defined within it. This information is subsequently displayed as output within the list view controls in the Windows form application. The schematic layout presented in Figure 4.6 indicates that Company Agent and Resource Agent population information is sent from AnyLogic federate to Simphony federate at the end of a simulation run.

A brief summary that details the information that was actually shared between federates is presented. A schematic layout is used as a means of communicating these details. This schematic is presented in Figure 4.6. It indicates the source and the receipt of the different information. Further details on the sequence in which this information is sent and received are illustrated in the sequence diagrams presented at the tail end of this chapter.

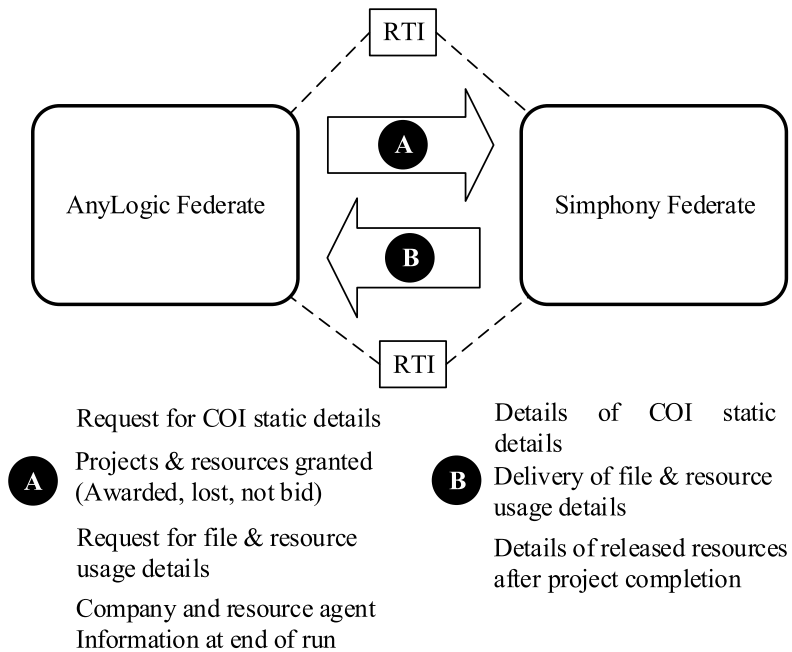


Figure 4.6: Details of Information Shared Amongst Federates in the Application

4.5 TIME MANAGEMENT

In a distributed simulation system, time is managed by the Run-Time-Infrastructure (RTI). This means that it receives requests from all joined federates to advance its time and grants them the permission to advance their time at the appropriate moment. The value for the current time within a federate is stored in a parameter known as a federate's *Logical Time*. At any moment, joined federates in the same federation execution can have different logical times. They can also implement different time schemes. The HLA provides for two time management schemes. These include:

- Event driven (Next Message Request—NMR) and
- Time stepped (Time Advance Request—TAR).

A federate may implement one of the two time management schemes throughout the life time of the federation, or it may opt to switch between both schemes in the course of the simulation. The event driven scheme is synonymous to the time management scheme implemented by discrete event simulation systems. In the next message request scheme, time is moved to only points in time at which events take place, i.e., the times that HLA messages are received by a federate. The time stepped management scheme on the other hand is similar to that implemented by simulation systems that support the continuous simulation modeling paradigm. The developer or modeller (user) defines the size of the time steps before hand, or as the simulation progresses. These time steps are then used to advance the logical time of the federate. The reader is referred to a book published about the HLA by Kuhl, Weatherly and Dahmann (1999) for further reading on these time management schemes.

When time is implemented in a distributed simulation federation, it becomes necessary to time stamp messages that are passed between federates so that they are delivered at the right time, since federates may have different values for their logical time at any given moment.

Federates implementing time in an HLA distributed simulation at any instance in the course of the simulation can take on one of two states. These include:

- *A time advancing state:* A federate enters a time advancing state as soon as it makes a request to the RTI to advance its logical time. It can be through a time advance request or a next message request. No computations are done in this state. Messages received in this state (typically RO if the federate has asynchronous delivery enables) are used to update the state of the federate. In other words, the data passed on to the federate is stored in a buffer.
- *A time granting state:* Federates enter a time granting state as soon as the RTI issues them permission to advance their logical time. This is where the processing (computations) of the federate are done. Data is retrieved from the federate's buffer and used in computations. Federates will typically receive TSO messages on entry into this state. Requests to advance the time of the federate forward are made in this state (typically as the federate is going to exit this state).

4.5.1 Implementation of Time and Message Exchange in the Federation

Time was implemented within the distributed simulation federation because the real life processes and constructs emulated within the application are tightly coupled with time and influenced by time. The federation was set up such that the AnyLogic federate had no time regulation while the COI/Simphony federate was time constrained. This meant that the AnyLogic federate would be the lead federate and determine the pace of the federation with respect to time advancement and the COI/Simphony federate would follow. Another implication of this is that the AnyLogic federate would be sending only time stamped messages and the COI/Simphony federate would be receiving time stamped messages only.

Details of the time related parameters that were enabled/disabled by each federate within the developed application are summarized in Table 4.2.

Table 4.2: Federation Time Management Settings Used

Parameter	Federate	Status of Parameter in the Federate
Time Regulation Enabled	AnyLogic	True
	COI/Simphony	False
Time Constrained Enabled	AnyLogic	False
	COI/Simphony	True
Asynchronous Delivery Switch On	AnyLogic	True
	COI/Simphony	True

This setup was adopted to ensure the successful achievement of the envisaged behavior of each federate. The behavior referred to is one in which the AnyLogic Federate sends Time Stamped Ordered (TSO) messages to the COI/Simphony federate and receives Receive Ordered (RO) messages from the COI/Simphony federate. Also, the COI/Simphony federate would be able to send (RO) messages to AnyLogic federate and receive TSO messages from the AnyLogic federate.

The asynchronous delivery switch turned on for the AnyLogic federate guarantees that messages sent by the COI/Simphony federate will be delivered regardless of whether it is in a time advancing state or a time granting state. This is necessary to ensure that messages are delivered as soon as they are sent by Simphony. For example, when Simphony federate is done engaging some resources on a specific project (i.e., releases them) and sends a message to AnyLogic federate indicating that these have been freed, AnyLogic should receive such a message instantaneously so that it replenishes the industry resource pool making resources available to other companies for use. Also, when the COI ambassador in the AnyLogic federate requires information from the COI/Simphony federate so that it can make a decision on whether or not to bid a specific project, it will need to receive such information as soon as it is required. Enabling asynchronous delivery switch for the AnyLogic federate makes this possible. This detailed message exchange between federates is summarized in Figure 4.7.

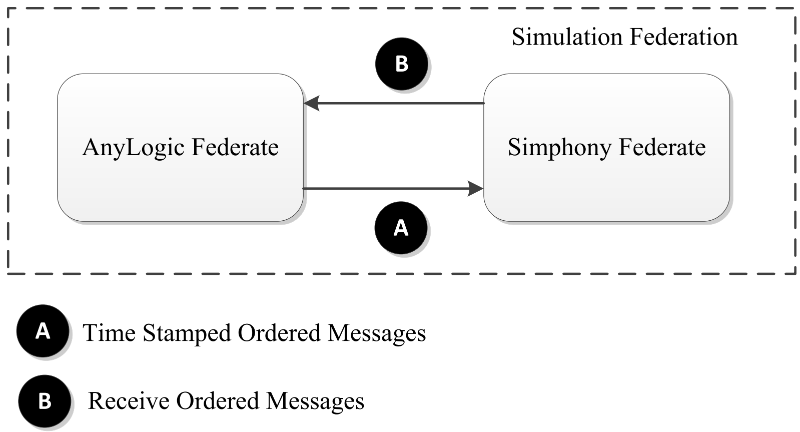


Figure 4.7: Time-Based Message Exchange Between Federates in the Application

The implementation of time within this application was guided by the state chart within the IEEE HLA standards – 1516.1-2000 Federation Interface specifications. It is important to note that there are two types of time that have to be carefully managed within each federate within the application. These include:

- The federate's logical time and
- The simulation engine's logical time.

The two times arise from the fact that each federate has an embedded simulation engine that is concurrently running alongside the federation execution. Details presented in this section only relate to the federate's logical time.

4.5.2 Time Units in the Simulation Application

Two time units were used in the developed simulation application. The COSYE RTI was written in such a fashion that it only supports simulation *time units of seconds*. The RTI connection element within Simphony, which serves as a basis for developing COSYE-aware special purpose templates, also only supports time units in seconds. It is for these reasons that these aspects were modeled using a time unit of seconds within the application. Other aspects within the AnyLogic simulation system were modelled using a *time unit of days* for convenience. The HLA component of this federate was modelled using seconds in order to be consistent with the COYSE RTI requirements. The appropriate time unit conversions were made in the course of implementing time management aspects. Table 4.3 summarizes the time units used in the implementation of time within the different components of the developed distributed simulation application.

Table 4.3: Time Units Implemented in the Federation

Component/Aspect of the Application	Time Unit
Federation	Seconds
Simphony federate (Simulation engine)	Seconds
AnyLogic federate (Simulation engine)	Days

4.5.3 Adopted Design Pattern for Managing Time in the AnyLogic Federate

Convention stipulates that a good design pattern for implementing time management involves making time advance requests within the time advance grant callback routine. This eliminates the possibility of a federate making other time advance requests while it is in a time advancing state. This requirement is specifically crucial in cases where the RTI used does not support zero look ahead (as was the case with the COSYE RTI). However, this convention may be violated with some additional modifications to the way time is implemented within the federate. Violation refers to implementations that make time advance requests outside the time advance grant RTI callback. Although discouraged, it can be a work around for complex federate developments, as was the case with the AnyLogic federate.

There were three places within the AnyLogic federate at which time advance requests were made. These included:

- COI Federate Ambassador Class (specifically when a callback was received that indicated that the federation had achieved the *"ReadyToExecute"* synchronization point).
- Bid Manager Agent (within the event scheduler nodes used to model new project arrivals).

- Construction Industry (CI) Agent (within the additional code snippet—in the routines meant to send interactions to the Simphony federate).

In order to guarantee a seamless implementation of Time Advance Requests and proper sending of Time Stamped messages to the Simphony Federate, two state variables were defined to track the state of the federate with respect to time aspects. It is know that any federate implementing time will transition between two states namely:

- Time advancing state and
- Time granting state.

One state variable was used to track the value of the last requested time for the federate, while the other tracks the state of the federate with respect to time. Requests for time advancement were then made only when the federate was in a time granting state. However, sending messages could have been accomplished in both the time advancing and time granting states. When in the time granting state, the federate was set up to time stamp messages with a value of the time that was to be requested from the RTI. On the other hand, messages sent while the federate is in a time advancing state would be time stamped with a value of the last requested time.

A complex federate in this context is defined as one that sends time stamped messages while the federate is in a time advancing and granting stated. In addition, it is a federate that is comprised of distributed components that execute concurrently with a significant number of these components participating in the different aspects of the federation. A typical example is a federate that is comprised of autonomous or semi-autonomous agents.

Most distributed simulation frameworks don’t provide for these two state variables in their implementations, hence transferring the burden of this implementation to the developer. A convenient place to incorporate these two state variables in any distributed simulation framework would be defining these as attributes/properties of the federate ambassador so that the RTI can update these as simulation progresses.

4.6 OWNERSHIP MANAGEMENT

Ownership management is an important concept within the HLA that is best suited for the development of distributed simulation federations that contain federates that share object instances.

There are three ingredients that must be present in order for ownership management to become a necessary part of a distributed simulation federation execution. These include:

- There must be at least one object instance that has at least one attribute.
- There should be two or more joined federates that intend to share the attribute(s) of the object instance(s).
- These federates must publish the attribute(s) that they plan to own at some point in the course of the simulation.

To avoid conflict in sharing object instance attributes, the HLA stipulates in its rules (the 5th rule) that an object instance attribute can be owned by only one joined federate at a time.

Registering and managing object instances between different federates in a distributed simulation federation execution is quite complex and creates a significant amount of application development overhead. A huge piece of this overhead usually arises from the management of ownership of objects.

Consequently, when a given distributed simulation can do away with object instances, it is advisable to go that route and implement communication through the use of interaction classes. This is usually possible in situations where constructs being modelled don't need to persist in the course of the simulation execution. Using interaction classes to convey messages within the federation removes that extra layer of complexity related to ownership management.

Fortunately, the application developed within this thesis was designed and implemented in such a way that there was no need to register object instances. The constructs that would otherwise have been modelled as object instances at the federation level were instead represented as proxy object, an instance within the respective federates (i.e., AnyLogic and Simphony federates respectively). The constructs that were represented in this fashion include shared resources in the pool at an industry level and projects created within the virtual construction industry.

Information about these was shared through the RTI by use of interaction classes and their associated parameters. As a result, no ownership management was implemented within the company performance management simulation federation developed within this thesis.

4.7 REQUIREMENTS TO RUN THE DEVELOPED PERFORMANCE MANAGEMENT FEDERATION

The performance management distributed simulation federation was developed using different simulation systems and development environments. These included:

- Simphony simulation system,
- AnyLogic simulation system,
- COSYE framework (.NET and Java APIs) and
- Visual Studio.

As a result, all these software would be required to run the distributed simulation federation. The detailed steps required to get the entire federation up and running are summarized in Figure 4.8. This detailed sequence of steps also enumerates what is required for the inputs.

4.8 DESIGN SPECIFICATIONS FOR THE FEDERATION

Prior to implementation of the federation within the various development environments, design specifications had to be created that detailed the envisaged behavior of the different components within the federation and the federation as a whole. Sequence diagrams were used to a large extent to achieve and communicate these designs.

4.8.1 Sequence of Events at Federation Start-up

The start-up of the distributed simulation federation is a crucial part of the model development because it is the phase in which federate(s) create the federation execution, join it, register synchronization points, achieve the synchronization points (i.e., synchronize the federation execution), initialize the state of each federate (i.e., assign initial values to variables), and start simulation engine execution. The detailed process is summarized in the flow chart shown in Figure 4.8.

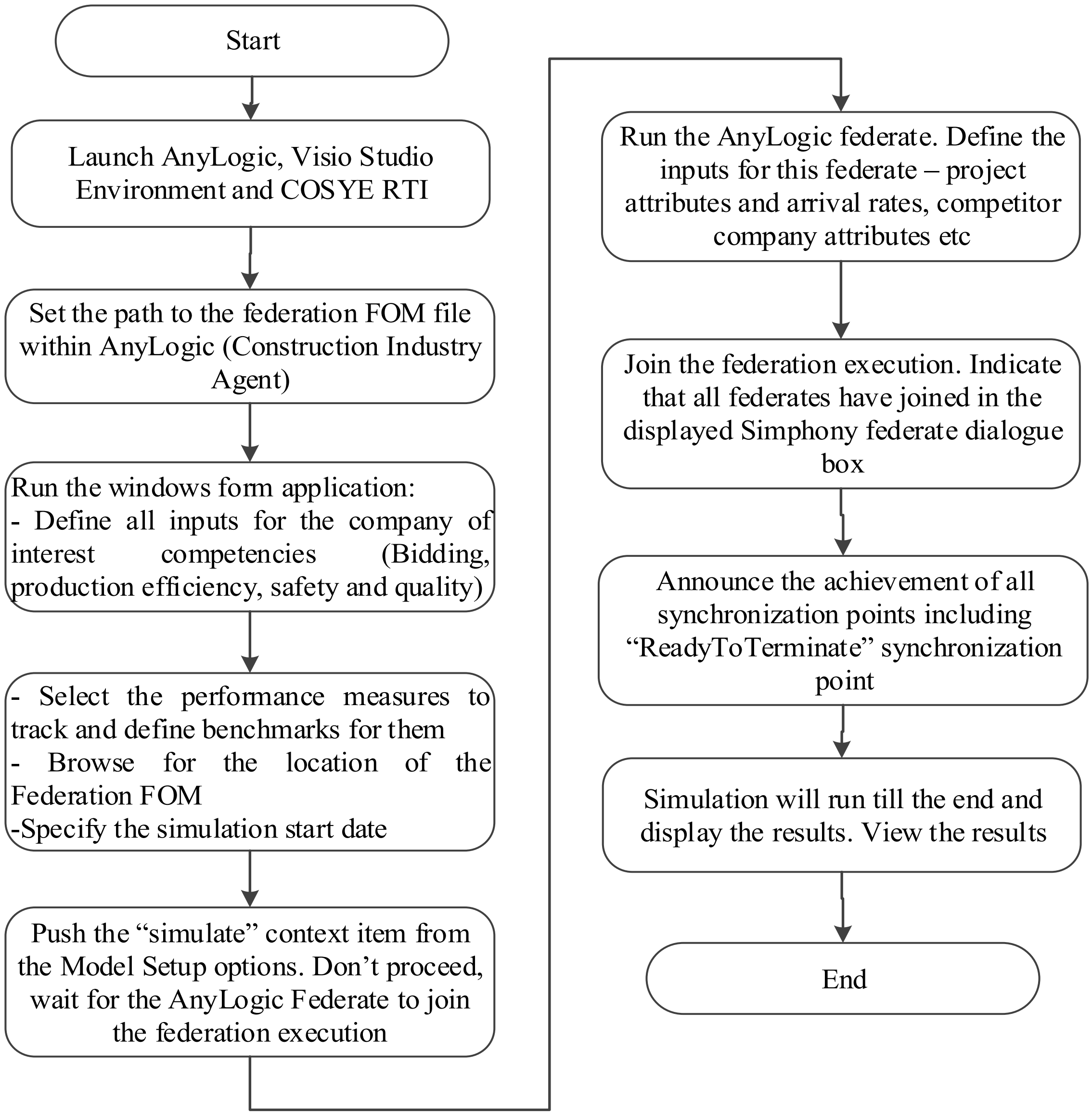


Figure 4.8: Summary of Sequence for Starting up the Federation Execution

This phase has to be managed properly to ensure that the simulation starts up gracefully so that every aspect that was envisaged to take part in the federation execution is joined in the federation by the time simulation execution commences. In order to guarantee that this portion is implemented appropriately while developing the model, this front-end of the simulation was designed and specifications were provided. A sequence diagram was used as a design aid to accomplish this. The sequence diagram developed for this is summarized in Figures 4.9 and 4.10. Two figures were used to avoid clutter.

At commencement of the simulation of the federation, the AnyLogic simulation engine automatically pauses itself when the code embedded within it is executed. This line of code is inserted to enable the user to edit the model inputs on controls placed on the Main Agent's and Bid Manager Agent's editor.

Examples of model inputs that can be defined include:

- The total number of companies in the industry (Main Agent's editor).
- The percentage of these companies that are small, medium size and large size (Main Agent's editor).
- The percentage of each company category that subscribes to a specific bidding strategy (Main Agent's editor).
- The types of projects that each company is willing to bid on (Main Agent's editor),
- Project inter-arrivals for small, medium and large size projects and their associated statistical distributions from which their properties are to be derived (Bid Manager Agent's editor).
- The resource pool for the company of interest i.e., the resource availabilities (Bid Manager Agent's editor).
- The resource requirements in man-hours for each project category i.e., small, medium and large size projects. The probability that a given resource is required by a project is also defined (Bid Manager Agent's editor).

A second reason for the AnyLogic simulation engine to be paused at simulation start-up is to prevent it from running before AnyLogic joins the distributed simulation environment as a participating federate. When in a paused state, the AnyLogic federate can gracefully join the federation execution, declare its intention to publish/subscribe to specific messages and send messages of resources to be initialized within the Simphony federate prior to simulation. It also receives a message of the static bidding information for the *Company of Interest* from the Simphony federate.

On the other hand, AnyLogic was written as a self-contained federate within the AnyLogic simulation system. The AnyLogic federate was responsible for the following:

- Joining the federation execution and notifying that all federates have joined (since it's always the last to join).
- Enabling time regulation, time constrained and asynchronous delivery (for itself).
- Achieving all registered synchronization points (for itself).
- Performing declaration management when federation is synchronized at *"ReadyToDeclare"*—publishing/subscribing interactions (for itself).

Likewise, Simphony performs a number of tasks within its initialize run method. This is a method provided by the developers of Simphony to enable modellers to do initialization work prior to simulation. Most initialization for the distributed simulation federation is done by the RTI Connection element. This is an element that provides connectivity to the Run-Time-Infrastructure (RTI) and the distributed simulation environment. This element has already been developed by the team in-charge of developing and maintaining Simphony. This study just made use of it within the Simphony model to achieve connectivity to the distributed simulation environment.

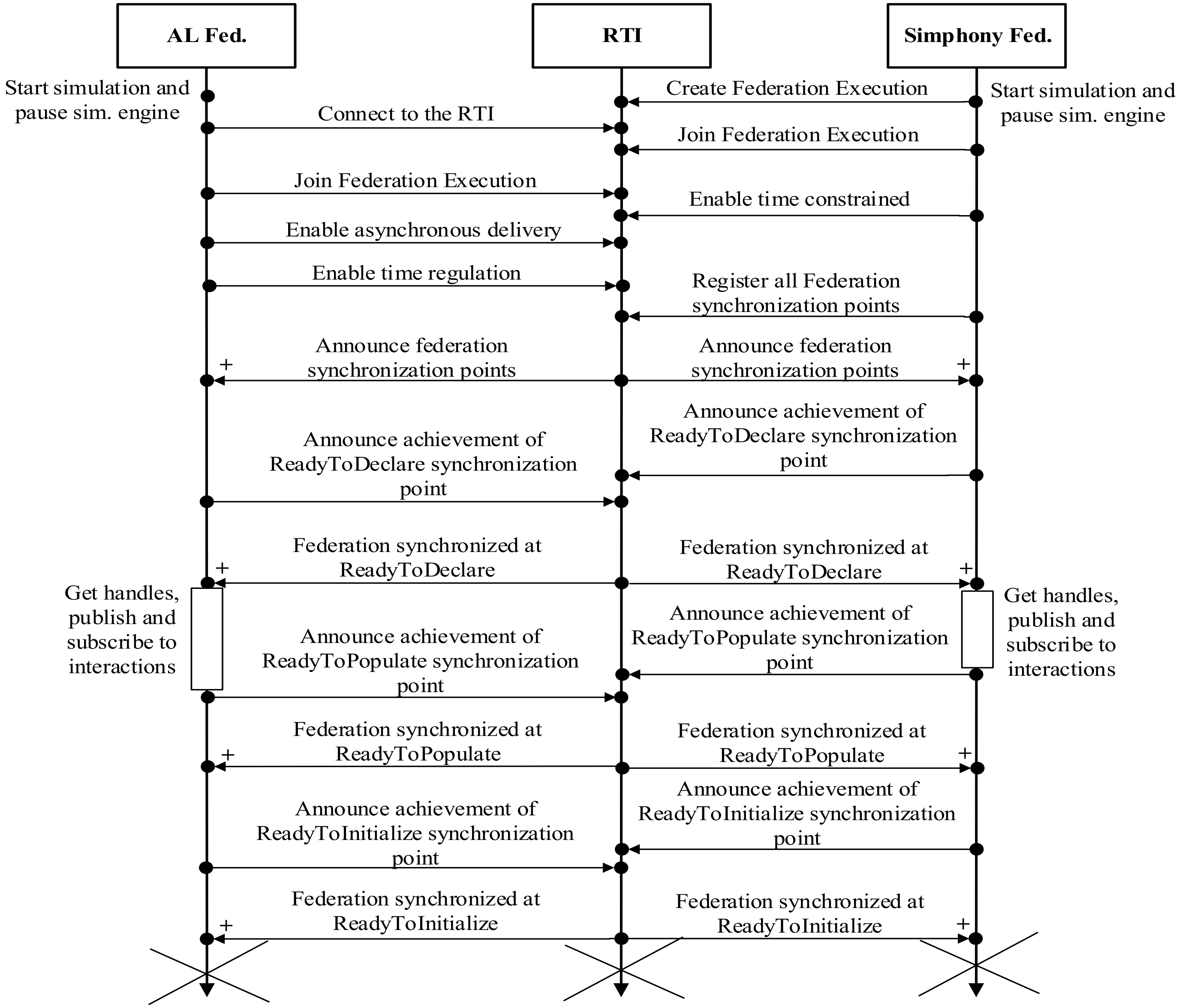


Figure 4.9: Sequence Diagram for Performance Management Federation Start-Up (Part I)

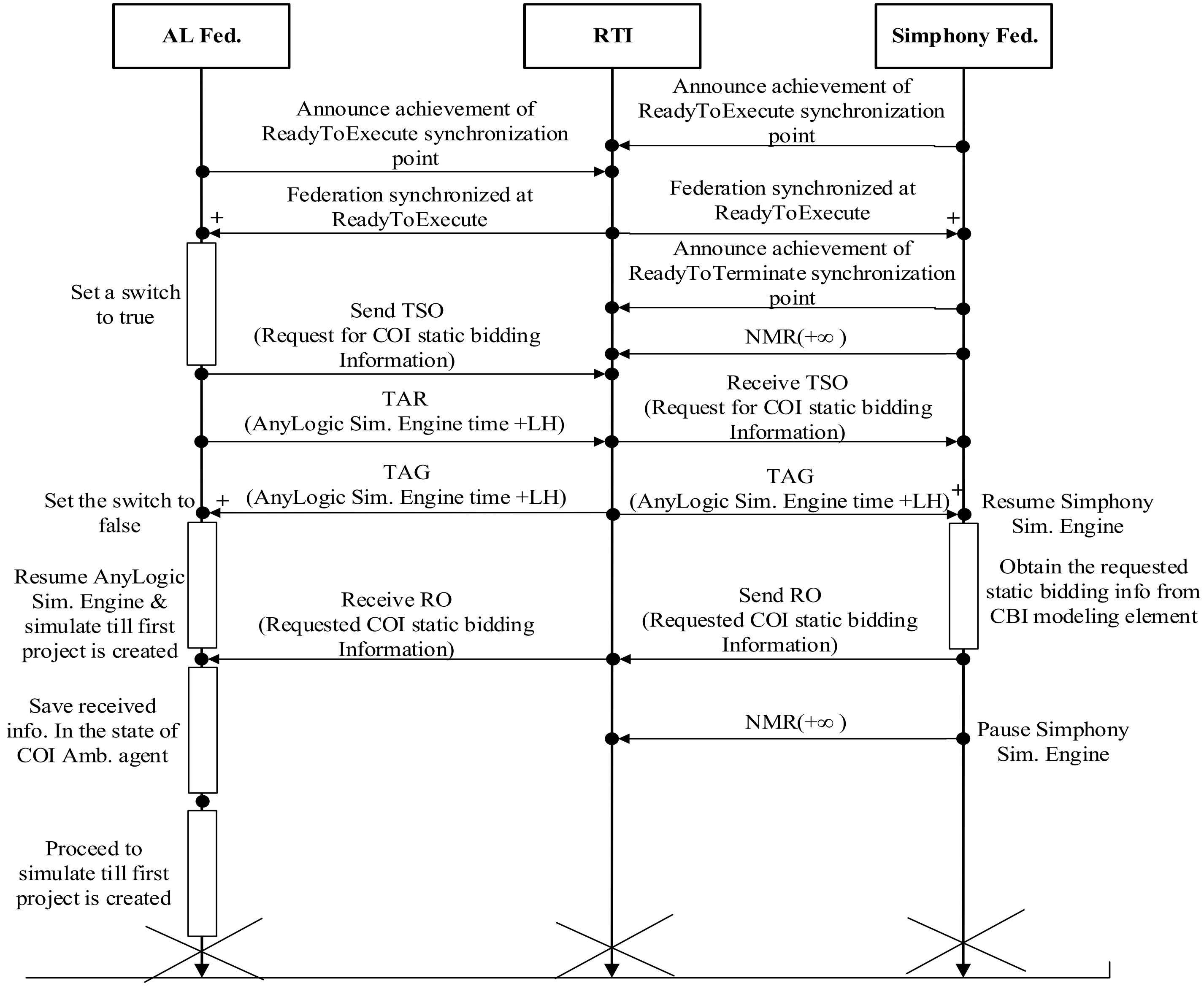


Figure 4.10: Sequence Diagram for Performance Management Federation Start-Up (Part II)

The RTI Connection element is responsible for a number of start-up events and shut down of the federation execution. These include:

- Creating the federation execution.
- Joining the federation execution and waiting for all other federates to join.
- Enabling time regulation, time constrained and asynchronous delivery (for itself).
- Registering all synchronization points (for the federation).
- Achieving all registered synchronization points (for itself).
- Performing declaration management when federation is synchronized at *"ReadyToDeclare"*—publishing/subscribing interactions (for itself).
- Resigning the federation execution (for itself).
- Destroying the federation execution.

In addition to the above, the Simphony federate also performs other initialization tasks to ensure that it has whatever it requires when the distributed simulation commences. These tasks are performed by the *Company Bid Information Element*. They include:

- Reading the user input stored within the controls of the Windows form application and storing them within public fields defined in this modeling element.
- Resizing the global attributes in the Simphony model to facilitate the storage of information generated during the simulation.

At start-up, the Simphony federate (through the RTI Connection element) automatically implements the first three listed tasks. The fourth is partially achieved. Complete synchronization of the federation at all points is achieved by the modeller/user appropriately announcing their achievement for the AnyLogic federate.

The start-up phase represents one of the two phases of communication streams that take place between the AnyLogic federate and the Simphony federate. Start-up communication occurs only once in the life time of the federation and that is at the beginning of the simulation execution. This phase is triggered by the commencement of the federation start-up but prior to simulation execution.

The other phase of communication takes place during simulation execution. It is triggered by the arrival of a new project. A sequence of communications will take place between the AnyLogic and Simphony federates each time a new project arrives. This sequence of communication associated with a single project arrival is referred to as one cycle of communication. Multiple cycles of communication therefore take place between federates for one simulation execution. The total number of cycles will depend on the new project arrival rate.

4.8.2 Sequence Diagram for Data Exchange during Simulation

Run-time data exchange between the AnyLogic and Simphony federates is triggered every time a new project arrival simulation event is being processed by the AnyLogic simulation engine.

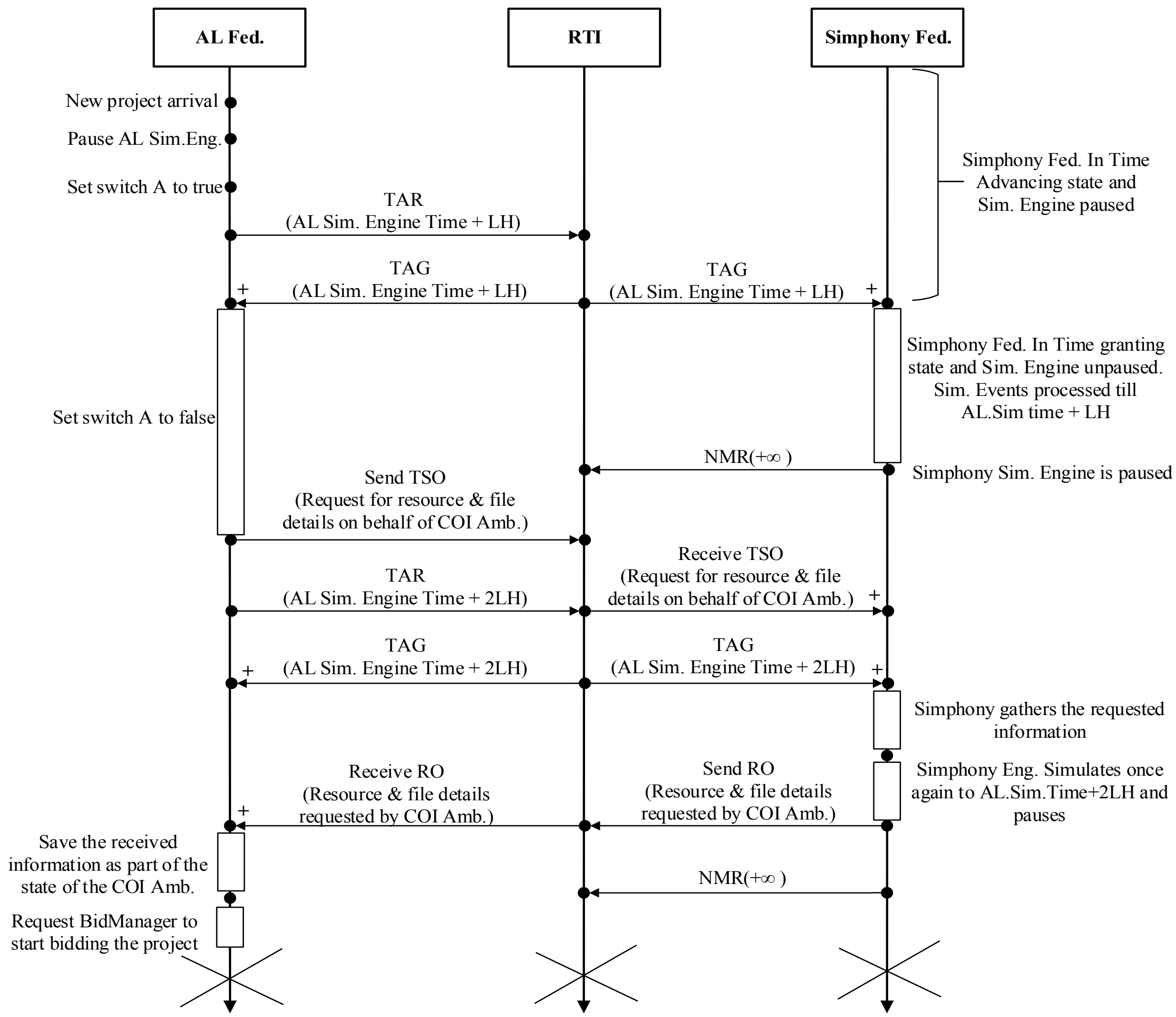


Figure 4.11: Sequence Diagram Summarizing Communication Protocols at Federation Start-up

Based on this sequence diagram, it can be seen that the AnyLogic federate regulates the pace at which the Simphony federate advances its time. The AnyLogic federate also determines when the federation execution should be terminated. This set-up is achieved by the Simphony federate achieving the *"ReadyToTerminate"* synchronization point just prior to simulation execution, and the AnyLogic federate achieving this same synchronization point only when its simulation has terminated. The termination of AnyLogic's simulation takes place when the user's pre-set time has been fully simulated.

A mechanism had to be devised for efficiently passing projects that have been awarded to any of the companies to the Simphony federate so that it could track the tendering performance of the company of interest in a timely fashion. A couple of scenarios were possible at the award of a project. These are enumerated next.

- A project is awarded to the company of interest and the project does not have any resource requirements.
- A project is awarded to the company of interest and the project has resource requirements.
- A project is awarded to a company agent other than the company of interest and it has no resource requirements.
- A project is awarded to a company agent other than the company of interest and it has resource requirements.

A design pattern was devised which would efficiently handle all four scenarios. A sequence diagram that explains how the second scenario was implemented within the developed application is presented here. It involved a sequence of messages passed between the resource agent, COI Ambassador Agent, the Bid Manager agent and the RTI (Simphony federate). Details of these including their chronological order can be viewed in the sequence diagram presented in Figure 4.11.

4.8.3 Sequence Diagram for Time Management in the Federation

Time is an important aspect of any simulation system but more especially for those that involve information exchange. This is because the design and implementation has to guarantee the time delivery of messages to parts of the model that they are required. The model developed in this thesis was intense with respect to communication between federates; therefore, time had to be managed well.

It has been previously mentioned that the time units used within the implementations of the AnyLogic federate and the Simphony Federate were different. This was done for convenience in the development process. AnyLogic federate implemented time in days and seconds while Simphony federate implemented time in seconds. However, since the federation implemented seconds as its time units, the appropriate conversions had to be done on the AnyLogic federate side, when sending or receiving messages.

Since each federate was developed around a simulation system, there were two types of time parameters tracked in each federate. The first type was related to the simulation engine of the system used (simulation engine time) and the other was related to the federate's logical time (relates to the distributed simulation federation). The logical time of the federate was kept as an internal state variable for purposes of properly managing time. Each variable dedicated to track a specific type of time had a unique execution thread associated with it. This is clearly shown in the sequence diagram presented in Figure 4.12.

It is important to point out that the federation was set up so that the AnyLogic federate would be time regulating, while the Simphony federate would be time constrained. This meant that the AnyLogic federate would always run ahead of the Simphony federate and would determine the pace at which the federation execution progressed. This pattern of time advancement was achievable (i.e., AnyLogic federate running ahead and the Simphony federate following) because distributed simulation frameworks that implement the HLA impose a constraint for time regulating federates (AnyLogic federate) to only send Time Stamped Messages (TSO), and time constrained federates (Simphony federate) to only receive Time Stamped Messages. This explains the sequence of events detailed in Figure 4.12.

The specifications in the sequence diagram (Figure 4.12) detail the use of a look ahead because the COSYE RTI did not support zero look ahead services at the time of development. Details summarized in this sequence diagram represent a complete cycle that is repeated throughout the course of simulation. The federation execution transitions two states. These include:

- Federation execution—upcoming projects scheduled and awaiting processing; processing of awarded project.
- Federation paused—new project created, bid (through a competitive process), and bid awarded.

The cycle summarized in the sequence diagram is one that occurs when the entire federation is paused. After a new project has been bid and awarded, the federation execution is resumed.

It is good practice to develop design specification for details envisaged to be implemented, especially if implementations relate to time management. This practice helps the federate/federation developer catch potential conflicts in implementation, hence avoiding frustrations resulting from run-time exceptions thrown by the RTI.

4.8.2 Sequence of Events Following Project Award

There are a series of communications that follow the award of any project. These were not included in Figure 4.12. All projects awarded are communicated to the Simphony federate so that it can track the tendering performance of the company of interest (COI). For cases where the COI is awarded a project, it communicates this so that it can be processed.

Every project awarded is communicated to the COI Amb. Agent (Company of Interest Ambassador Agent). When this happens, the AnyLogic simulation engine execution is resumed (not shown in Figure 4.13). Also, the COI Amb. Agent sends an interaction with project details to the Simphony federate. The Simphony federate queues this project. The awarded company requests the resource agent population for resources required by the project and is queued. When these required resources become available, they are granted to the company, which then starts processing the project (if project was awarded to another competitor). Details of this project are once again sent to the Simphony federate, which then takes the appropriate action.

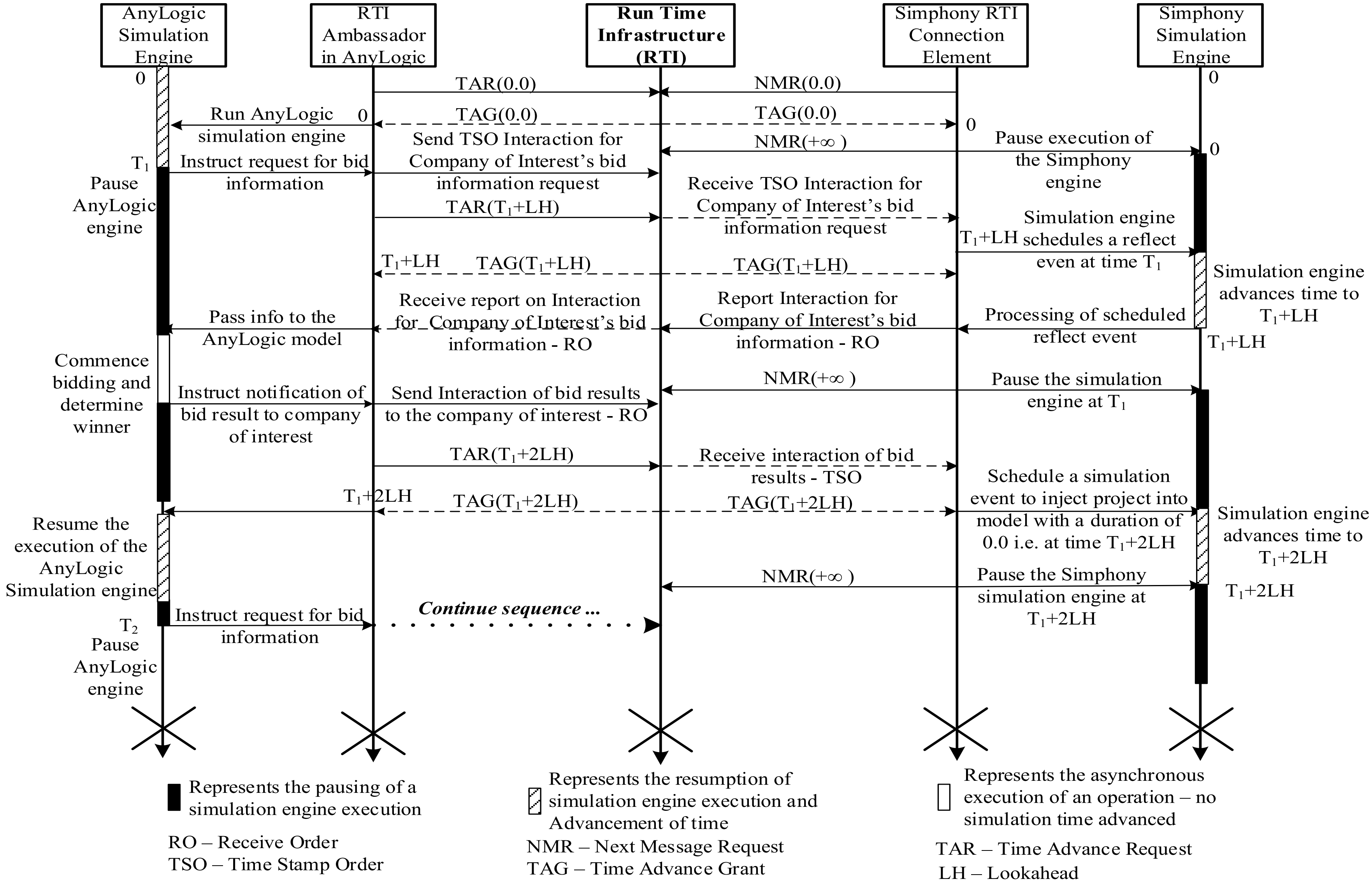


Figure 4.12: Sequence Diagram Devised Prior to Implementing Time Management Aspects in the Distributed Simulation Federation

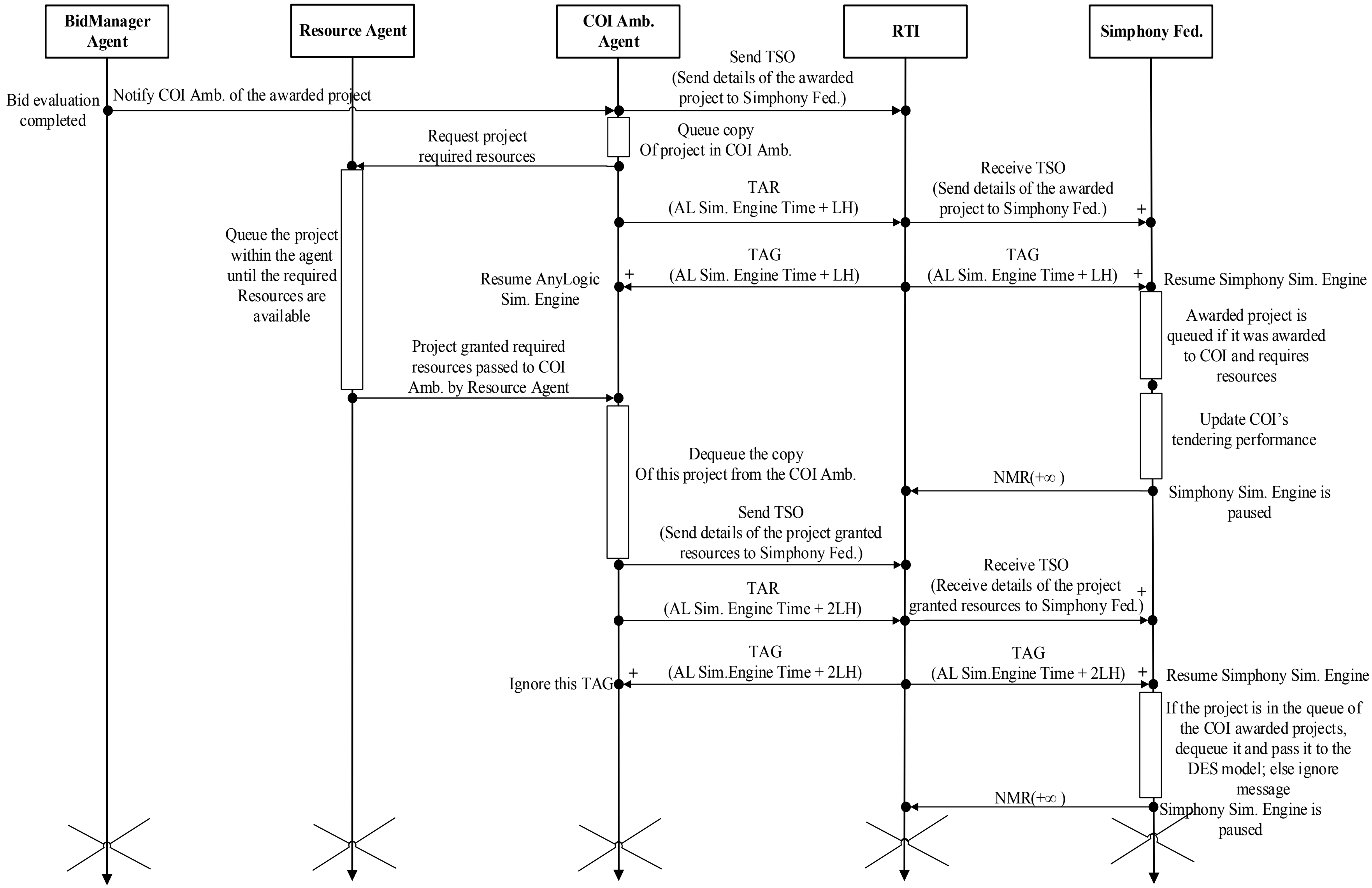


Figure 4.13: Sequence Diagram for Passing Projects from the AnyLogic Federate to Simphony Federate at the End of Bidding

4.9 SUMMARY OF CHAPTER FOUR

Chapter four was meant to provide insights into the architecture of the distributed simulation federation created in this thesis. This has been well presented and discussed in a concise manner. The design specification (sequence diagrams) used to guide the development work that related to federate communication and time management have also been summarized. The chapter also highlights the fact that development work for large-scale complex simulation systems should commence with the development of design specification that guide the implementation phase. Sequence diagrams are just one design tool that can be used in this process.

CHAPTER FIVE – THE ANYLOGIC FEDERATE

5.0 INTRODUCTION TO CHAPTER FIVE

This chapter discusses a model that was developed to emulate the competitive bidding process through which the majority of construction contractor companies acquire their work. It was necessary to model this process explicitly because it influences the performance and ultimate competitiveness of a construction contractor company. To achieve this, a virtual construction industry had to be represented on a computer and experimented with. A simulation model was used to achieve the representation on a computer. A number of things take place within any real construction industry, which was also formalized in the model (and virtual construction industry) that was created. These include:

- Creation of projects (large, medium and small projects) in a virtual construction industry based on variable inter-arrival times.
- The competition for these projects by companies that are operating within the industry.
- The request for resources required by projects and subsequent release and replenishment of the resource pool after their engagement on a project.
- Exchange of information between itself and the Simphony/COI federate.

For the model to achieve all of the above, an agent-based simulation approach was adopted. This approach was adopted because of the nature of the problem that was being dealt with. Constructs in the system that was being formalized were characterized by the following:

- They are autonomous/semi-autonomous.
- They each have unique and somewhat complex behaviors and states.
- They execute concurrently e.g., community of owners go about their business independent of construction companies. The same applies to individual construction companies.
- There is some degree of interaction between the constructs. When a project is being bid, companies interact with the owner or representative of the owner.

These characteristics qualified the problem for analysis using an ABM approach. A number of agents were created and commissioned to thrive within the model. They included:

- A construction industry agent (CI Agent).
- A bid manager agent.
- Company agents (small company, medium company and large company agents).
- Company of Interest Ambassador Agent (COI Amb. Agent).
- Resource Agents.

Figure 5.1 shows a schematic diagram illustrating the topology adopted in the implementation of the ABM. The resource agent is conveniently left out as it is an autonomous agent that thrives within the CI agent and interacts with company agents. The intention was to demonstrate that the interactions don't exist between company agents throughout the simulation. The communication that exists is between them and the Bid Manager agent during the bidding process. Bid Manager agent is referred to as a super-agent in this topology because it regulates the bidding process and the company agents follow. The COI Ambassador agents fit in this topology in the same way as company agents.

With the exception of the CI agent, COI Amb. Agent and the Bid Manager agent were formalized as singletons while other agents were linked to an agent population within which they thrived. The CI agent was the top-most level agent that represented the virtual construction industry. All other agents thrived within this CI agent in a fashion similar to what takes place in a real life setting. Each of these agents is discussed in more detail within this chapter, i.e., their design and implementation. The design specifications of the agents were presented using different aides', i.e., state charts, sequence diagrams and activity diagrams. The implementation of the agents was presented by showing screen shots of the modeling constructs used.

The model was developed in such a way that it was able to send and receive communications from another model (the Simphony/COI federate), i.e., it would operate as a standalone federate in a distributed simulation system.

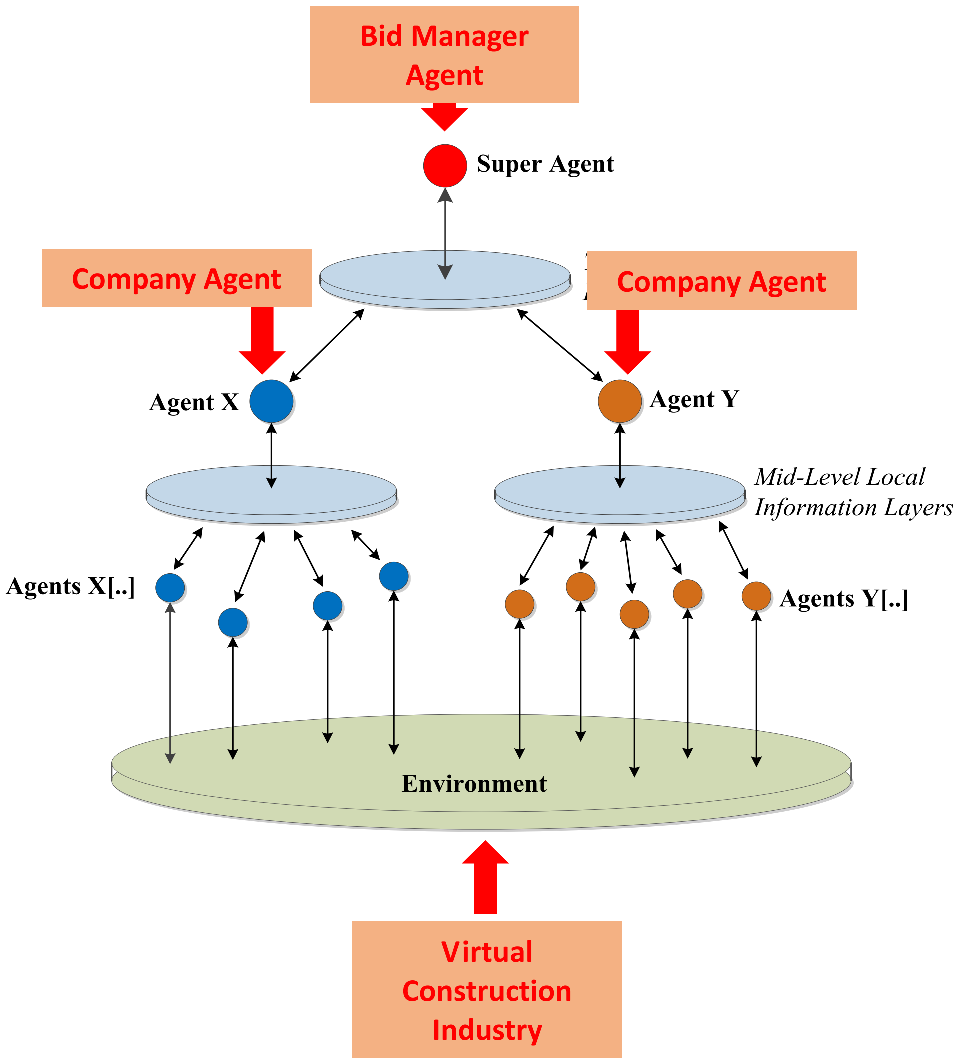


Figure 5.1: Topology Adopted for Bid Manager—Company Agent Interaction

The AnyLogic simulation system was chosen for the development of this model because it had a very advanced and easy-to-use ABM environment. In addition, the architecture of the AnyLogic simulation system was built in such a way that it facilitates the reusability and interoperability of models developed within it. This made the implementation easy for the HLA connectivity details to facilitate the model to behave as a standalone federate in a distributed simulation.

In the next sections, a discussion of the agents that exist within the model is presented. After a comprehensive discussion of these agents, it is believed that the architecture of the model will have been fully covered.

5.1 CONSTRUCTION INDUSTRY AGENT

The construction industry within which the construction contracting companies operate was modelled as a virtual environment. It is formalized, abstracted and represented as the top-most level agent in the model and was setup to handle a number of things such as:

- All the creation of company agents at start-up of the simulation.
- All the run-time communication between AnyLogic and Simphony through the RTI (HLA connectivity).
- Embedding controls used to facilitate the analyst to define their inputs prior to simulation and view their outputs after simulation has been completed.
- Serving as a container for all other agents—bid manager, company agents and company of interest.

In order to structure the presentation of this agent, it will be discussed in two themes, i.e., its basic roles and its advanced roles in the simulation.

5.1.1 HLA Connectivity

The AnyLogic model developed made use of the ABM paradigm to emulate the typical bidding behavior of companies within the construction industry. However, in order to achieve the overall objective of this study, a DES model that mimics the detailed processing of projects at the company of interest had to be developed and considered as an integral part of the larger model. In order to achieve this, a distributed simulation approach was used in which the AnyLogic model was a standalone federate and so was the DES model developed in Simphony.

For the distributed federation to execute seamlessly there needed to be communication between these federates. In the AnyLogic model, all the HLA connectivity details were embedded within the CI Agent. Code snippets that would facilitate a linkage to the RTI were written which could then be invoked anywhere within the AnyLogic model. Button controls were also provided which also made reference to these code snippets for HLA connectivity. Figure 5.2 shows these buttons.

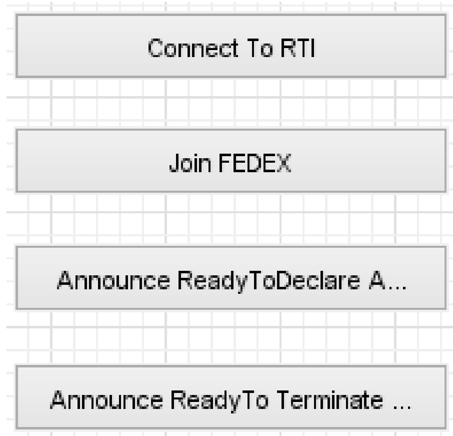


Figure 5.2: Buttons Embedded within the CI Agent for HLA Connectivity

A more in-depth discussion of the messaging sequence across the HLA framework was discussed in the chapter on the distributed simulation federation.

5.1.2 Agents Thriving within the Virtual Construction Industry

The construction industry agent (CI Agent) is a top-level agent that serves as an environment within which other agents operate, for example the resource agents, the Bid Manager agent, the company agents and the Company of Interest Ambassador (COI Amb. Agent). Figure 5.3 illustrates the constructs for each of these agents embedded within the CI Agent.

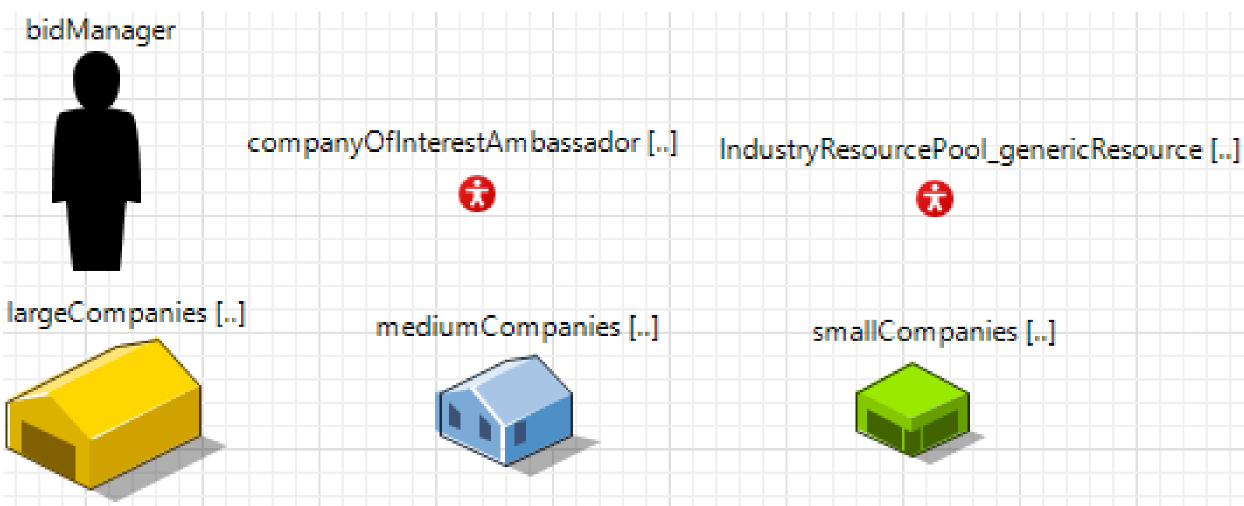


Figure 5.3: A Screen Shot of all the Agents Embedded within the CI Agent

5.1.3 Creation of Company Agents

During the pre-simulation phase, the analyst is expected to specify the total number of competitors that exist within their virtual construction industry. They are also expected to specify the proportions of those competitors that are small companies, medium size and large companies. These details are specified within controls embedded within the editor for the construction industry agent (see Figure 5.4).

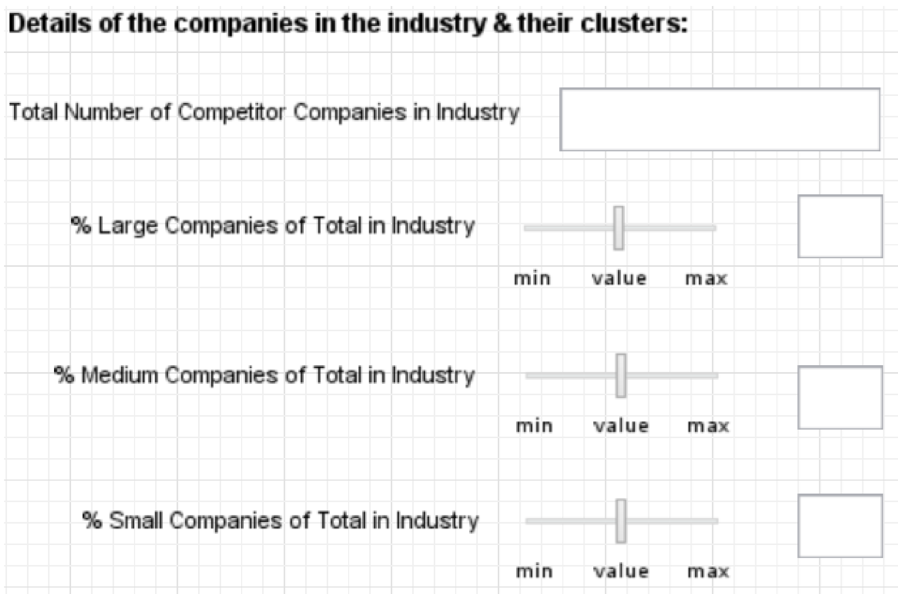


Figure 5.4: Controls used to Specify the Number of Company Agent Type to Create

5.1.4 Assignment of Attributes to Company Agents

At start-up of the simulation, the Agent that represents the construction industry creates all these companies. It then assigns some of the attributes to these companies. Other attributes that were set by the user prior to simulation are acquired by the company agents themselves. These attributes influence the behavior of these company agents. The attributes that are assigned by the construction industry agent are those which affect the bid/no bid behavior of the company. Examples include:

- The company's bidding strategy/pricing criteria.
- The company's appetite for competition.
- The company's tolerance levels for different owner traits.

- The company's affinity for project safety risks.
- The company's ability to take on complex projects.
- The company's tolerance for projects with varying engineering quality.
- The company's policy on the availability of all required resources for a project in order to bid.

The majority of these attributes are defined prior to simulation using a linguistic rating scale. The analyst also defines the proportions of small, medium size and large size companies that subscribe to each of the respective ratings. The objective was to have companies decide whether or not to bid a project that has the same attributes but unique values, based on all these criteria. The first criterion, i.e., the company's appetite for competition, was used to determine the bid/no bid decision in the final bid decision phase. The other four criteria were used to facilitate a bid/no bid decision in the initial bid decision phase. The assignment of values to these attributes for projects in the course of their creation is discussed in the section for the "Bid manager" agent. The features included within the CI agent to facilitate the assignment of some of these attributes are discussed in the following sub-sections.

5.1.4.1 Competition Appetite

A company's appetite for competition was one of the parameters used by the company agents to decide whether or not they were to bid a project.

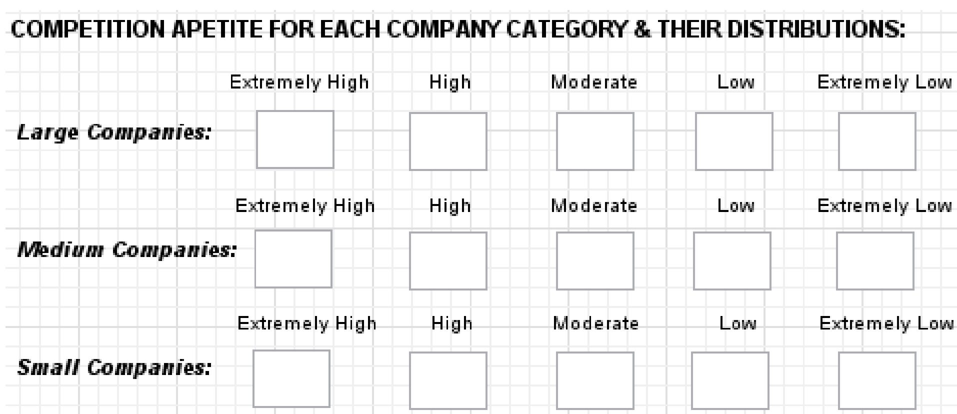


Figure 5.5: Controls for Defining the Competition Appetite for Company Agents

If the number of competitors exceeded the company's internal threshold for the maximum number of companies it can bid against, then the company would not bid the project. Figure 5.5

shows the controls that were provided within the CI agent to facilitate the user to specify the competition appetite for the company agent populations. An approach that made use of linguistic variable inputs was adopted for the convenience of the analyst that would be using this model. These linguistic variables are translated into Beta distributions, which are subsequently used to draw random deviates that are then used to derive each company agent's maximum number of competitors that can be tolerated. The total number of competitors to this company that exist within the virtual industry is used with the random deviate to generate a value for this parameter.

The analyst would have to specify the percentage of companies within each agent population that subscribe to a given value of the competition apetite linguistic variable. The analyst enters a number that represents the percentage value into the appropriate edit box.

5.1.4.2 Company Bidding Strategy

In the course of generating a bid price for a project that a company has decided to bid, there is a criteria followed. This may be one of the following:

- Maximize the chance of winning the project.
- Maximize the potential profit in case the project is awarded to the company.
- Maximize both the chance of winning the project and the potential profit from the project.

The strategy that the company adopts depends on the individuals responsible for running the company and the conditions prevailing at the company and in the industry at the time of bidding the project. A simplifying assumption was made in which the conditions prevailing at the company and within the industry at the time of bidding are assumed not to influence the strategy adopted when generating a price to carry with a bid. In addition, the strategy adopted by a company at the start of a period of operation is assumed not to change. These assumptions made it easy to incorporate the effects of bidding strategy on the bidding process for each company agent.

The controls shown in Figure 5.6 were embedded within the CI agent so that the modeller could define the strategy that the different company agents would adopt, prior to simulation. In this setup, the analyst would specify the percentage of company agents that subscribe to each bidding strategy (maximize chance of winning, maximize profit, or maximize chance of winning and profit) for each agent population (large companies, medium size companies and small companies). The slider controls are used to specify that percentage. The specified value then gets displayed within the edit box.

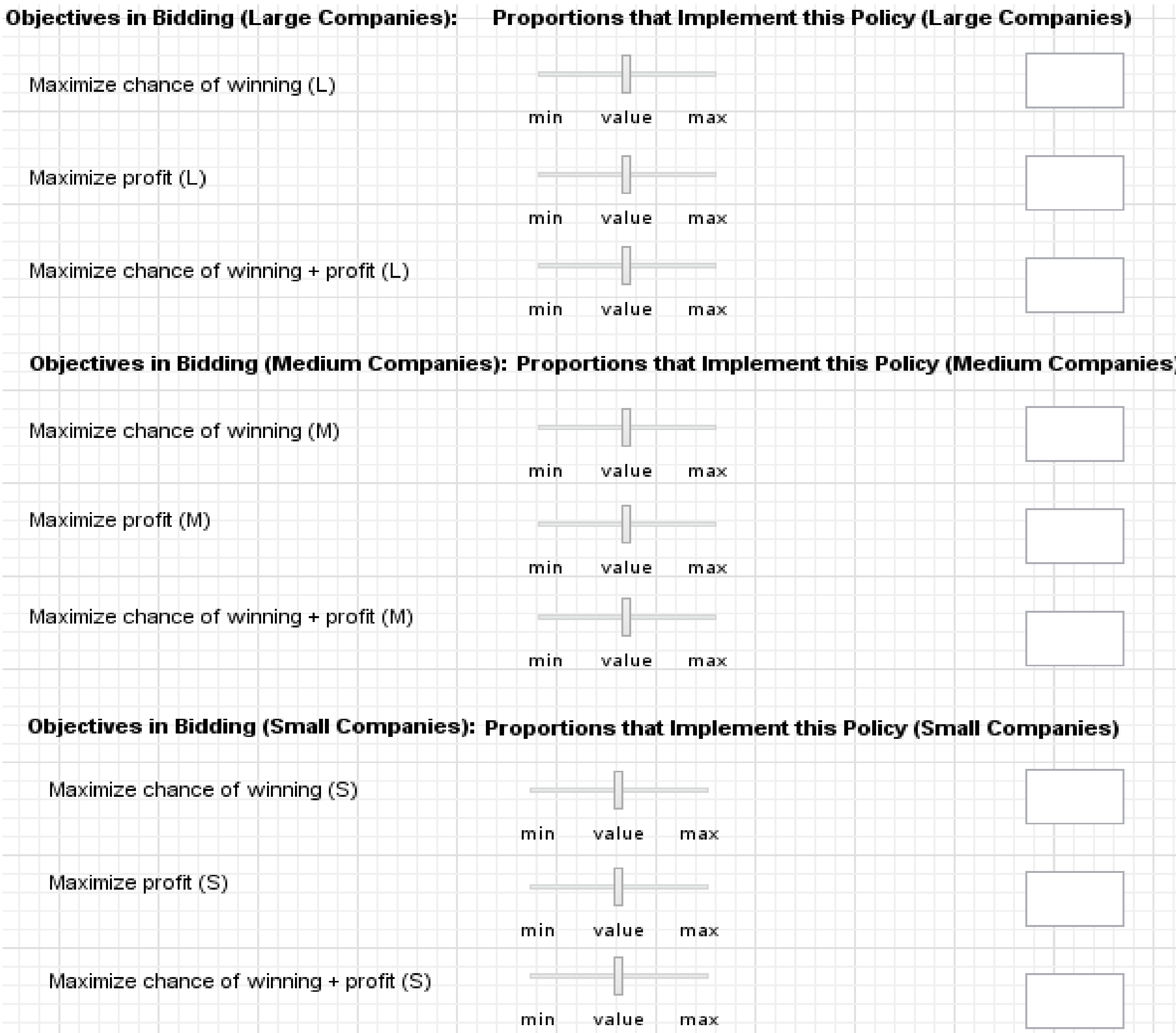


Figure 5.6: Controls for Defining the Bidding Strategy for Company Agents

5.1.4.3 Owner Trait Preference

The trait of the owner of a project to be bid can influence a compay's decision to bid or not to bid a project. This is typical in cases where the company has past experience with a returning owner. Owner trait was modelled as an index between 0.0 (horrible owner) and 1.0 (excellent owner). It was assumed that owners with bad traits were those that keep interrupting the work, generate lots of change orders, issue late payments and cause other interuptions in the course of executing their project. When projects are created by the Bid Manager, they are assigned an owner trait. Also, company agents would have an internal attribute for the owner trait preference against which they would check when deciding whether or not to bid a project.

Prior to simulation, the analyst would be expected to specify the distribution of owner trait preference amongst agents within specific company agent populations. The user controls presented in Figure 5.7 were setup to achieve that. A number indicating the percentage number of agents that subscribe to a specific owner trait tolerence was entered into the appropriate edit box. The agents that got assigned an extremely high tolerence for owner trait meant that they would tend to bid projects that had owners with poor traits. On the other hand, the company agents with an extremely low tolerance would tend to only bid projects with an owner that has excellent traits.

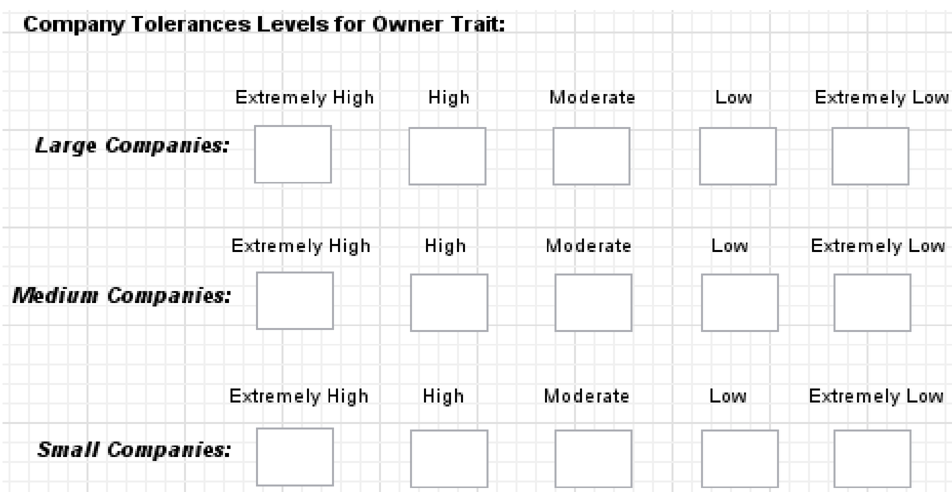


Figure 5.7: Controls for Specifying the Owner Trait Tolerence for Company Agents

5.1.4.4 Project Size Preference

The decision of company agents to bid or not to bid a project was constrained by another parameter referred to as the company's preference for a specific project size. This parameter was specified prior to simulation by the modeller using the controls shown in Figure 5.8 which were also embedded within the CI agent.

The setup of the controls for the rest of the attributes was similar to those presented and is therefore not discussed.

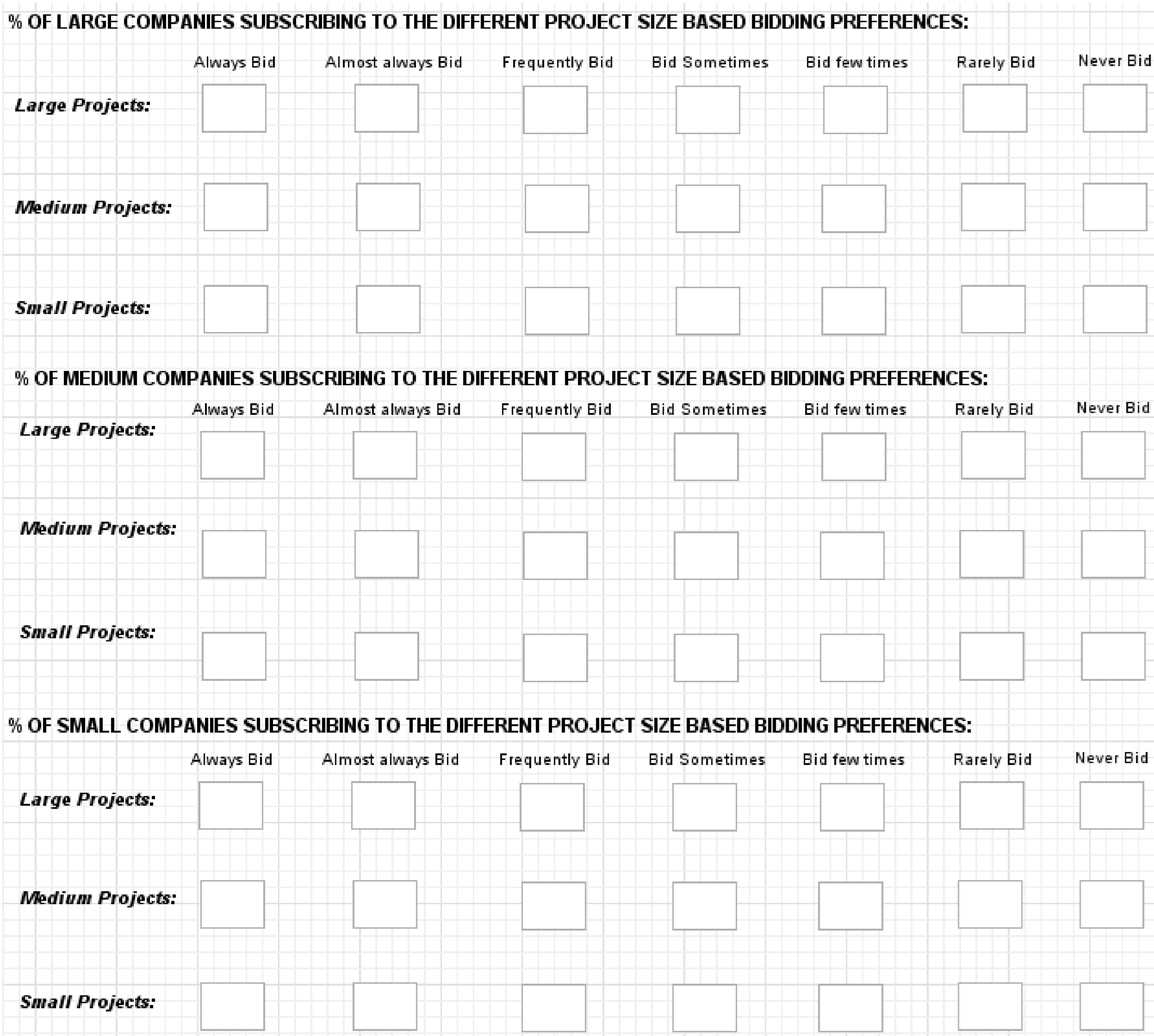


Figure 5.8: Controls for Specifying the Project Size Preference for Company Agents

5.1.4.4.1 Use of Liguistic Inputs in the Simulation

When simulation commences, these ratings converted into quantitative values on a scale that ranges from 0% to 100%. First, the linguistic variable assigned is translated into a Beta statistical distribution. To accomplish this, the findings in AbouRizk's (2013) MSc thesis were utilized. Table 5.0 summarizes details of these translations from linguistic variables to statistical distributions. The values for the linguistic variables used in AbouRizk's thesis are modified without distortions to suit this study.

Table 5.0: Linguistic Variables and their Corresponding Beta Distributions

Value of Linguistic Variable	Corresponding Beta Distribution
Extremely High	Beta(3.50,2.00,0.70,1.00)
High	Beta(2.60,2.40,0.50,0.90)
Somewhat High	Beta(2.70,2.80,0.35,0.75)
Low	Beta(4.00,3.99,0.00,0.60)
Extremely Low	Beta(2.00,3.60,0.00,0.25)

5.2 RESOURCE AGENT

In a real life setting, a typical construction industry will have a pool of resources from which companies will draw when they acquire work that requires resources to be performed. This shared resource pool setting is typical of industries that support the open shop or closed shop type of resource polling and replenishment. The open/closed shop arrangement is one in which the worker is not permanently stationed or owned by a specific company, but rather belongs to an association of sorts, such as a union for the case of a closed shop arrangement. Although companies tend to hire and retain highly skilled technical staff such as Engineers and project managers and a small pool of trades that they believe they can sustain, this scenario is not modelled in the application developed for this thesis. This was an assumption made to simplify the modeling process. Another assumption made was that the resource pool defined prior to simulation remains static throughout the simulation and does not change. In reality, workers leave the trade or profession and others enter. The rate of departure and entry vary from time to time causing the resource pool to fluctuate. However, this dynamic was not modelled.

Resources were explicitly represented in the model developed in this thesis because they affect the business operations of companies in two ways. These include:

- Their availability or unavailability constrains a company from bidding a project. A resource availability index is evaluated every time that a new project is created. If this index is above a company agent's internal threshold, then it bids the project, otherwise it does not.
- Unavailability of resources required by a project awarded to a company agent (at the time of award) delays the start of the project execution.

Resources within the industry were represented as autonomous agents. An agent population was created that represented the pool of resources at the industry level. The different agents represented the different trades or careers of works. Resources were represented as agents because it was easy for companies within the virtual construction industry to request, utilize and return resources with this approach. The quantity for each resource agent, i.e., the number of workers in a given trade was represented by an attribute of each agent referred to as *"servers."* Note that any resource that the modeller believes constrains the operations of construction companies could easily be represented using a specific resource agent. Workers are used as an example to explain how the resource agents were set up. Just like any other Agent, the resource agents were designed and implemented to exhibit a specific behavior. This entailed:

- The ability to register resource requests.
- The ability to be allocated and engaged on a specific project for a pre-defined amount of time.
- The ability to be returned and replenish the resource pool once the engagement on the project is completed.
- The ability to track the number of workers in its trade, those that are engaged on projects, and those that are free at any point in time.

There were two sets of constructs that were provided within the model to ensure the proper use of the resources defined within the resource agent population. One set of these constructs were embedded within the editor of the CI agent (at the top level), while the other set were embedded within the resource agent itself. The constructs embedded within the resource agent were meant to track the state of the agent, i.e., their extent of usage and availability. Figure 5.9 presents a screen shot of the constructs embedded within the resource agent.

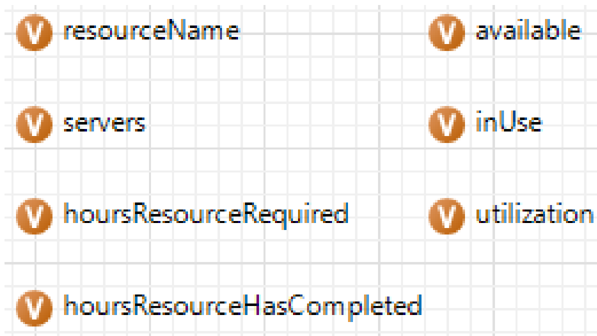


Figure 5.9: Modeling Constructs within the Resource Agent for Tracking its State

On the other hand, the constructs embedded within the CI agent were meant to mimic the placement of resource requests, queuing of these requests and fulfilling these requests (resource allocation). There were also constructs dedicated to tracking resources agents that had completed their engagement on projects and using these to replenish the resource pool. Figure 5.10 shows the constructs that were created to achieve this behavior.

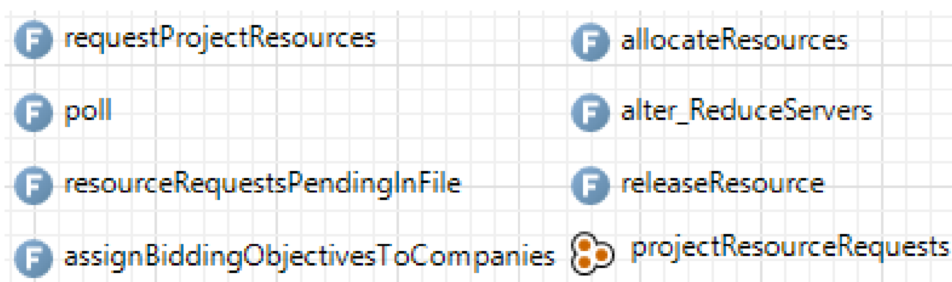


Figure 5.10: Modeling Constructs used to Mimic the Usage of Resources (Defined in CI Agent)

The creation and effective utilization of resource agents in the model was made possible through collaboration between the virtual construction industry agent, the bid manager agent and company agents. An activity diagram (see Figure 5.11) is used to summarize the interaction between the resource agent and all the other agents. Detailed explanations of each phase in the life cycle of the resource agents are discussed in sections that follow.

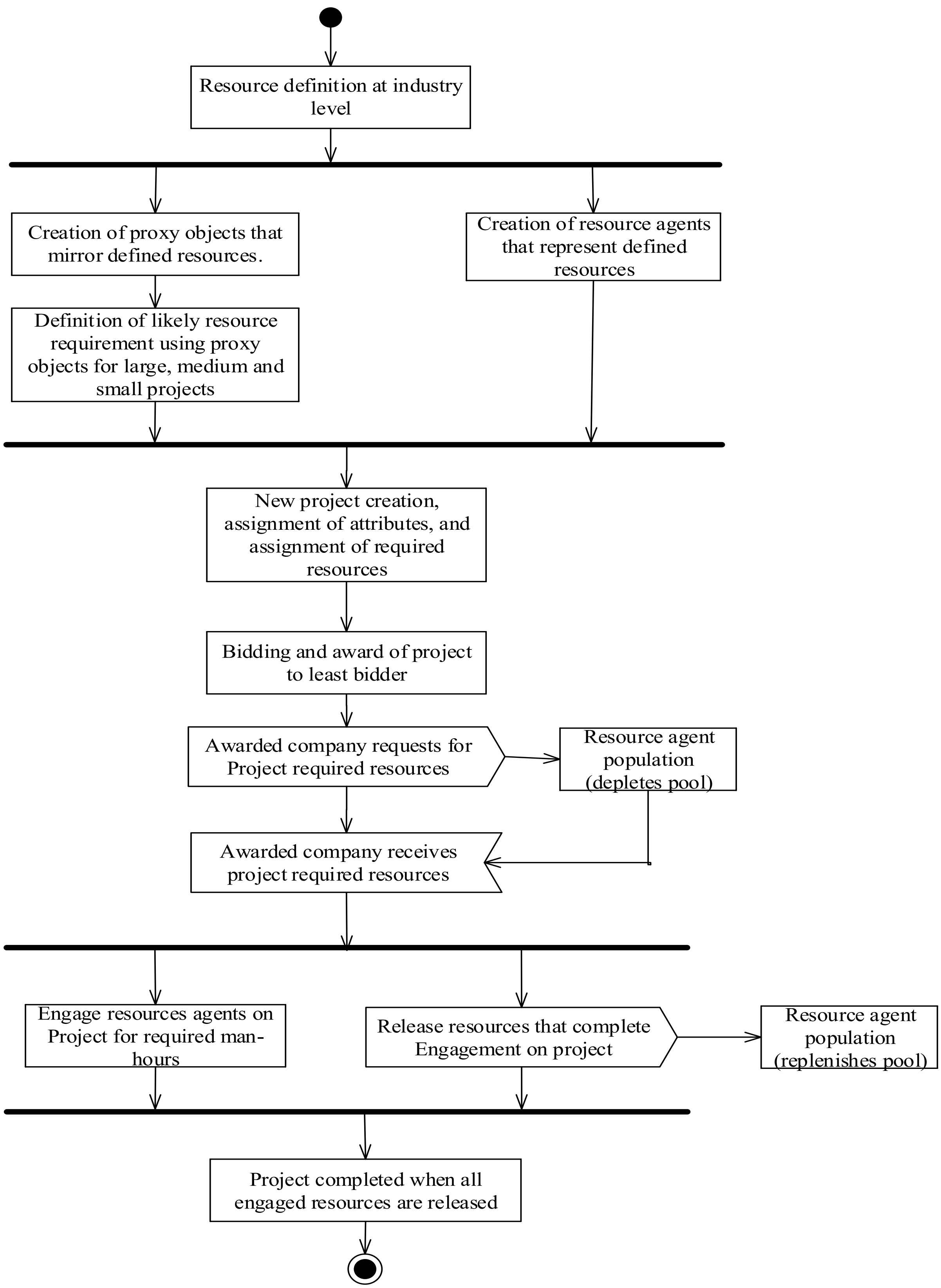


Figure 5.11: An Activity Diagram Summarizing the Lifecycle of an Agent in the Model

5.2.1 Definition of the Resource Pool for the Industry

The development of the model had to account for provisions that would facilitate the modeller to define the resources that constrain the operations of companies prior to simulation execution. In order to achieve this, a number of user controls were embedded within the editor of the CI Agent. Figure 5.12 shows a screen shot of these controls (list box, text box, slider and buttons).

This list box was provided for purposes of displaying the list of resources defined by the analyst at the virtual construction industry level at any point in time. This would enable analyst edit the resources to suit their needs. Providing this list box was also useful for purposes of verifying that the resources defined were actually created as agents in the simulation execution.

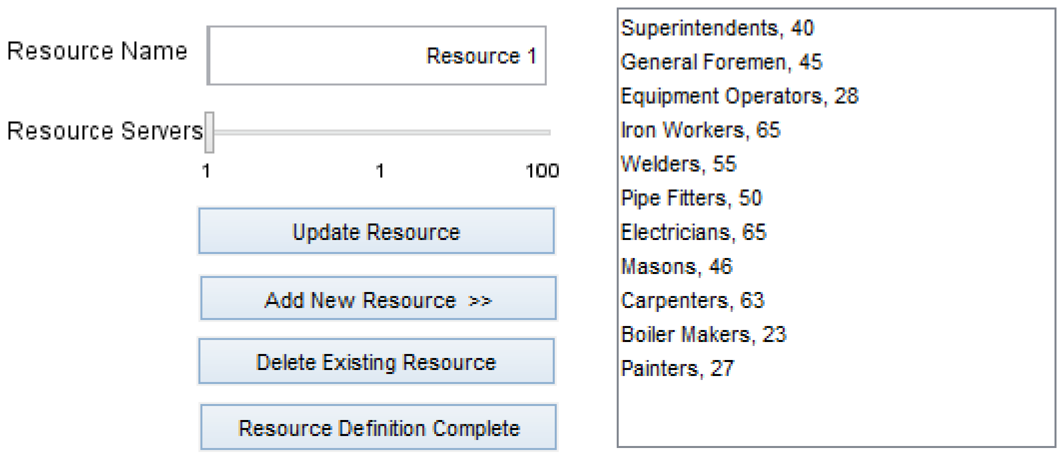


Figure 5.12: User Controls Embedded within the CI Agent for Defining and Editing the Resource Pool for the Virtual Construction Industry

The resource details (name and quantity available in the construction industry) defined within the list box control in Figure 5.12 represents the defaults setup in the application for the convenience of the analyst/modeller. In case these don't match the modeller's needs, they can be edited or deleted all together and replaced with other resources that the modeller would like to use.

The definition of likely resource requirements for the small, medium and large size projects was based on the resources defined at the industry level. No likely project resource requirement could be defined outside this set of resources. Details of how resource requirements were setup are discussed next.

5.2.2 Definition of Likely Project Resource Requirements

The purpose of modeling resources explicitly was to constrain the operations of the construction company. In order to successfully achieve that, the supply/availability of resources needed to be defined along with the demand for resources. The provisions made in the model for the availability aspect of resources have just been discussed in the previous section. This section discusses the provisions made for the demand side for resources that would complete the requirements for implementing the resource constrained strategy for company operations.

The demands for resources were expressed as project resource requirements. Prior to simulation, the analyst would have to define what the likely project requirements were envisaged to be for small, medium size and large projects. A complete definition of what these resource requirements would be involved specification of the following parameters.

- The name of the resource
- The likelihood that the resource would be required by a given project category (small, medium, and large)
- The likely quantity of that resource that would be required i.e., the number of servers
- The duration that the resource would be engaged on the project i.e., the man-hours

After the resources have been fully defined using these controls, they are committed to the model through the use of the *"Resource Definition Complete"* button. Once this button is clicked, no further changes can be made to them. Once definitions have been finalized, an agent is created for each resource trade defined. At the same time proxy object instances that reflect the resources defined for the construction industry are created and saved in collections for small, medium size and large projects. These proxy objects represent the likely resource requirements for newly created projects.

These likely resource requirements have probabilities associated with them which represent the chance that that specific resource will be required by a project. They also have statistical distributions that define the number of a given resource that a project will require and another statistical distribution used to define the number of man-hours that a given resource would be engaged on a project if identified as required. Figure 5.13 shows the controls that are used to edit the likely resource requirements for small projects. The medium size and large projects each have similar controls.

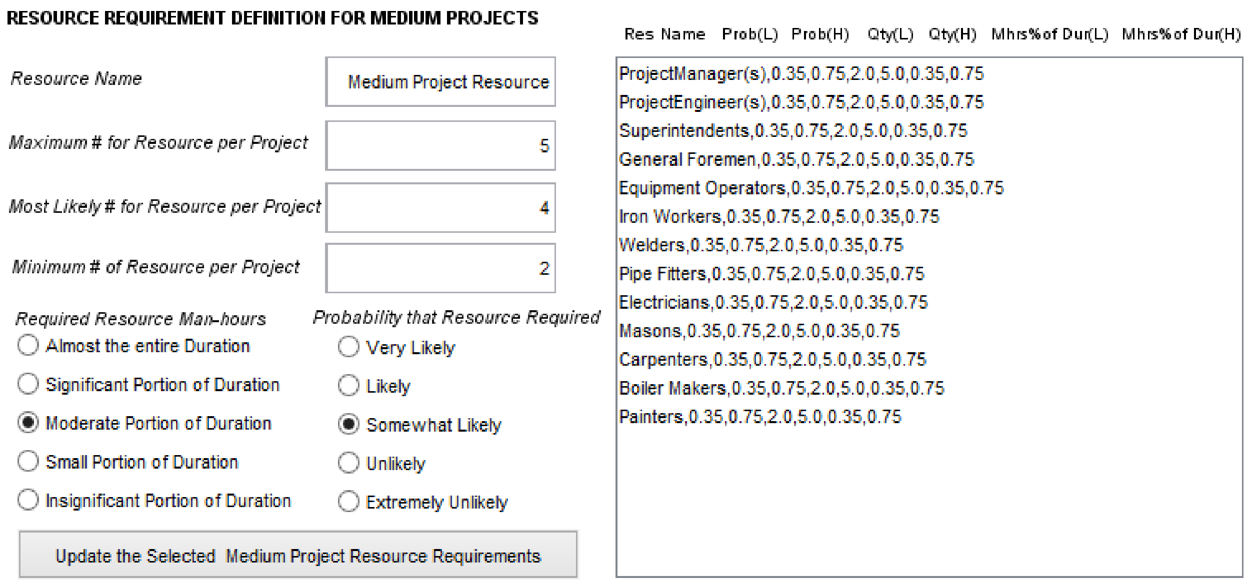


Figure 5.13: User Controls Embedded within the CI Agent for Defining and Editing the Likely Resource Requirements for Medium Size Projects

5.2.4 Resource Agent Engagement and Subsequent Release

The execution phase of projects in a real life setting requires resources at different points in time. Some resources are required at the start of the project while others at different times while the project is underway. Modeling the engagement of resources on a project at different times was possible but it would create a lot of computing overhead as a result of capturing and releasing resources at different times. Instead, the developments in this thesis adopted a simplistic approach which assumes that all resources are required at the project start. The release of resources is maintained as is in a real life setting i.e., resources are released at different times when their engagement on the project is completed.

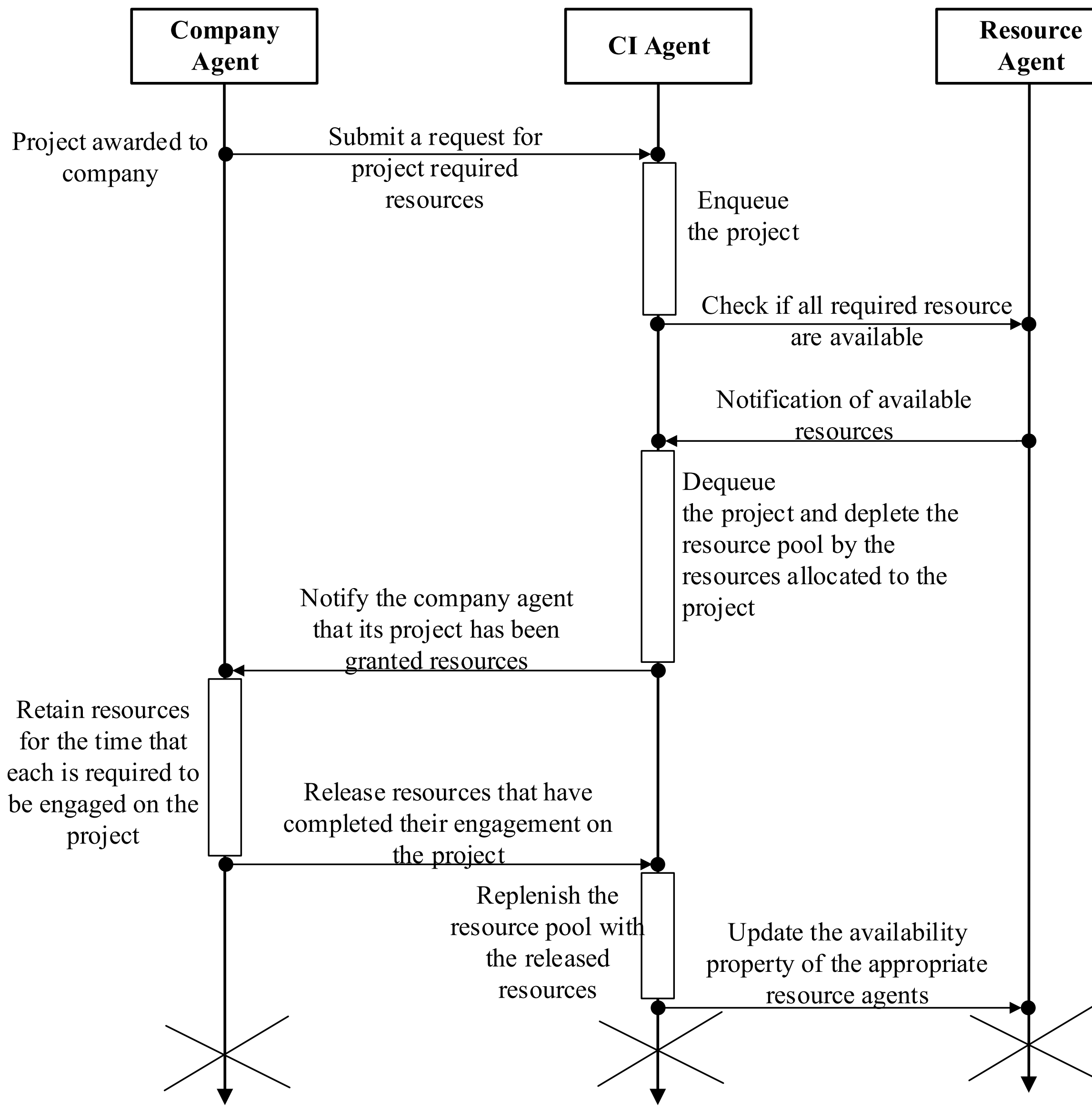


Figure 5.14 :Sequence Diagram Showing Resource Agent Capture, Utilization and Subsequent Replenishment

When projects are created, they are assigned resource requirements. Details include the quantity of each resource and the man-hours that the resources would be retained on the project. It was assumed that all servers belonging to the same resource i.e., quantities would be concurrently engaged on the project for the man-hours specified. A resource request would be queued until the requested quantity of each resource was available. Partial resource fulfillment was not accounted for in the modeling approach used.

The sequence diagram shown in Figure 5.14 shows the steps followed from project award, to resource request, resource allocation and usage and subsequent replenishment of resources. This sequence diagram (Figure 5.14) does not show details of the occurrence of parallel events to resource request soon after a project is awarded to a company. The activity diagram presented in Figure 5.15 bridges this gap.

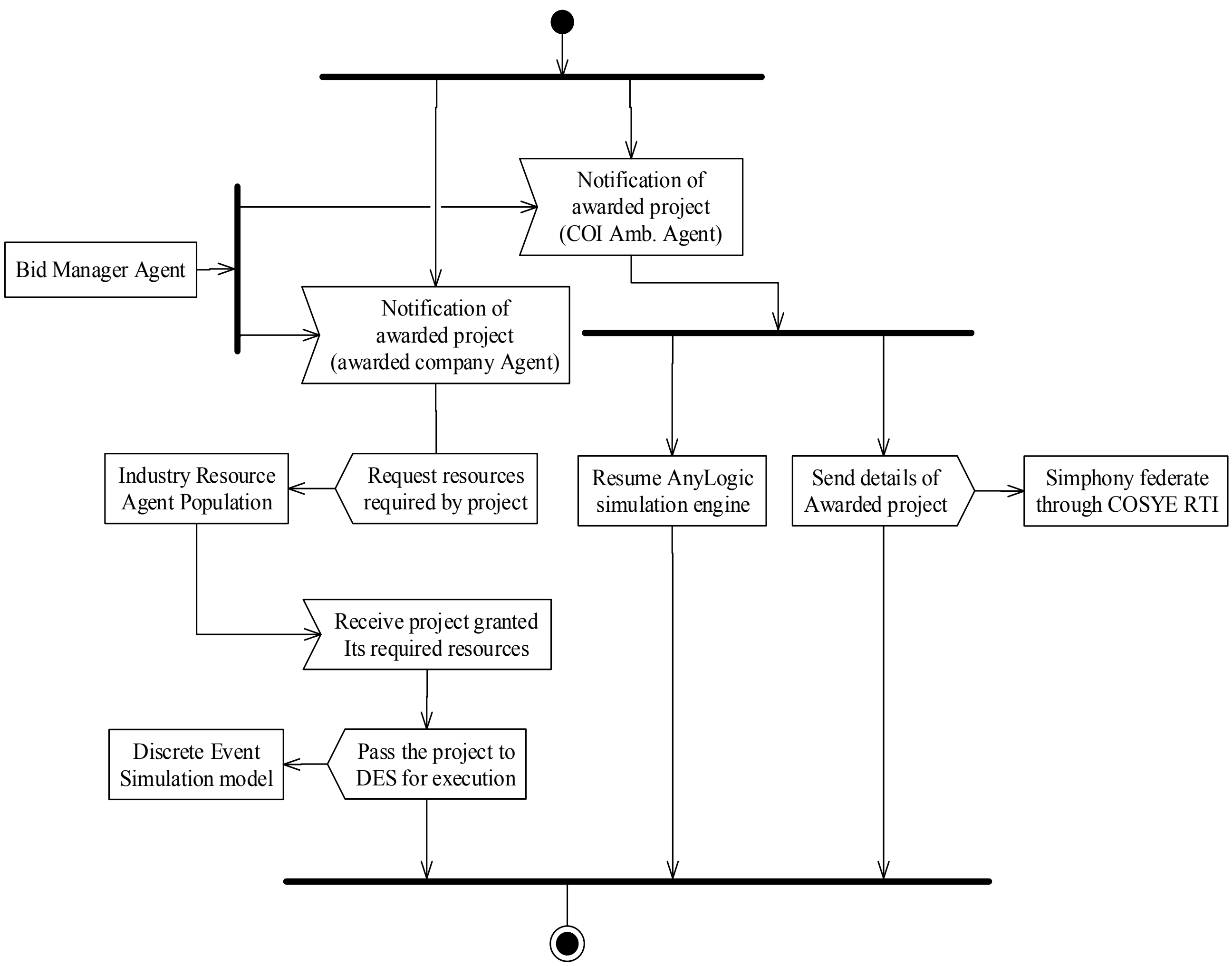


Figure 5.15: An Activity Diagram Showing Details of Events that Commence soon after a New Project is awarded

Soon after a new project is awarded, three events are simultaneously triggered. These include:

- The company that was awarded the project requests for required resource agents
- The AnyLogic simulation engine execution is resumed
- Details of the awarded project are sent to the Simphony federate

Each of these three execution threads (events) is discussed.

After a bidding cycle, the execution of the AnyLogic federate (and simulation engine) resumed the simulation progresses and leads to other new projects being created and those already created to be performed.

Details of the awarded project are passed on to the COI Amb. Agent which sends them to the Simphony federate. These project details are stored in a buffer awaiting a send communication indicating the allocation of resources to the project and awarded company.

At the time of project award, the company agent awarded the project requests for resource agents that are required to perform the project. This request is made by invoking a method embedded within the CI agent. Once this request is received, it is queued until the requested resource agents become available. The industry resource agent population then passes this project back to the company that requested resources. When available, the resource agent is passed to the company that made the request so that project execution can commence. Details of the granted resources are also passed to the COI Amb. Agent so that it can notify the Simphony federate. In cases where the Simphony federate was the awarded company, it pulls the project out of the buffer and starts executing it.

5.3 BID MANAGER AGENT

The Bid Manager Agent is a construct that was used within the model to represent the behaviors of the community of owners and their consultants. The community of owners within the construction industry plan and procure construction projects. They usually contract consults to handle the technical aspects of this procurement. Both constructs i.e., the owners and consultants are aggregated and formalized into the Bid Manager Agent.

This Agent is a singleton within the model and is responsible for the following:

- Scheduling the arrival of new projects
- Communicating details of these new projects to company agents
- Manage the bidding process and award the project to the least bidder

The project creation process and bidding processes will each be discussed in detail. First, a discussion is presented on how the behaviors of the Bid Manager Agent were formalized and implemented within the model.

A convenience way to view the behavior of any agent is to enumerate the envisaged states that the agent is likely to transition throughout its life time. From a simplistic standpoint, it can be said that in the course of the simulation, there will be times when the Bid Manager Agent will be busy and other times that it will be idle. For the times that it will be busy, there are other different possible states that it can assume. See "2" and "3" in Figure 5.16.

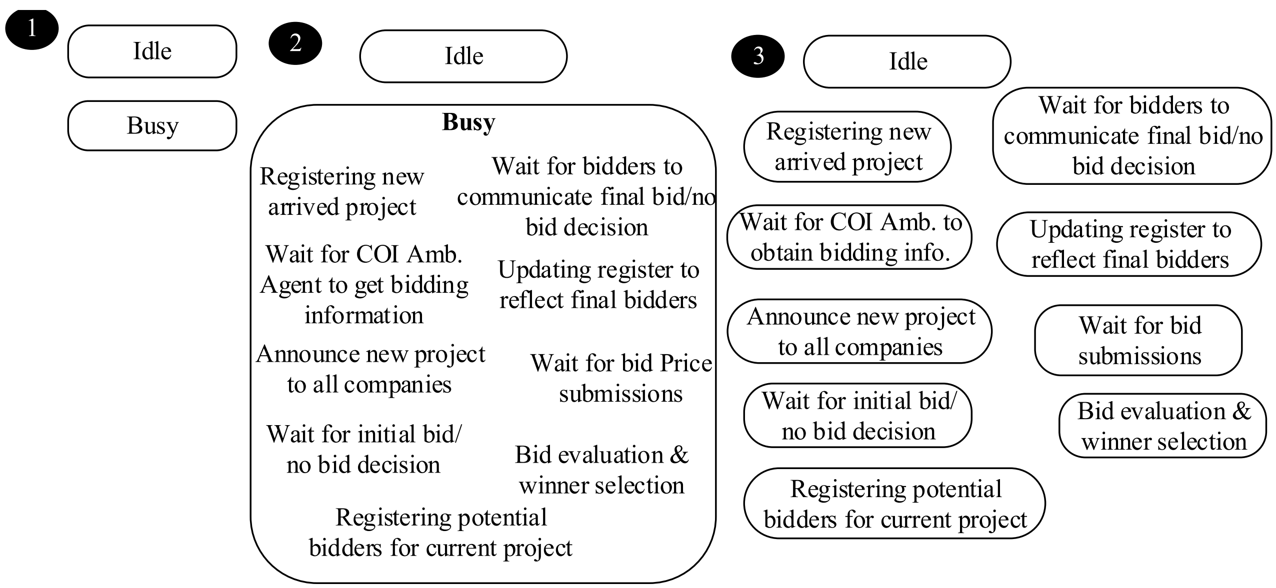


Figure 5.16: Initial State Transition Model for Bid Manager Agent in Bidding Problem

Illustrating the transitions between all these states using a figure would lead to one that is cluttered. In order to simplify the illustrations and explanations of the transition of the Bid Manager Agent through the various states, the states presented in Figure 5.16 are aggregated into four states that are still representative of the Agent's behavior. These include: a pure idle state, new project creation state, Pseudo Bidding state and a pure bidding state (See Figure 5.17). The pseudo and pure bidding states both represent the Bid manager Agent engaged in bidding. These states are triggered on new project creation. The Bid Manager is within the pseudo bidding state from the time a project is created and the company of interest ambassador agent sends a request to the Simphony federate, to the point in time that the company of interest ambassador agent receives information about the resource and file details. The end of the pure bidding state coincides with the award of the project competed for to the least bidding company agent.

Figure 5.178 is more comprehensive than Figure 5.17 in that it shows all possible transitions between states. It also demonstrates the hierarchical nature of the states that the Bid Manager Agents transitions through as the simulation progresses.

These state charts served as design aides and as a basis for modeling the behavior of the Bid Manager Agent. There are various options that could have been taken from this point onwards in implementing the Bid Manager's behavior. However, the one adopted is presented in the following paragraphs. The flow chart presented in Figure 5.17 illustrates how the Agent's behavior is all tied together and the logical sequence in which it exhibits its behavior.

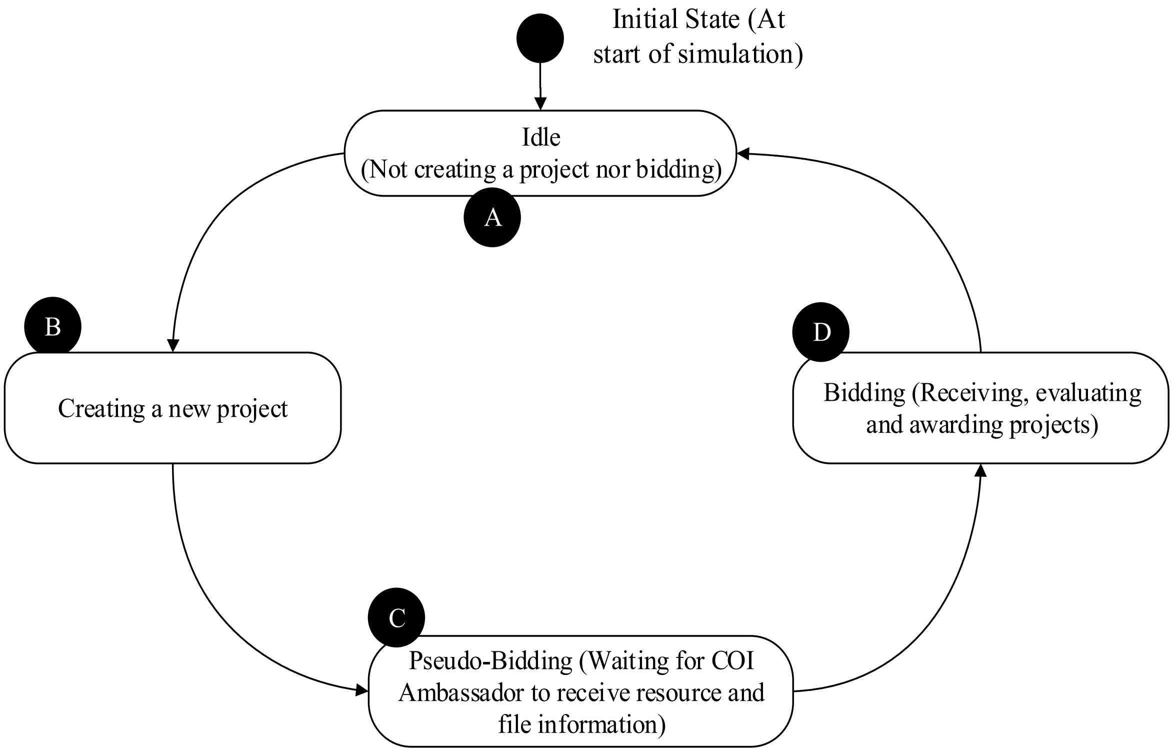


Figure 5.17: High-Level State Diagram for the Bid Manager Agent

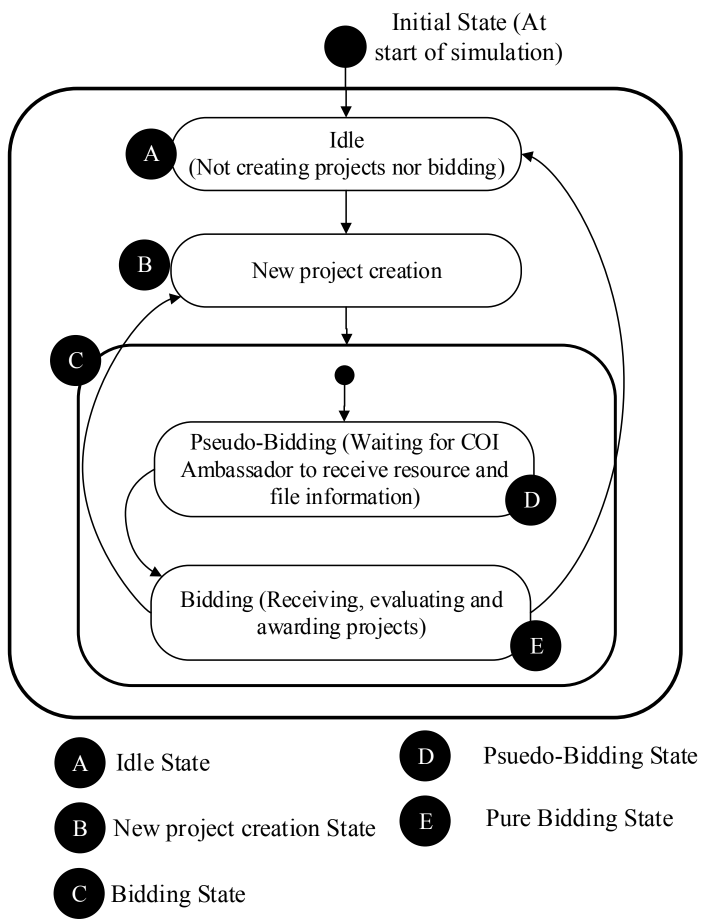


Figure 5.18: Hierarchical More Detailed State Diagram for the Bid Manager Agent

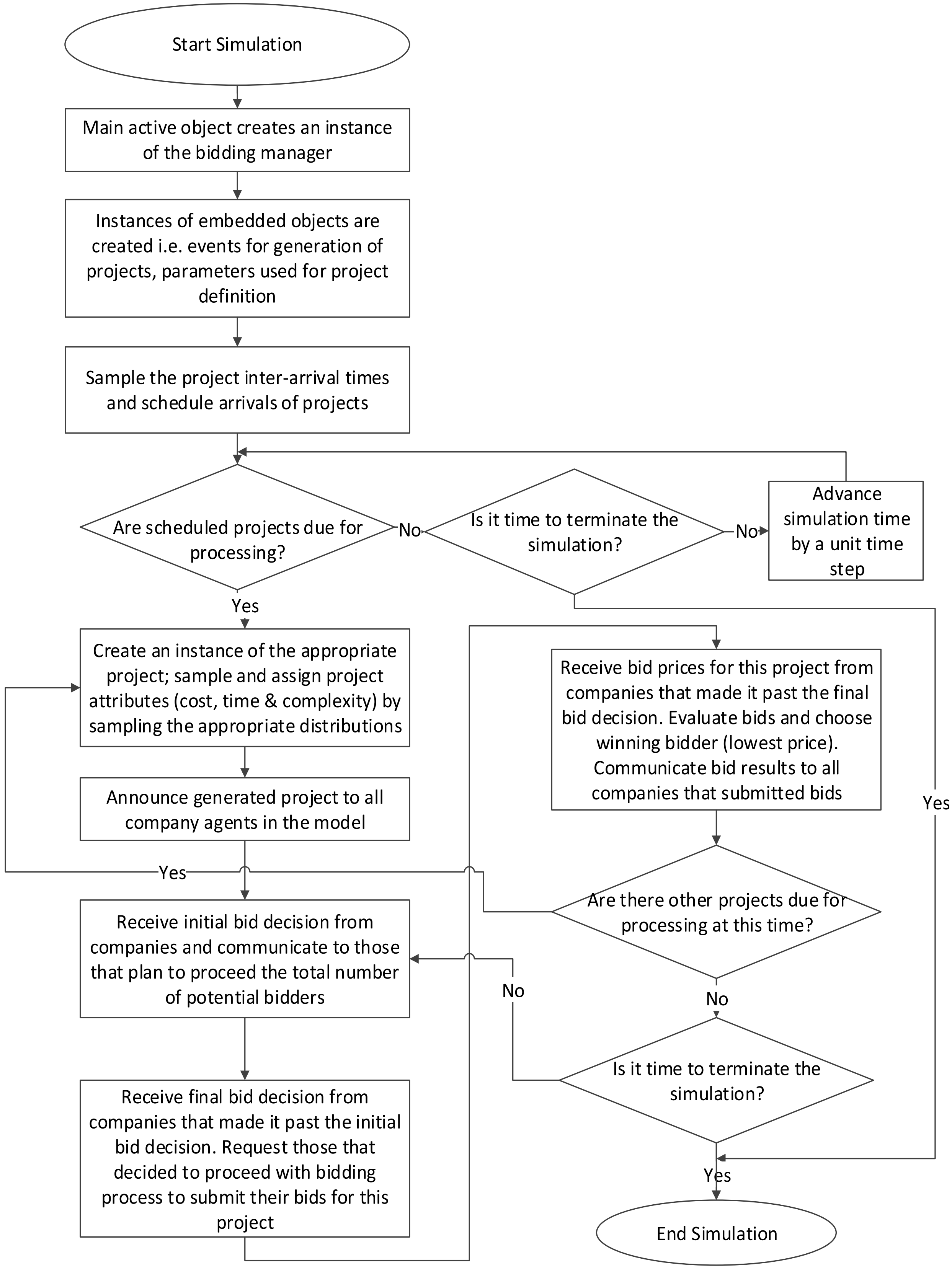


Figure 5.19: Process Logic for the Behavior of the Bid Manager Agent

5.3.1 Project Creation

Construction projects were sub-categorized into small, medium and large projects based on size, cost and resource requirements. The Bid Manager Agent schedules each of these independently by drawing values for their inter-arrival times from statistical distributions. When a project is created, it is assigned a number of attributes. These are drawn from the inputs defined by the modeller prior to simulation. These include:

- A name
- Cost
- Duration
- Size
- Resource requirements (quantities and Man-hours)
- Complexity
- Engineering quality
- Owner trait
- Safety risk

A screen shot that shows the modeling constructs responsible for medium size project creation is presented in Figure 5.20.

The model was setup to facilitate the analyst to define their inputs linguistically and in a Program Evaluation and Review Technique (PERT) type format. The parameters serve as place holders for the PERT inputs which are then transformed into Beta distributions that are then made use of during the simulation.

PERT provides for the analyst to provide their best guess for the optimistic value, pessimistic value and most likely value. Experts in PERT assert that these point estimates are easy to estimate for most people. The mathematical formulations proposed by Malcolm et al. (1958) were used to achieve the mappings from PERT point estimates to Beta distribution parameters. These are summarized in the Equations 5.0 and 5.1.

$$\alpha = \left[1 + \lambda \left(\frac{Most\ Likely\ Value - Minimum\ Value}{Maximum\ Value - Minimum\ Value} \right) \right] \tag{5.0}$$

$$\beta = \left[1 + \lambda \left(\frac{Maximum\ Value - Most\ Likely\ Value}{Maximum\ Value - Minimum\ Value} \right) \right] \tag{5.1}$$

After computing the shape parameters, a standard Beta distribution of the form shown in Equation 5.2 is constructed.

$$Beta\ Distribution = (\alpha, \beta, 0.0, 1.0) \tag{5.2}$$

This Equation is scaled to fit the boundaries defined by the PERT point estimates. The resulting Beta distribution is used in simulation computations. A typical value of 4.0 is typical for the λ parameter (Malcolm, Roseboom, Clark, & Fazar, 1958); (Herrerias-Velasco, Jose, Herrerias-Pleguezuelo, & Rene, 2010).

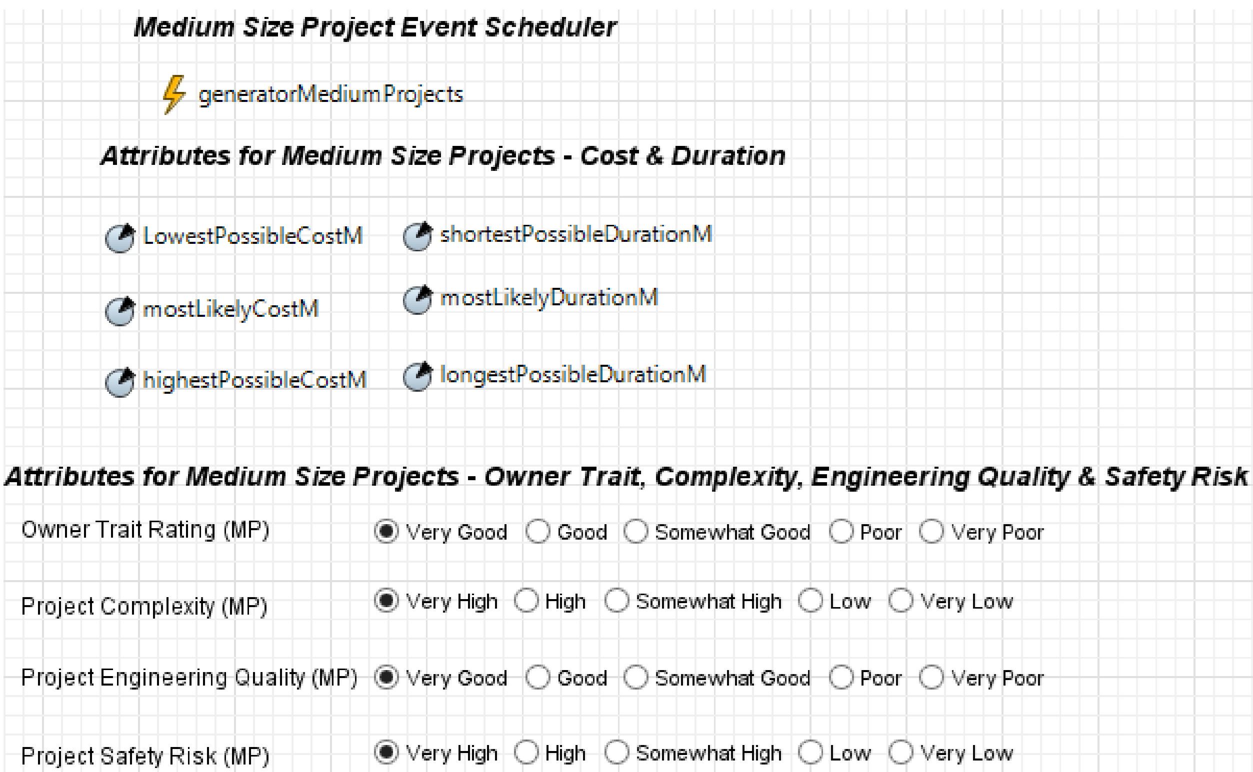


Figure 5.20: Controls for Definition of Medium Size Project Attributes in the Bid Manager Agent

The controls provided for linguistic input were also used as a basis for generating Beta distributions that were also made use of in the simulation. Although the linguistic variables used by AbouRizk (2013) don't exactly match these, the same idea can be applied here for their mapping to Beta distributions (See Table 5.1).

Table 5.1: Linguistic Variable Definition and Corresponding Beta Distributions Based on AbouRizk (2013)

Linguistic Variable	Corresponding Beta Distribution
Very Good	Beta(3.5,2.0,0.7,1.0)
Good	Beta(2.6,2.4,0.5,0.9)
Somewhat Good	Beta(2.7,2.8,0.35,0.75)
Poor	Beta(4.0,3.99,0.0,0.60)
Very Poor	Beta(2.0,3.6,0.0,0.25)

The linguistic variable approach and that of PERT were both used to simplify and facilitate the user in expressing their knowledge into model inputs. The Beta statistical distributions were then used to draw the values that were then assigned to newly created projects.

The resource requirements for the newly created projects are determined from the likely resource requirements for each respective project category i.e., small, medium and large size projects. The likely resource requirements are defined by the analyst prior to simulation using list box controls embedded within the Construction Industry Agent.

Next, the details of the bidding process are discussed. The bidding process follows a systematic logical sequence every time that a new project is created. This process culminates in the award of the project to the winning bidder i.e., the company that submitted the lowest bid price.

5.3.2 Solicitation, Evaluation of Bids and Project Award

After projects have been created, companies have to strive to acquire them through a competitive bidding process. This process has to be well regulated by the Bid Manager Agent. In order to do so, a number of modeling constructs were embedded within this Agent that emulated a behavior in line with that envisaged. These modeling constructs are summarized in Figures 5.21.

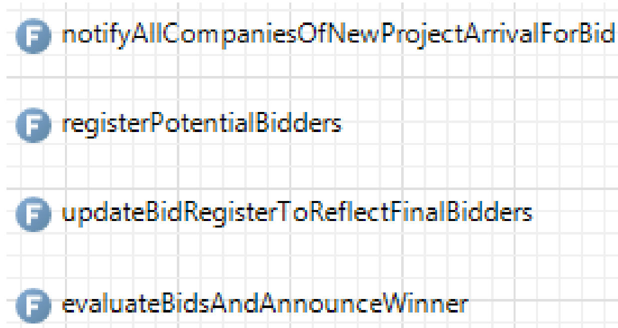


Figure 5.21: Modeling Constructs in the Bid Manager Agent for Modurating the Bidding Process

All these modeling constructs represent the interaction between the Bid Manager and the company agents within the virtual construction industry. The most convenient and efficient way to design and implement such interaction starts with the development of a sequence diagram. One was developed in this thesis and used as a basis for implementations in the application development. This same diagram was used in the implementation of the company agent too. In order to avoid a repetition in the discussion, the presentation of this sequence diagram is differed to the section that discusses the company agent development.

5.4 COMPANY AGENTS

Construction contractor companies that carry out their business operations within the construction industry were represented as autonomous/semi-autonomous agents in the model. This is because they execute concurrently as the simulation advances and they each have unique behaviors and attributes. For the sake of convenience, companies were sub-categorized into small size, medium size and large companies. It was envisaged that these would be set apart by their project production capacity. These company agents represent the legitimate competition that the company of interest to the modeller has to face.

The company whose performance the modeller/analyst is interested in tracking is referred to as the Company Of Interest Ambassador Agent (from now on referred to as COI Amb Agent). This name was conveniently chosen because this agent represents the interests and behavior of the company that the analyst is interested in tracked at a detailed level. In addition, this company was modelled as a standalone agent because in addition to the ordinary behavior that the other company agents exhibit, it is engaged in retrieving information from and sending information to the Simphony federate through the COSYE RTI from time to time as the simulation progresses. This COI Amb Agent was also modelled as a singleton because it can only be one for every model instance that is executed.

The Simphony federate represents the operations that take place at the company of interest (also referred to as COI) at a more detailed level. This federate keeps track of the performance of COI as it goes about its business operations. Details of the information exchanged between the COI Amb Agent and the Simphony federate include:

- Projects awarded to the COI Amb Agent by the Bid Manager Agent
- Projects awarded to the another company Agent (i.e., a competitor to the COI Amb Agent) by the Bid Manager Agent
- Static bidding information – these include attributes that are internal to the COI that don't change as the simulation progresses. They could be threshold values for maximum number of competitors that the COI can bid against, thresholds for complex projects or high risk projects.
- Dynamic bidding information – these include details of the COI that change during simulation that affect its decision to bid/not to bid projects. They could be resource availabilities and utilizations, prevailing work load etc.

In the following sections, details of agent behaviors that are shared and unique the company agents (competitors) and COI Amb Agent are presented and discussed.

5.4.1 Shared Behavior amongst Company Agents

There are behaviors and attributes that are common amongst all construction contractor companies when viewed from a high level. This is because they share similar objectives, and ideologies to guarantee their existence. Examples of behaviors that are shared include:

- Their Bid/no bid decision sub-process and
- The bid price generation sub-process

These two constitute the bidding process for any construction project. In order to set the stage for discussions on the two sub-processes, the overall behaviors into which these sub-processes fit are discussed.

5.4.1.1 The Bidding Process

The bidding process is triggered by the arrival of a new project within the virtual construction industry. As soon as a project arrives, the COI Amb Agent retrieves information that it requires to make a bid/no bid decision from the Simphony federate. Thereafter, the Bid Manager Agent notifies all company agents within the virtual industry of the arrival of this project. This marks the commencement of the bidding process. The process is regulated by the Bid Manager Agent. All company agents are just participants in the process. After each company agent has been notified of the new project arrival, it goes through a sequence of sub-processes. These include:

- Making an initial bid/no bid decision
- Making a final bid/no bid decision
- Bid price generation
- Notification of an awarded project and request for project required resources
- Commencement of project execution when required resources are granted

Each of these sub-processes is summarized in the methods shown in Figure 5.22. The parameters provide represent the unique attributes of each company agent. This affects the behaviors of the company agent when it is engaged in a bidding process for an agent.

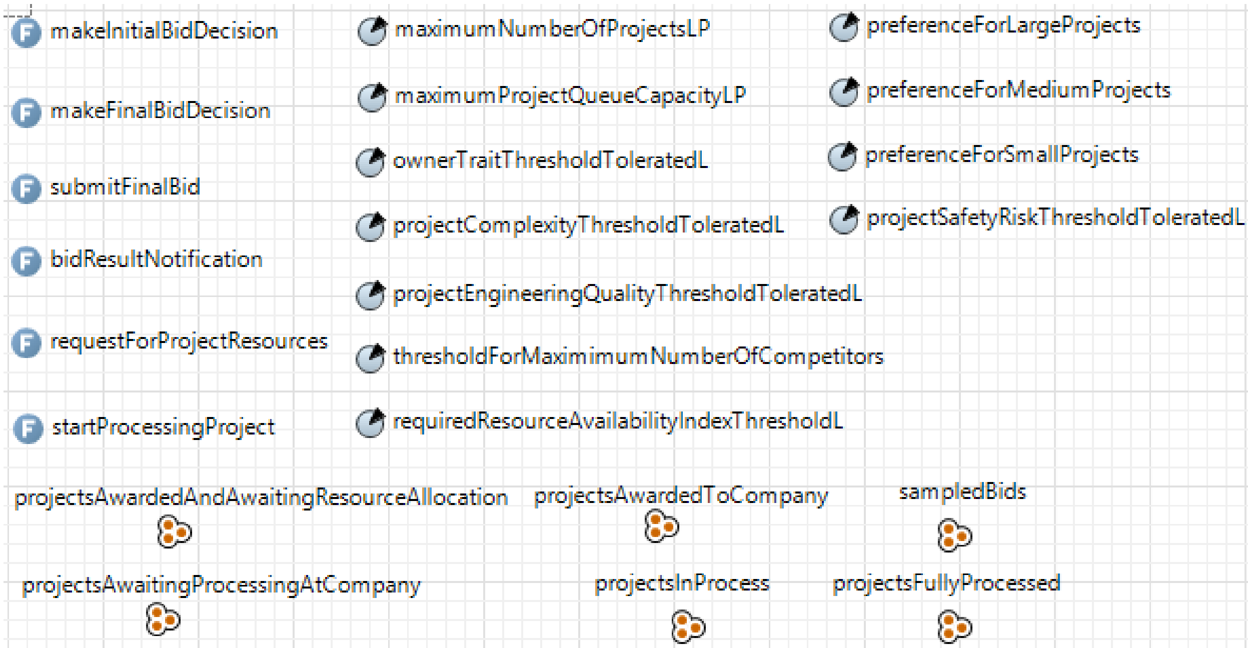


Figure 5.22: Controls within a Company Agent for Modeling its Behavior

5.4.1.2 Sequence Diagram for the Company Agent-Bid Manager Agent Communication

The communications that take place in the course of a bidding process are between individual company agents and the Bidmanager agent. The logical sequence for the communication protocols are summarized in the sequence diagram presented in Figure 5.23.

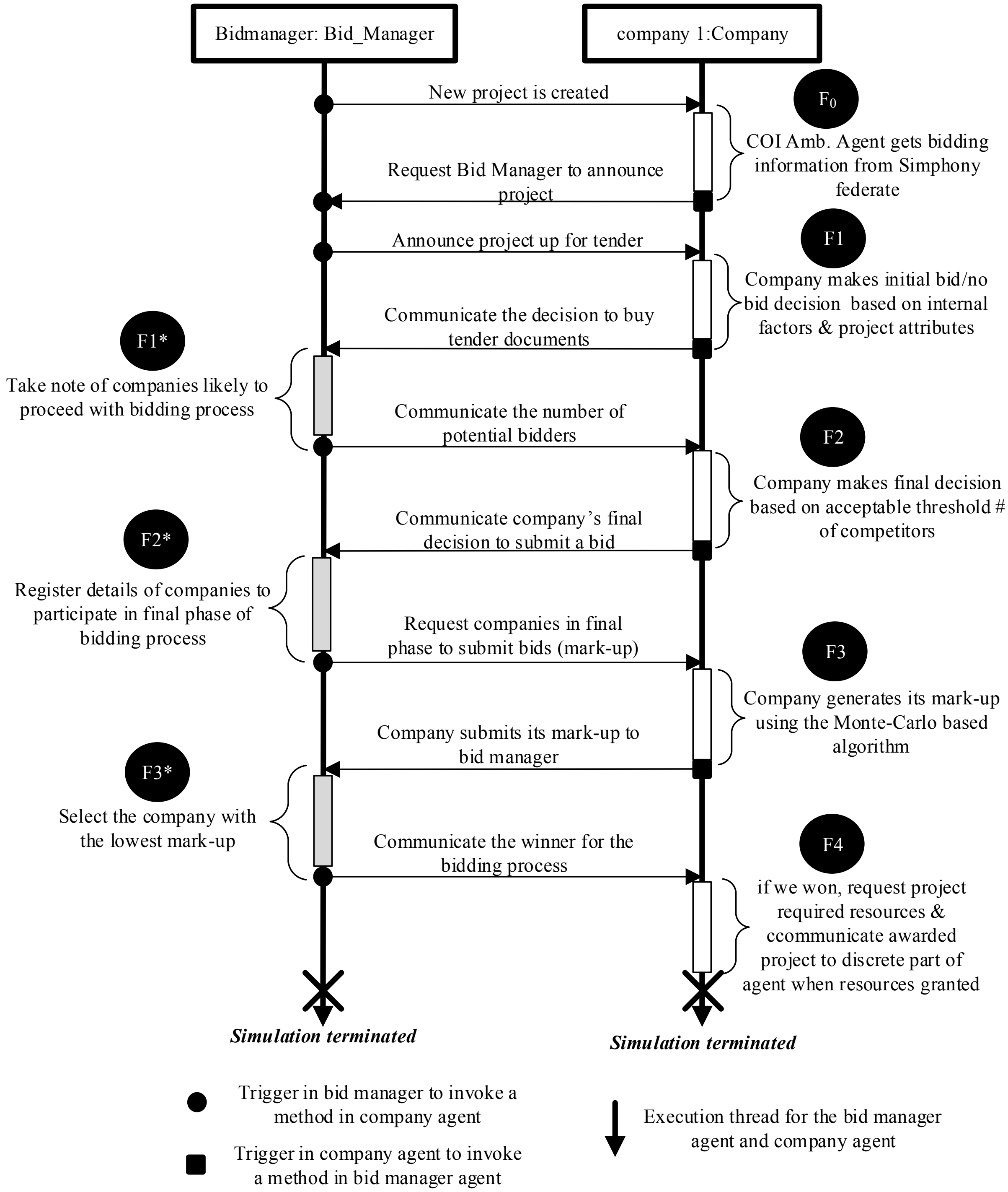


Figure 5.23: Message Sequence Diagram used to implement the Bidding Process Behavior

Designing and detailing the Communication between agents is a critical part of the development process of an agent-based model. In the bidding problem, communication exists only between the "bid manager agent" and the "company agents". There is no communication between the “company agents”. This is consistent with real life practice in the construction industry because communication between companies engaged in a competitive bidding process would amount to collusion or bid rigging, something that is illegal (Doree, 2004) (Davis, 2004) (Brockmann, 2011).

Details of the sequence followed during “bid manager agent”-“company agent” interactions is summarized in Figure 6.23. This graphical notation helps us to effectively design and detail agent communication and event ordering in this system. At the start of simulation, instances of the “big manager agent” and “company agents” are created. The event elements within the “big manager agent” are turned on and generate new project arrivals based on inter-arrivals sampled from a statistical distribution. When a new project arrives, it triggers the interaction between the “bid manager” and the “company” agents.

A new project arrival acts as the stimulus within the “bid manager agent” to start the bidding process. However, before the bidding process is started, the project is passed on to the COI Amb. Agent do that it stores it within an internal buffer. On receiving this project instance, the COI Amb. Agent saves it as part of its state and then sends a request to the Simphony (COI) federate for information that it can use to facilitate it in making the bid/no bid decision. This request is sent via the COSYE RTI. This type of information is that which changes as the simulation progresses, hence it should be obtained at the point in time when the bid/no bid decision is to be made. The Simphony (COI) federate gathers the required information (illustrated as F_0 in Figure 5.23) and sends it to the COI Amb. Agent.

As soon as the COI Amb. Agent receives this information; it updates its state and notifies the Bid Manager Agent. This notification involves the COI Amb. Agent passing the Bid Manager agent a reference to the newly created project that it had stored in its buffer and a request for the Bid Manager to commence the bidding process.

The formal bidding process is then started by the Bid Manager Agent invoking the “makeInitialBidDecision” method of each “company agent”, including the COI Amb. Agent. This method is labelled “F1” in the sequence diagram and is one that each replicated object of the company agent will execute. The argument for this method is the new project that just arrived in the market. The code within this method (F1) internally assesses the current situation in the agent in order to make a preliminary decision whether to bid or not to bid on the newly arrived project. This initial decision is based on the capacity of the company to handle concurrent projects, the project number currently under process at the company including those that are queued and resource utilization thresholds. In *AnyLogic simulation software*, when the numbers of entities arriving at a port exceed the element's capacity, a run-time exception is thrown. To avoid this, we make sure that there is no likelihood of projects entities exceeding the capacity of the discrete modeling agents within each company agent. This justifies our decision criteria in the initial bid/no bid decision from a simulation perspective. From a real life point of view, construction companies will tend not to commit to projects that cause them to exceed their production capacities because it could result in performance failure and default in contract performance. Once a decision has been made on whether to bid or not to bid, this decision is set to a new cloned project instance by setting one of its properties to *"true"* or *"false"*. Also, the replicated object tags the cloned project with its name so that the bid manager can get back to it in case communication is to proceed i.e., the agent has opted to proceed with the bidding process. Thereafter, the replicated object of the company agent calls the *"registerPotentailBidders"* method within the bid manager agent. It passes the cloned project to the *"registerPotentailBidders"* method (F1*) as an argument. All this takes place within the *"makeInitialBidDecision"* method (F1). Thereafter, the "F1" method is exited.

The F1* method adds the communication received from the company agents to a linked list within the project instance. If an agent indicated that it would like to proceed with the bidding process (initial bid decision property of the project is "true"), its name is added to the project instance linked lists. This function (F1*) keeps doing that until all the replicated objects in the small company cluster, medium size company cluster and large size company cluster have all sent in their communication. A logic "if statement" within the F1* method is used to achieve that behavior. After communication from all agents is received, number of potential bidder's property is updated for the project instance. Thereafter, the bid manager loops through all the replicated objects that expressed their interest in proceeding with the bidding process and it invokes the *“makeFinalBidDecision”* (F2) method within these company agents. F2 is passed the project instance as an argument.

Each replicated object that receives this message (invoking F2), then looks at the potential number of bidders and checks this value against its internal threshold of the maximum number of companies it can compete against. If the potential number of bidders exceeds this threshold, the replicated company agent will not bid and will therefore set the final bid decision property of a cloned project to “false”. Otherwise, it will set it to “true”. It will also set its name to the cloned project. Thereafter, it will call the *“updateBidRegisterToReflectFinalBidders”* method (F2*) of the bid manager agent. In making this call, it passes its cloned project to F2* as an argument. Method F2 is exited at this point.

When F2* method is called, the bid manager will create an updated list of companies that made a final decision to proceed with the bidding process. The bid manager agent adds the names of replicated company agents that opt to proceed with the bidding process, to a list of final bidders for the project being bid. This is also a property of the project instance. When the manager has received communication from all the potential bidders for that project, it updates that project instance with the number of final bidders. It then loops through all the replicated company agents, identifies the ones that want to proceed with bidding that project and it calls the *“submitFinalBid”* method (F3). Then it exits the F2* method.

The F3 method is where the company replicated objects that are participating in the final bidding process estimate their markup for the project being tendered. The method receives the project as its argument and this project indicates the number of final bidders. The markup estimation algorithm uses a Monte Carlo Simulation based algorithm to estimate the markup. This is explained in further detail in the following section. The generated markup depends on the number of final bidders and the objective of the company when bidding the job. After the estimate for markup is made, the agent clones the project, sets its name, final number of bidders and the generated markup and then passes it as an argument while invoking the F3* method (*“evaluateBidsAndAnnounceWinner”*) within the bid manager agent. The F3 method is then exited.

When the bid manager agent's *"evaluateBidsAndAnnounceWinner"* method is invoked, it registers the bidder name under the list of submitted bids for that project. It also registers the corresponding markup. The bid manager keeps track of the number of bids submitted (equal to the number of F3* method calls that have been made by the company agents for that particular project). When the expected final number of bids is submitted, the bid manager agent evaluates the bids and selects a winner i.e., the bid with the lowest markup. The bid manager then writes the details of the bid winner name and their markup into the appropriate project properties. The bid manager agent then sends the bid results to all the final bidders by calling the *"bidResultNotification"* method (F4) of those company agents. When making the call to method F4, the project instance is passed as an argument.

The last method in the company agent (F4 – "*bidResultNotification*") involves the agent object checking whether it won the bid by interrogating the "*name of winning bidder*" property of the project. In situations where the company agent instance has won the bid, it makes a resource request to the Agent that represents the resource pool at the industry level. This request is queued within the resource agent until the requested resources are made available to the project. When the resources are granted to the project, the resource agent passes the project back to the company agent. It is at this point that the company agent creates a project entity that represents the awarded project and passes it on to the Discrete Event Simulation model that is embedded within it.

There is an exception to this for the COI Amb. Agent. When awarded a project, it sends an interaction to the Simphony federate via the COSYE RTI that contains information about this project. When this federate receives information of this awarded project, it creates an instance of a project entity which it also passed to the DES model embedded within the federate. Once the DES receives the project entity, it commences execution when there is sufficient production capacity. The COI Amb. Agent also passes information about projects that have been awarded to its competitors, to the Simphony federate so that it is in position to track the tendering performance of the company of interest.

5.4.1.3 State Transition Models for Company Agents

As the simulation progresses, companies are engaged in the bidding process while at other times, there are idle with respect to the bidding process. To illustrate this, a number of State charts are presented in Figures 5.24 and 5.25. State transition models typically communicate all the possible states that an agent can assume and the fashion in which it transitions between these states. State charts serve as a basis for modeling the behavior of any agent.

From a bidding perspective, we are certain that a company will either be engaged in a bidding process or will not be engaged in a bidding process. These represent the two states that any company agent will be in during the execution of the simulation. In order to explicitly represent the behavior of the agents during their life span, one needs to understand the different sub-states that the agent transitions during simulation. The high-level and detailed states that the company agent transitions through are summarized in Figure 5.24.

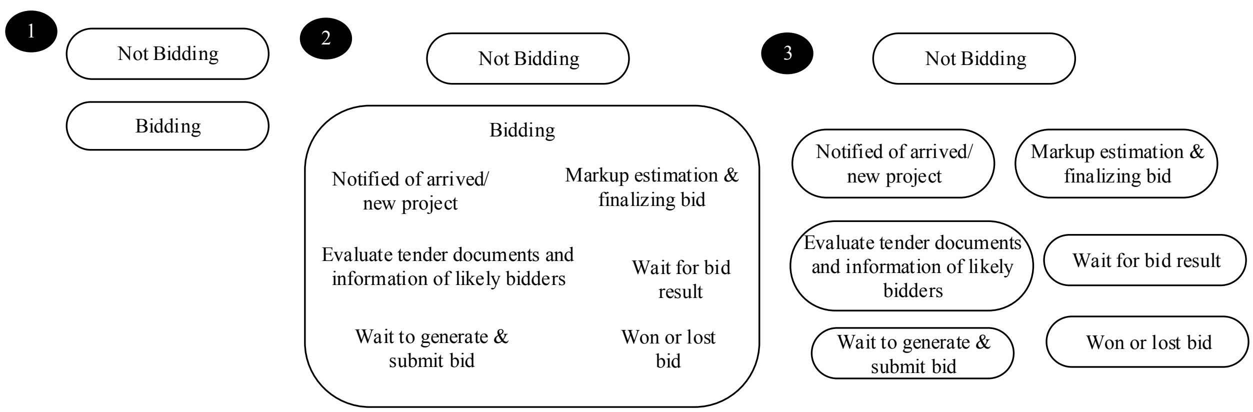


Figure 5.24: Initial State Transition Model for Company Agents in Bidding Problem

Figure 5.24 represents the initial state transition for the company agents during their life span. This is the product of the first phase of formalizing the agent's behavior (Part 3 of Figure 5.24). The initial transition model (Part 3 of Figure 5.24) starts with detailing part 1 of Figure 5.24. This sub-model is then detailed further to get the sub-model shown in part 2 of Figure 5.24. This is achieved by sub-dividing the bidding state into server other states. These include: (1) notification of a new project in the market, documents, (2) evaluation of tender documents and information of the likely bidders, (3) wait to generate & submit bid, (4) markup estimation and bid generation, (5) waiting for the bid result and (6) notification of winner i.e., bid loss or win. Then the final sub-model is generated (shown in part 3 of Figure 5.24) by doing away with the "bidding state" given that all the 6 enumerated states represent the "bidding state" of the company agent.

This initial state transition model shown in Figure 5.24 is very limited because it does not give any information about how the agent transitions from one state to the other. This information is provided by generating a final transition state model (See Figure 5.25) for the bidding problem. Another piece of information that is important is the stimulus that triggers the transition of the agent from one state to another. Figure 5.25 is an embellishment of Figure 5.24, sub-model 3. It illustrates the sequence of transition between states and some high level information of what triggers this transition.

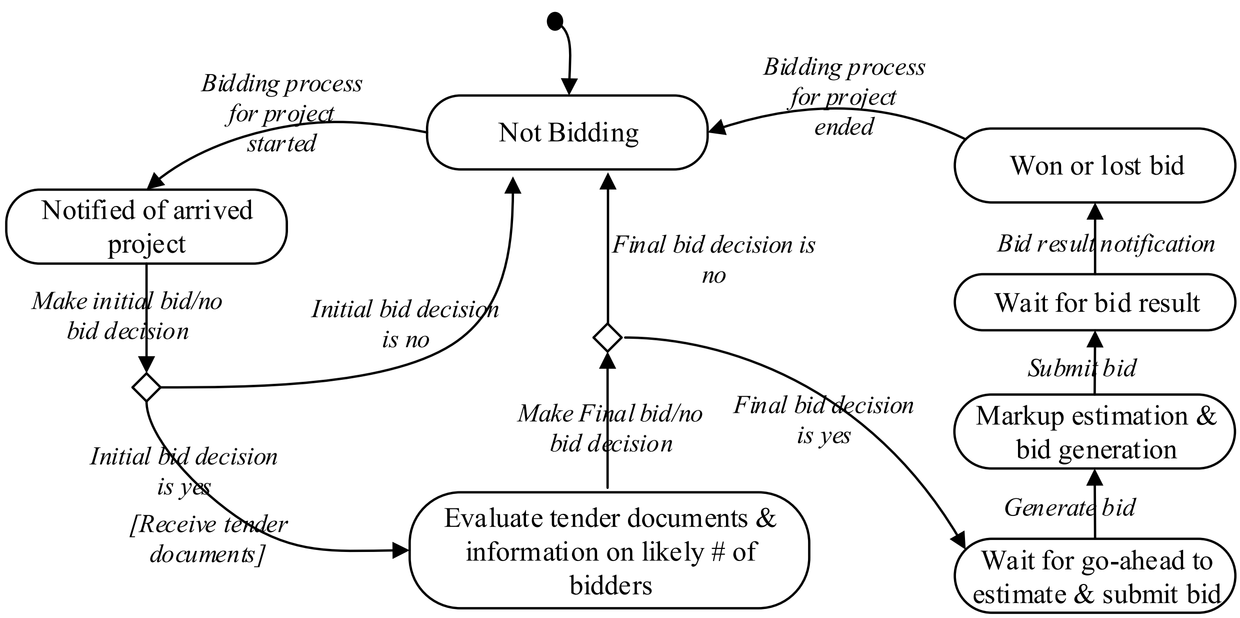


Figure 5.25: Final State Transition Model for Company Agents in Bidding Problem

Next, details of each of the sub-process that are entailed in the bidding process are discussed. It involves a decision to bid or not to bid and bid price generation for the companies that decide to bid.

The decision to bid a project is a highly complex and dynamic one and there have been several studies done on this subject. This thesis adopts a novel approach that considers the company's internal business strategies and the attributes of the project. This approach was adopted because it closely represents what takes place in reality and was easy to implement within the developed simulation application.

For purposes of convenience, the bid decision process was sub-divided into two phases; an initial bid/no bid decision phase and a final bid/no bid decision phase. Each of these are discussed in the following sections.

5.4.1.4 Initial Bid/No Bid Decision

The initial decosion to bid a project is based on a number of criteria that depend on a company's internal strategies and the attributes of a project. The attributes of the project are assigned by the Bid Manager Agent on project creation. The internal strategies of a company agent on the other hand are defined prior to simulation and remain static throughout the simulation. These are defined as threshold values for project attributes. If project attributes don't meet the company's criteria (i.e., are not within the company's threshold values), the company does not bid that project. Each criteria is evaluated as shown in the flow chart in Figure 5.26 and each has to be fulfilled. The criteria evaluated in the initial bid phase include:

- Project size
- Project owner trait
- Complexity
- Engineering quality
- Safety risks
- Availability of Project required resources

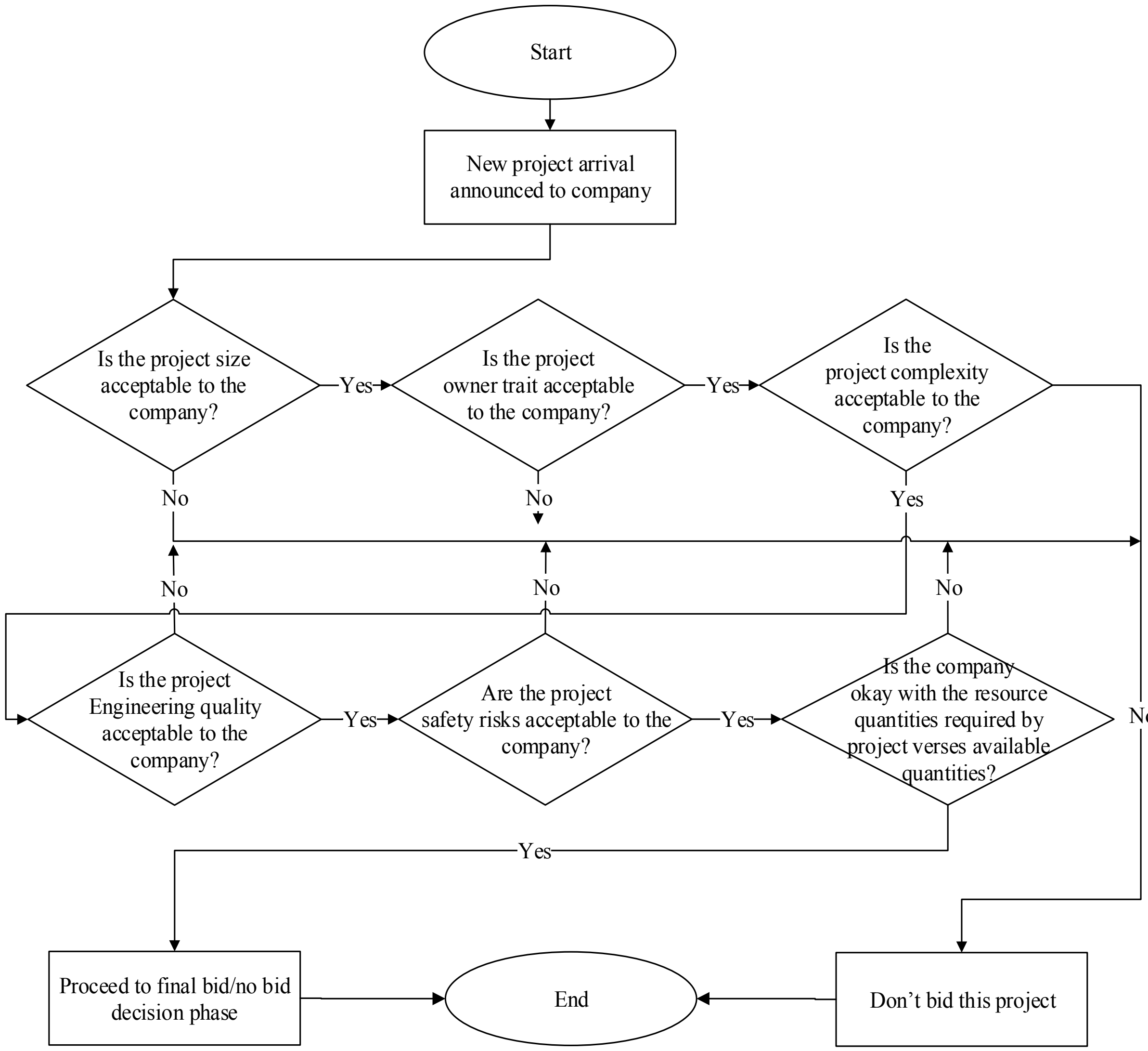


Figure 5.26: Process Flow Logic used By Company Agents to make their Initial Bid/No Bid Decision

Company agents that realize all their criteria fulfilled proceed to the final bid decision phase, otherwise, they do not bid the project. The final bid/no bid decision phase is discussed next.

5.4.1.5 Final Bid/No Bid Decision

If the company opts to proceed with the bidding process, it moves on to the final bidding process. The final bid or no bid decision is made based on two criteria namely:

- The maximum threshold number of competitors and
- The utilization of the resource units processing that type of project (i.e., the need for work).

The maximum threshold number of competitors represents the nature of competition which the company is willing to take part in. A small threshold valve for the number of competitors implies that that company is not willing to take part in stiff competition. A large number on the other hand implies that the company is open to taking part in furious competition. If the potential number of bidders on a specific project is less than the company's maximum number of competitor's threshold, the company will opt to proceed past final bid decision phase to the actual bid completion. In cases where the potential number of bidders exceeds the company's maximum number of competitor's threshold, an assessment is done to establish the company's need for work. The final bid decision is then based on this need for work.

The mean utilization of the resource units gives an indication the company's appetite for work. This parameter is used in such a way that a high mean resource utilization value implies a reduction in the company's need for work. On the other hand, a low mean resource utilization implies that the company's resources have been redundant most of the time and it is therefore in great need for work. So, when the potential number of bidders exceeds the company's maximum threshold number of competitors, a random number that is compared to the mean utilization to model those unique cases in which company's aggressiveness for work is not reduced by high utilization of its resources. It is okay to implement this stochastic component because it is consistent with the behavior of some companies in practice. The random number is sampled between 0.0 and 1.0 and if it is greater than or equal to the mean utilization of resource units, the company disregards the threshold number of competitor's violation and proceeds to bid on the project. The rationale is that the company is in urgent need for this specific work and can disregard its threshold number of competitor constrain. In case the threshold number of competitors is exceeded and the sampled random number is less than the mean utilization of resource units that process that type of project, the company opts not to proceed with the bidding process because there is a violation on the level of competition intensity that the company can afford to be a part of and also the company is not in urgent need of work because it has been busy on average.

If the company decides to proceed past this bidding phase, it moves on to implement the algorithm for generating an optimal bid for that project. The next section of this paper is dedicated to discussing the algorithm that will be used for generating a bid price.

5.4.1.6 Bid Price Generation

The price that construction companies carry within their bids depends on numerous factors. It will depend on the attributes of the project that the company is bidding, the competition that the company is facing; it's current and recent past workload and its access to resource to perform the project. Some companies will also consider more sophisticated criteria such as the anticipated number of project in the future. The bid price also depends on the people that are involved in the cost estimation process and those that make the ultimate decision of the price to carry in the company's bid.

In this thesis, an approximate method is proposed for purposes of quantifying the likely price that a company would carry if it were to bid for a project in varying conditions. In order to simplify this process of quantifying this value, some assumptions were made. The effects that individuals estimating the project would have are ignored and not explicitly modelled. In order to represent the uncertainty that comes along with the other factors, a statistical distribution is used from which random variates are drawn that represent the bid that the company would submit under the prevailing conditions.

Beta distributions are used to achieve this because they are very robust in terms of their scale and shape parameters. At the start of this bid generation process, an initial beta distribution is constructed that has shape parameters alpha and beta set to 1.6. A value of 1.6 was selected to ensure that the distribution has closed ends (any value greater than 1.0 could have been selected). The values for the alpha and beta shape parameters are made equal so that the distribution is symmetric at the start of the process. Figure 5.27 shows the shape of a standard beta distribution with shape parameters 1.6.

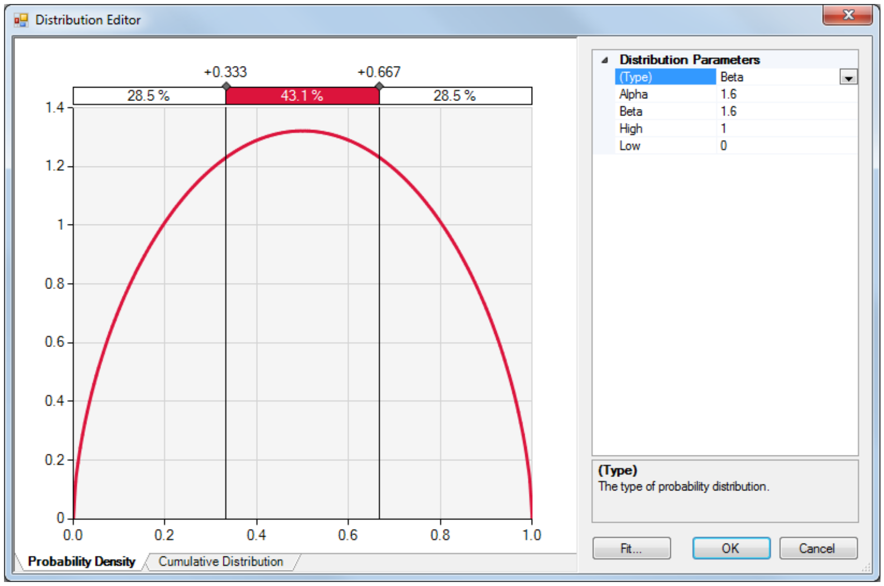


Figure 5.27: The Shape of the Standard Beta Distribution (α=1.6, β=1.6) used as a Starting Point for obatining a Bid Price Estimation Distribution

5.4.1.7 Constructing the Beta Distribution for Bid Price Generation

In an attempt to construct a unique beta distribution for each company for each project that it intends to bid, different criteria were considered. This section discusses how the parameters based on these criteria and intended for use in construction the beta distribution are estimated.

The criteria assumed to affect the bid price generation process at companies when encountered with a bid situation are summarized. The same factors are used in constructing the Beta distribution used in bid price estimation. The criteria include:

- The utilization of company resources
- Availability of the resources required by the project
- The nature of competition
- The attributes of the project bid

The utilization of company resources (i.e., for performing large, medium and small projects) indicate the appetite that the company has got for work. When companies go for long spells without work, they become desperate and tend to bid low on most projects so that they can have work to do and cash inflows to cover their basic needs. A low utilization would imply that the company has not had much work and would therefore be eager to get work hence exhibiting tendencies to submit low bids. Utilization values for the corresponding size of project are retrieved directly from the discrete event models embedded within the agent.

Before companies bid a project, they check out details of resources required by the project that they are considering to bid. If there is easy access to resources required by the project, they will not tend to escalate their prices. In case there is an apparent shortfall in the availability of resources within the industry, companies tend to add a premium to their bid which accounts for the risks and associated inconvenience of failing to access specific resources when they require them to perform a project. The ease of access to resources required by a project was quantified using a resource availability index. Equation 5.3 presents the mathematical expression used to quantify this index.

$$\text{Resource Availability Index} = \frac{1}{n} \times \sum_{i=1}^{n} Min\left[1.0, \left(\frac{\text{Number of this Required Resource Available at Industry Pool}}{\text{Number of this Resource that the Project Requires}}\right)\right] \quad (5.3)$$

The quotient on the right hand side of this equation is evaluated for each resource required by the project and an average obtained. In the general case, it is assumed that there are n resources required by the project.

Companies tend to submit lower bid prices when competition is stiff especially if they are eager to get the project awarded to them. The stiffness of the competition is relative and will vary from company to company. In order to model a company's perspective on how stiff given competition is, a threshold value was defined within each company agent which gives an indication of the number of competitors that that company would not bid against for the same project. This parameter also gives an indication of how stiff a company perceives the competition for a specific project to be. Equation 5.4 summarizes a mathematical expression used to quantify this based on the total number of competitors and the company's maximum competitor threshold.

$$\text{Competition Stiffness Index} = Min\left[1.0, \left(\frac{\text{Number of Competitors for a Project}}{\text{Company's Maximum Competitor Threshold}}\right)\right] \quad (5.4)$$

The value of the competition stiffness index can only range from 0.0 to 1.0. A value of 0.0 or tending towards 0.0 implies that the company perceives the competition as low. Values tending towards 1.0 imply high or extremely stiff competition.

Attributes of a project qualify a project as either good or bad or too risky or not so risky. A number of attributes for a project are compared to the company's tolerances or thresholds for those attributes and an index calculated. The project attributes considered to affect the bid pricing include:

- Project engineering quality
- Project complexity
- Project Owner trait
- Project safety risks

A "poor project quality index" is quantified by comparing the company's threshold values of each of the attributes to those of the project. Equations 5.5, 5.6, 5.7, 5.8, and 5.9 illustrate this.

$$\begin{aligned} Poor\ Project\ Quality\ Index = & \left(0.25 \times Project\ Engineering\ Quality\ Component\right) + \left(0.25 \times Project\ Complexity\ Component\right) \\ & + \left(0.25 \times Project\ Owner\ Trait\ Component\right) + \left(0.25 \times Project\ Safety\ Risk\ Component\right) \end{aligned} \tag{5.5}$$

Each of the components is computed from the following Equations.

$$Project\ Engineering\ Quality\ Component = 1.0 - Min\left[1.0, \frac{Project\ Engineering\ Quality}{Company\ Engineering\ Quality\ Threshold}\right] \tag{5.6}$$

$$Project\ Complexity\ Component = Min\left[1.0, \frac{Project\ Complexity}{Company\ Complexity\ Threshold}\right] \tag{5.7}$$

$$Project\ Owner\ Trait\ Component = 1.0 - Min\left[1.0, \frac{Project\ Owner\ Trait}{Company\ Owner\ Trait\ Threshold}\right] \tag{5.8}$$

$$Project\ Safety\ Risk\ Component = Min\left[1.0, \frac{Project\ Safety\ Risk}{Company\ Safety\ Risk\ Threshold}\right] \tag{5.9}$$

The next step involves aggregating all these indices into one metric that can be used to obtain the other limit of the Beta distribution. An aggregate index known as the *"Bid Price Escalation Index"* is used for this purpose. In the aggregation operation, all the indices were assumed to have equal weight. Equation 5.10 shows the expression used to compute the *"Bid Price Escalation Index"*.

The *"Poor Project Quality Index"* was set up such that it causes the price escalation index to increase proportionately. As it increases, the escalation index value increases. The "competition Stiffness index" was also setup in such a fashion that it influences the "Bid Price Escalation Index" through a direct proportionate relationship. The "Resource Availability Index" and "Resource Utilization" were setup such that they affect the "Bid Price Escalation Index" in an indirectly proportionate fashion. As the values for these indices increase, the "Bid Price Escalation Index" value reduces.

$$\begin{aligned} Bid\ \mathrm{Pr}ice\ Escalation\ Index = & \left[0.25\times\left(1.0-\mathrm{Re}source\ Utilization\right)\right]+\left[0.25\times\left(1.0-\mathrm{Re}source\ Availability\ Index\right)\right] \\ & +\left[0.25\times Competition\ Stiffness\ Index\right]+\left[\left(0.25\times Poor\ \mathrm{Pr}oject\ Quality\ Index\right)\right] \end{aligned} \tag{5.10}$$

The "Bid Price Escalation Index" Equation (Equation 5.10) generates values that range from 0.0 to 1.0. These values are then mapped onto a range from which a "Bid Price Adjustment Factor" is drawn. Table 5.2 summarizes the mappings of the "Bid Price Escalation Index" to ranges used to build a uniform distribution used to sample the "Bid Price Adjustment Factor" value.

Table 5.2: A Table Summarizing Mappings for Bid Price Escalation Indices to their Bid Price Adjustment Factors

Bid Price Escalation Index	Range for the Bid Price Adjustment Factor
0.0 – 0.25	-1.60 – -1.20
0.25 – 0.50	-1.40 – 1.10
0.50 – 0.75	1.00 – 1.40
0.75 – 1.00	1.20 – 1.60

Deriving the Low and High Values for the Beta Distribution

Two points were estimated and used to derive the low and high value for the Beta distribution to be used for bid price generation. For convenience, the first point was assumed to be equal to the estimated project cost. This value was assigned in the course of the simulation on creation of the project.

The second point was estimated from the "Bid Price Adjustment Factor". This factor was derived from the factors that were believed to affect the price that a typical construction company carries within its bid. The process of quantifying this factor was discussed in the previous section. Equation 5.12 was used to calculate this factor.

$$Second\ Point = \begin{cases} Estimated\ ProjectCost \times Sampled\ BPAF;\ BPAF \geq 0.0 \\ Estimated\ ProjectCost + \left[Estimated\ ProjectCost \times \left(1 + Sampled\ BPAF\right)\right];\ BPAF < 0.0 \end{cases} \tag{5.11}$$

$$BPAF = Bid\ Price\ Adjustment\ Factor$$

The low value and high value for the Beta distribution are then computed from the two points using the following Equation 5.12 and 5.13 respectively.

$$Low\ Value = Min\left[First\ Point, Second\ Point\right] \tag{5.12}$$

$$High\ Value = Max\left[First\ Point, Second\ Point\right] \tag{5.13}$$

Deriving the Shape Parameters for the Beta Distribution

In order to further represent the effects of the prevailing conditions on the price that a company carries in its bid, one of the shape parameters for the Beta distribution used to represent the possible bid price was adjusted from the initial value of 1.60 using a random variates sampled from the “Bid Price Adjustment Factor” range. The shape parameter, alpha, was chosen for adjustment while the beta shape parameter was kept fixed at a value of 1.60 at all times.

This resulted in an effect that saw the company generating low bids from the custom Beta distribution when conditions were favorable for the company and high bid values when the company was faced with unfavorable conditions at the time of bidding. The expression used to adjust the alpha shape parameter for the Beta distribution is summarized in Equation 5.14.

$$Alpha = Max\left[0.001, \left(1.60 + Bid\ Price\ Adjustment\ Factor\right)\right] \tag{5.14}$$

5.4.1.8 Generating the Bid Price

The bid price was generated through a Monte-Carlo sampling process that utilized the Beta distribution created through a process described in the previous section. The model was setup to perform 1,000 simulation runs. A unique Beta distribution is created and assigned to each of the competitors for the project. In doing so, it was assumed that the conditions at each of the competitor companies, although unknown to us and possibly different in a real-life setting, are identical to those of the company intending to bid. A hypothetical scenario is then projected in which we try to obtain the bid prices that each of the competitors would submit if they were to bid the same project 1,000 times. This results in a matrix with two degrees of freedom. The first degree of freedom being the number of competitors and the other being the number of simulation runs performed. The uniqueness in the Beta distribution assign to each competitor can be guaranteed by drawing a random variate from the range of the "Bid Price Adjustment Factor" and using that to compute the low, high, and alpha values. Figure 5.28 how typical layout of the values generated for bid price from the Monte-Carlo process. This 2-D matrix represents all the possible bids that the competitors are likely to submit.

$$
i = \textit{Simulation iterations} \left\{ \overbrace{\begin{bmatrix} P_{1,1} & P_{1,2} & \bullet & \bullet & P_{1,j} \\ P_{2,1} & P_{2,2} & \bullet & \bullet & P_{2,j} \\ \bullet & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet \\ P_{i-1,1} & P_{i-1,2} & \bullet & \bullet & P_{i-1,j} \\ P_{i,1} & P_{i,2} & \bullet & \bullet & P_{i,j} \end{bmatrix}} \right.
$$

Figure 5.28: Matrix Setup of Possible Bids

In order to guarantee that the company carrying out this analysis gets the project awarded to it, it needs to submit a bid price that is less than or equal to the lowest in this 2-D matrix. The process that involves seeking the least bid from those of the competitor is identical in many ways to an optimization process. For simplicity, the process is sub-dived into two phases.

The first phase involves generating a vector of *"sub-optimal bids"* from the 2-D matrix. This is achieved by taking the minimum value in each row of the matrix. This is illustrated in Figure 5.29.

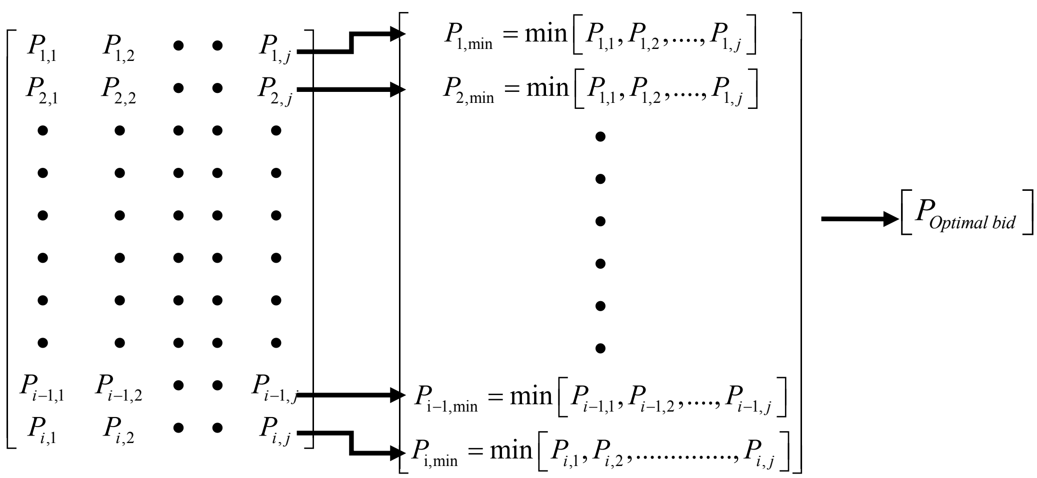


Figure 5.29: Process of Generation the Vector of Sub-Optimal Bids from the Matrix of Possible Bids

The second (last) phase involves generating the optimal bid from this vector of sub-optimal bids. This step makes use of the company's adopted bidding strategy. Table 5.3 summarizes the mathematical operations to apply on the vector of *"sub-optimal bids"*, in order to generate the optimal bid.

Table 5.3: Arithmetic Computations Corresponding to each Possible Bidding Strategy

Bidding Strategy	Arithmetic Operation to Generate Optimal Bid
Maximize the chances of winning the project	Take the minimum value of values in *"sub-optimal bids"* vector
Maximize the likely project if we win the project	Take the maximum value of values in *"sub-optimal bids"* vector
Maximize the chances of winning the project and likely resulting profit	Take the average value of values in *"sub-optimal bids"* vector

The approach presented was inspired by an existing algorithm presented by Wayne in 2001, in his book on modeling uncertainty using @RISK (Wayne, 2001). Wayne's algorithm could not be applied as is because it would fall short for our purposes. Consequently, a number of additions were made to this algorithm to obtain the one presented here. For example, the concept of bid strategy is introduced and incorporated in the algorithm used for generating an optimal bid.

5.4.2 Project Award and Execution

After a the bid prices have been generated and submitted by the bidding companies, the Bid Manager agent evaluates them and identifies the company agent with the least bid. It then awards the project to that company agent. When a company agent is awarded a project, it requests for the resources required by the project from the resource agent. Once granted the required resources, the company agent passes the project to the DES model that is embedded within it for processing. Then project execution can commence when there is available production capacity. Figure 5.30 shows the layout of the DES model embedded within company agents for processing awarded projects.

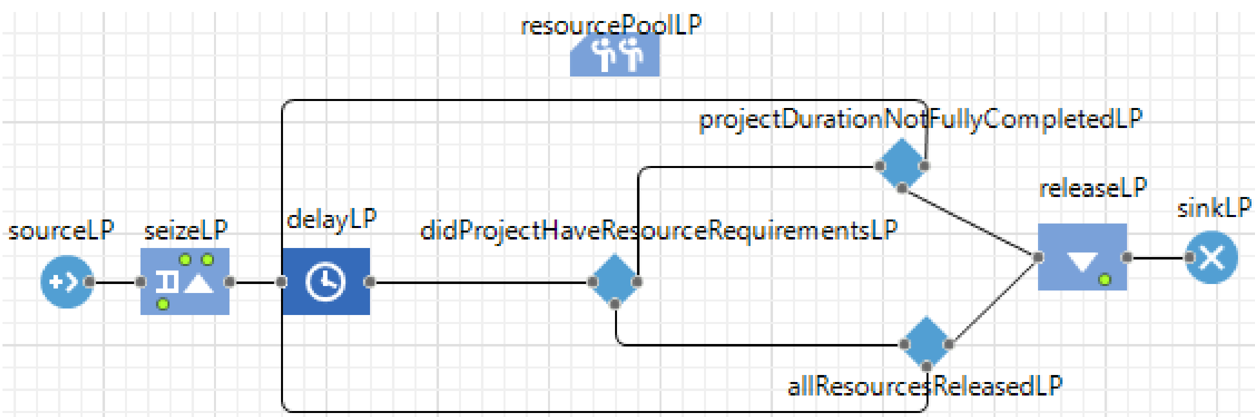


Figure 5.30: Discrete Event Process Interaction Model Embedded within Company Agent for Executing Awarded Projects

Two layouts of the model shown in Figure 5.30 are embedded within each company agent. The first one is responsible for the process-interaction modeling of large projects while the second processes both medium size and small projects. The number of resources in the resource pool represents the concurrent production capacity for that size of project. The suffix *"LP"* in each modeling element name represents Large Project. The small and medium project DES modeling elements have a suffix *"SMP"*.

When a project is awarded to a company and has been granted its required resources, it is passed on to the embedded DES model within the agent that is identical to the one shown in Figure 5.30.

The source modeling element then fires out an entity that represents the project. This entity then flows into the seize element where it requests for one resource from the resource pool and is then queued. When granted the resource, it is transferred into a delay modeling element which delays its flow for one calendar day. This delay mimics the processing of the project. This is because the scope of work to be done on a project is reperesented in the form of time.

The time left to project completion is then reduced by one day (for projects that did not initially have resource requirements) and so are the man-hours (for projects that initially had resource requirements) that each resource is expected to be engaged on the project. When the delay time elapses, the entity flows into a branch element which routes it out through the top (for projects that did not initially have resource requirements) or bottom port (for projects that initially had resource requirements).

As the project entity that initially had resources enters the branch element named *" allResourcesReleasedLP"*, a check is made to identify the resources that have completed their engagement on the project. Those resources are released and returned back to the resource agent that was defined within the virtual construction industry agent. The project agent is then checked to findout whether it still has resources that it is engaging. In case it does, it is sent back to the delay modeling element named *"delayLP"*. Otherwise, the execution of this project is deemed complete and the entity is transferred into the release modeling element names *"releaseLP"*. Projects that didn't initially have requirements are sent to another branch element named *"projectDurationNotFullyCompletedLP"*. A check is made to see if the cumulative time that the entity was delayed within the delay element adds up to the total project duration. If it does, the project execution is deemed complete and the entity is routed into the release modeling element names *"releaseLP"*. Otherwise, it is returned to the delay modeling element for further processing.

When finalized projects enter the release element, they release the company resource freeing one space of production capacity for that project size. The released resource is returned to the resource pool modeling element named *"resourcePoolLP"*. The project entity is transferred into a sink modeling element *(sinkLP)* where it is destroyed.

The explanation given for the processing of the projects awarded to the company agent indicates that they are processed using a high-level approach that relies on representing the work scope as a duration. Using other techniques would be quite cumbersome as the activities, their sequence, work scope and resource requirements vary significantly from project to project.

5.5 COMPANY OF INTEREST AMBASSADOR (COI AMB.) AGENT

A special agent was created and embedded within the virtual construction industry agent. This agent represented the company whose performance the analyst was interested in closely tracking. To a large extent, the attributes and behaviors of this agent were similar to those of the other company agents. For purposes of distinguishing this agent from the other company agents, the term competitors is used to refer to the company agents that are not the COI Ambassador.

At the time of design, it was envisaged that the COI Amb. Agent was compete for new projects created in the virtual industry, then track the projects that it is awarded and those that it lost. Both types of projects were tracked so that the tendering performance of the company of interest could be precisely tracked. In addition to this, the awarded project was tracked so that these could be performed and the resulting performance from project execution operations generated.

Details of the execution of awarded projects were not modelled within this agent but rather a separate standalone module was developed which was dedicated to handling this task. The implementation of an architecture that involved a COI Amb. Agent competes for projects and another module executes the awarded projects, required communication protocols to be implemented between the COI Amb. Agent and this module. The COSYE framework was used to implement these communication protocols. This module that executes awarded projects was referred to as a Simphony/COI federate because it was build off of a Simphony model and represented the operational level processes at the company of interest to the analyst.

Static bidding information required for the COI Ambassador Agent to effectively engage in bidding processes were defined by the analyst in the Simphony/COI federate. This federate had a windows form application sub-component with a user interface that served as a place hold for this information from the analyst prior to simulation. Also information that kept changing within the Simphony/COI federate that was necessary for bidding had to be passed on to the COI Amb. Agent whenever a new project was announced for tender.

The communication protocols were setup such that the COI Amb agent would request for information that it required from the Simphony/COI federate by making calls to the RTI through the CI agent. Also, whenever a project was awarded to a company, COI Amb. Agent sent details of this project to the Simphony/COI federate. When information required for the bidding process was received by the COI Amb. Agent, it would be stored within variables embedded in it (See Figure 5.31). The state of this agent gets updated whenever new values are assigned to these variables.

Besides competing for projects, requesting and sending information via the RTI, the COI Amb. Agent was also responsible for making resource requests for projects that it was awarded and resuming the execution of the AnyLogic simulation Engine whenever a project was awarded to a company.

5.5.1 Modeling Constructs within COI Amb. Agent

The model development phase utilized the modeling constructs that exist within the AnyLogic simulation system to achieve the behavior discussed within the agent. This section presents screen shots of the modeling constructs embedded within this agent for this purpose. It is worth-noting the choice and creation of these constructs were based on design specifications detailed in an activity diagram. The details of these design specifications are presented in the section that follows this one.

Figure 5.31 shows the buffers that define the state of the agent. This aspect of the state of the agent remains static from start to the end of the simulation but influences the agents behavior especially in bidding.

On the other hand, Figure 5.32 shows the buffers that hold the dynamic information that defines the other part of the state of the COI Amb. Agent. These are updated as simulation progresses results in the state of the agent to get updated too. These buffers are updated whenever a new project has been created in the industry that is to be bid.

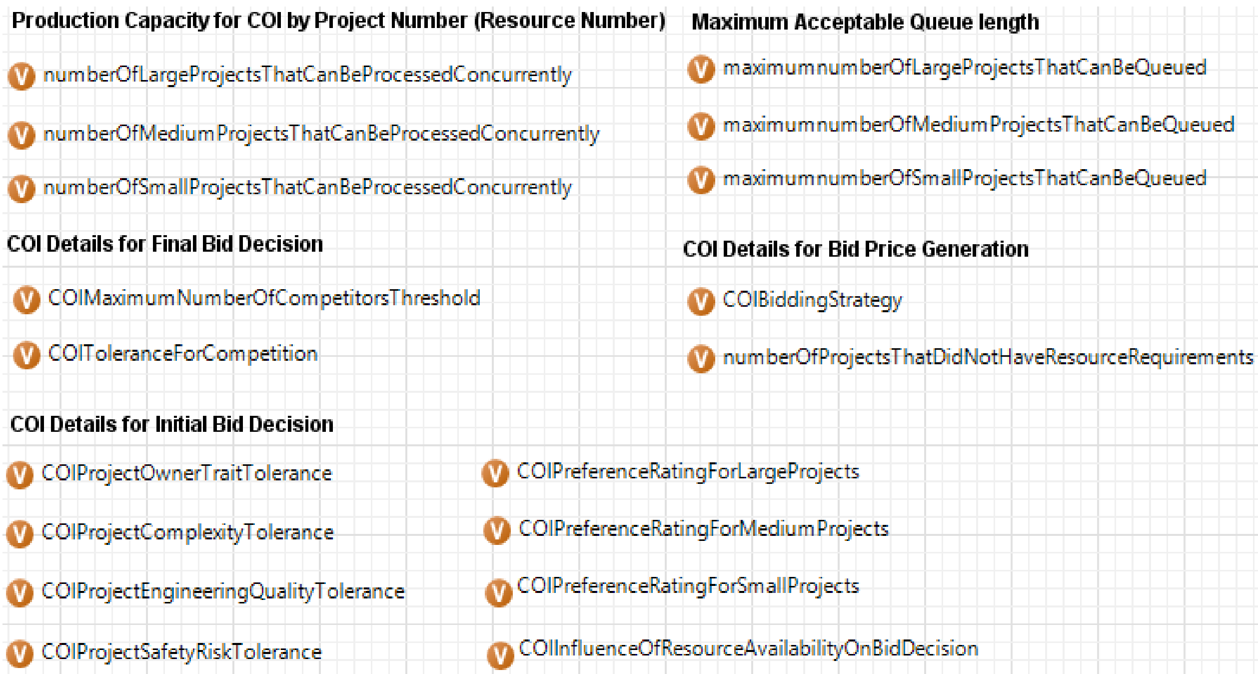


Figure 5.31: Buffers that Define the Static State of the COI Amb. Agent

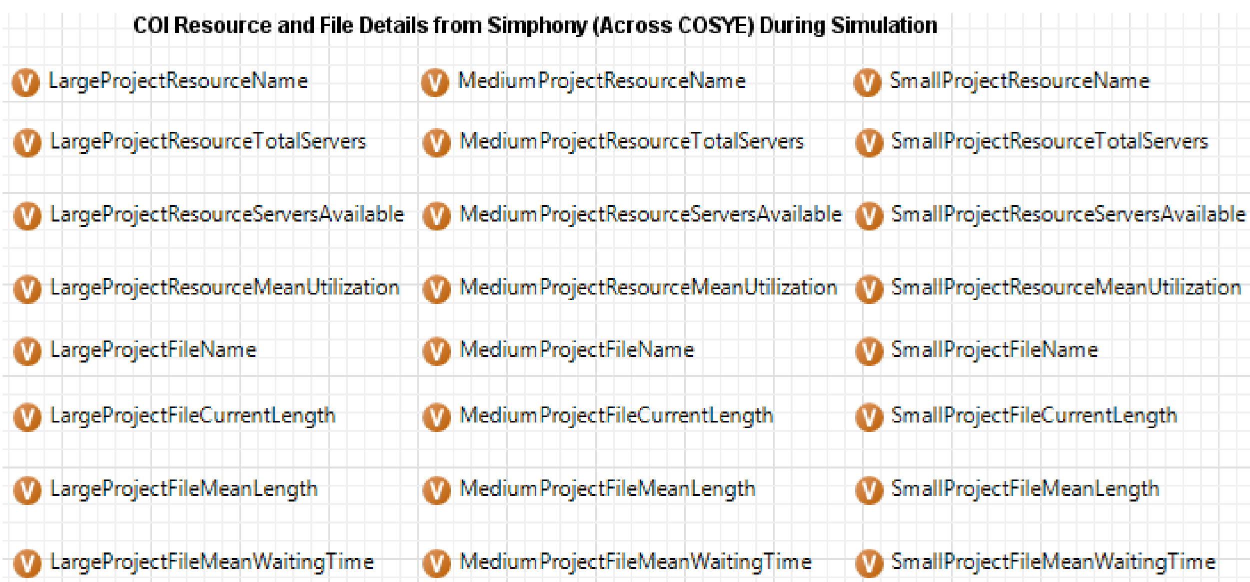


Figure 5.32: Buffers that Define the Dynamic State of the COI Amb. Agent

Figure 5.33 shows the constructs that implement the behavioral aspects of the COI Amb. Agent. These are conveniently sub-categorized into those that emulate the bid behavior and those that request for resources required by projects awarded to COI Amb. Agent. The second sub-category is also responsible for resuming the simulation execution of the AnyLogic simulation engine at the end of each bidding cycle for a project.

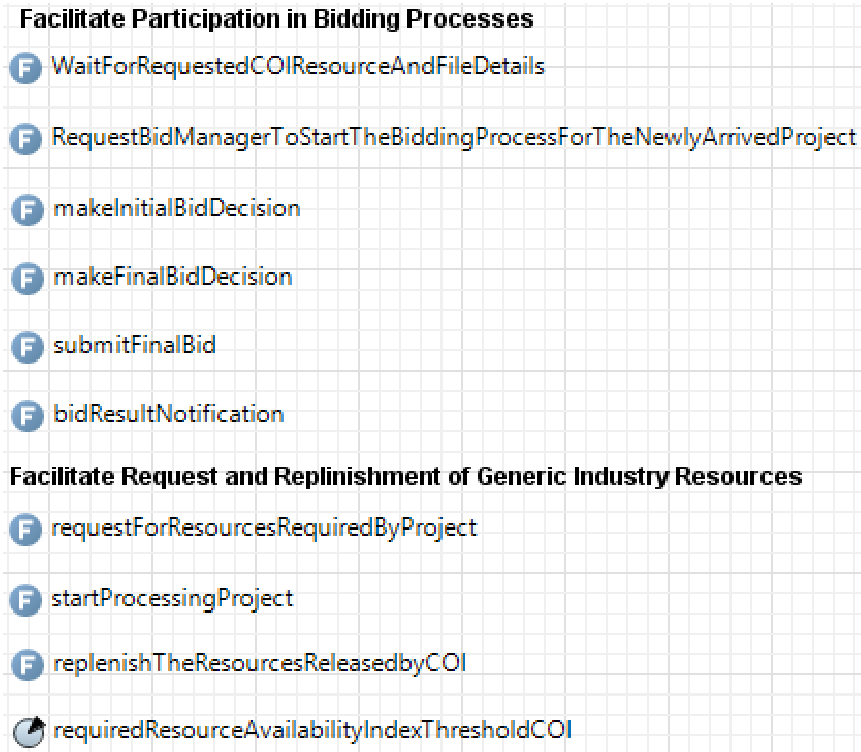


Figure 5.33: AnyLogic Modeling Constructs that represent the Behavior of the COI Amb. Agent

5.5.2 Concept Design for the COI Amb. Agent

Prior to implementing the model development aspects of the COI Amb. Agent, design specifications that detailed the envisaged behavior and states of the agent were created in a formal documented format. An activity diagram approach was used as a design aide in this task.

The behaviors summarized within this activity diagram (Figure 5.34 and 5.35) are similar to those discussed in this section of the chapter. The activity diagram was split into two parts with a node labelled “A” introduced as a point of continuity between the parts. This was done to ease readability of the diagram.

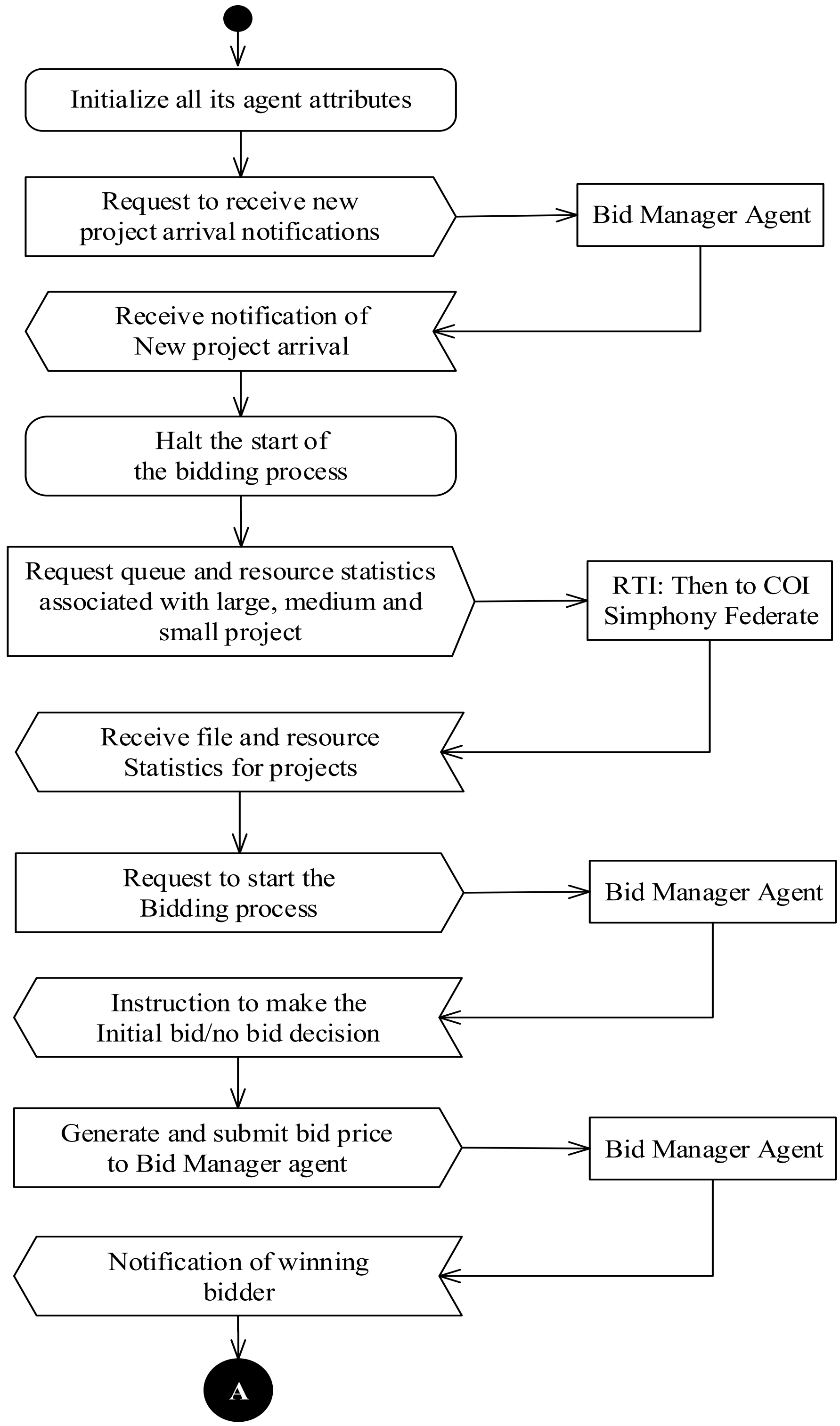


Figure 5.34: Activity Diagram Summarizing the Behavior of the COI Amb. Agent (Part I)

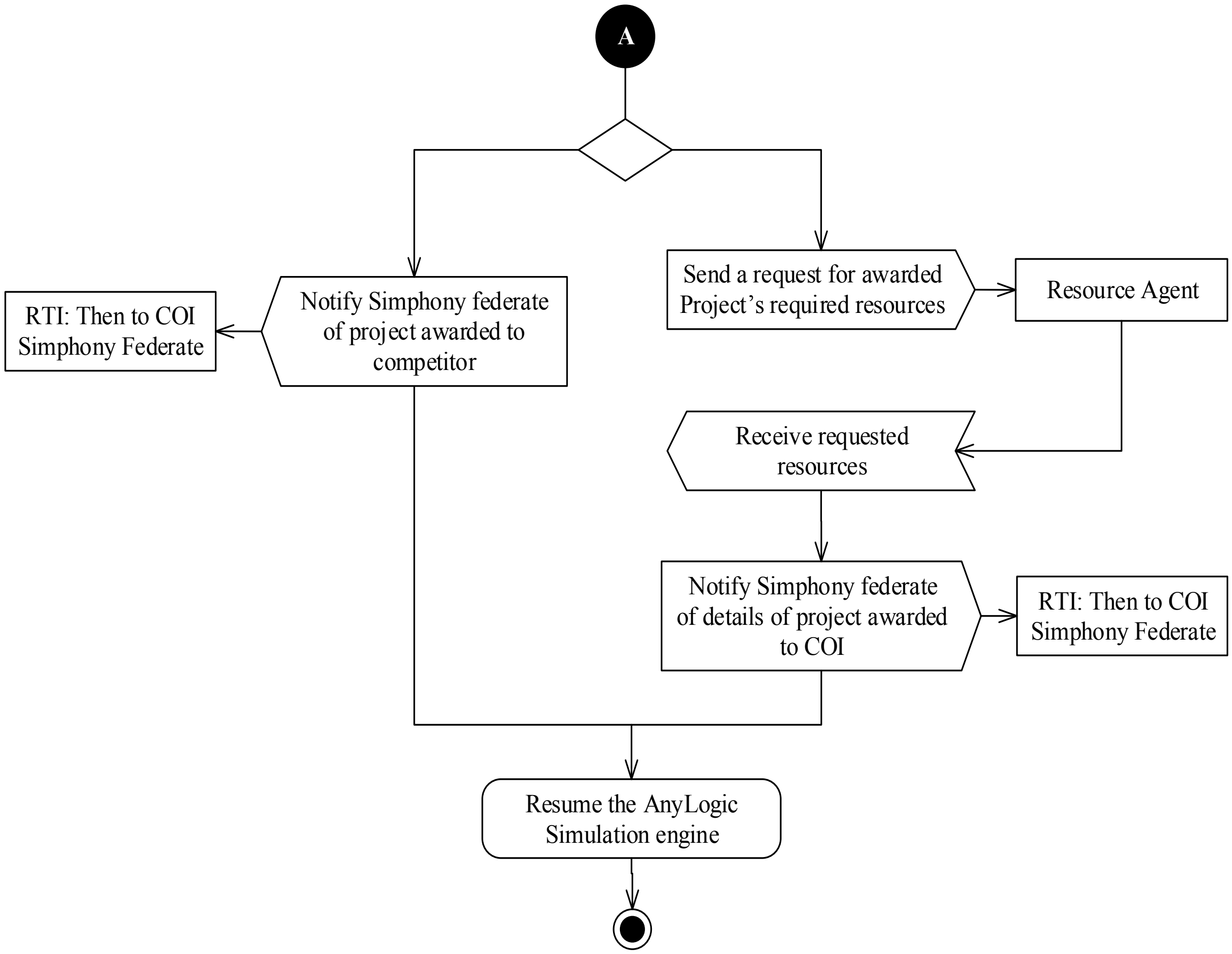


Figure 5.35: Activity Diagram Summarizing the Behavior of the COI Amb. Agent (Part II)

The behaviors represented in the activity diagram include:

- Request and receipt of information required for bidding every new project arrival is announce
- Request to Bid Manager to start the formal bidding process for newly arrived projects after COI Amb. Agent has received the information (from Simphony/COI federate) that it requires to engage in the bidding process
- Engagement in the bidding process – bid/no bid decision and bid price generation
- Communication of project awarded to competitor to the Simphony/COI federate
- Request of resources required by projects awarded to the COI company
- Communication of the projects awarded to COI when they are granted resources

- Resuming the AnyLogic simulation engine execution every time the bidding process for a project comes to an end (i.e., a project is awarded to a company).

5.6 SUMMARY FOR CHAPTER FIVE

One of the crucial modules (federate) within the distributed simulation federation has been successfully discussed. This module was developed to operate as a standalone federate i.e., the AnyLogic federate. This federate was dedicated to emulate the typical high-level behavior of a construction industry i.e., creation of new projects, competition for projects by companies operating in the industry.

Autonomous and semi-autonomous simulation agents were created to emulate this behavior because of its the highly distributed nature and that fact that each component is self-executing resulting in a system that was multi-threaded or that can also be regarded as being characterized by concurrent execution. A total of 5 types of agents have been discussed. These include:

- *A construction industry agent*: This agent represented the virtual construction industry in which all other agents thrived i.e., was a container that nested all modeling constructs
- *Resource agent population*: This agent population comprised different types of agents that each represented a unique operational constrain for the companies operating in the industry
- *Bid manager agent*: This agent represented the community of owners and their representatives (i.e., consultants). It generated projects for the virtual construction industry, and managed the competitive bidding process for each of the created projects.
- *Company agent populations*: Three agent populations were created i.e., large companies, medium companies and small companies, with each population having unique attributes. However, the overall behavior of all three agent populations were identical i.e., they competed for new projects and processed those that they were awarded. Each agent population had company agents that thrived within it. These company agents derived their attributes from the range of values defined for their respective agent population. These company agents represented the competition to the company of interest.

- *Company of interest ambassador agent*: This agent had behaviors similar to those of other company agents. It was meant to represent the company that was of interest to the modeller or analyst. This agent was also responsible for sending details (via the COSYE RTI) of projects that were awarded to it to the Simphony federate for detailed processing. The Simphony federate was responsible for processing projects awarded to the company of interest and tracking the performance that would result in the course of processing. Details of projects not awarded to the company of interest (COI) were also sent by this agent to the Simphony federate so that it (the Simphony federate) would track the tendering performance of the COI. Details of the Simphony federate are discussed in the following chapter (Chapter 6).

Prior to implementing each agent, their desired behavior and envisaged interaction were documented using sequence diagrams and activity diagrams through a formal design process. These design aides (sequence diagrams and activity diagrams) were presented in this chapter and used as a basis for discussing each agent.

An agent-based model was used to implement all details discussed in this Chapter. This model was developed within the AnyLogic simulation system and configured to serve as a standalone federate in the larger distributed simulation model.

CHAPTER SIX – THE SIMPHONY (COI) FEDERATE

6.0 INTRODUCTION TO CHAPTER SIX

This chapter presents details of the components within the company of interest (COI) federate, also known as the Simphony federate. This federate was configured to mimic the core operations that take place at the COI. It is worth mentioning that this federate is tightly coupled to the AnyLogic federate within the distributed simulation federation. It receives details of awarded projects from the AnyLogic federate and processes those that have been awarded to the company of interest. This federate is also responsible for tracking the performance of the company of interest (COI) at a strategic (tendering) and operations (project execution) level.

The chapter starts by presenting an overview of the structure of the federate along with the behavior it was envisaged to emulate. This federate was developed using a windows form application and a Simphony discrete event simulation model. Details of each of the components are discussed.

The windows form application was developed in Visual Studio 2010 using C# programming language. The Simphony model on the other hand was built using general purpose template modeling elements alongside modeling elements that were developed in a special purpose Simphony template.

A special purpose template developed for use in the application was discussed. The roles and features of each modeling element are presented. The layout of the model developed from this special purpose template and embedded within the Windows Form Application, was discussed.

A number of sections towards the end of the chapter were presented that describe how the performance of a contractor company was abstracted and represented in the model. These details were introduced by presenting an overview of what performance of a contractor company means.

6.1 AN OVERVIEW OF THE SIMPHONY FEDERATE

The Simphony federate is one of two federates that form the performance management simulation federation. The roles of this federate within the simulation federation execution included:

- Provide an interface of the user or modeller to specify inputs that define the specifics of the *company of interest* (COI)
- Provide an interface for displaying the simulated results for the company of interest
- Track projects acquired by the COI and those lost
- Process projects awarded to the COI using a resource constrained discrete event simulation approach
- Track and collect data on project level performance as projects are being executed and roll those up to generate company performance
- Connectivity to a Run-Time-Infrastructure to facilitate participation in a distributed simulation federation

In order to achieve all the above, it was decided that development a windows form application that encapsulates a simulation model of some sort, would be right approach to take. Such an application would at the least behave as specified above and would have extensibility capabilities.

Simphony was the simulation system chosen for use in this development work because it is an advanced, easy-to-use and extensible simulation software. In addition, the developers of Simphony were a part of this research study and could easily add more functionality to Simphony when needed to achieve specific simulation behaviors. Simphony also provides an easy way for achieving connectivity to a RTI so that it can participate as a federate in a distributed simulation system. A discrete event simulation modeling approach was selected for modeling the project execution because it is the most suitable paradigm modeling for process-centric operations especially when resource constrains play a huge role in the operation.

Visual Studio was chosen as a development environment for the windows form application that would provide the desired interfaces. This is because it is a DOTNET based development environment hence making it easy to embed Simphony services within it.

Consequently, this federate is referred to as a *Simphony federate* within this thesis. This *Simphony federate* represents the attributes and operational capabilities at the *company of interest* (COI). Figure 6.1 summarizes the components developed within this federate, in a hierarchical fashion.

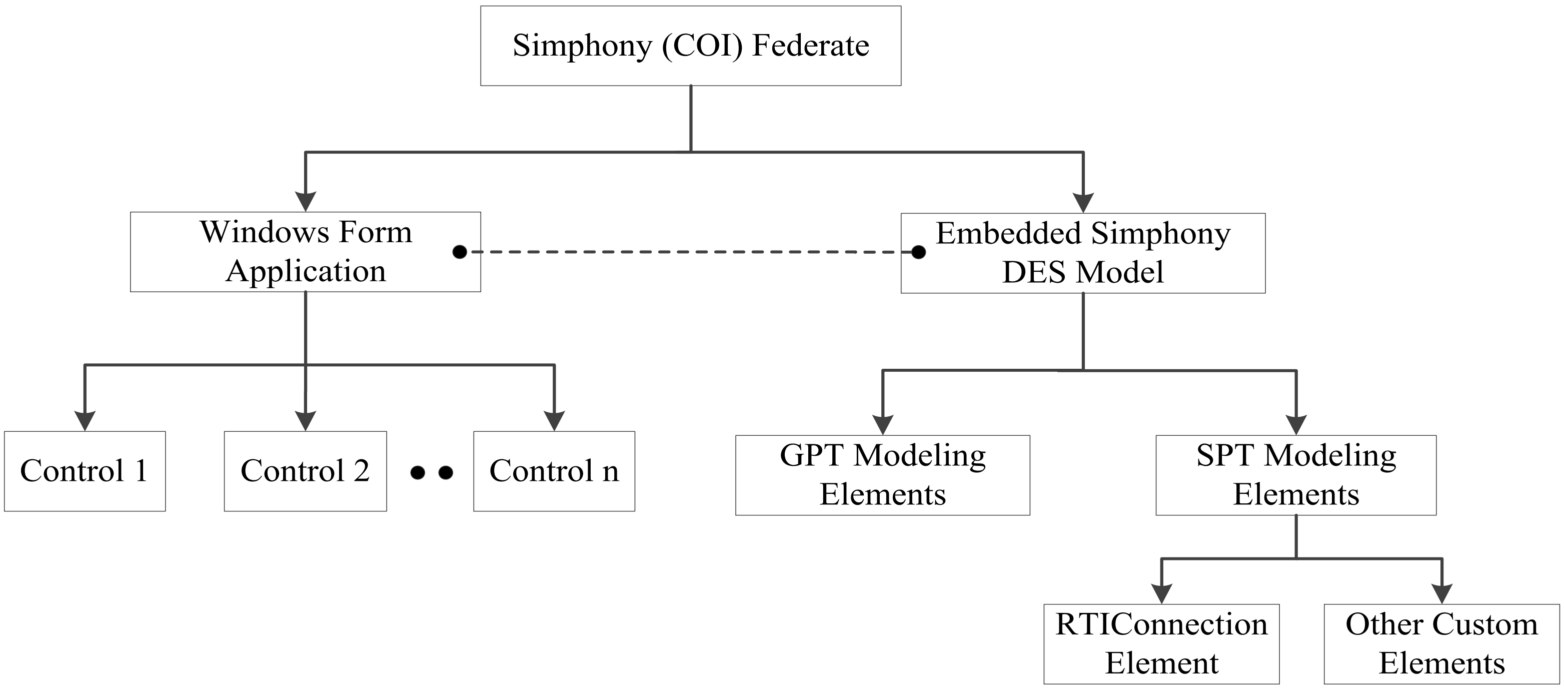


Figure 6.1: Components in the Simphony (COI) Federate

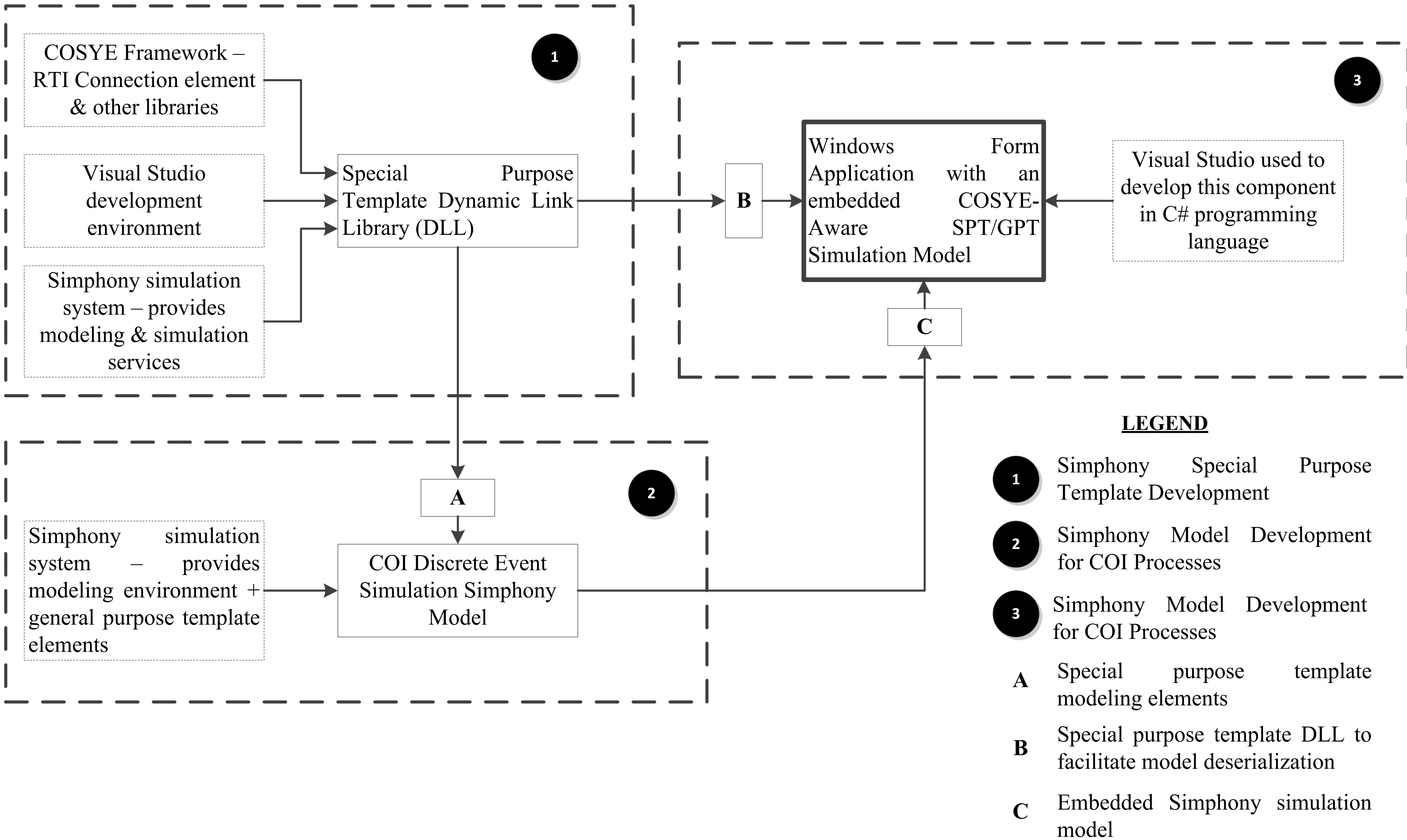


Figure 6.2: Development Process of the Simphony (COI) Federate

The design and development of this federate was done in-line with these components. The phases undertaken in the development of the federate include:

- Development of a windows form application
- Special purpose template development
- Simphony Model development for COI operations

A schematic layout that details the conceptual design developed prior to federate development is shown in Figure 6.2. This conceptual model illustrates how the different components relate with each other. Subsequently, the development of the actual *Simphony federate* followed this layout. Details involved in the development work of each phase are discussed in detail within the following sections.

6.2 WINDOWS FORM APPLICATION

Software developers give a lot of thought to the expected interaction between their products and the end-users. It is natural to provide end-users with a convenient way to enter inputs into a developed system and view outputs. The obvious route to achieve this in the DOTNET development environment is to design and develop a windows form application. This approach was adopted in this thesis work in order to achieve similar objectives.

The windows form application comprises of the interface (with numerous controls) and methods/code behind the scenes, to do the computation work. First, details will be presented about the interface and then the computational behavior that the windows form application was designed to emulate.

6.2.1 User Interface of the Windows Form Application

The user interface was designed keeping in mind the input variables expected for purposes of modeling the tendering and project execution behaviors and capabilities of the COI. Emphasis was also placed on the type and form of the output expected after a simulation run.

In order to simply the setup of the form, a context menu strip and tab controls were used to create a hierarchical layout that would permit users to easily enter their model inputs and view outputs. The context menu strip comprised of a “Model Setup” menu, a “Model Results” menu, and an “Overall Company Ratings” menu. Within the “Model Setup” menu, there is an option to/for

"Model Inputs", "Run Simulation" and "Close the Application". The option for "Model Inputs" has five tabs that are shown and discussed in the following sub-sections. The "Run Simulation" option starts simulation execution and the "Close the Application" shuts down the application.

The "Model Results" menu and an "Overall Company Ratings" menu also have tabs for displaying model results. These will not be presented nor discussed here. They will be deferred to chapter eight.

6.2.1.1 Inputs – COSYE Setup and Performance Measure Details

This interface contains controls that facilitate the modeller to specify the settings of the distributed simulation federation they intend to join (See Figure 6.3).

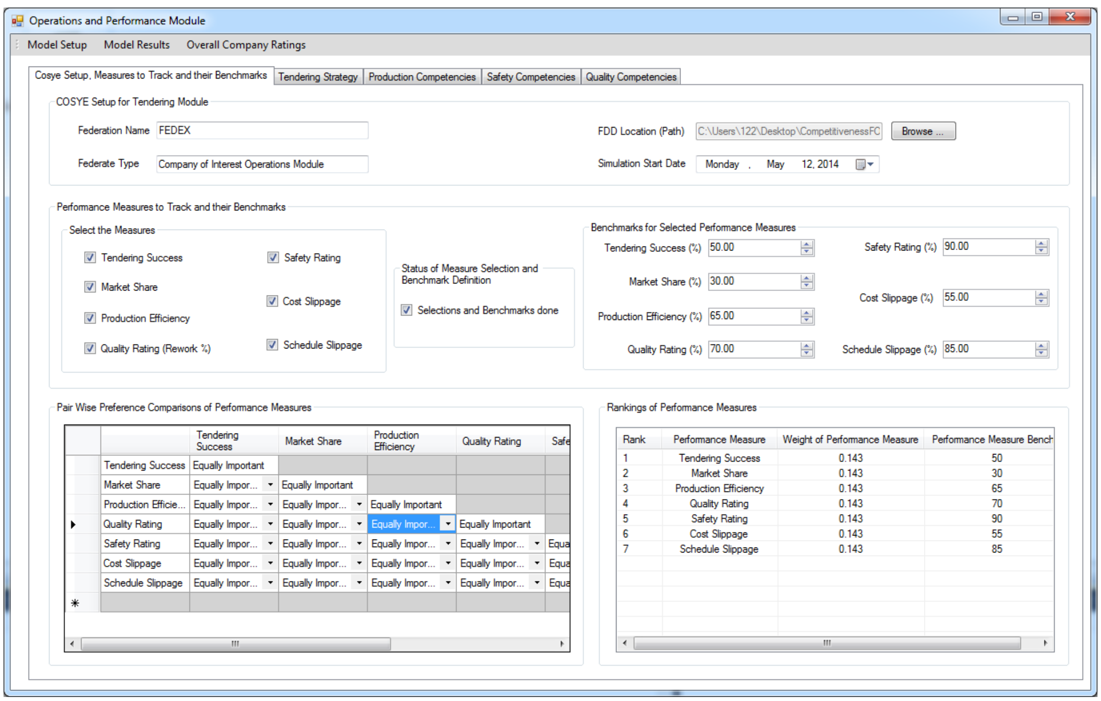


Figure 6.3: A Screen Shot of the Setup and Performance Measure Tab in the Windows Form Application

The interface also serves as a means by which the modeller specifies the performance measures that they would like to track, their relative importance (defined in the pair-wise data grid view control) and the benchmarks for those performance measures. The weights generated from the pair-wise comparisons are displayed in the list view in the bottom right corner of Figure 6.3.

6.2.1.2 Inputs – Tendering Strategy

The tab labelled "Tendering Strategy" is used to capture the values of variables that can be used to derive a bidding strategy for the company of interest. This strategy is used to acquire work through a competitive bidding process. The tab also contains a control in which the modeller can specify the work that was in process at the time that they would like to commence the simulation (see Figure 6.4).

Variables provided for that aide in the bidding strategy include:

- The objective in bidding any project (maximize the chance of winning, maximize possible profit or both)
- The percentage of times that a specific type of project (large, medium or small) should be bid
- Maximum number of competitors the COI can bid against
- Threshold values for the project to bid based on project attributes such as complexity, owner trait, engineering quality and safety risks.

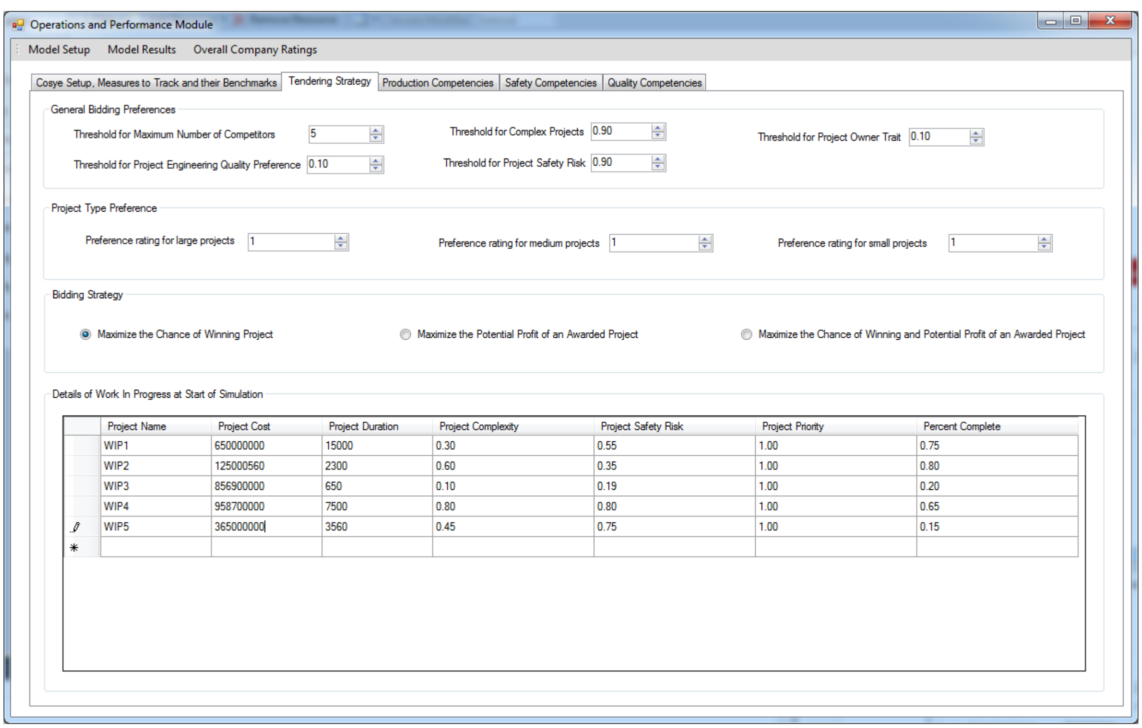


Figure 6.4: A Screen Shot of the Tendering Strategy Tab in the Windows Form Application

6.2.1.3 Inputs – Production Capacity

The variables specified within the "Production Competency" tab contribute to the static component that is used to compute the production efficiency (See Figure 6.5).

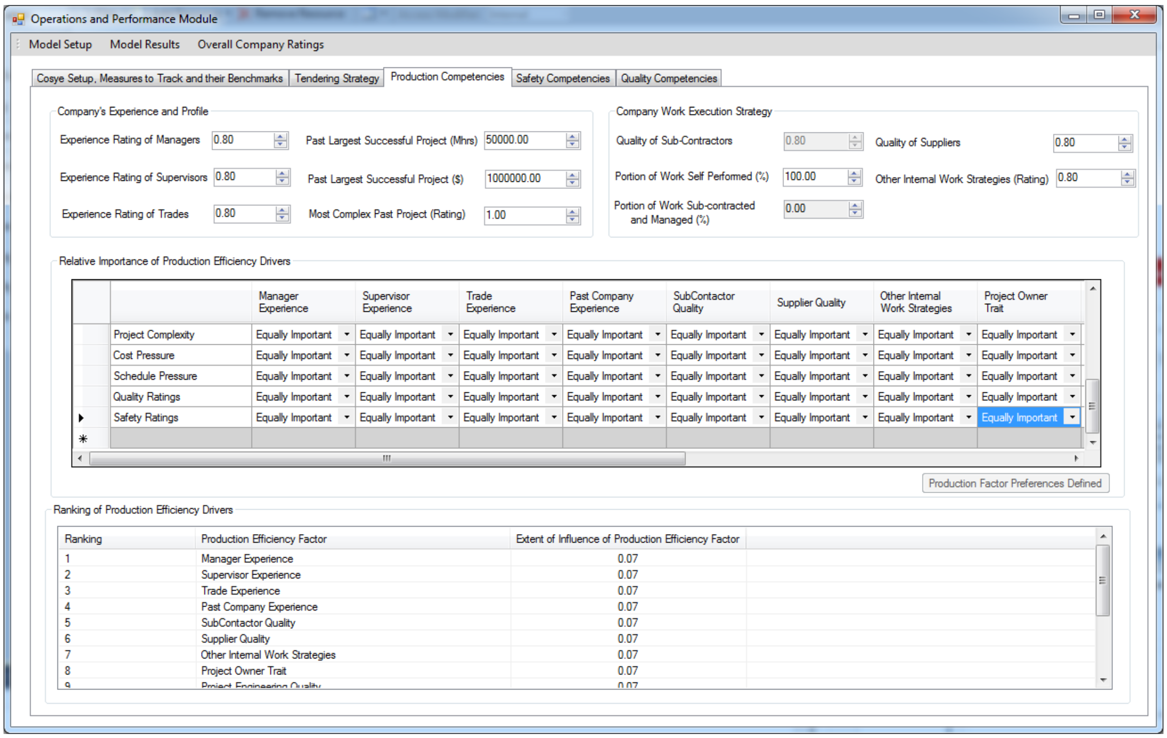


Figure 6.5: A Screen Shot of the Production Competency Tab in the Windows Form Application

Production efficiency is assumed to be affected by competencies at a company that are static and another dimension of competency that is dynamic and depends on the project attributes and the prevailing work conditions e.g., schedule pressure, cost pressure, amount of rework and safety ratings. The static production efficiency competencies explicitly captured in this interface and are sub-divided into two—company experience and effectiveness of work methods.

The interface also provides a data grid view that can be used by the modeller to perform pair-wise comparisons of factors that are believed to affect the production efficiency. These factors belong to either the static production efficiency drivers or the static ones.

Details of the production capacity, i.e., number of critical resources at the COI, are defined within the AnyLogic federate (Bid Manager), discussed in Chapter 5, and are considered behind the scenes; hence, there is no representation for that in the interface.

6.2.1.4 Inputs – Safety Competencies

Safety performance at any construction company will be affected by numerous factors. In this study, these have been limited to the following:

- The safety practices and systems at the company
- The Worker attributes
- The project attributes and
- The prevailing work conditions

These are high level factors that can further be aggregated into static safety drivers and dynamic safety drivers. These same enumerated factors can be further sub-divided into different sub-factors.

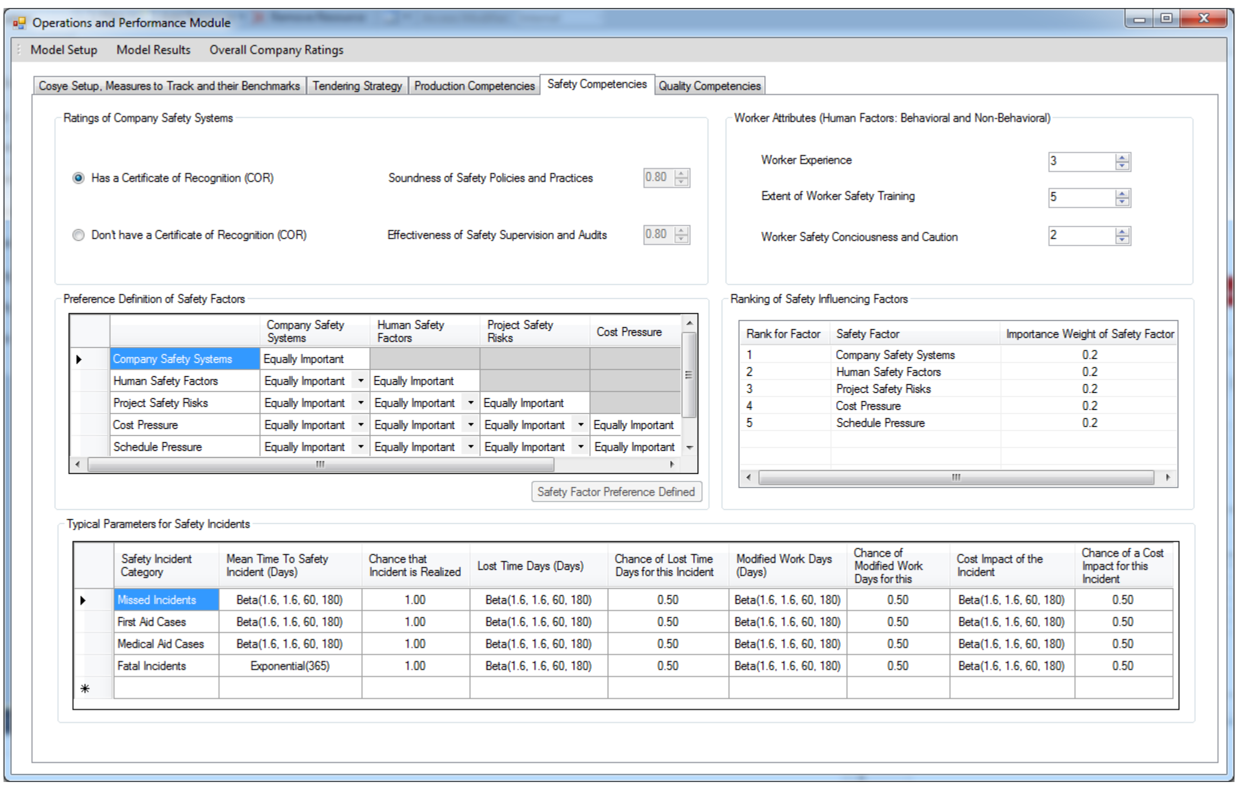


Figure 6.6: A Screen Shot of the Safety Competency Tab in the Windows Form Application

The interface in Figure 6.6 provides controls that capture input values for the different aspects that have been explicitly modeled to affect safety performance. It also provides a control that facilitates the definition of the extent that each factor is believed to influence safety performance.

The interface further provides a data grid view control into which the modeller can specify attributes for the different safety incident categories i.e., *missed incident, first aid cases, medical aid cases* and *fatal incidents*. Attributes defined for each include:

- Mean time to safety incident
- Chance that incident is realized
- Lost time associated with incident and the chance of having a lost time for any incident
- Chance of modified work days and the amount if modified work days are realized for an incident
- The chance of a cost impact and the amount of cost impact if it is realized for an incident

6.2.1.5 Inputs – Quality Competencies

Quality is a performance metric that is also affected by both static and dynamic aspects. The interface for defining quality competencies provides for the user to rank the factors that belong to each driver category using a pair-wise approach. The modeller has an option to choose the factors they believe will affect the quality performance.

The interface also provides for the definition of the base or typical chance of rework experienced at the company, the proportion of work items affected each time there is a rework incident and the extent of rework (in man-hours) for work items affected. Details of the interface can be viewed in the screen shot shown in Figure 6.7.

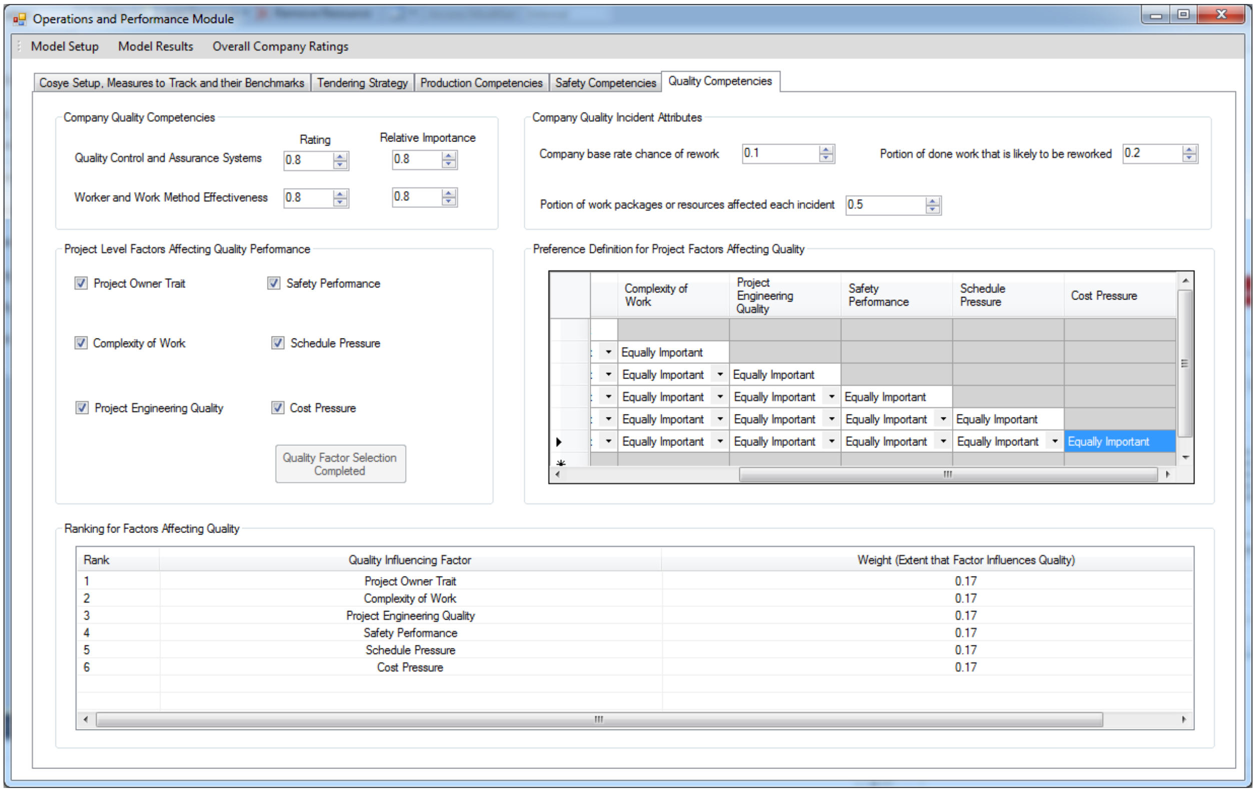


Figure 6.7: A Screen Shot of the Quality Competency Tab in the Windows Form Application

6.3 FORMALISM OF THE PROJECT EXECUTION PROCESS AT THE COMPANY OF INTEREST

To a large extent, construction is a project-based industry that formulates teams around projects and tasks them to properly execute projects. These project teams manage and control project execution using standard tools that are usually made available to them. One of the key responsibilities of project teams is to track and control how well projects perform. In addition to the standard project management tools available to them, project teams make use of the information that they gather on projects to fulfill this responsibility of project performance.

The performance and competitiveness of companies is largely dependent on the performance of the projects that it executes. Relying on this on this premise allows us to model the performance of a company based on the aggregation of the performance of projects. In turn, the performance of projects can be modelled by explicitly representing the process of project execution by the use of a standard process interaction model layout. In this option is adopted, projects could then be formalized as tokens (entities) that flow through this model triggering a sequence of events to occur.

Subjecting every project to a standard simulation model layout and yet each project is unique (hence warranting a custom modeling approaches) initially seemed inconceivable. Nonetheless, this was overcome by modeling the project execution process at a high level. This meant that the scope of work in projects was to be represented in terms of man-hours that resources would be engaged on projects rather than as tasks with logical sequencing relationships between them and that each have a resource requirement and a quantity of work defined that needs to be performed (e.g., m^3 of concrete to cast, m^2 of wall to erect, DI of pipe to weld etc.).

This high-level approach meant that the required resources would be retained on the project until their required time elapses, after which they would be released. In a real life setting, projects capture and release different types of resources at different times in the course of the project execution. This phenomenon was abstracted and represented in the developed application with a slight adjustment to simply the modeling process. All resources that were required by a project were to be made available at before the project starts. This meant that the engagement of all resources on the project would commence at the start of the project and not at varying points in time in the course of project execution, as is the case in real life. However, resources were released after their required time on a project elapsed. This implies that they are released at different times if they are retained on a project for varying lengths of time. This matched the situation in real life.

It is also important to distinguish between project resources and company resources in the context of this study. Project resources refer to the workers (trades), supervisors, managers and Engineers that would be required to perform the work. These are modelled explicitly as an agent population within the AnyLogic federate using a resource pool defined at an industry level. Companies that require these resources to perform projects awarded to them, compete for these projects. Company resources on the other hand were used to represent an aggregation of all the constraints to the production capacity of a company. It is known that it is not practical for any construction company to be assumed to have an infinite production capacity if the project resources are not a constraining factor to its operations. If this is the case, some form of constraint that is specific to a company had to be defined. The concept of company resources was formulated to represent this constrain. Some examples of these constrains include:

- Bond capacity
- Access to credit
- Equipment and Plant
- The finite number and capacity of sub-contractors and suppliers
- Administrative and support staff

These company resources were represented in the form of the number of large, medium size and small projects a company can concurrently perform if not constrained by the availability of project resources. Each resource was associated with a unique waiting file whose queuing capacity was also defined. The definition of this metric influenced the company's ability to bid or not to bid projects and the also affected the price that the company carried in its bids. These company resources also gave an indication of the prevailing workload at the company and its past work load (i.e., by scanning the resource utilization values).

In conclusion, a special purpose simulation template was developed within the Simphony environment. A model that mimics the execution of awarded projects by the Company of Interest was then built using the special purpose template. This template comprised a number of custom modeling elements and a custom entity. The entity was developed in such a way that it encapsulates the behavior of a typical construction project and project team. Details of the special purpose template (entity and modeling elements) will be discussed in the following sections. The model created using this special purpose template is also discussed.

6.3.1 Special Purpose Template

This section is dedicated to presenting the modeling elements that were created as part of the special purpose template. The custom entity was defined as a flow unit to transit models built using these special purpose template modeling elements. The graphical appearances of the modeling elements that constitute this special purpose template are summarized in Table 6.0.

Table 6.0: Graphical Layout of Modeling Elements in the Performance Management Special Purpose Template

Graphics of Modeling Element	Graphics of Modeling Element
GLP ▶ GenerateLargeProject	▶ PP ▶ ProjectProcessor
GMP ▶ GenerateMediumProject	▶ PM ▶ PerformanceMeasurement
GSP ▶ GenerateSmallProject	▶ RPF ▶ RegisterProjectFinish
▶ RLPR+ ▶ RequestLargeProjectResource	▶ RPR- ▶ ReleaseProjectResource
▶ RMPR+ ▶ RequestMediumProjectResource	 RTI Connection
▶ RSPR+ ▶ RequestSmallProjectResource	CBI CompanyBidInformation
▶ RPS ▶ RegisterProjectStart	

6.3.1.1 Special Purpose Simulation Modeling Elements

A special purpose discrete event simulation template was created as part of the development work. This template was used to construct a model that would emulate the project execution phase at the company of interest. The constructed model was made up of a mix of general purpose simulation template modeling elements and the special purpose template elements.

In designing these special purpose template elements, there were two behaviors targeted. These include:

- The ability to play a contributing role to the execution of awarded projects in a resource constrained fashion.
- The ability to receive or send messages from or to the AnyLogic federate as execution of awarded projects progresses.

Details of the behavior embedded within each modeling element in the special purpose template are summarized in Table 6.1.

Table 6.1: Behavior of the Special Purpose Template Modeling Elements

Modeling Element/Construct	Label	Role
Large Project Generator	GLP	Receive interactions of awarded large projects (from the AnyLogic federate), create a corresponding project entity instance, assign it attributes from received data and send off the entity into the model. The element also tags the entity with the company resource type (i.e., Large company resource) that it will require and subsequently release
Medium Project Generator	GMP	Receive interactions of awarded medium size projects (from the AnyLogic federate), create a corresponding project entity instance, assign it attributes from

Modeling Element/Construct	Label	Role
		received data and send off the entity into the model. The element also tags the entity with the company resource type (i.e., medium company resource) that it will require and subsequently release
Small Project Generator	GSP	Receive interactions of awarded small projects (from the AnyLogic federate), create a corresponding project entity instance, assign it attributes from received data and send off the entity into the model. The element also tags the entity with the company resource type (i.e., small company resource) that it will require and subsequently release
Request Large Project Resource	RLPR+	Places a formal request for one large project company resource for every large project entity that arrives at it. This element also registers the project amongst the list
Request Medium Project Resource	RMPR+	Places a formal request for one medium project company resource for every medium project entity that arrives at it.
Request Small Project Resource	RSPR+	Places a formal request for one small project company resource for every small project entity that arrives at it.
Register Project Start	RPS	It transfers the project entity arriving at itself from the list of projects queued for company resources into a list of projects

Modeling Element/Construct	Label	Role
		in process
Project Processor	PP	Delays the flow of entities transferred into it by one calendar day. This mimics the execution/performance of the project for 24 man-hours for all resources engaged on the project. Advancement made in performing a day's work is dependent on the day's production efficiency. Releases the project resources whose engagement on the project is complete. It sends an interaction of these resources to the AnyLogic federate so that the resource agent population can be replenished
Performance Measurement	PM	Invokes all entities transferred into it to evaluate the work day's performance. This element also serves as a buffer for all the measured project performance observations. Keeps a record of the daily performance of all projects processed at the COI in fields defined within this element.
Register Project Finish	RPF	Project entities arriving at this entity get transferred from the list of projects that are in process to the list of completed projects
Release Project Resource	RPR-	Releases the company resource that is

Modeling Element/Construct	Label	Role
		attached to any project entity that is transferred into it. Represents the replenishment of the production capacity at COI for that project size
Company Bidding Information	CBI	Receives a request from AnyLogic federate at the start of federation simulation, to create COI/Simphony federate company resources and waiting files and does so instantaneously
		Receives requests for Company resource and file details from COI Ambassador in AnyLogic federate and sends that information instantaneously
		To read input values from the UI controls of the windows form application and store them within its fields
		Avail input values that it read from the UI at simulation start-up, to other modeling elements during simulation execution
RTIConnection	-	It is responsible for managing the connectivity protocols between the RTI and the other special purpose template elements

The special purpose template was created in such a way that it is compatible with other general purpose modeling elements in the Simphony simulation system. The graphic appearance of each of these modeling elements is summarized in Table 6.2.

6.3.1.2 Special Purpose Template Simphony Model

A model was created using the special purpose modeling elements that had been created. The model was setup in such a way that it emulated the typical project execution processes at the company. This involved the capture and engagement of the appropriate company resources, retaining them and their subsequent release. This model was embedded within the Windows Form Application so that it was an integral part of the Simphony federate. A screen shot of the model layout is presented in Figure 6.8. The paragraphs that follow are dedicated to discussing how the model was used within the Simphony federate, the process logic represented in the model layout and the role that each element contributes to that.

As soon as the simulation execution of the distributed simulation federation is commenced, the embedded Simphony model is deserialized by the Windows form application and reference made to the "CBI" modeling element. All input values defined by the analyst/modeller in the User Interface of the Windows form application are then transferred from their respective controls to the corresponding public fields defined within the "CBI" modeling element. This "CBI" element serves as a place holder (buffer) for information required by modeling elements (and the project entities) in the course of the simulation execution.

As the simulation progresses, new projects are created in the virtual construction industry. All companies operating within this industry are notified of these new projects so that those interested in acquiring the project can submit their most competitive bid for it. Bids received are evaluated and the company with the least bid gets awarded the project. This competitive bidding process takes place within the AnyLogic federate. The Company of Interest (COI) is represented within the virtual construction industry in the AnyLogic federate by a COI ambassador agent (COI Amb. Agent).

In cases where the COI Amb. Agent is awarded a project, it requests for the project required resources from the resource pool defined at the virtual construction industry level. Once granted the required resources, COI Amb. Agent sends details of this project to the Simphony federate via the COSYE RTI. Projects that are not awarded to the COI Amb. Agent also get sent to the Simphony federate by this agent.

All messages of projects sent by the AnyLogic federate (by the COI Amb. Agent) are first received by the RTI Connection modeling element on the Simphony federate side. Received messages containing project details are then passed on to the appropriate project generator modeling elements by the RTI Connection element. This implies that received messages of small projects are passed onto the "Generate Small Projects" modeling element, those of large projects are passed on to the "Generate Large Projects" modeling element while those for medium size projects are passed on to the "Generate Medium Projects" element.

On receiving this message type (i.e., containing details of a project), the respective project generator modeling element creates a proxy object instance (project entity) that encapsulates the received information of the project. In situations where the project was not awarded to the Company of Interest (Simphony federate), this object instance is stored in a list that contains lost projects. If the project was awarded to the Company of Interest, details of this project instance are stored within a list that contains projects awarded to the Company of Interest. Thereafter, this object instance (project entity) of the awarded project is transferred into the Simphony model.

Such a project entity may either arrive at a "request large project resource", or "request medium project resource", or "request small project resource" modeling element. Details of these modeling elements are shown in Figure 6.8. The Figure summarized the layout of the Simphony DES model that was embedded within the windows form application to mimic the project execution operations at the company of interest.

The model contained embedded resources and waiting files which were intended to mimic the company resource constrains at the company of interest. The modeling elements representing these resources are labelled as follows: "LARGEPROJECTRESOURCES", "MEDIUMPROJECTRESOURCES", and "SMALLPROJECTRESOURCES". The waiting file elements are those labelled as follows: "LARGEPROJECTQUEUE", "MEDIUMPROJECTQUEUE", and "SMALLPROJECTQUEUE".

Project entities arriving at the resource request modeling elements have their resource requirements queued within the waiting files (project queues) and the flow of the entity halted. These entities are stored amongst the list of projects awaiting company resources so that they can be processed. When the resource requirements are fulfilled, the request is removed from the waiting file and the entity transferred out into the "Register Project Start" modeling element. This element moves the project from the list of projects waiting for resources to a list of projects that are in process at the company of interest. Thereafter, the project entity is routed out into the next three modeling elements ("Project Processor", "Performance Measurement", and "Conditional Branch") where it flows in a cyclic fashion to mimic its execution until it is considered complete.

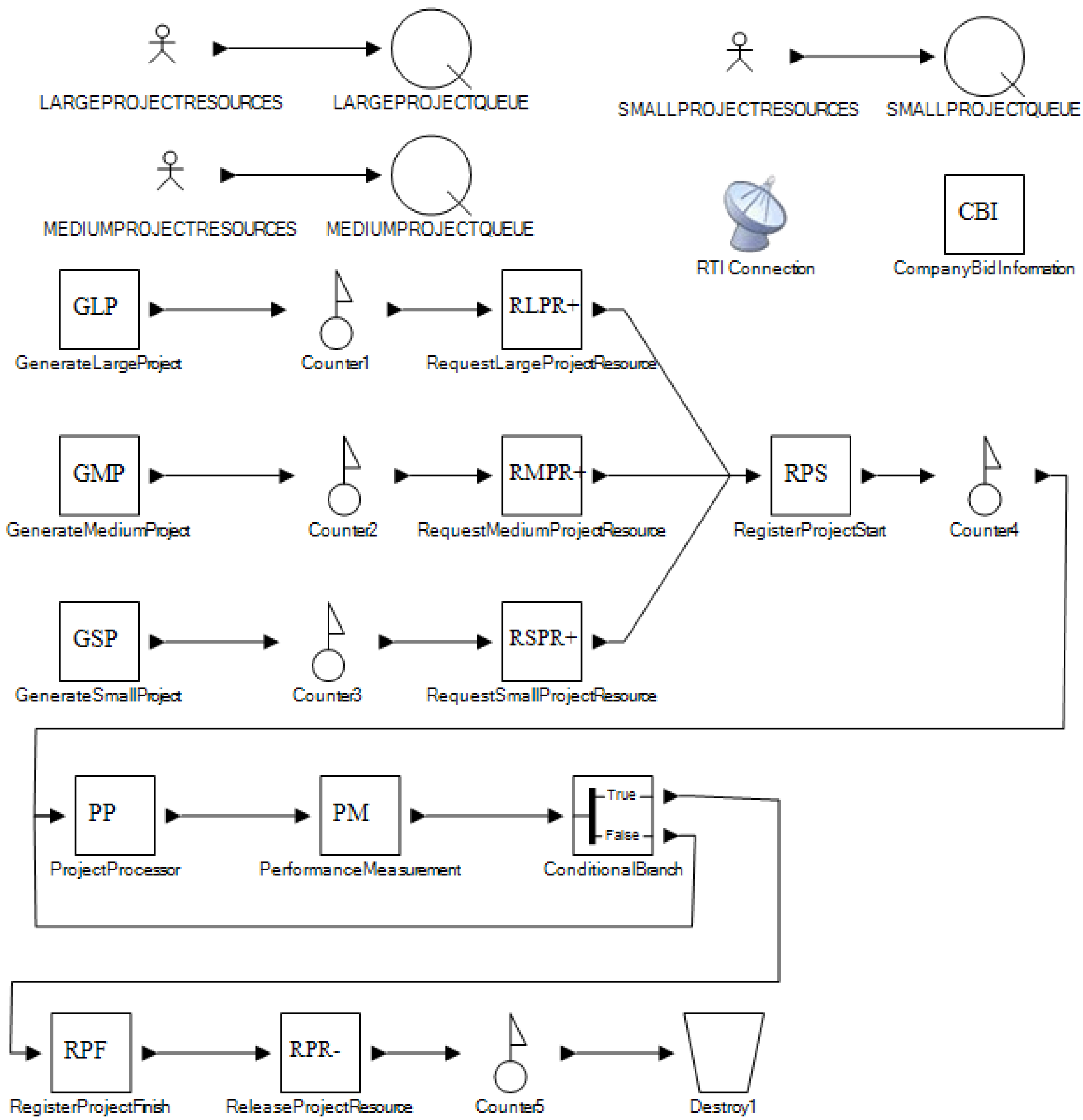


Figure 6.8: Discrete Event Simphony Model Layout that was embedded within the Windows Form Application

Every project entity transferred into the "Project Processor" modeling element is delayed by the equivalent of one calendar day. This delay was meant to mimic the execution of the project for that work day. The work to be performed on a project was expressed in terms of man-hours (for each project required resource) therefore, the amount of work left to complete would be less that the effective day's hours worked. This modeling element reduces the quantity of man-hours that each resource engaged on a project is left with before it can be released. It should be noted that the time unit used within the Simphony model is ***seconds***. This was the case to enforce the IS unit requirement imposed by the HLA for the development of distributed simulation systems. The last thing that this modeling element does when an entity is transiting is that it checks for resources that have fulfilled their required man-hours on a project and releases those resources. This element finalizes their release by sending an interaction of the released resources to the resource agent in the AnyLogic federate via the COSYE RTI. When this Agent receives the interaction, it replenishes the resource pool defined at the virtual construction industry by the number released.

Thereafter, the project entity is transferred out of the "Project Processor" element and into the "Performance Measurement" element. This modeling element prompts the project entity to evaluate its performance for the just completed day's work. The behavior of the project entity mimics that of a real project and its project team. Project teams evaluate and track their project's performance as it progresses in a real life setting. The project entity was configured to emulate a similar behavior for every work day completed. The project entity evaluates its production efficiency for a given work day and the effective progress made in a day. It also evaluates its cost performance, whether safety and quality incidents are due to occur and processes these events. The day's performance observations for each project entity are then stored within the "Performance Measurement" modeling element and appropriately time stamped. This modeling element also checks to see if all project resources have completed their engagement on the project. In case all resources have completed their engagement on a project, that project is assumed to be completed. The floats attribute (ProjectEntity.Floats[0]) for the project entity is then set to 1.0 to indicate that the processing of the project is completed. The project entity is then transferred into the conditional branch modeling element.

When project entities arrive at the conditional branch element, they are queried to establish whether they have been fully processed. Entities that have been fully processed have their Floats[0] attribute set to 1.0 and are routed out through the top output port. Entities whose processing is still on-going are recycled back to the "Project Processor" modeling element for another iteration of processing. This cyclic process continues until the execution of the project is completed.

Projects that are completed are routed into the "Register Project Finish" modeling element. This element transfers the project entity from the list of projects in process at the company of interest to a list of completed projects. The project entity is then transferred into the "Release project resource" modeling element. This modeling element releases the company resource that the project entity was granted to facilitate its processing. The release of this company resource replenishes the available production capacity for that project size at the company of interest. Thereafter, the project entity is routed through a counter then into a "destroy" modeling element where it ends its journey through the model.

6.3.1.3 Special Purpose Simulation Entity and Performance Measurement

A custom entity was created for use within the special purpose simulation template. This entity was meant to represent construction projects and their respective project management teams. In order to effectively do that, a number of behaviors were embedded within the custom entity. Examples include: the ability to track the daily performance of the project, the ability to track resource usage on a project and update progress made, and the ability to know when resources should be released and when the project is completed.

In this section, the algorithms used to evaluate and track the project performance, are presented. Mathematical formulations and flow charts are used to summarize these algorithms.

6.4 MODELING THE PERFORMANCE OF A CONSTRUCTION COMPANY

Performance of a construction company is modelled basing on the philosophy that performance is a hierarchical phenomenon. This means that the performance at any given level is dependent on the performance at lower levels. Therefore, aggregating the lower level performance details facilitates the roll up of performance details to higher levels. In this thesis, performance was tracked at a project level and rolled up to obtain company performance. It was not feasible to drill down and track individual worker performances given that they mostly work in groups/teams (and don't produce anything measureable as individuals). Also, tracking individual worker performance would be expensive in terms of simulation modeling and computing effort. Consequently, project level performance was tracked and considered as the bottom line.

The project entity used within the special purpose simulation template was customized to behave in a fashion similar to a project. This meant that all the performance tracking details were encapsulated by this project entity construct.

In the paragraphs that follow, an overview of company performance is presented and the discussion narrowed down to the performance sub-category modelled within the special purpose simulation template.

The performance of a contractor company in the construction industry was categorized into two i.e., strategic performance and operational level performance. Figure 6.9 presents a schematic layout that illustrates how each of these performance categories feed into overall company performance. Each of these performance categories are discussed next.

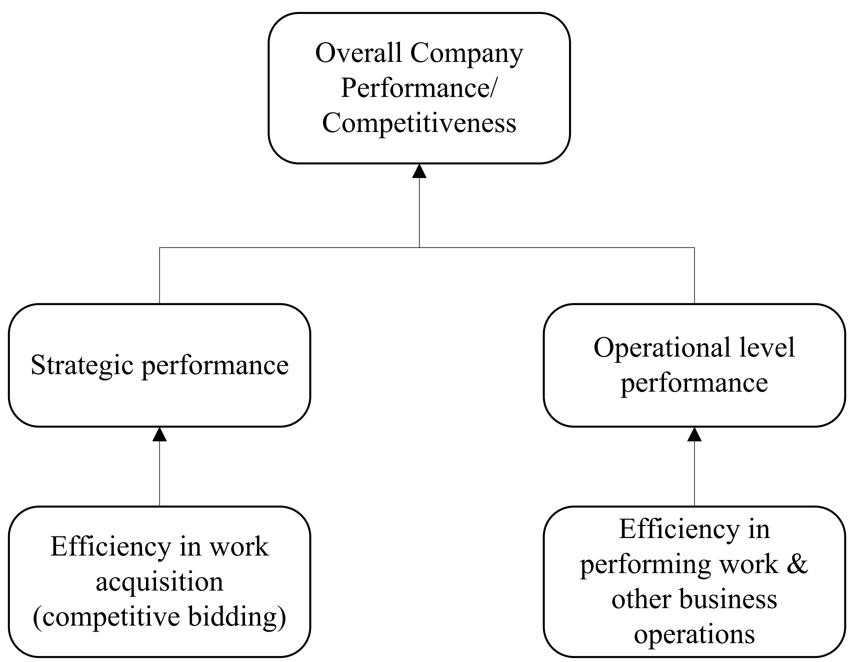


Figure 6.9: Performance Sub-Categories that Feed into Overall Company Performance

6.4.1 Strategic Level Company Performance

Strategic level performance represents the effectiveness of a company in acquiring work within a competitive environment. This metric is believed to be dependent on two measures that are summarized in Figure 6.10. Each of these two measures is discussed further and the mathematical equations used to calculate their values are presented.

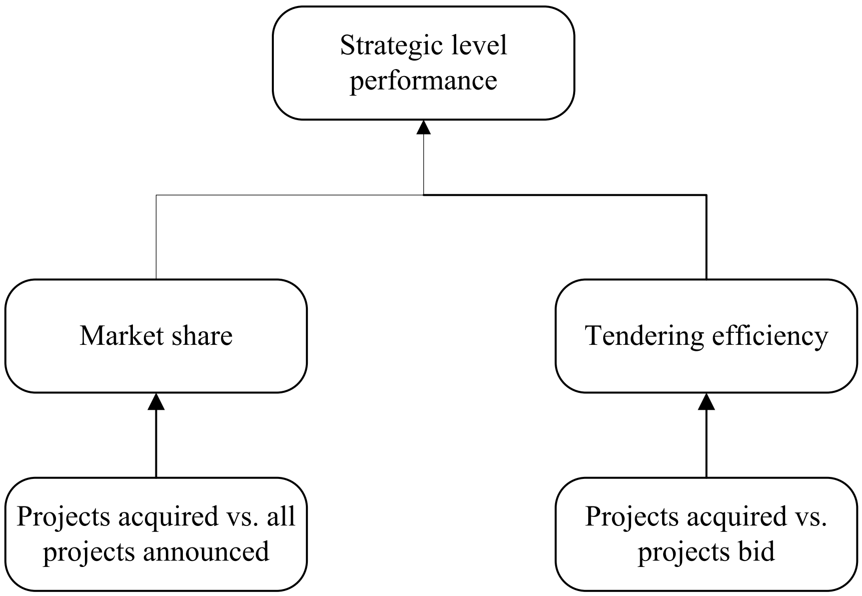


Figure 6.10: The Different Performance Types at a Company Operational Level

This type of performance i.e., strategic level performance was modelled explicitly within the AnyLogic federate using an agent-based modeling approach. Construction projects were modelled as tokens while companies (modelled as semi-autonomous agents) operating within a virtual construction industry was setup to compete for these projects as a means of acquiring work. Details (of this competitive bidding process) that evolved in the course of simulating this federate were passed on to the Simphony federate via the COSYE RTI. The Simphony federate then evaluated values of the market share and tendering success for the company of interest at the end of the simulation, based on this received information.

6.4.2 Market Share and Tendering Success

Tendering is a phase in the project life cycle in which an owner selects and awards a project to a contractor that they feel is best suited to perform the work. The majority of construction contracts are awarded after a competitive bidding process. This is the only way that a project is assumed to be awarded to a general contractor in this study.

Of all the business operations that general contractors in the construction industry undertake, competitive tendering is one that exposes companies the most to direct competition with peers that operate within the same industry. The success of a company at this stage influences the potential of the company to make money and grow. As a result, the performance of a company with respect to how it efficiently it acquires work relative to its competitors, is a good measure of how competitive the company is.

In this study, two measures are proposed to measure the degree of competitiveness of a company with respect to tendering. These include:

- Market share and
- Tendering success

Market share is an indicator that reveals the proportion of projects that the company has acquired relative to those announced in the industry within which it operates for a specific period of time. Equation 6.0 shows how this metric is computed. This metric measures how efficient the company is in making a right decision on whether to bid or not to bid a project. Market share is believed to be a function of both the relative number of projects acquired to those announced and the relative value of acquired projects and those announced. It was therefore expressed as such.

$$Market\ Share = 0.5 * \left[\left(\frac{Number\ of\ projects\ awarded}{Total\ number\ of\ projects\ announced}\right) + \left(\frac{Dollar\ value\ of\ projects\ awarded}{Total\ dollar\ value\ of\ announced\ projects}\right)\right] \quad (6.0)$$

Tendering success on the other hand is a metric that indicates the efficiency of the company in acquiring a job that it decides to bid (See Equation 6.1). In other words, it indicates the efficiency of the estimators in generating bid prices that out-compete other bidders. Tendering success focusses on what was won of those that were bid and is also a function of number of projects and dollar value of projects.

$$Tendering\ Success = 0.5 * \left[\left(\frac{Number\ of\ projects\ awarded}{Total\ number\ of\ projects\ bid}\right) + \left(\frac{Dollar\ value\ of\ projects\ awarded}{Total\ dollar\ value\ of\ projects\ bid}\right)\right] \quad (6.1)$$

Both measures are provided for in the performance management system that was developed because they indicate different things about a company's performance. Market share indicates how dominant a company is within its industry. Tendering success indicates how efficiently the company is utilizing its resources in going after projects in a competitive bidding process.

The 0.5 multiplier is used to obtain the average performance of the company with respect to number of projects and value of projects.

The modeller would then specify within the inputs of the application, whether they would like to track both performance measures, one of them or none of them. If there are to track at least one of these measures, they would then proceed to specify the relative importance that they attach to the measure through the use of a pair-wise comparison scheme. These relative importance values would be used to calculate the weights which would in turn be used to calculate the overall company competitiveness index.

6.4.3 Performance at an Operational Level

The performance of a construction at an operational level was quantified using traditional performance measures. Examples of these included:

- Production efficiency
- Cost slippage
- Schedule slippage
- Quality rating and
- Safety rating

It is common knowledge within the construction domain that each of these performance measures is affected by shared or unique factors. Comprehensive identification of all of the factors that affect each measure and subsequent refinement to redefined number of those that strongly influence the performance measure, is non trivial. In fact, some similar studies have been done for some of these measures. The challenge in most of these is that they have largely been context specific i.e., based on information about a specific construction industry which may not necessarily apply to all. Also, commissioning a comprehensive fact finding study of these factors for each measure would be counterproductive and not feasible within the boundaries of this thesis. Other options were therefore considered.

A mixed approach was used in the identification of factors. This involved reviewing and summarizing the top frequently reported factors in the literature and having formal and informal discussions with practitioners in the construction industry within Alberta. The factors that seemed to be in agreement with information obtained from both sources were considered in the model. The factors defined in the model are static in the sense that the user can not add to those factors for a given measure. They can strategically remove from that list be stipulating a negligible influence (i.e., factor weight) of the factor on its performance measure.

The relative influence of the factors on a given performance measure is another important aspect that needed to be resolved. This was left to the modeller/analyst (i.e., user of the application) to define as inputs prior to model execution. The assumption made in doing this was that the analyst would be a domain expert that knows the operational details of typical contractor companies operating within the construction industry. Alternatively, it was assumed that the analyst would be guided by a domain expert in the definition of the relative extent to which different factors affect a given performance measure. This approach was deemed acceptable as it captures and makes use of domain expert knowledge that is has not been well documented in the literature. In addition, the approach avoids a situation in which the developed application would be context specific. The approach mimics the calibration of a model to fit a given context by the analyst inputting knowledge of their context (i.e., operations) in a convenient fashion.

Data grid view controls were provided within the Windows Form Application component of the Simphony federate. Each performance measure had a unique DataGrid view provided for it. These datagridview controls were setup to facilitate the analyst to define their knowledge of the extent to which factors affect a given performance measure through a pair-wise comparison scheme. The pair-wise comparison scheme was used because each measure had more than two factors identifying as influencing them. Literature shows that effectively ranking many factors or criteria can best be achieved through a pair-wise comparison scheme. The comparisons are defined linguistically by the analyst after which an Analytical Hierarchical Process (AHP) is used to generate a comparison matrix of crisp values and crisp values for factor weights that represent the extent to which each factor affects its performance measure.

The last two paragraphs discussed how the relative influence of factors affecting a given performance measure are quantified. Next, the quantification of the factor itself is discussed.

Quantifying factors that affect performance measures is non-trivial because the values of these factors change dynamically as the work conditions at the company vary. In addition, it is extremely difficult to know beforehand what the different conditions at a company are going to be as this depends on other events (e.g., how much work the company is awarded) and the sequence in which these events occur. Simulation is robust enough to analyze phenomena of this nature in a realistic fashion. Simulation explicitly overcomes the problem of tracking variables that depend on previous event occurrences and that change as time progresses. It is for this reason that simulation was used in the quantification of these factors as time progresses. The simulation system quantifies these influencing factors at discrete points in time. Details of the mathematical and numeric formulations used in the quantification of these factors are discussed in the following paragraphs.

Insights have been gained into how to quantify the value of the factors and the extent of influence they have on their performance measure. It is worthwhile noting that these evaluations are done at a project level and rolled up to a company level. As projects get executed, their performance is impacted by things that occur at a project level and certain things that are characteristic of the company that is performing the project. This implies that performance measure influencing factors can be sub-categorized into two. These include:

- Project specific influencing factors
- Company specific influencing factors

The two factor sub-categories are summarized in Figure 6.11.

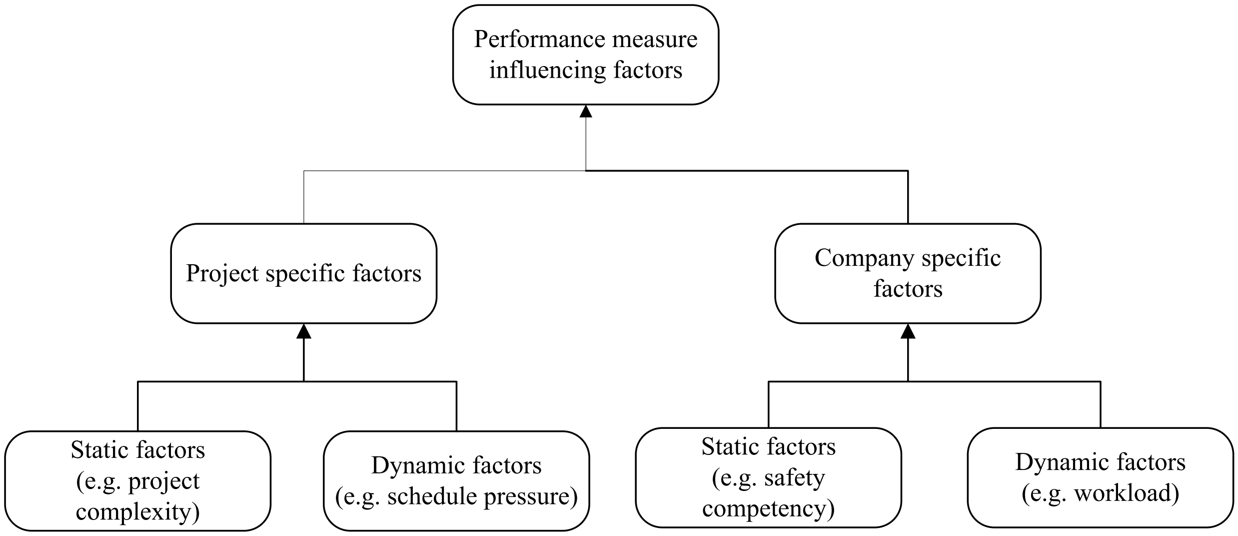


Figure 6.11: Categories of Performance Measure Influencing Factors

All project specific and company specific factors can both either be static or dynamic. Most static factors at the company represented the competencies that exist in each of the performance measures. In a real life situation, most of these are dynamic as the competency can be boosted through trainings or hiring. However, in this thesis, these were assumed to remain static throughout a given simulated period.

Static factors for the company were defined by the analyst for the company of interest prior to simulation execution. These were regarded as the attributes of the company of interest. The static attributes for the projects were also derived from the attributes of the project e.g., project complexity, project safety risk, project engineering quality, and project owner trait. These were assigned as attributes at the time of project creation by the Bid Manager Agent within the AnyLogic federate. Values for the dynamic performance measure influencing factors were computed on the fly was the simulation progressed. Figure 6.12 summarizes the steps for obtaining the weights for the influencing factors.

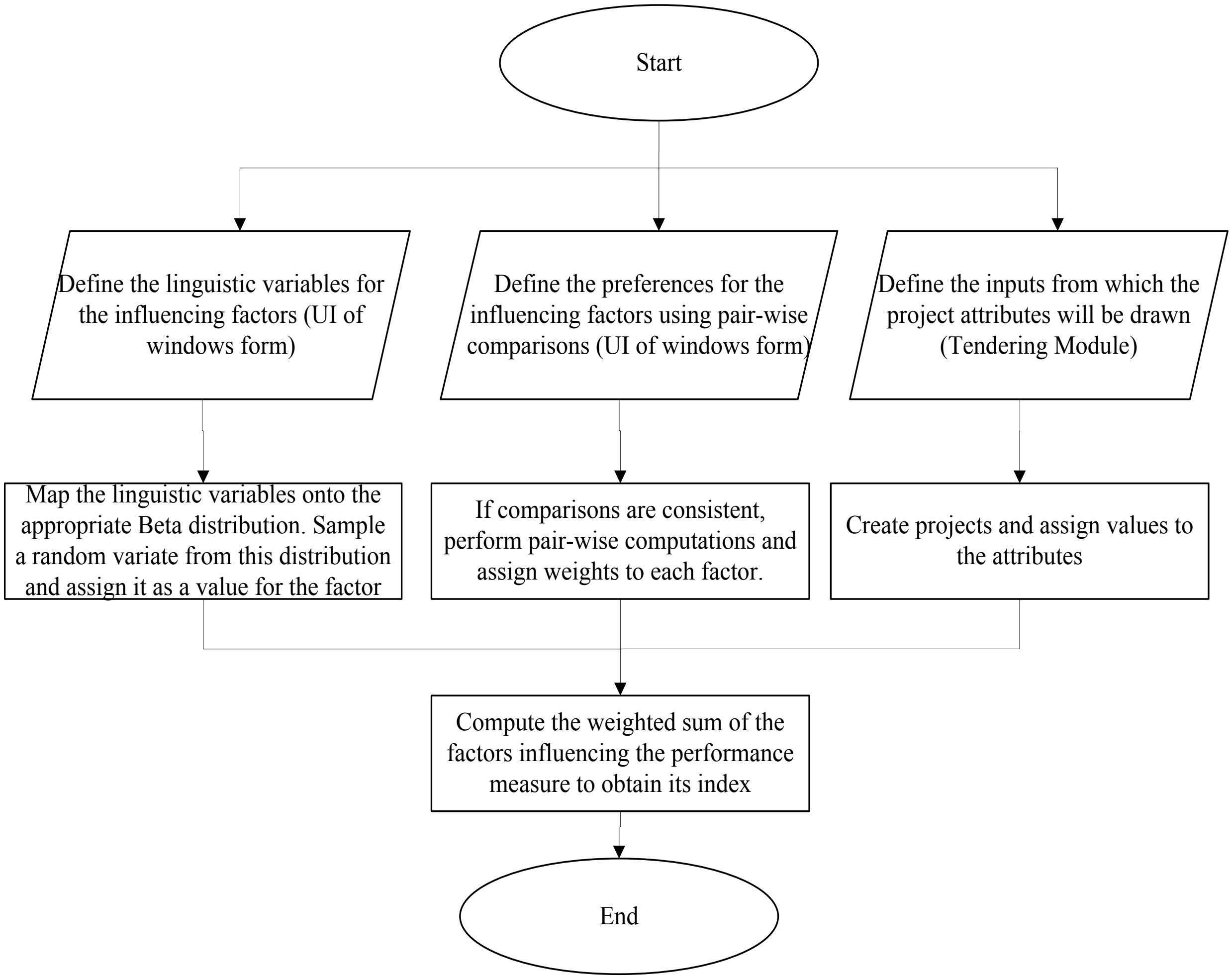


Figure 6.12: Schematic Layout of Process for Computing Weights for Performance Measure Influencing Factors

The performance measure values were assumed to be scaled on the range zero to one. This was done for convenience in the calculations. At a project level, the value of each performance measure then calculated as a simple weighted average of the respective influencing factors.

This is expressed mathematically in Equation 6.2.

$$Performance\ Measure\ Value = \sum_{i=1}^{n} Inflencing\ \ factor\ value_i \times factor\ weight_i \tag{6.2}$$

Project level performance measures were then rolled up to obtain company level performance for the company of interest. The mathematical operations made use of in the roll up of performance measures include summation and averages. Figure 6.13 summarizes a schematic layout that illustrates the hierarchical setup of a company's performance based on projects, performance measures and their influencing factors.

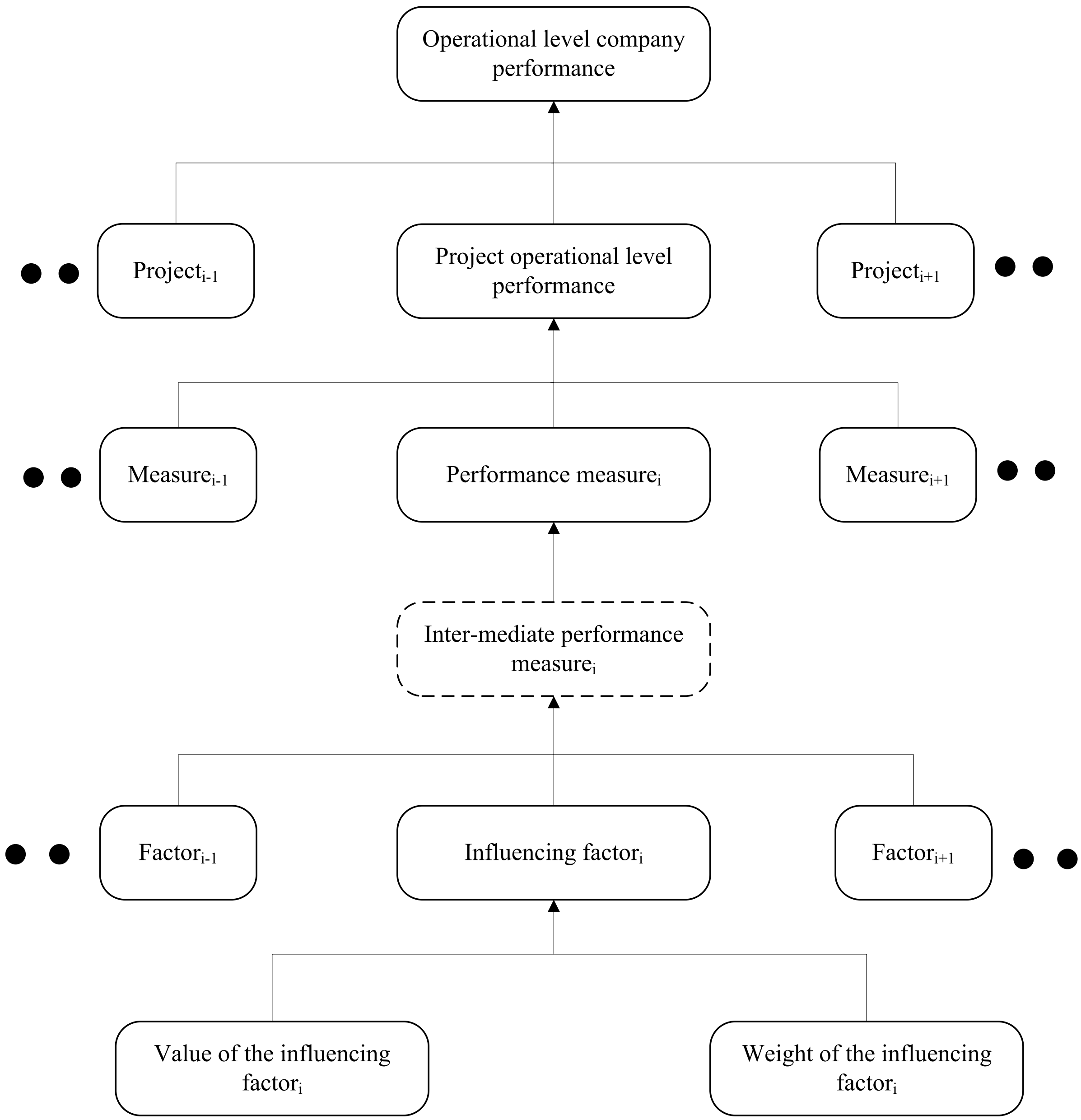


Figure 6.13: Hierarchical Approach used to Model the Performance of a Typical Construction Company

This entire process was mapped onto a simulation model. This first step in this mapping process involved creating a custom simulation entity referred to as a ***"project entity".*** This entity was setup to represent a unique project and its management team. This implies that it had the ability to encapsulate the attributes of the project that it represents and the performance measurement roles that the project team is typically tasked with. Other constructs i.e., modeling elements were setup to trigger the measurement of performance of a given project entity soon after a specific work period. In addition, the modeling elements served as place holders (buffers) for collected performance observations. Details of how performance measures and aspects that relate to performance measures were evaluated in the simulation model are presented in the sections that follow.

6.4.4 Sequence for Calculating Performance Measures

Performance measures are calculated throughout the course of the simulation for projects that are currently being executed. Projects whose execution was completed and those queuing for their turn to be processed are not evaluated for performance. A design pattern is adopted in which the projects are modelled as flow units (entities) which evaluate their performance on a daily basis. This was done to mimic project control operations performed by project management teams in a real life setting.

The company level performances are also evaluated on a daily basis as long as there is at least one project in process. Three lists were kept in order to facilitate all these computations seamlessly. A list was dedicated to projects queuing to be processed, another for those that are in process and the last for those for which processing has been completed. These lists were stored at a global scenario level within the Simphony simulation modeling environment. This was the case in order to facilitate any modeling element to access the lists for these projects. Also, at the end of the simulation period, this would facilitate the windows form application to retrieve these projects, obtain values from them and display them in its User Interface.

There were a number of performance metrics to be computed on a daily basis. This was done by strictly following a logical sequence summarized in the flow chart in Figure 6.14.

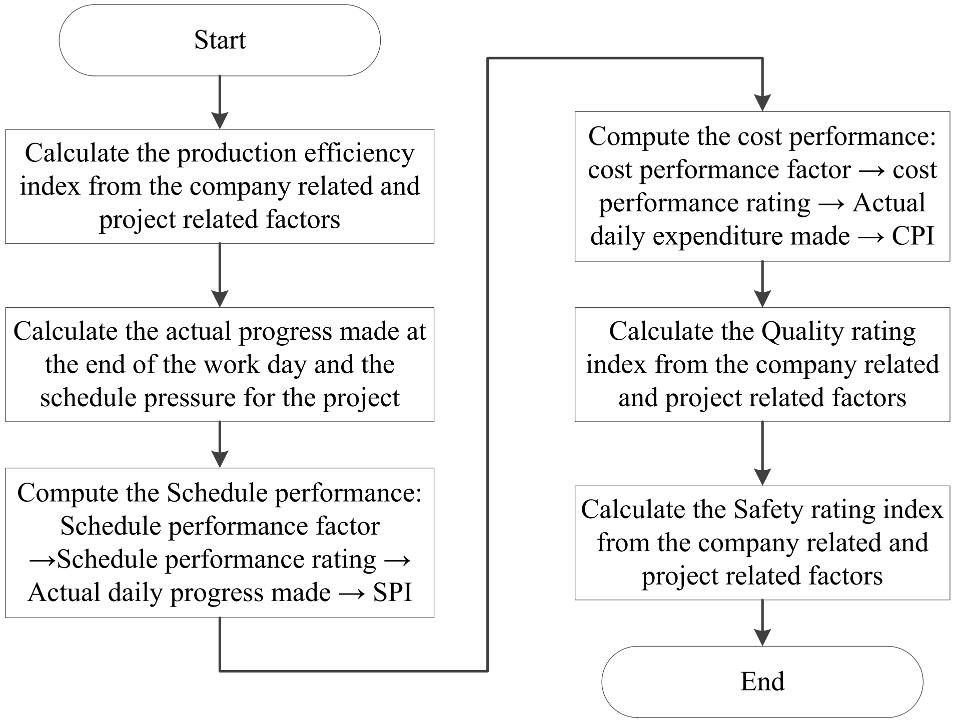


Figure 6.14: Sequence with which performance indices and measures are calculated in the model

The discussion that follows presents details of how the different indices and performance measures were computed. This discussion is not based on the order in which these variables are calculated within the model. First, details for the calculation of safety and quality performance are presented.

6.4.5 Modeling Quality and Safety Incidents

Every company in the construction business stands the risk of the occurrence of safety and quality incidents as it goes about its day to day business operations. For a very long time, safety and quality in construction and many other industries had been viewed as qualitative phenomena that cannot easily be represented and analyzed in a quantitative fashion.

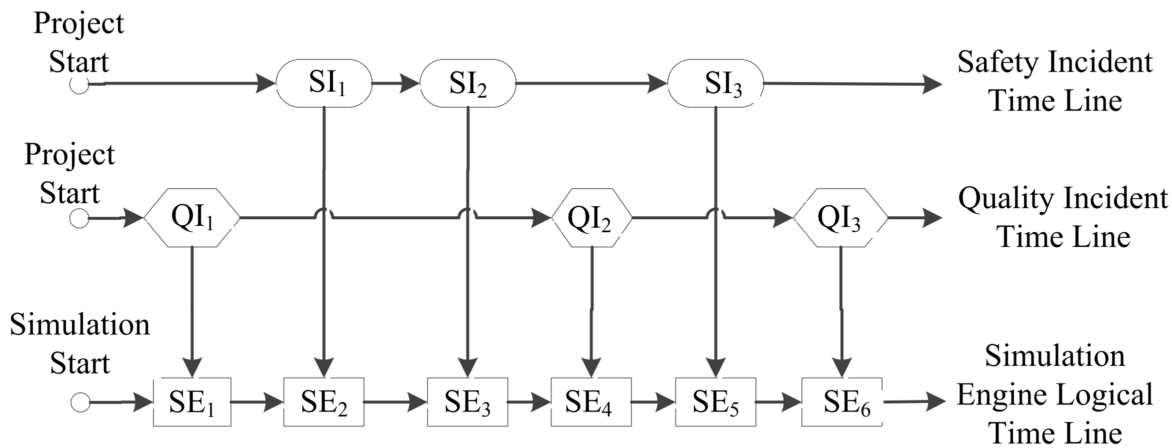
In this thesis, an approach was devised to formalize these two phenomena for purposes of representing them in a computer simulation model. Simulation modeling of these two performance measures is discussed together in this section because the same philosophy is adopted in the formalism of both.

The approach perceives quality and safety as phenomena that can be accurately represented by their corresponding incidents. These incidents (i.e., safety and quality incidents) can then be directly mapped onto simulation events as a first step in the formalism process. This implies that any simulation event related to safety or quality that is processed can be related to the occurrence of a safety or quality incident on a real project or at the company. It is also known that these incidents are typically spread out and can be said to be separated by a metric referred to as an interval or time between incidents. It can be stated that this time between incidents varies depending on the conditions prevailing at the company, the attributes of the projects being performed and the static attributes (competencies) of the company.

Furthermore, the value of the time between incidents is highly variable that it is proper to refer to it was a random variable. There are several ways of representing random variables for example; fuzzy logic may be used to represent the uncertainty in the variable or probability and statistics. This thesis adopted a statistics approach because it has been extensively used in simulation for modeling uncertain phenomenon.

Statistical distributions were therefore used for presenting the time between incidents (for safety and quality). Every time that an incident occurs, a new random variable (incident inter-arrival time) is drawn from that statistical distribution. As the simulation progresses, the time that has passed since the last incident is subtracted from the time to the incident to obtain the time left to the next incident. A quality of safety incident occurs when the time left to the next incident is zero.

The time left to the next incident is then lengthened or shortened as the simulation progresses to mimic the influence of the prevailing work conditions on the occurrence of quality and safety incidents. As work conditions deteriorate, the time left to the next incident is shortened so that an incident materializes earlier. On the contrary, the time left to the next incident is increased as prevailing conditions improve so that the occurrence of incidents is pushed further ahead in to the future. Figure 6.15 utilizes a schematic layout to summarize the concept of safety and quality incidents being mapped directly onto simulation events within a simulation model.



SI_i - Safety Incident

QI_i – Quality Incident

SE_i – Corresponding simulation event

Figure 6.15: A Schematic Layout Showing the Mappings of Quality and Safety Incidents onto Simulation Events

The prevailing conditions with respect to quality and safety were each represented in the simulation model by a metric known as quality rating index and safety rating index respectively. These indices were evaluated based on the values and weights of the factors that were perceived to affect quality and safety. The quality and safety indices were used to influence the modeling of quality and safety incidents in two ways. These include:

- Lengthening or shortening the time left to the next incident
- Determining whether an incident actually occurs at the time of processing the incident i.e., time left to the next incident is zero.

The quality and safety rating indices were computed at a project level at the end of each work day. The computation of these was based on the values of their influencing factors and their weights.

Factors that were explicitly considered to influence the quality rating include:

- Quality control and assurance systems at the company of interest
- Worker and work method effectiveness
- Project owner trait

- Complexity of the work
- Project engineering quality
- Safety performance (safety rating)
- Schedule pressure
- Cost pressure

The factors that were considered to influence safety rating include:

- Soundness of safety policies and practices at the company of interest
- Effectiveness of safety supervision and audits
- Worker experience
- Extent and relevance of worker safety training
- Worker safety consciousness and caution
- Project safety risks
- Schedule pressure
- Cost pressure

These indices would only be used to modify the time left to the next incident when there was a change in index value from the previous work day to the current. The percentage change in the rating index was used as a guide for modifying the time left to the next incident.

At the end of each work day, the time left to the next time incident was updated i.e., reduced by one calendar day. When this value reached a value of zero, a corresponding incident was processed. The occurrence of the incident was dependent on the value of the incident's corresponding rating index. A random number between zero and one was used along with the rating index value to decide whether it was actually a realized incident or a missed incident. Regardless of what it turned out to be, a new value for the time to the next incident was drawn from the statistical distribution used to represent the time between incidents. The cycle is then repeated until the simulation comes to an end.

In situations where the incident was realized, the impacts related to that incident were explicitly modelled. In the case of a quality, the amount of rework arising from the incident was modelled. This was sampling a value from a user defined statistical distribution for the amount of rework associated with an incident. Prior to simulation, the analyst would define portion of the completed work since the last incident that would be sent for rework when a quality incident occurs.

For safety incidents, impacts related to lost time and cost were modelled. A chance of occurrence was associated with each impact. This was defined prior to simulation by the analyst using linguistic variables such as extremely likely, unlikely or somewhat likely. These were provided in the user interface of the windows form application. The safety incident impacts related to modified work were ignored/not explicitly modelled in this thesis study because it was assumed to be a form of lost time impact.

It is worthwhile noting that each incident type was explicitly modelled separately using the approach described i.e., quality and safety incidents. Safety incidents were further sub-categorized by incident type (i.e., missed incident, first aid incident, medical aid incident and fatal incident) then each explicitly modelled using this same approach.

In the following two sub-sections, details of how the performance for safety and quality were represented and tracked in the model are presented.

6.4.5.1 Evaluating Safety Performance

An index referred to as safety performance index was used to indicate the prevailing safety conditions on a project and a company as a whole. This index was evaluated using the factors identified as influencing safety and the occurrence of safety incidents. Equation 6.3 was used to calculate the safety performance index.

$$Safety\ performance\ index = \sum_{i=1}^{n} safety\ \mathrm{inf}\ luencing\ \ factor\ value_i \times factor\ weight_i \qquad (6.3)$$

The Safety performance index was also updated whenever a safety incident occurred. This was done to emulate the fact that the safety conditions at the time of occurrence of an incident are not conducive. In order to achieve this, each safety incident type was mapped onto a Beta distribution (See Table 6.2) which would be used to obtain (through a sampling process) a percentage value for reducing the safety performance index.

Table 6.2: Safety Incident and Corresponding Safety Performance Reduction Factor

Safety Incident Type	Distribution for Safety Performance Reduction Factor
Missed Incident	Beta(3.50,2.00,0.90,1.00)
First Aid	Beta(2.60,2.40,0.60,0.80)
Medical Aid	Beta(4.00,3.99,0.10,0.50)
Fatal Incident	Beta(2.00,3.60,0.00,0.50)

Safety performance index for a project was evaluated on a daily basis. Whenever this value changed, the margin by which it changed was used to modify the time to the next safety incident. In cases where this safety index increased (i.e., the project became safer), the time to all safety incidents would be increased hence postponing the occurrence of safety incidents to later date in the future. The likelihood of the incidents occurring and their impacts would also be adjusted accordingly.

It is important to note that although this metric was tracked throughout the simulation and was representative of safety performance, it was not used in rating the overall performance of the company. A more tradition metric (i.e., number of safety incidents realized in 1,000,000 man-hours) was used for this purpose. Safety incidents that occurred in the simulation were tracked and compared at the end to the benchmark value provided in this form, to obtain the company's performance rating with respect to safety.

6.4.5.2 Tracking Quality Performance

Quality performance was tracked in more or less a similar way as safety performance. A quality performance index was evaluated on a daily basis for projects that were in process at the company of interest from its influencing factors. This index was derived from its influencing factors using Equation 6.4.

$$Quality\ performance\ index = \sum_{i=1}^{n} Quality\ influencing\ factor\ value_i \times factor\ weight_i \qquad (6.4)$$

The typical values obtained from this equation ranged from 0.0 to 1.0. These values were transformed into a percentage when reporting model outputs. This metric was used to determine whether or not a quality incident occurs at the point in time that the time to the next quality incident elapses. The quality performance index was also used to rate the company's performance with respect to quality.

6.4.6 Time Lost in a Day

The time lost in a day was an aggregation of lost time from three sources. These include:

- Time lost due to production inefficiency
- Lost time impacts due to safety incident occurrence
- Rework arising from quality incidents.

The contribution of the quality incident and safety incident occurrences to the time lost in a day were scaled to a daily contribution rather than a lump sum contribution to time loss for the project. The Equations 6.5 and 6.6 were used to achieve this.

$$Lost\ time\ contribution\ of\ safety = \frac{\sum lost\ time\ from\ all\ incidents\ since\ project\ start}{Total\ planned\ project\ duration} \qquad (6.5)$$

$$Rework\ from\ quality\ incidents = \frac{\sum lost\ time\ due\ to\ rework\ since\ project\ start}{Total\ planned\ project\ duration} \qquad (6.6)$$

Equations 6.5 and 6.6 represent a realistic estimation of the impacts of quality and safety incidents on the lost time on a project on a daily basis.

The time lost which was as a result of production inefficiencies on a given work day were quantified based on the production efficiency index value. This value was translated into corresponding uniform distributions from which the percentage time lost were to be drawn. The ranges for production efficiency and their corresponding uniform distributions are summarized in Table 6.3.

Table 6.3: Production Efficiency Index Ranges and their Corresponding Percentage Time Lost Margins

Production Efficiency Index	Percentage Time Lost
0.00 – 0.25	Uniform(0.15,0.50)
0.25 – 0.50	Uniform(-0.10,0.35)
0.50 – 0.75	Uniform(-0.35,0.10)

Production Efficiency Index	Percentage Time Lost
0.75 – 1.00	Uniform(-0.50,-0.15)

The time lost on a given work day was then estimated using Equation 6.7. The planned progress on any given work day was assumed to be equal to the total number of work hours in a day (i.e., 24 man-hours).

$$\text{Time lost due to production inefficiency} = \text{Percentage time lost} \times \text{Planned daily progress} \quad (6.7)$$

Time saved was registered as a negative value while time lost was taken as a positive observation. This was guaranteed by the fashion in which the uniform distribution mappings had been setup. The Overall time lost in a specific work day was then found as an aggregation (i.e., summation) of the time lost from all these three sources (production inefficiency, safety incidents and rework).

6.4.7 Money lost on a given Work Day

The money lost on a given work day was calculated in using a procedure that was similar to that used in the *"time lost on a given work day"*. The sources of losses were also assumed to be the identical to those of lost time i.e., production inefficiency, quality incidents (arising from possible material wastage) and safety incidents (e.g., arising from higher insurance premiums, cost of treatment or compensation). The Equations 6.8, 6.9 and 6.10 were used.

$$\text{Money lost due to safety} = \frac{\sum \text{Total costs from all safety incidents since project start}}{\text{Total planned project duration}} \quad (6.8)$$

$$\text{Money lost due to rework} = \frac{\sum \text{Total costs from quality incidents since project start}}{\text{Total planned project duration}} \quad (6.9)$$

$$\text{Money lost due to production inefficiency} = \text{Percentage money lost} \times \text{Planned daily expense} \quad (6.10)$$

The percentage money lost for a specific work day was obtained from mapping identical to those summarized in Table 6.4. These mappings were setup so that profits were registered as negative values and losses as positive values. The Overall money lost on a specific work day was then an aggregation of the money lost from all these three sources (production inefficiency, quality and safety incidents).

The two metrics i.e., daily time and money lost were used in subsequent computations of schedule performance and cost performance. The computation of these two performance measures are discussed next.

6.4.8 Schedule and Cost Pressure

Falling behind a schedule or overrunning the budget for components completed during project execution can cause lots of stress which could result in secondary effects on the performance of the project, not only with respect to schedule but also other performance dimensions. These effects are gradual and do not occur suddenly but rather cumulate on a daily basis. Metrics referred to as ***"schedule pressure"*** for schedule performance and ***"cost pressure"*** for budget performance are used to quantify these effects in this thesis. There have been quite a number of definitions and formulae proposed for quantifying both metrics. However, in this thesis, a simplified version that effectively quantifies and represents them is adopted.

6.4.8.1 Schedule Performance

The computation of schedule performance commences with the evaluation of the actual progress made in a day. The Equation 6.11 was used for this evaluation. Schedule performance is computed on a daily basis at the end of each work day. Man-hours were used in this computation.

$$Actual\ daily\ progress = Planned\ progress\ for\ the\ day - Lost\ time \tag{6.11}$$

The planned progress each work day (Mhrs) was assumed to be equal to 24 hours (24 hour calendar used for simplicity). The subtraction operator is used to generate a small value when time is lost on a given work day (i.e., a positive value for lost time) and a large value greater than 24 when time on the project is saved. The actual daily progress represents the work scope completed for the project in man-hours. Details for the computation of lost time are summarized in the section on lost time computation.

The concept Earned Value Management concept for schedule performance index was then used to compute a schedule performance index. The Equation 6.12 was used to compute the un-normalized schedule performance index.

$$\text{Unnormalized schedule performance index} = \frac{\sum_{\text{Project start}}^{\text{Current date}} \text{Actual daily project progress}}{\sum_{\text{Project start}}^{\text{Current date}} \text{Planned daily project progress}} \tag{6.12}$$

A value greater than 1.0 is good (i.e., implies that we are ahead of schedule) while a value less than 1.0 means we are behind schedule which is bad. 1.0 means the project is on schedule. This value of schedule pressure was then normalized so that it ranges on a scale from 0.0 to 1.0. The assumed maximum un-normalized SPI was taken as 1.2.

6.4.8.2 Cost Performance

Cost performance calculations were based on the values of the daily losses registered on the project from the day of commencement. Its computation made use of the earned value management philosophy. The actual expense on a given project for a specific work day was computed using the monetary loss for that day. The Equation 6.13 was used to evaluate this value.

$$\text{Actual daily project expense} = \text{Planned daily expense} + \text{Money lost for the work day} \tag{6.13}$$

The addition operator is used to reflect the fact that more money is spent on a day when a loss (positive value for money lost for the work day) is made and less money spent when a profit is made. Values obtained from evaluating Equation 6.13 were collected on a daily basis and used to calculate a cost performance index (see Equation 6.14).

$$\text{Unnormalized cost performance index} = \frac{\sum_{\text{Project start}}^{\text{Current date}} \text{Planned daily project expenses}}{\sum_{\text{Project start}}^{\text{Current date}} \text{Actual daily project expenses}} \tag{6.14}$$

The value of cost pressure obtained from Equation 6.14 was normalized and fitted onto a scale that ranges from 0.0 to 1.0. The maximum value of the un-normalized cost performance index was assumed to be 1.2.

6.4.9 Production Efficiency Computation

The efficiency of projects was dependent on numerous factors. Prior to simulation, the modeller would enter the values of the static factors (mostly attributed to the company's properties and work methods). The weights (relative influence on production efficiency) of each factor would also be defined by the modeller. A weighted aggregation mathematical formulation (shown in Equation 6.15) is then used to compute the value of the production efficiency.

$$Current\ production\ efficiency = 0.5 \times \left(\Pr evious\ production\ efficiency + \sum_{i=1}^{n} factor\ value_i \times factor\ weight_i \right) \quad (6.15)$$

The value of the prevailing production efficiency was calculated as an average of the current value and the previous value. This was done to incorporate the effects of previous performance on the current. The factors used in the evaluation of the production efficiency are listed below:

- Subcontractor quality value (portion of work typically subcontracted and the rating of sub-contractors is used)
- Company past project experience (project complexity, cost and scope attributes are utilized)
- Schedule pressure
- Cost pressure
- Quality performance/rating
- Safety performance/rating

Some of these factors are static (remain the same throughout the simulation) while others are dynamic and were evaluated on a daily basis for each project that was being processed.

6.5 SUMMARY OF CHAPTER SIX

This chapter presented details of development work that was done to produce the Simphony federate. Components that make up the Simphony federate (Windows form application and the embedded Simphony) were discussed. Development of each component and their specific roles in the federate has also been presented. The numeric and mathematical formulations used to model the performance of the company of interest at a strategic and operations level have also been presented and discussed.

CHAPTER SEVEN – EXPERIMENTING WITH THE DEVELOPED SIMULATION APPLICATION

7.0 INTRODUCTION TO CHAPTER SEVEN

Chapters four, five and six presented details of the design and development of the simulation application that was created for modeling the performance of a contractor company in the construction industry. This chapter focuses on demonstrating the validity and reliability of the development process and design specifications adopted. It also goes ahead to show that the results generated by the application are reasonable.

As the complexity and scale of a simulation model increase, the verification and validation process exponentially increase. This is particularly common for simulation models that are distributed in nature and are concurrently executing. Typical examples are multi-agent simulation models and distributed simulation models. The simulation model developed in this thesis belonged to both categories. Attempts to verify and (or) validate such systems as one unit could be extremely difficult, frustrating and may not yield any reasonable outcome. Consequently, a piece-wise verification and validation approach was adopted in which components were tested in isolation or in combination with fragments of other components. Failure to prove that these components were unreliable or invalid could be assumed that the entire system would be reliable and generate accurate results when these components are put together to operate as a unit.

There is no single way to validate simulation models. There are several ways to achieve this hence the different types of validation. Data-driven validation is the preference for most simulation modellers especially in cases where the data is obtained from the actual system that is simulated. However, simulation modellers especially within the construction domain don't have the luxury of using this approach because of several constrains that relate to obtaining a good data set.

The other approach entailed a review of results generated by the model. Each component of the application was run separately and the result presented and discussed. First, the AnyLogic federate was run alone. Thereafter the company of interest (Simphony) federate was run alongside the AnyLogic federate. All competitor agents within the AnyLogic federate were left out in this scenario so that the company of interest was the only one operating in the industry. This scenario was meant to show the type of result (i.e., company performance) that could be generated by the Simphony federate. The last scenario run involved executing the entire system as one unit in a fashion that it was intended for use when deployed. This scenario involved subjecting the company of interest to competition from other company agents.

The last section (section 7.8) in this chapter presents details of probabilistic sensitivity analysis that was carried out for purposes of demonstrating that the developed simulation application generates valid and reliable results.

7.1 APPLICATION DEVELOPMENT PROCESS

The development of the application in this thesis followed systematic steps in order to guarantee its reliability and accuracy. The steps followed in the development process are summarized in Figure 7.0.

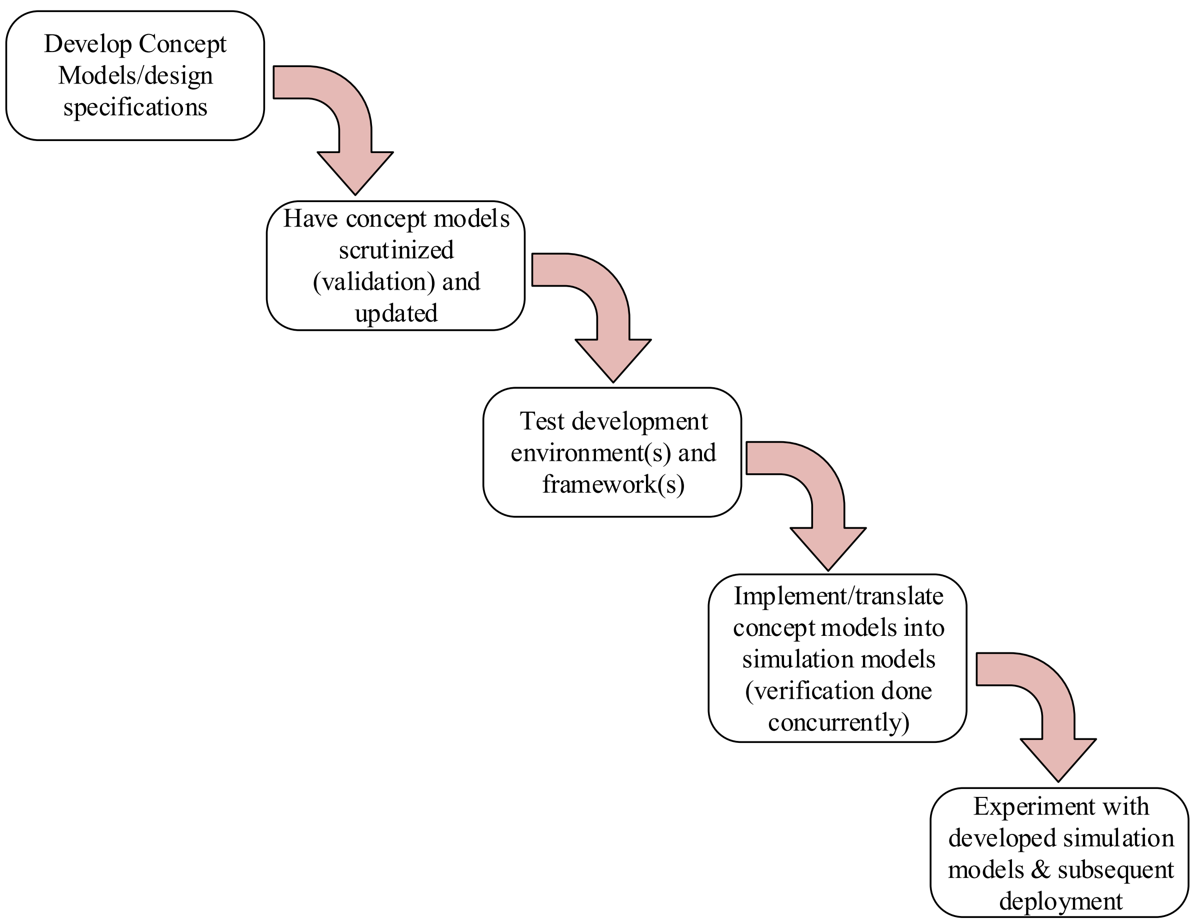


Figure 7.0: Systematic Model Development Process Adopted in this Thesis Study

First, concept models were developed which summarized specifications of components of the application and the envisaged behavior of each component. These were presented in the form of sequence diagrams, activity charts, flow charts, and traditional concept models. The process of developing these documents is often referred to as formalism of a real world system or may also be referred to as an abstraction process.

Next, these designs were validated. Details of this process are elaborated in section 7.2. Thereafter, the designs were translated into simulation models and computer programs that made up the envisaged application. This application was then experimented with to confirm that it generates realistic results.

7.2 VERIFICATION AND VALIDATION STRATEGIES APPLIED

7.2.1 Validation Work Done

The option of a data driven approach to model validation was ruled out because of the challenges associated with the nature of the model and other domain specific problems. These included:

- It would be quite complex and difficult to validate the model as a whole as a result of the breadth in of the scope of the model. It contains several sub-components that perform specific tasks that would each need to be validated. Although doable, it is not feasible especially when constrained by time.
- Validating the model using data would require data on bidding processes of companies within the construction industry, the projects that where bid within a specific period, their attributes, and the rate at which these were created. Data about the performance of companies along with the conditions that were prevailing at the companies when that performance data was collected would be required. Furthermore, several years of data would be required to perform a credible validation of the model. To a large extent, such data may not exist within the industry and for cases where it exists, it may not be precise and may be proprietary to the companies that are tracking it.

A combination of the above factors made it unfeasible for a data driven validation to be adopted in this thesis study. Instead, other validation approaches were chosen to prove that the model generates realistic results. The first of these involved validating design specifications (concept models) for the simulation application.

Concept models serve as the basis for building real models regardless of their form. Once these have been well developed, the burden shifts to using these to develop models. Concept models are represented in different ways. Examples of these (that were made utilized in this thesis) include:

- Flow charts
- Activity charts
- State charts
- Sequence diagrams
- Class diagrams (used in preliminary stages of development but not documented in this thesis)

Soon after these had been developed, they were subjected to rigorous scrutiny. This scrutiny served as the first form of validation that was meant to proof that the abstraction of the real world system was done accurately. The experience, knowledge and skills of practitioners in the construction industry along with that of individuals from academia, were utilized in this validation process. The professionals from industry were engaged informally during industrial chair progress meetings and at other academic events related to the industrial chair (e.g., research poster sessions at annual forum events). Individuals engaged from academia included my professor/supervisor, other professors in the Hole School of Construction Engineering at the University of Alberta, my colleagues (fellow graduate students) and technical support staff working for my supervisor. The concept models were revised/updated to accommodate the feedback obtained from all these sources. This approach is useful for simulation studies especially those that are large-scale and complex. It serves as a viable option where data-driven approaches cannot be adopted. The approach has also been strongly advocated for by Sargent (2013).

Once concept models had been validated i.e., concept models shown to be doing the right thing with respect to the actual system, the bulk of the work left was related to translating these concept models into actual models that mimic the real system behavior. This required that things be done right i.e., making sure that the actual model matches the concept model. The process of confirming this is referred to as verification. There were various techniques applied in verifying model developments. Those utilized in this thesis are discussed in the following sub-section.

7.2.3 Verification Strategies Applied

Simulation models are verified to confirm that they are reliable and match the intended design behavior. The process of achieving that includes eliminating syntax and logical errors that may exist in modeling blocks and code snippets.

Verification of the application developed in this thesis as a single unit proved to be challenging because of the scale and complexity of the application. Consequently, a piece-wise approach was adopted in which component behavior was tested and verified. The components tested include:

- The agent-based model within the AnyLogic federate
- The individual special purpose modeling elements (Part of the Simphony federate)
- The simulation model built using these special purpose template modeling elements. These tests were performed when the model was embedded within the windows form application (Part of the Simphony federate).
- The COSYE framework (Part of the Simphony federate).

The techniques used in the verification of the numerous components included:

- *Breakpoints:* These were used together with stepping options (step into, step over, and step out) to scheme through blocks of code. These were mainly used for the windows application in the Simphony federate. This approach was used to query values stored in collections and within other simple and complex data types

- *Message Boxes:* Message boxes were used to trace interim results generated in the course of simulation. This technique was specifically useful in confirming the behavior of Simphony modeling elements in the developed special purpose template and the model developed from these. This is because the trace environment provided within Simphony was not accessible given that the model was embedded in the windows form application.

- *Trace Log:* Several simulation events are processed in the course of simulation execution. The sequence of occurrence of these events is based on the logic embedded within the simulation model. One way to determine the events that occur and their sequence is to trace details of these and then review the trace logs to verify the correctness of the model. These were specifically useful in the verification of the multi-agent model embedded within the AnyLogic federate. This technique was applicable because the AnyLogic simulation system (interface) was run in an exposed mode so it was easy to view details traced as the simulation progressed.

- *Unit Tests:* Before using software to develop a model, it is usually good practice to test and confirm that the software does whatever it was intended to do. One software development environments (Visual Studio), two simulation systems (Simphony and AnyLogic) and one development framework (COSYE), were used to implement developments. Unit tests were carried out on these (with the exception of Visual Studio) to verify their behavior.

- *Executing the entire model or segments of the model:* Running a model that has errors, especially syntax errors, will result in exceptions being thrown. These problems could then be addressed as they appear. This approach was applied and proved to be useful given the application developed in this thesis was large in scale.

In the implementation of all these verification techniques, the model components tested were expected to run with no exceptions thrown. Also, the generated results were supposed to precisely match the expected results. Achieving this confirmed that there were no syntax and logical errors in the implementation of design specifications.

The following sections discuss approaches for verifying the framework and software development environments used in the thesis. This was followed by scenarios run of individual components of the developed application. The last scenario run involved the execution of the entire application in a form that it would be deployed when solving a real world problem.

7.3 TESTS TO VERIFY THE COSYE FRAMEWORK BEHAVIOR

Prior to translating these concept models into simulation models, the development environments (AnyLogic simulation system and Simphony simulation system) and framework (COSYE) were subjected to preliminary tests to confirm that they were reliable. Preliminary testing was done on AnyLogic simulation system using its trace log and its performance was found to be acceptable. Simphony simulation system had been rigorously unit tested in earlier years of my PhD program and it was also found to be accurate in simulating systems. Most of the testing done in this thesis was related to COSYE primarily because the development of one of the Application Programing Interfaces (i.e., the Java API) had been concluded in the course of this thesis work. Details of the tests performed are presented in this chapter. This API along with other components within COSYE was found to perform accurately.

Concept models were then translated into simulation models that made up the application created. This involved extensive simulation modeling and computer programming (implemented in CSharp and Java). Verification tests were performed in the course of this implementation to make sure that the concept models were accurately translated. Details of how this was done are also summarized in this chapter.

Prior to implementing any serious developments in a distributed simulation environment, it is good practice to verify that the framework used is reliable and conveys messages in an accurate and timely fashion. This practice was adopted in the development procedures followed in this thesis study. Details of the tests carried out are discussed in the following sub-sections.

7.3.1 Set up of the Java and Dot Net COSYE API Unit Tests

The communication and exchange of data between federates developed in this thesis was crucial for the success of the development work. This is because the performance of the COI was to be tracked based on the information received and compared to that of competitors within the AnyLogic federate in certain respects. Furthermore, most of the results in the application were displayed within the COI federate which relied on information received from the AnyLogic federate.

Unit tests were devised and implemented to test the reliability and validity of the information exchanged between the two APIs. It was expected that the information sent from a federate on one end using one API would be the same information received on the other end of the federation i.e., at the other federate utilizing another API. The test setup used is summarized in Figure 7.1 and Figure 7.2. The Java federate was responsible for creating and destroying the federation execution in both setups.

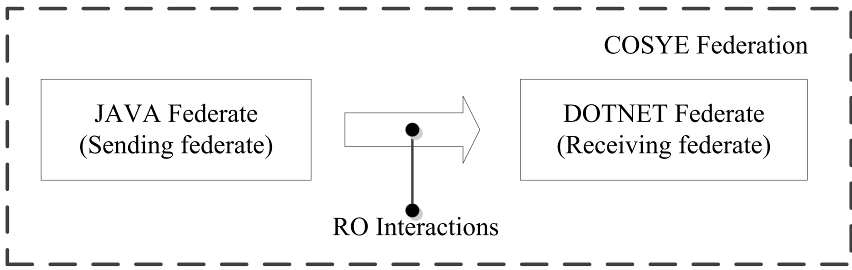


Figure 7.1: Unit Tests Setup I for the COSYE Java and Dot Net APIs – Java Sending and Dot Net Receiving Interactions

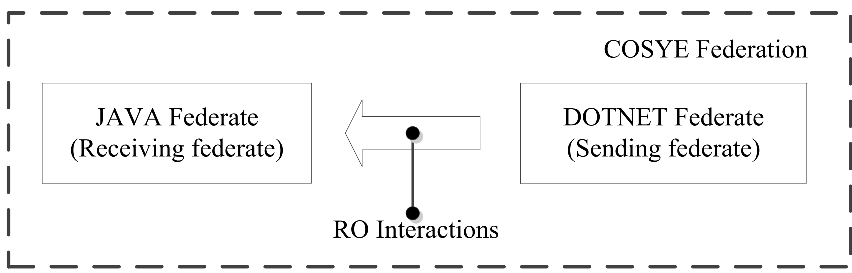


Figure 7.2: Unit Tests Setup II for the COSYE Java and Dot Net APIs – Java Receiving and Dot Net Sending Interactions

One could opt to test the exchange of all simple data types to confirm the accuracy of the APIs. However, a more efficient approach that was adopted in this thesis utilized the most extreme or complex data type that can be exchanged between these APIs. This data type represents the extreme case that would stress the APIs and cause them to fail if they were not implemented accurately. In addition, this data type made use of the primitive data hence served as implicit tests for these as well. Last, this data type was made use of in the developments implemented in this thesis. Confirming their accuracy would serve as a proof that the framework used to develop the application is reliable and so is the application.

The data types used in tests were variable arrays of fixed record type. The fields of the fixed record were mixed to represent the entire spectrum of primitive data types. In this thesis, information was packaged in variable arrays of fixed record type and passed from the AnyLogic federate (made use of the JAVA COSYE API) to the Simphony/COI federate (which made use of the DOTNET COSYE API) and vice versa. This information included:

- Project resources
- Details of the COI resources and waiting files
- Data about the company agents in the AnyLogic federate (sent/received at the end of simulation)
- Information about the resource agents defined in the virtual construction industry within AnyLogic federate (sent/received at the end of simulation)

The tests demonstrated the exchange of only the project resource and COI resource & file details. Also, tests were limited to the use of interaction classes because this was the method used for data exchange within the developed simulation federation.

7.3.2 Unit Test Results for the COSYE APIs

Tests done to verify the behavior of software or a given framework should run without exceptions being thrown. In addition to this, there should be a zero error in the result obtained i.e., the expected and actual result must match. This is not the case in Validation where we would be trying to approximate the real system modeled as close as possible.

In order to mimic the AnyLogic federate behavior, a program was written in Java and the COSYE Java API embedded within it. The Simphony federate on the other hand was emulated using a console application. This console application made use of the Dot Net COSYE API. The Java and Dot Net federates were written in such a way that they had both sending and receiving capabilities. When the tests were run, they gave the precise result that was expected. Screen shots showing details of implementations and output within the programs are summarized in the following Figures.

The traces were generated by each federate when unit test setups summarized in Figures 7.1 and 7.2 were implemented. Details traced by the sending federate in both test setups are presented first. These are followed by trace results displayed by receiving federates in both test setups.

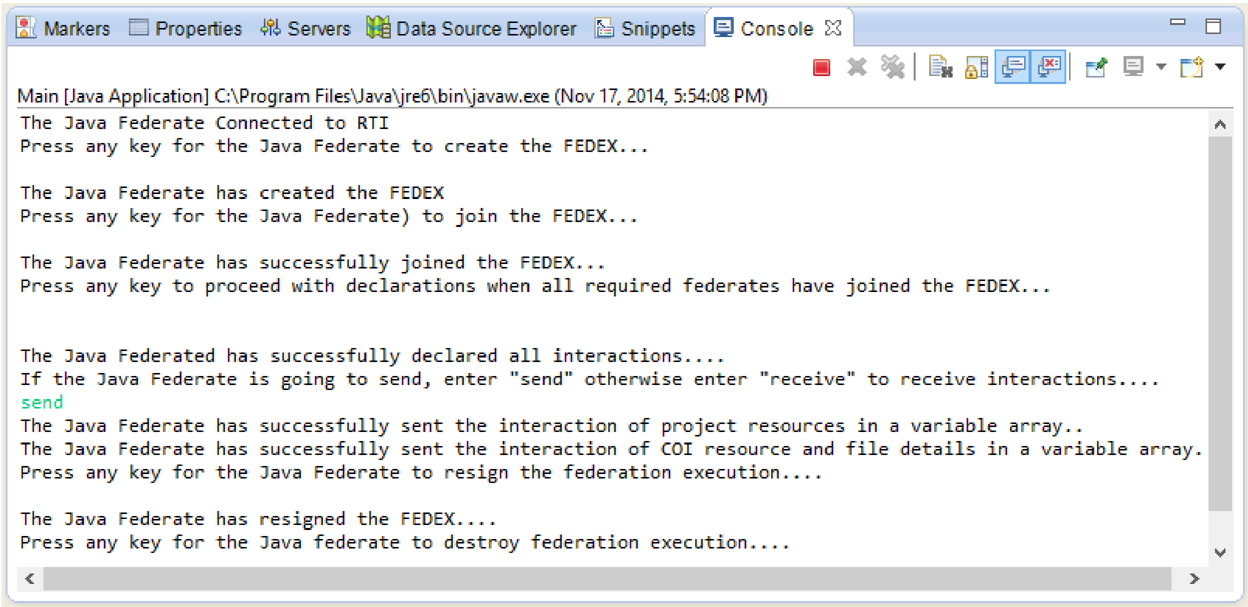


Figure 7.3: Trace Log Generated by Java Federate in the Course of the Unit Test Setup I: (Java – Sending Federate)

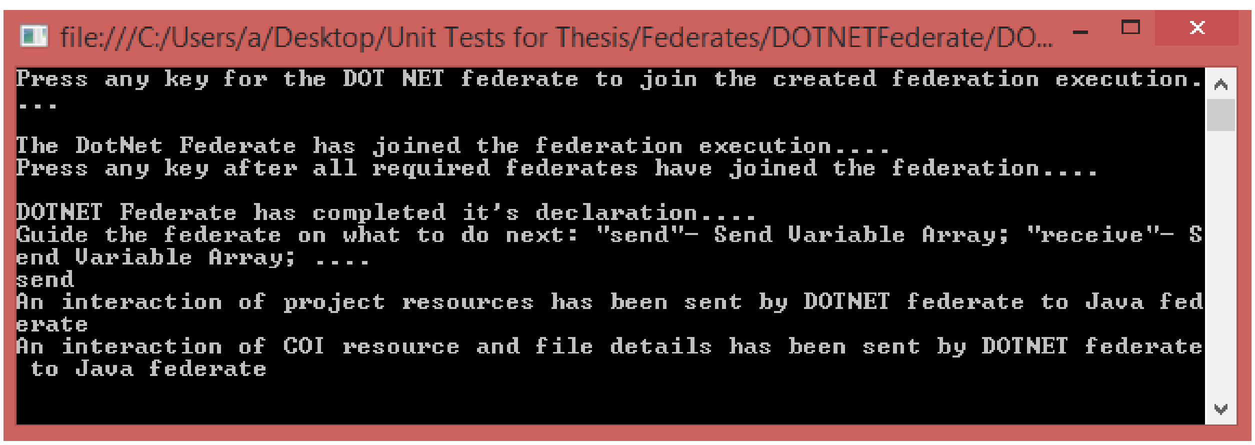


Figure 7.4: Console Application Details for the Dot Net Federate in the Course of the Unit Test Setup II (Dot Net – Sending Federate)

The values sent within the variable arrays were the same in both federates. Receipt of the same values sent in both test setups would serve as confirmation of the reliability of the COSYE framework for implementing distributed simulation federations developed using a mix of Java and Dot Net APIs. Details in each sending federate i.e., Java and the Dot Net are summarized in Figures 7.3 and 7.4 respectively.

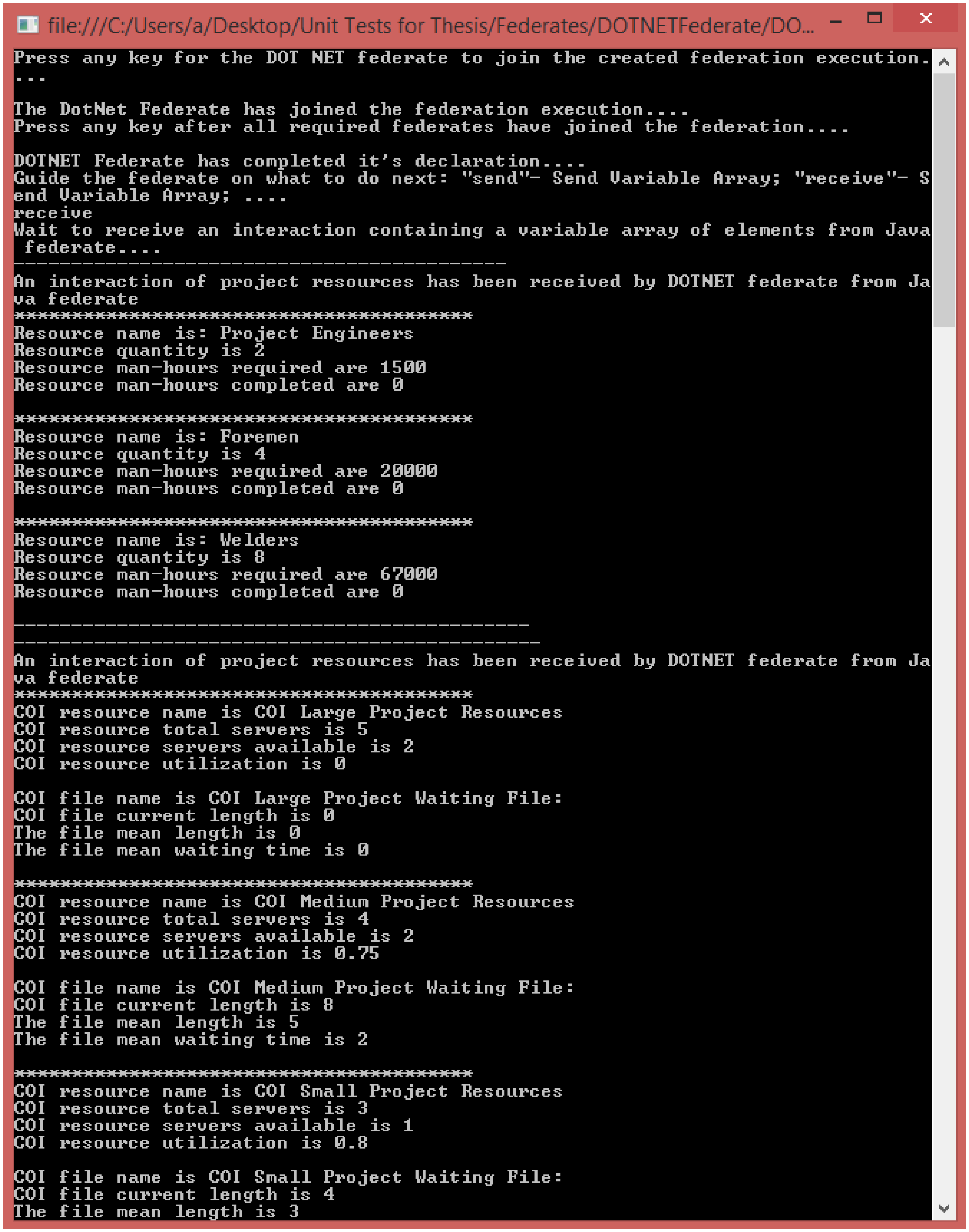


Figure 7.5: Console Application Details for the Dot Net Federate in the Course of the Unit Test Setup I (Dot Net – Receiving Federate)

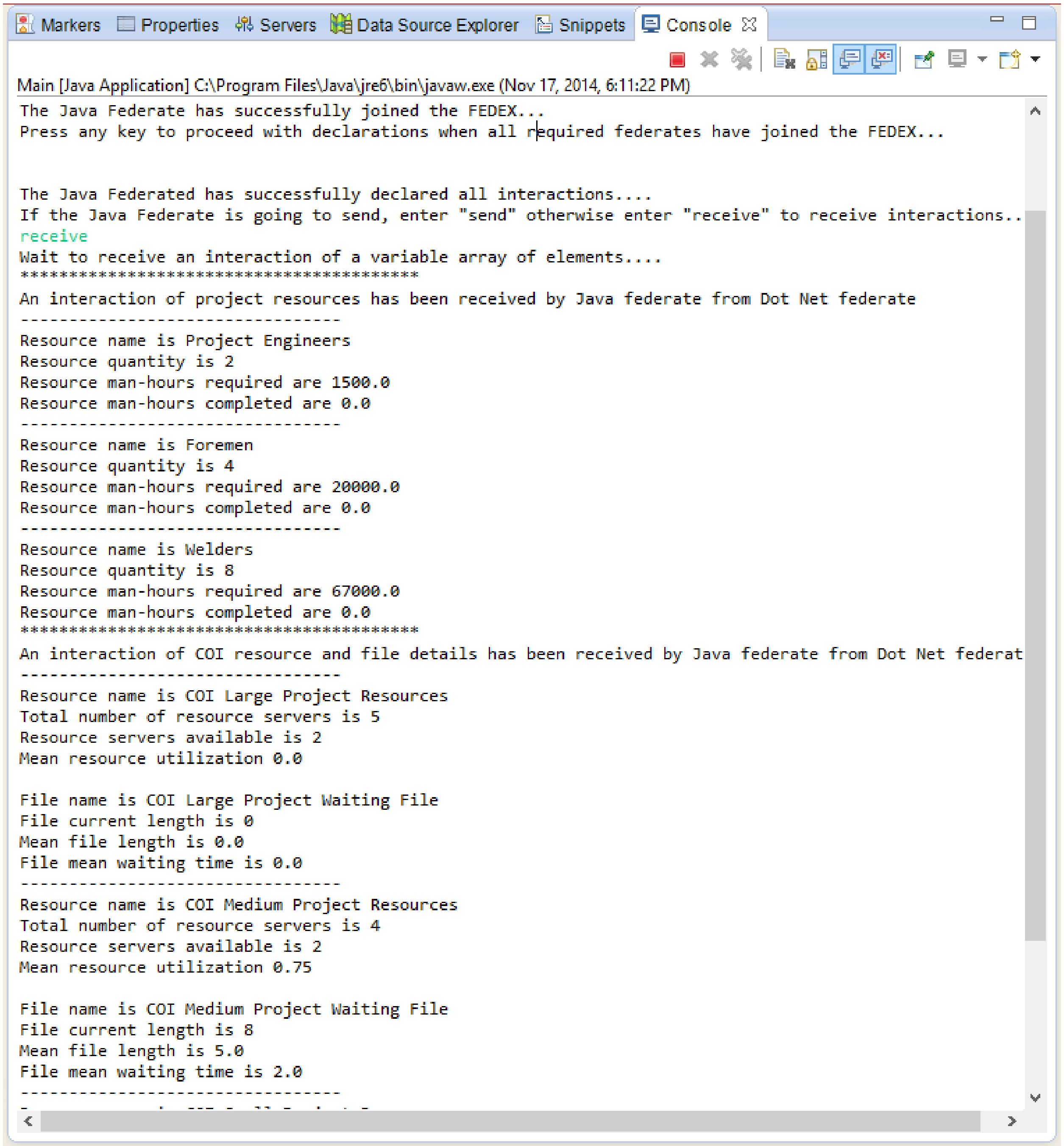


Figure 7.6: Trace Log Generated by Java Federate in the Course of the Unit Test Setup II (Java – Receiving Federate)

Screen shots showing the result from the JAVA and DOTNET federates are presented in Figures 7.6 and 7.7.

The results show that whatever was sent on one end of the federation was precisely received on the other end. This confirmed that the framework and APIs that were used to develop the application are accurate. It also confirmed that the implementer (the author of this thesis) of these tests has decent skill/knowledge set for developing complex simulation components within the framework (i.e., COSYE). This same skillset and process was applied to the translate concept models into the application developed in the actual thesis study. Consequently questions/concerns related to the verification of the application produced in this thesis were partly addressed.

7.4 VERIFICATION OF DISTRIBUTED SIMULATION MODEL BEHAVIOR USING TRACE LOGS

7.4.1 Creation of a New Project

As part of the verification work, it was decided that it was necessary to confirm the implementation of the project arrival and creation process. There were three different types of projects explicitly modelled in the application; small size, medium size and large size projects. These were each modelled independently by an event scheduler modeling node in the AnyLogic Simulation system. These nodes sampled the project inter-arrival times from a statistical distribution defined by the modeller prior to simulation. On creation, project instances were assigned attributes and details of these traced within the console of the AnyLogic system (shown in Figure 7.8).

The first large project was setup to be created at day15. The other inputs defined by the modeller prior to simulation which were to be used for modeling large project details are summarized in Table 7.0.

Table 7.0: PERT and Linguistic Input Variable Definitions for Project Creation in the AnyLogic Federate (ABM)

Parameter	Type of Input	Value
Duration (days)	PERT (L,M,H)	730; 900; 1460
Cost ($)	PERT (L,M,H)	40,000,000; 80,000,000; 250,000,000
Project Complexity	Linguistic Variable	Very high
Project Engineering Quality	Linguistic Variable	Very Good
Project Owner Trait	Linguistic Variable	Very Good
Project Safety Risk	Linguistic Variable	Very high

All input forms (PERT and linguistic variables) were translated into their corresponding Beta distributions at the start of simulation and used in modeling project attributes. Projects resource inputs were defined separately. The likely resource requirement definitions for large projects were presented in the list box control shown in Figure 7.7.

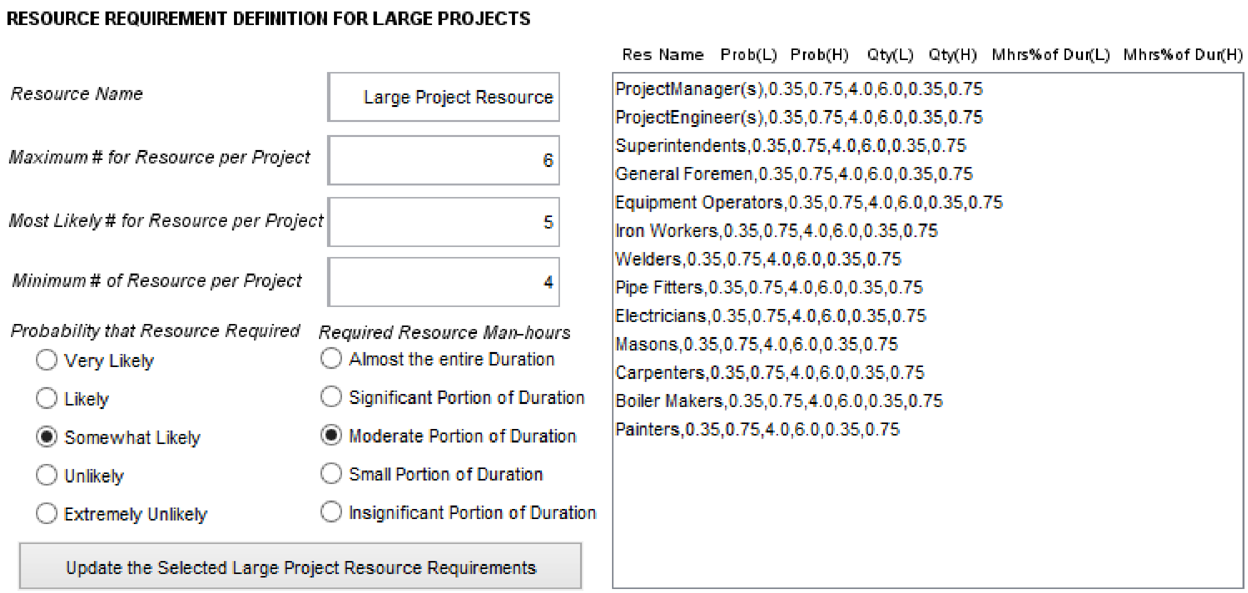


Figure 7.7: Controls used to define the Likely Resource Requirements for Large Projects in the AnyLogic Simulation System

This Figure shows the controls used for editing the likely resource requirements for large projects prior to simulation. The list box details the attributes of the resource requirements that were used in the actual assignment when projects were created. The attributes include a name, chance that a resource will be required by a project, a range for the likely quantity of resource that would be required and a range for the likely man-hours for which the resource would be engaged on the project.

The attributes of the large project created were traced in the console. An extract of these details were retrieved and packaged in a presentable format in Figure 7.8. This was meant to serve as proof of proper implementation of the project creation process.

```
NPNPNPNPNPNPNPNPNPNPNPNPNP - A NEW LARGE PROJECT ARRIVAL - NPNPNPNPNPNPNPNPNPNPNPNPNP

A new large project named largeSizeProject1 [Budget:$1133.6863309618893,
Schedule:7.729703128897287E7 days]has just arrived in the industry at day: 15.0

------------------- -DETAILS OF THE NEW LARGE PROJECT- --------------------------

largeSizeProject1 has a complexity rating of: 0.14192583581509202
largeSizeProject1 has an engineering quality rating of: 0.5783777928146783
largeSizeProject1 has an owner trait rating of: 0.946924051709407
largeSizeProject1 has a safety risk rating of: 0.5999986206677361
The list of resource requirements for largeSizeProject1 are:
largeSizeProject1 requires 4 of ProjectManager(s) for 14493.4906632533 man-hours.
largeSizeProject1 requires 5 of ProjectEngineer(s) for 15892.734684749401 man-hours.
largeSizeProject1 requires 4 of Superintendents for 14134.72981265047 man-hours.
largeSizeProject1 requires 5 of Equipment Operators for 15861.597749515473 man-hours.
largeSizeProject1 requires 4 of Iron Workers for 13117.823569741406 man-hours.
largeSizeProject1 requires 5 of Welders for 14730.35062769975 man-hours.
largeSizeProject1 requires 4 of Masons for 12838.709014753214 man-hours.
largeSizeProject1 requires 5 of Carpenters for 11362.694222992988 man-hours.

------------------- -DETAILS OF THE NEW LARGE PROJECT-| --------------------------

NPNPNPNPNPNPNPNPNPNPNPNPNP - A NEW LARGE PROJECT ARRIVAL - NPNPNPNPNPNPNPNPNPNPNPNPNP


PPPPPPPPPPPPPPPPPPPPPPPPPP -AL SIMULATION ENGINE PAUSED- PPPPPPPPPPPPPPPPPPPPPPPPPPPP

The AnyLogic Simulation Engine has been paused so that the tendering of this project
is fully completed by the Bid Manager Agent.

PPPPPPPPPPPPPPPPPPPPPPPPPP -AL SIMULATION ENGINE PAUSED- PPPPPPPPPPPPPPPPPPPPPPPPPPPP
```

Figure 7.8: A Trace log Generated in AnyLogic Federate whenever a New Project is created

Quick comparisons of the values traced out by AnyLogic during the simulation execution with the inputs defined by the modeller prior to simulation indicate a strong correlation. This confirms that the project arrival and creation process was adequately modelled in the developed application.

7.4.2 Setup for the Bidding Process

After the AnyLogic simulation engine has been paused to facilitate the bidding process to take place, a number of things take place before the formal bidding process. Most of these relate to the Company of Interest Ambassador accessing information (file and resource details) about the operations in the Simphony federate. A detailed sequence of events is traced out and summarized in Figure 7.9.

```
A switch has been set by BidManager Agent to TRUE that permits the CI Agent
to send a request for resource and file details to Simphony on behalf of
COI Ambassador.
Bid Manager Agent has set the Time Granting State of the AnyLogic Federate
is set to: false
The BidManager Agent is going to send sending a Time Advance Request to:
15.000000000001156
COIFederateAmbassador Class has set the Time Granting State of the AnyLogic
 Federate to: true
The AnyLogic Federate has been granted a TAG to:15.000000000001156
AnyLogic system is going to send a request to Simphony for resource and
file details within a TAG..
CI Agent in AnyLogic is going to send an interaction requesting for resource
& file information, with a time stamp: 15.000000000002313
CI Agent in AnyLogic has sent an interaction requesting for resource & file
information, with a time stamp: 15.000000000002313
The value of the federate time granting state is:true
CI Agent is making a small time advance request to 15.000000000002313 ....
CI Agent has set the Time Granting State of the AnyLogic Federate to: false
CI Agent has made a small time advance request to 15.000000000002313 ....
COIFederateAmbassador Class has set the Time Granting State of the AnyLogic
Federate to: true
The AnyLogic Federate has been granted a TAG to:15.000000000002313
A COI Resource and File details Interaction received from Simphony Federate
Federate Ambassador has requested COIAmb Agent to instruct BidManager Agent
to start the bidding process..
```

Figure 7.9: Trace log showing the Communication between AnyLogic and Simphony Federate to obtain Resource and File Details Prior to Bid Process Commencement

The communication involves a set of time advance requests to ensure that time stamped messages (request for file and resource information) sent by the AnyLogic federate get delivered to the Simphony federate. This information keeps changes as simulation progresses hence making it necessary to be accessed every time a new project is created. For every time advance request or time stamped message to be sent, a look ahead was added for consistency because the COSYE RTI does not support zero look ahead. This explains the several significant figures traced in the log for time.

7.4.3 Tracking the Bidding Process

Every time that a project got created in the model, it would be subjected to a bidding cycle in which all companies interested in that project go through and initial and final bid decision making phases. Successful companies submit their most competitive bid and the project get awarded to the least bidder. This particular scenario is not as interesting as typical ones because the bidding process is comprised of only one company (the company of interest) and no competitors. However, it confirms the occurrence of the envisaged events and their chronological order.

```
COI Amb. has received instruction to request bid manager to start the
bidding process for largeSizeProject1
-------------------------------------------------------------------------
companyOfInterestAmbassador[0] submitted a: true initial bid decision
to buy tender documents for largeSizeProject1
-------------------------------------------------------------------------
companyOfInterestAmbassador[0] submitted a : true final bid decision for
largeSizeProject1
-------------------------------------------------------------------------
Bid Manager has requested the COI Amb. to generate and submit a bid price..
companyOfInterestAmbassador[0] has submitted a bid of $7.729703128897287E7
for largeSizeProject1
--------------------------------------------------------------------------
companyOfInterestAmbassador[0] has been informed that largeSizeProject1 has
been awarded to companyOfInterestAmbassador[0]
```

Figure 7.10: A Trace log from AnyLogic Federate Showing a Bidding Cycle for one Project

The trace log in Figure 7.10 shows the engagement of the company of interest ambassador agent as a sole bidder for a large project and its subsequent acquisition of the project.

7.4.4 Resource Request and AnyLogic Simulation Engine Resumption

Following the award of a project, the company awarded a project was setup to request for that project's required resources. This request was to be submitted to the Construction Industry Agent and queued there until all the required resource agents became available. In the scenario that was run and gave the results presented in the trace in Figure 7.11, there was an abundance of resource agents defined in the virtual construction industry hence there would be not delay between placement of request and allocation of resources. The detail of the awarded project that is allocated resources is communicated to the Simphony Federate via the COSYE RTI.

```
<<<<--- COI Amb. Agent is going to request the resource agent(s) for resources
required by largeSizeProject1 on day 15.0
--->>> COI Amb has been granted project resources for largeSizeProject1 on day 15.0
companyOfInterestAmbassador[0] is requesting CI Agent to send an interaction of
this project to Simphony federate..
CI Agent in AnyLogic has sent an interaction on the award of a newly arrived
project, with a time stamp: 15.000000000004624
The AnyLogic Federate has been granted a TAG to:15.000000000003467
UPUPUPUPUPUPUPUP -AL SIMULATION ENGINE UNPAUSED- UPUPUPUPUPUPUPUPUPUPUPUPUPUPUP
The AnyLogic Simulation Engine has been unpaused within the AL TAG callback method.
```

Figure 7.11: Trace Log showing Events that Follow Project Award

Regardless of whether the required resource is granted or not, the AnyLogic simulation engine is resummed so that the federation excution can proceed. This was triggered by making a time advance request to the COSYE RTI and receiving a corresponding time advance grant. The tracelog shows that this sequence was respected in the resumption of the AnyLogic engine execution. All events traced and their chronologic order of occurrence are consistent with the concept designs discusssed in Chapter four of this thesis. This verifies the implementations carriedout in this thesis.

7.4.5 AnyLogic Federate exit from the Federation

The AnyLogic simulation model (federate) terminates execution when it gets to the maximum time set by the modeller prior to simulation execution. It is at this point that this federate sends information about the company agents (competitors) and the resource agents to the Simphony federate for display as outputs to the modeller. Thereafter, the AnyLogic federate announces the achievement of the *"READYTOTERMINATE"* synchronization point and then resigns the federation execution gracefully.

The sequence of these events was filtered from the trace log that was generated from AnyLogic federate and summarized in Figure 7.12.

```
CI Agent (from Simulation Experiment @ end of run) is going to send a
Time Advance Request to: 30.000000000001158
CI Agent (from Simulation Experiment @ end of run) set the Time
Granting State to: false
COIFederateAmbassador Class has set the Time Granting State of the AnyLogic
Federate is set to: true
The AnyLogic Federate has been granted a TAG to:30.000000000001158
CI Agent in AnyLogic is going to send an interaction of resource agents details
with a time stamp: 30.000000000002316
CI Agent in AnyLogic has sent an interaction of resource agents details, with
a time stamp: 30.000000000002316
The value of the federate time granting state is:true
CI Agent is making a small time advance request to 30.000000000002316 ....
CI Agent has set the Time Granting State of the AnyLogic Federate to: false
CI Agent has made a small time advance request to 30.000000000002316 ....
COIFederateAmbassador Class has set the Time Granting State of the AnyLogic
Federate is set to: true
The AnyLogic Federate has been granted a TAG to:30.000000000002316
The button has been clicked that announces that the READYTOTERMINATE
Synchronization point has been achieved..
The AnyLogic Federate has successfully resigned the federation execution.
```

Figure 7.12: AnyLogic Federate Trace Log at Federation Shut Down

This trace log was for a scenario in which the COI was the only company operating in the virtual construction industry. Also, the application was executed for only a 30-day period.

7.5 EXPERIMENTING WITH THE ANYLOGIC FEDERATE

This section marked the first of two sections that detailed the implementation of the piecewise approach to experimenting with the developed application. The front end component of the distributed simulation federation i.e., the AnyLogic federate is discussed here. This federate was designed and implemented in such a way that it could be execute independent of the rest of the distributed simulation federation by simply turning of all the HLA (i.e., distributed simulation federation) federation management and communication switches.

The main purpose of experimenting with the AnyLogic federate alone was to show case the input and outputs associated with the federate in a simplistic fashion. Details presented in this section are based on a conference paper presented at the summer simulation conference in the September 2014 (Ekyalimpa and AbouRizk, 2014).

Specific objectives for this experiment included: 1) to verifying the creation of projects in the course of the simulated period; 2) review the distribution of created projects by type i.e., relative number of small, medium size and large projects; 3) confirm that the quality (attributes) of projects that were created were consistent with inputs defined prior to simulation (with respect to owner trait, complexity, engineering quality and project safety risks); 4) establish the distribution/allocation of created projects to companies operating within the virtual construction industry; and 5) track the reason(s) for the distribution/allocation of projects to these companies in the industry.

In order to set up and run the experiment, hypothetical values were chosen for the different model parameters and set prior to simulation. First, values used to model the rate of project arrivals were specified. Then distributions used to generate the attributes for each created project were summarized. Finally, the tolerances that guide Companies' behavior in making decisions on which projects to bid or not were defined. The values for these parameters were specified as statistical distributions to ensure that there were variation in project instances and decisions made by companies, which was in-line with the phenomenon abstracted from a real-life setting.

7.5.1 Federate (ABM) Setup and Details of the Virtual Industry

The AnyLogic federate was comprised of an agent-based simulation model developed within the AnyLogic simulation system. This federate of the distributed simulation federation was be experimented with independently by turning off all HLA communication switches. This meant that the federate was made to run without need for the distributed simulation federation. The unit of measure for time used in the ABM was *days*. The model was set up to run for *1,000 days* after which it terminated.

The construction industry was represented by a unique agent referred to as a *virtual construction industry agent*. All other agents within the model were configured to thrive within this agent. Details of the construction industry such as the number of companies operating within the industry were defined at this agent. Small, medium size and large company agents were embedded within the construction industry agent. The bid manager agent was also defined. The company of interest ambassador agent was not represented in this model setup.

Inputs used to model the project creation process were defined within the bid manager agent. These inputs included details of distributions used to draw project attribute values and their inter-arrival times. Uniform statistical distributions were used in each of these cases but any other statistical distribution could have been utilized. Table 7.1 summarizes the ranges for these distributions.

Table 7.1: Model Inputs – Attributes for New Projects and their Inter-arrival Times

Parameter	Unit of Measure	Small Projects	Medium Projects	Large Projects
Inter-arrivals	Days	14~100	90~180	180~540
Cost	Million $	10~100	100~300	250~1000
Duration	Days	300~540	450~750	600~1200
Owner Trait	Scale (0-1)	0.00~0.50	0.40~0.80	0.75~1.00
Complexity	Scale (0-1)	0.00~0.40	0.25~0.75	0.65~1.00
Engineering Quality	Scale (0-1)	0.80~1.00	0.25~0.85	0.00~0.30

Parameter	Unit of Measure	Small Projects	Medium Projects	Large Projects
Safety Risks	Scale (0-1)	0.00~0.70	0.35~0.85	0.70~1.00

In the scenario tested, a total of 20 company agents were assumed to operate within the virtual construction industry. Of these 50% were large, 30% medium size and 20% were small companies. This translated into a total of 10 large company agents, three medium size company agents and two small company agents. Each of these thrived within their respective company agent populations. The attributes for these company agents are discussed in the following sub-section.

7.5.2 Input Definition—Project Type Preference for Company Agents

Each company agent had unique attributes. These attributes influenced the bidding behaviors of the company agents. The attributes were also referred to as the company's tolerances to specific types of projects. Examples of these tolerances included: owner trait, complexity, engineering quality, and safety risks. Table 7.2 summarized the range of values used for defining inputs for these attributes in the scenario simulated. Next, details of what each of these attributes were meant to emulate and how they influenced the company agent's behavior are discussed.

Table 7.2: Model Inputs – Company Tolerances for Projects

Parameter	Unit of Measure	Company Agent Population		
		Small	Medium	Large
Owner Trait	Scale (0-1)	0.00~1.00	0.40~1.00	0.80~1.00
Complexity	Scale (0-1)	0.00~0.40	0.00~0.85	0.00~1.00
Engineering Quality	Scale (0-1)	0.15~1.00	0.30~1.00	0.60~1.00
Safety Risks	Scale (0-1)	0.00~0.50	0.00~0.75	0.00~0.90

Owner trait was meant to indicate the degree of unnecessary interruptions to the contractor's work rhythm caused by the owner during project execution. This property was thought to affect the contractor's morale and in turn their productivity. Informal discussions with experienced practitioners in the construction industry revealed that this attribute plays a significant role when a contractor is deciding whether or not to bid on a project.

The extent to which a project was engineered prior to construction and the quality of the engineering work done on the project were abstracted and represented in such a way that they influence the production efficiencies during project execution. This attribute also influences the likelihood of quality incidents occurring while the project was being performed. The engineering quality for a project under tender was setup to influence a company agent's decision on whether or not to bid on the project.

The complexity of a project along with the potential safety risks associated with executing a project were set up to influence company agent's decision on whether or not to bid on a project. For the complexity attribute, a value of 0.0 indicated a straight forward project while a value of 1.0 indicated an extremely complex project. A value of 0.0 for project safety risk indicated low likelihood of safety incidents while a value of 1.0 indicates a very high likelihood of safety incidents occurring.

High values for engineering quality and owner trait for any given project are good, while low values are bad. On the other hand, high values for complexity and safety risk are bad, while low values are good.

The semantics discussed above were applicable to the demand side i.e., for projects. On the supply side, issues to do with company tolerances with respect to each of these attributes are dealt with. Rating scales identical to those used for the projects are used for each attribute (from 0 to 1). A low value (0.0) indicated that the company had an extremely low tolerance for the attribute, while a value of 1.0 indicated a very high tolerance for the adverse side of the attribute.

Both the demand and supply side values for these parameters were setup to influence a company's decision on bidding on a project-by-project basis. Under normal circumstances, a company would decide to proceed to bid on a project based on the engineering quality and owner trait criteria, when the ratings of these attributes for the project were higher than the tolerances set by the company agent. On the other hand, the company would proceed to bid when the ratings for the project, with respect to complexity and safety risks, are lower than the tolerance levels for the company.

At the start of the simulation, Company Agents were created by the Main Agent. Each Company Agent was assigned tolerance values through a process that involves randomly sampling from a statistical distribution. A uniform distribution was constructed from the ranges provided in Table 7.3 and used to sample the values for ratings assigned to the company. This meant that companies would end up having different tolerance levels for each of the criteria, something that is evident in practice.

7.5.3 Simulation Results from the ABM in AnyLogic Federate

Results generated by the simulation model were displayed in the form of charts to ease their interpretation. The results indicated a number of things: the number of projects generated the quality of these projects and who these projects were awarded to, as well as the reasons why they were awarded in that fashion. Details of each of these, for the simulated scenario, are discussed in the following sub-sections.

7.5.3.1 Details of Created Projects

A total of 11 projects were created within the virtual construction industry in the course of the period simulated. Figure 7.13 summarizes a trendline that illustrates the project creation process as the simulation progressed.

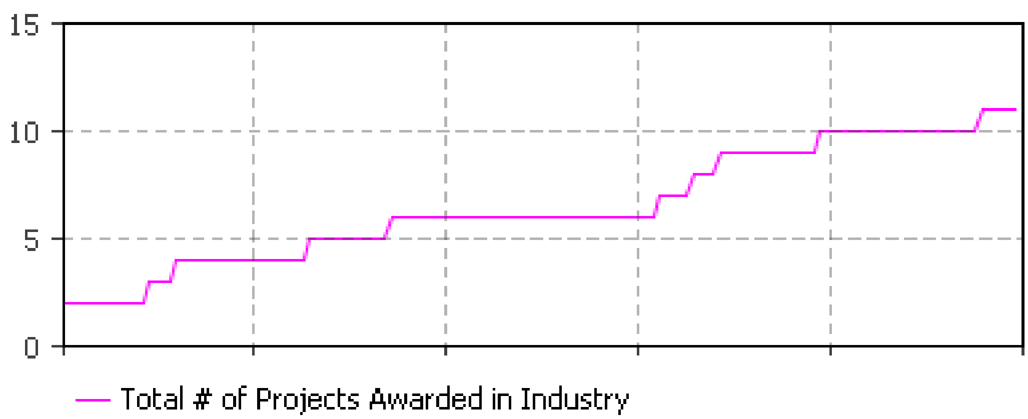


Figure 7.13: Total Number of New Projects Created in the Course of the Simulation

Besides details of the number of projects created in the virtual construction industry in the course of the simulation, the attributes of these projects were tracked. From these details, an analyst could infere whether the type of projects created where generally bad, good or moderate. The type of projects created influenced the company agent's decision on whether to or not to bid the projects. In the case of the company of interest, these project attributes were carried along so that they influenced the performance of the company in the project execution phase. The company of interest was not considered in this scenario setup in order to keep things simple.

Data tracked in the course of simulation on project attributes were plotted on bar charts. These were categorized to reflect mean attributes values for small, medium size and large projects.

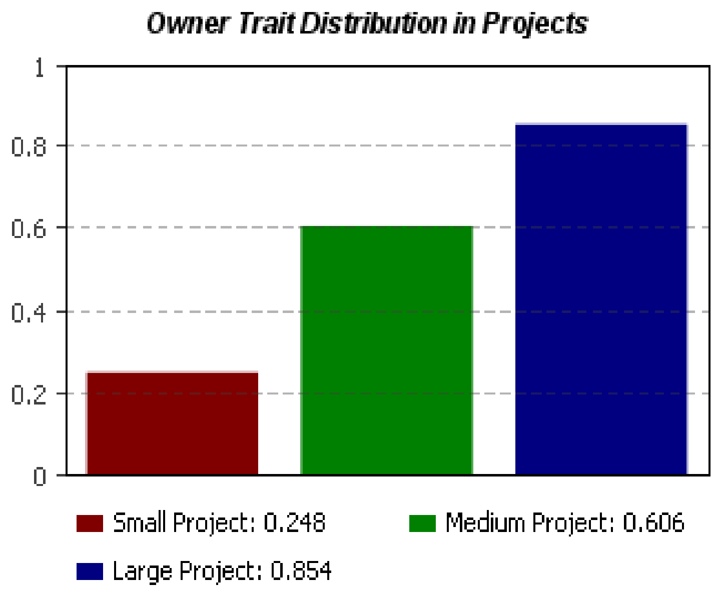


Figure 7.14: Owner Trait Attributes of Generated Projects

Figure 7.14 summarizes the mean owner trait attributes for the projects. Owner trait attribute represented possible owner initiated interruptions, change orders, possible delayed payments e.t.c in the course of performing the project. For the projects generated, the large size projects had the highest average owner trait while small projects had the lowest mean value. This meant that large projects had a better quality with respect to this attribute.

Complexity of a project was another attribute tracked. This represented the likelihood of challenges the company would face in executing the project as a result of its unique features and scale. Average values for projects created are summarized in Figure 7.15.

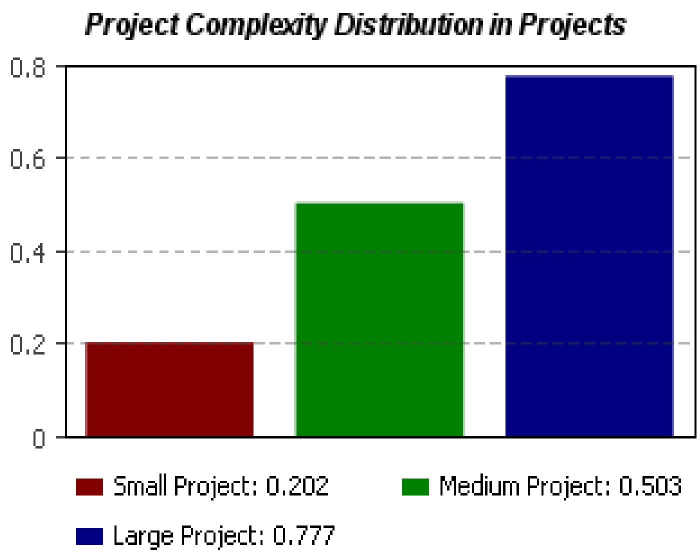


Figure 7.15: Complexity Attributes of Generated Projects

On average, large projects were the most complex while small ones were the least complex.

The other two attributes tracked included project engineering quality and project safety risk. Details of these are summarized in Figures 7.16 and 7.17 respectively. The graphs indicate that on average the large projects had the poorest engineering quality and the highest project safety risks compared to medium and small projects. These results are consistent with the input data summarized in Table 7.3 and used in the model.

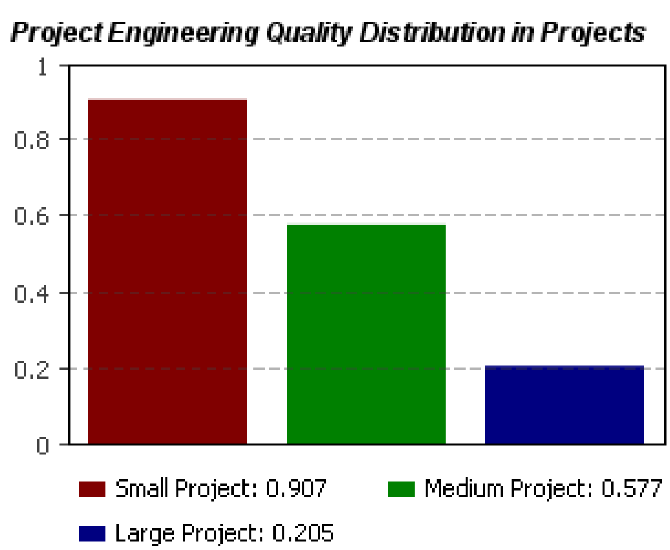


Figure 7.16: Project Engineering Quality Attributes of Generated Projects

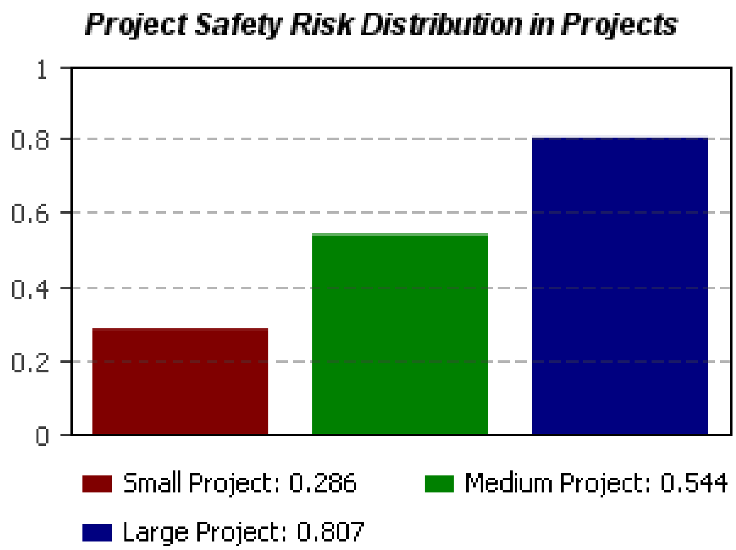


Figure 7.17: Project Safety Risk Attributes of Generated Projects

The trends indicate that small projects have extremely good attributes with the exception of the owner trait. Medium size projects on the other hand possess moderate attributes across the board. All attributes of large projects are bad with the exception of the owner trait. This implied that a company agent that was inclined to acquire more large projects than small ones would be more susceptible to failure. Values indicated were averages obtained for the projects generated during the 1,000 day simulated period.

7.5.3.2 Projects Awarded to Company Agents

One of the objectives of experimenting with the ABM in this fashion was to demonstrate that for a specific set of inputs, projects created within the virtual construction industry would be awarded to the different company agents based on their configurations (i.e., how tolerant they were to acquiring the different types of projects and their competitiveness in bidding).

In order to avoid clutter that could arise if details of awarded projects were discussed at an individual company agent level, results presented in this section relate to company agent populations i.e., small companies, medium size companies and large companies.

For the scenario simulated large companies were awarded a total of 3 projects, medium size companies, 6 projects, and small companies, 2 projects. These details are summarized in the bar chart presented in Figure 7.18.

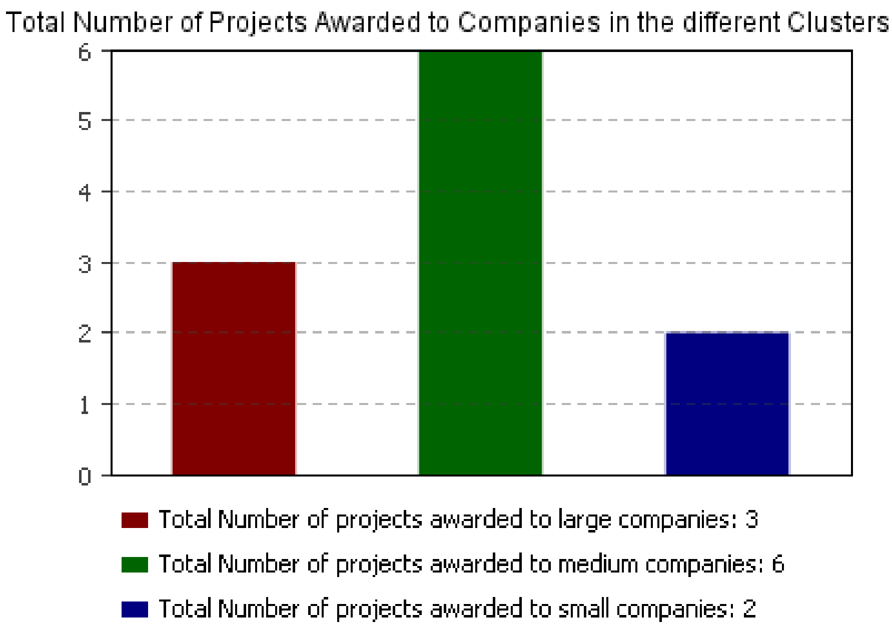


Figure 7.18: Details of Projects Awarded to the Different Company Agent Populations

Results displayed in Figure 7.18 indicate that the bulk of the projects were awarded to medium size companies. Small companies were awarded the least number of projects.

The model also tracked details of the value of the projects awarded to the different individual company agents and the agent populations that they thrived in. Figure 7.19 summarises the results obtained from the simulated scenario.

It is evident from both result sets (number of projects awarded and the value of these projects) that despite the intermediate number of projects awarded to the large company agent population, it had the largest share with respect to the aggregated value of projects i.e., $1.492 billion. $754 million and $124 million worth of projects were awarded to the medium and small companies, respectively.

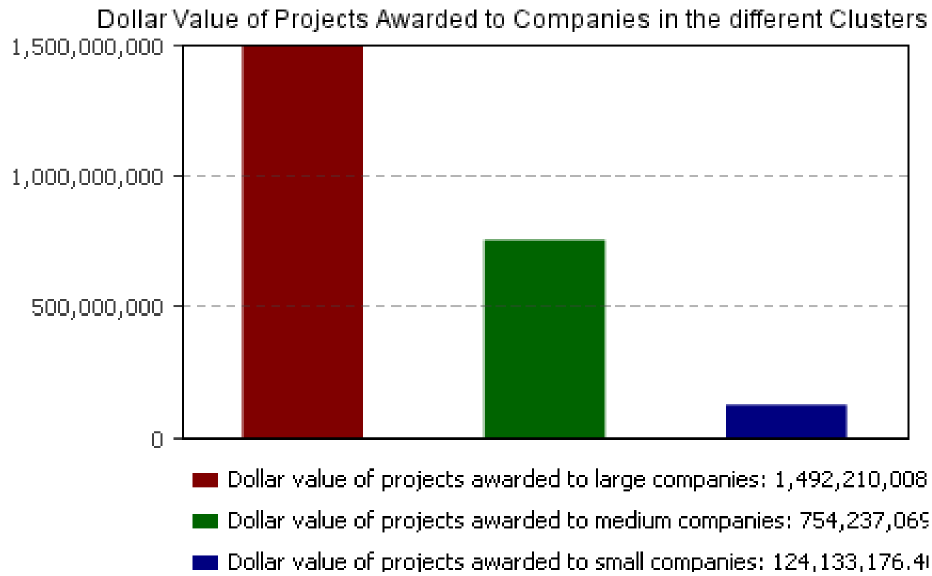


Figure 7.19: Total Value of Projects Awarded to the Different Company Agent Populations

7.5.3.3 Tracking Tendering Performance of Company Agents

The agent-based model was setup to track behaviors of each company agent along with the rationale behind selected behaviors. The bid/no bid behavior for company agents was tracked at the agent and agent population levels. This implies that at the end of simulation, the modeller could drill out to a specific company agent and establish the reasons why it did not bid for projects that were created in the virtual industry in situations that the agent did not bid some projects. This scrutiny of agent bidding behavior could also be rolled up to the agent population.

An example is presented in Figure 7.20 in which the reasons as to why the small company agent population did not bid projects were summarized as a pie chart. A similar chart was created for medium size and large company agent populations.

These reason tracked for agent bid/no bid behavior included the availability of sufficient production capacity at the company, competitor number exceeded company's internal threshold value, bid was not competitive, owner trait very bad, projects too complex, project engineering quality poor, and project safety risks too high.

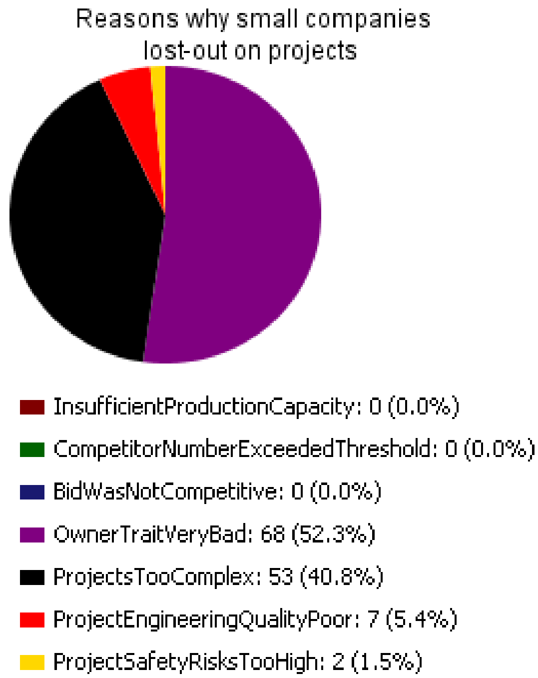


Figure 7.20: A Summary of the Rationale for Small Company Agent Population Bid/No Bid Behaviors

For this specific case, all companies opted not to bid on specific projects mainly because the trait of the project owner was bad. The project owner trait in these cases was worse than the tolerances that were predefined by the company. There were no cases in which the companies could not bid on a project due to insufficient production capacity.

7.5.3 Summary of Experimentation with AnyLogic Federate

This section has discussed details an experiment set up that was run using the AnyLogic model. The results obtained were realistic and emulated constructs and phenomena that exist in the real world of construction. The model can be put to various uses. From an owner's perspective, the model can serve as a tool to guide when to invest in projects based on the available company resources in the industry. From a construction contractor's perspective, the model can be used to gain insights into the amount and type of work to go after. These insights can then be used to devise appropriate strategies that ensure that a reasonable work balance is achieved which guarantees good company performance. In its current state, the federate would serve the purpose for which it was developed within the performance management distributed simulation system.

7.6 EXPERIMENTING WITH THE SIMPHONY FEDERATE – NO COMPETITORS IN THE VIRTUAL CONSTRUCTION INDUSTRY

Unlike the AnyLogic federate, the Simphony federate could not be run independent of the distributed simulation federation. This was because it needed to receive projects from the virtual construction industry which was represented within the AnyLogic federate. Running the entire federation while testing the Simphony federate also enables the communication protocols between both federates to be tested.

For purposes of testing the Simphony federate, a unique fictiteous scenario was setup. It involved creating a virtual construction industry in which the Company of interest operated as the only construction company. All other agents were present except competitors to the company of interest.

The objective here was to test the performance of the communication between AnyLogic federate and the Simphony federate without complicating the simulation model through the addition of many uncertainies e.g., competitor company agent behaviors. The other objective was to test the execution of the Simphony federate to verify that it actually generates reasonable results at the end of simulation execution. This test was setup to confirm the proper modeling of the different performance phenomenon.

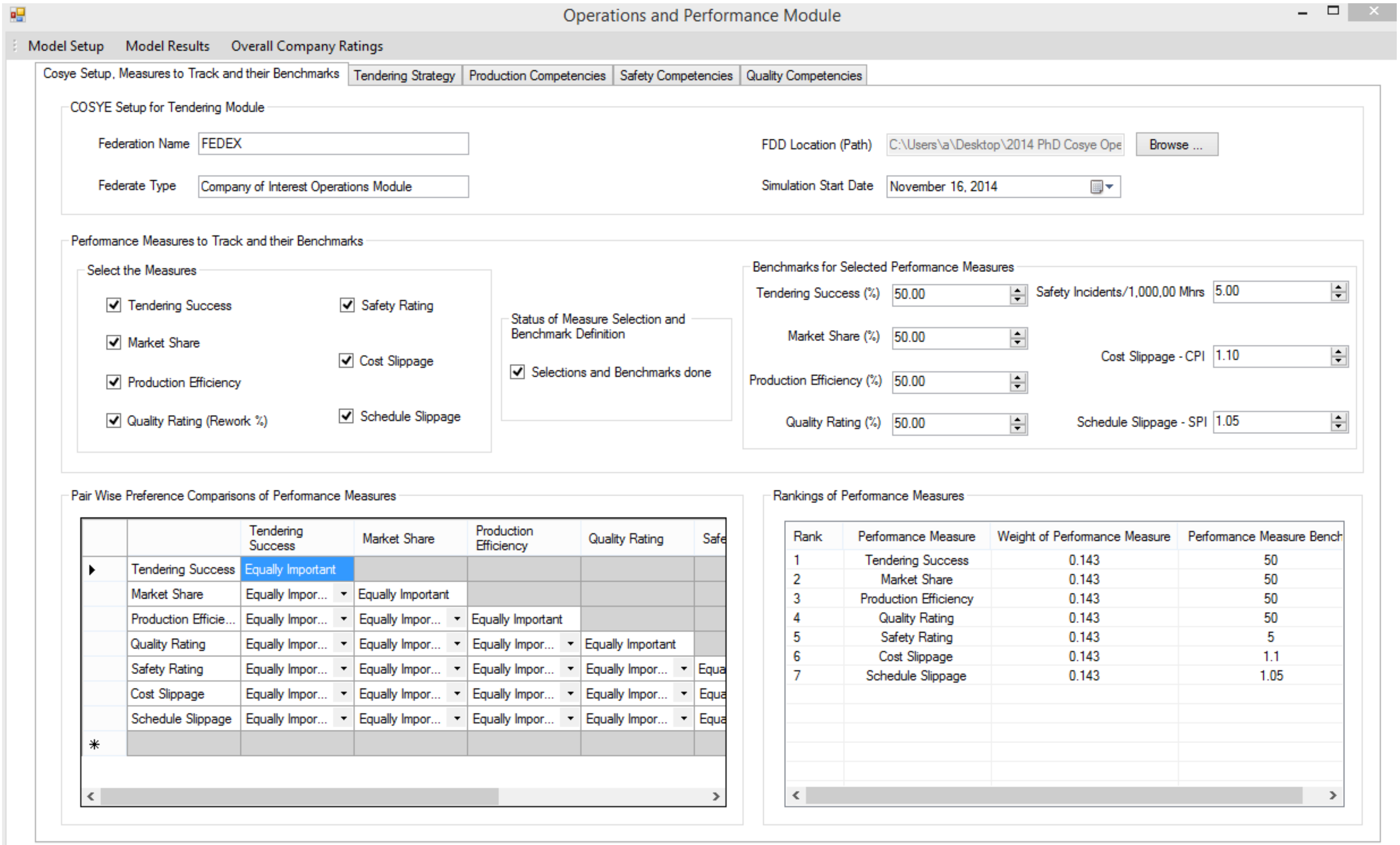
Results from this simple scenario would confirm that the model is reliable and is worthy of use in more complex test cases prior to final deployment.

7.6.1 Scenario Setup

This scenario was meant to simulate 30 calendar days. The Bid Manager Agent within the AnyLogic federate was configured to create its first small project at day 5, medium project at day 10 and large project at day 15. The rest of the project arrival times are based on inter-arrival times sampled from three different statistical distributions.

It was envisaged that if the application behaves as adesigned the Company of interest would be awarded all the three projects since there is no competition. Default inputs were for the Simphony federate were used. This are summarized in the following Screen shots (Figures 7.21, 7.22, 7.23, 7.24, and 7.25).

The model was setup to track all the performance measures provided for. All influencing factors for these performance measures were assumed to have an equal influence on the performance measure. This decision was taken to simplify the test case scenario. Another assumption was made when setting up the inputs for the company of interest. No work in progress was existent at the company at the start of simulation exceution.



	Tendering Success	Market Share	Production Efficiency	Quality Rating	Safe
Tendering Success	Equally Important				
Market Share	Equally Impor...	Equally Important			
Production Efficie...	Equally Impor...	Equally Impor...	Equally Important		
Quality Rating	Equally Impor...	Equally Impor...	Equally Impor...	Equally Important	
Safety Rating	Equally Impor...	Equally Impor...	Equally Impor...	Equally Impor...	Equa
Cost Slippage	Equally Impor...	Equally Impor...	Equally Impor...	Equally Impor...	Equa
Schedule Slippage	Equally Impor...	Equally Impor...	Equally Impor...	Equally Impor...	Equa

Rank	Performance Measure	Weight of Performance Measure	Performance Measure Bench
1	Tendering Success	0.143	50
2	Market Share	0.143	50
3	Production Efficiency	0.143	50
4	Quality Rating	0.143	50
5	Safety Rating	0.143	5
6	Cost Slippage	0.143	1.1
7	Schedule Slippage	0.143	1.05

Figure 7.21: COSYE Setup Details and Inputs for the Company of Interest – Performance Measures to Track and Their Benchmarks

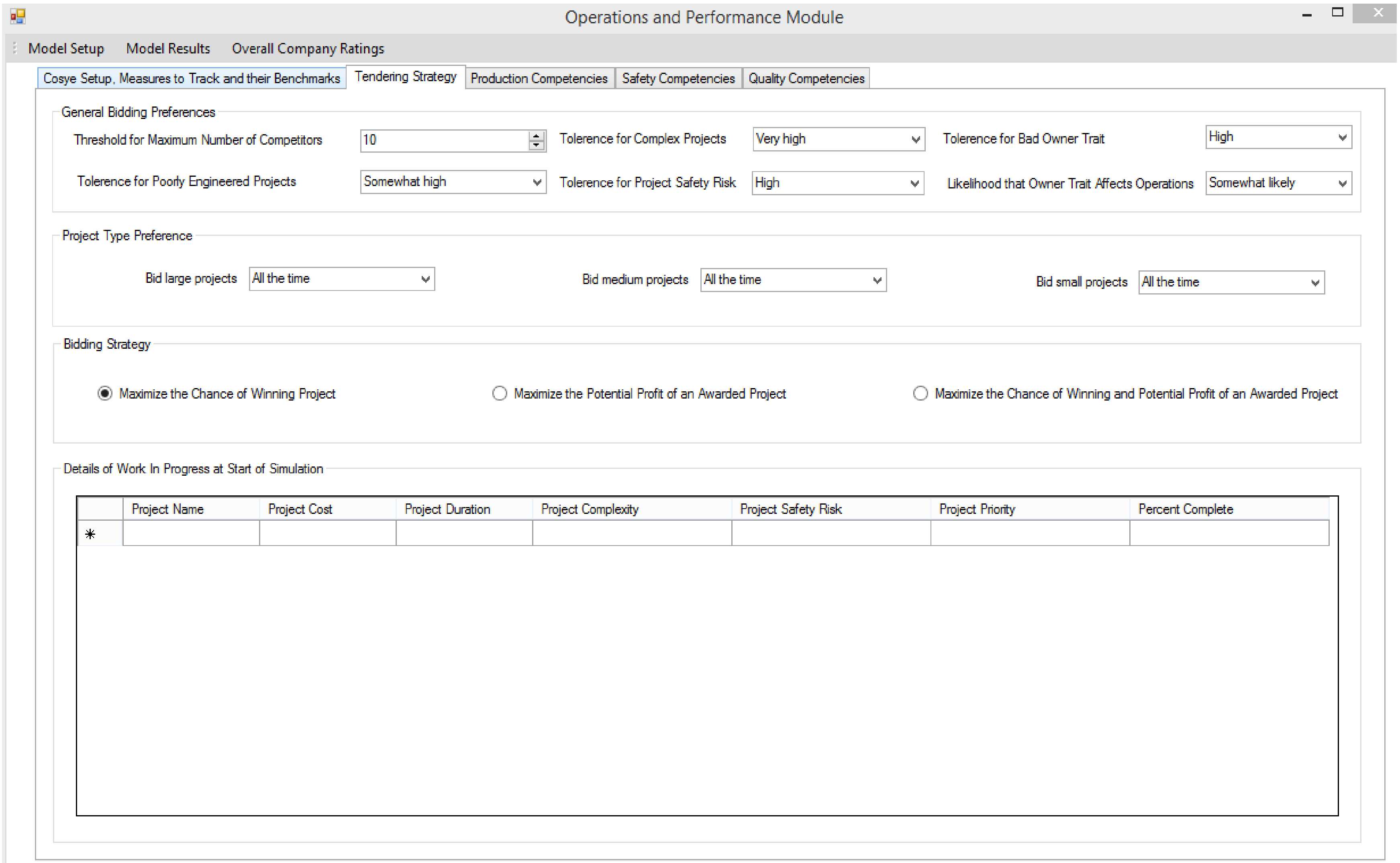


Figure 7.22: Inputs for the Company of Interest – Tendering Strategies

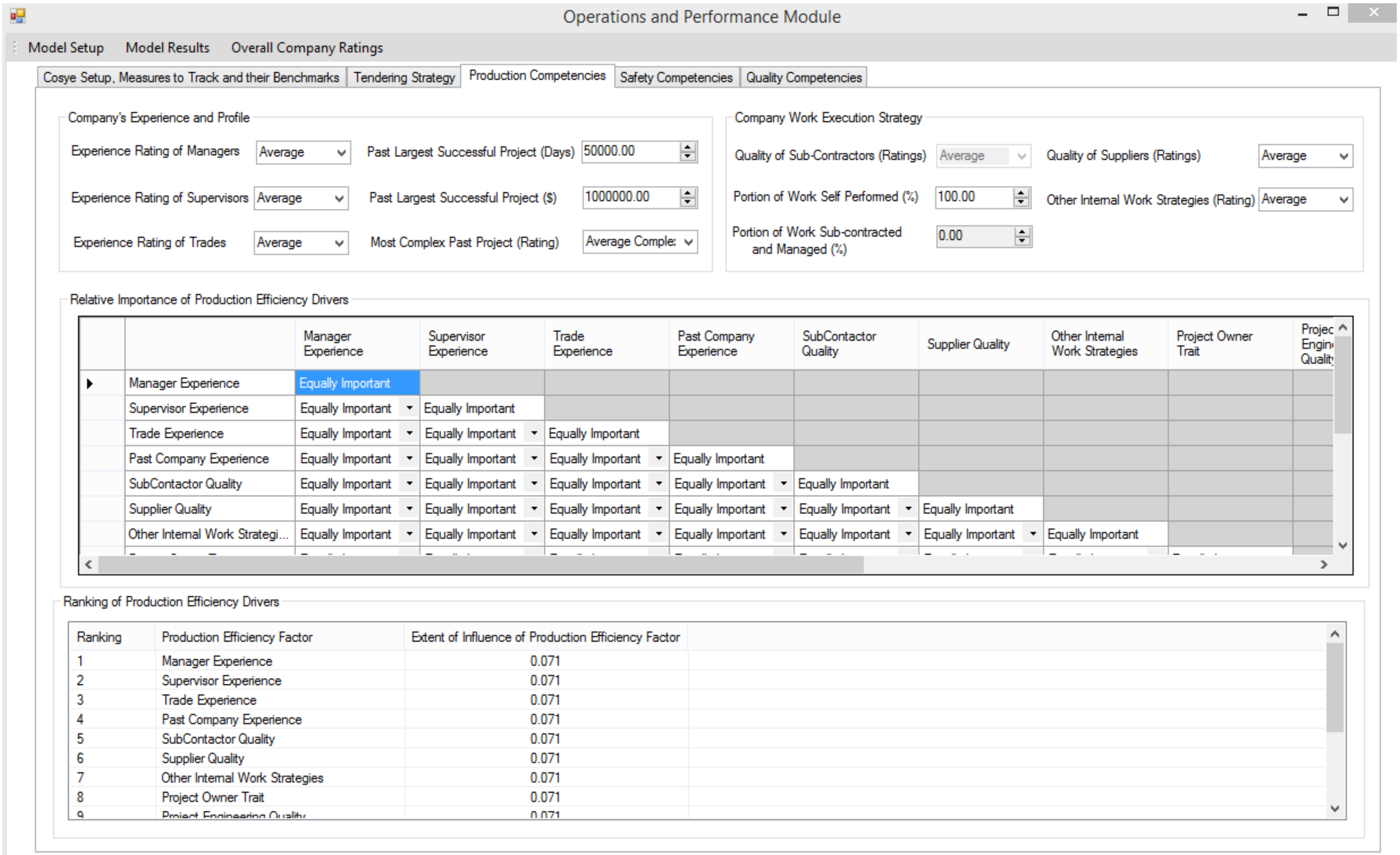


Figure 7.23: Inputs for the Company of Interest – Production Efficiency Competencies and Influencing Factors

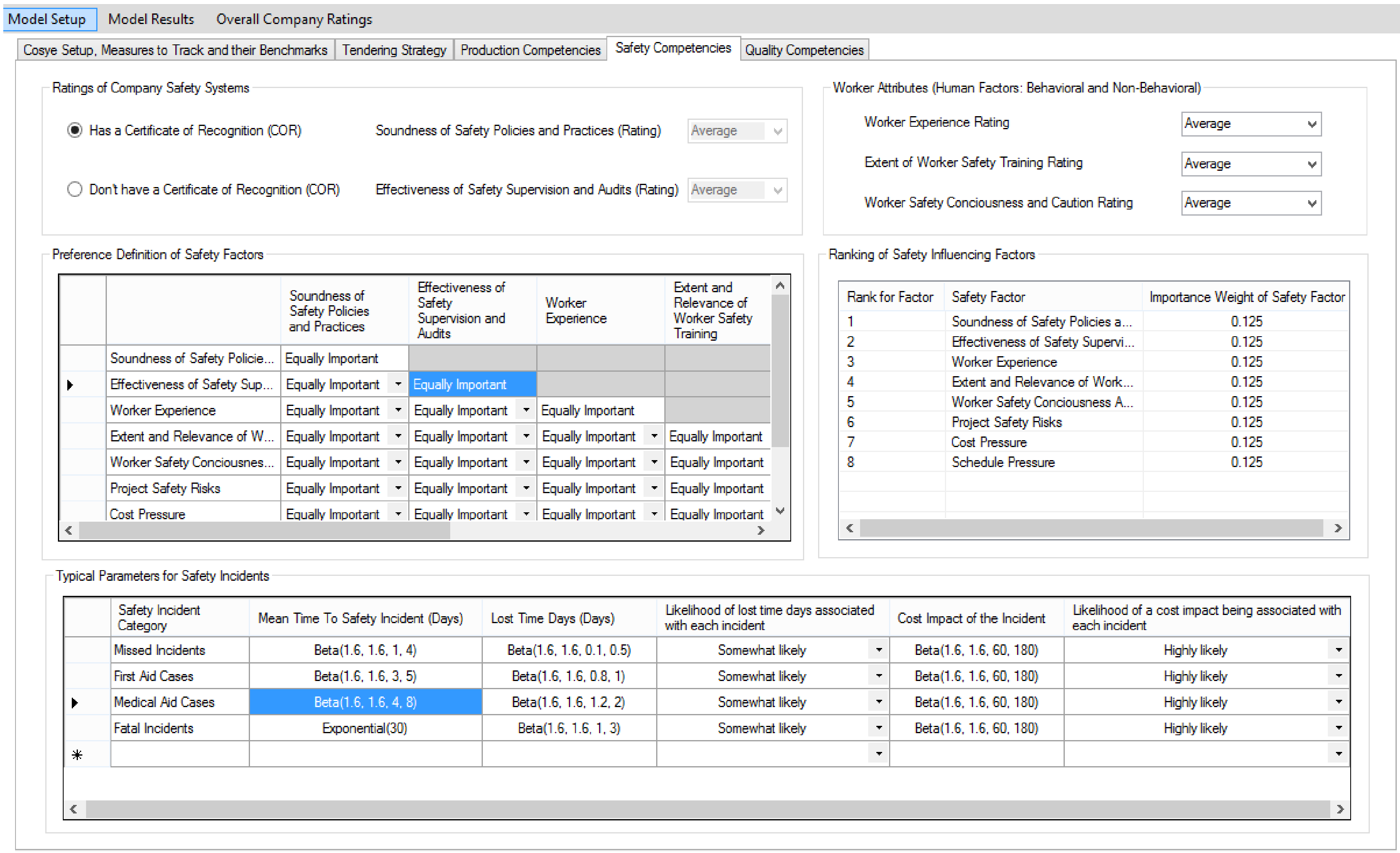


Figure 7.24: Inputs for the Company of Interest – Safety Competencies and Influencing Factors

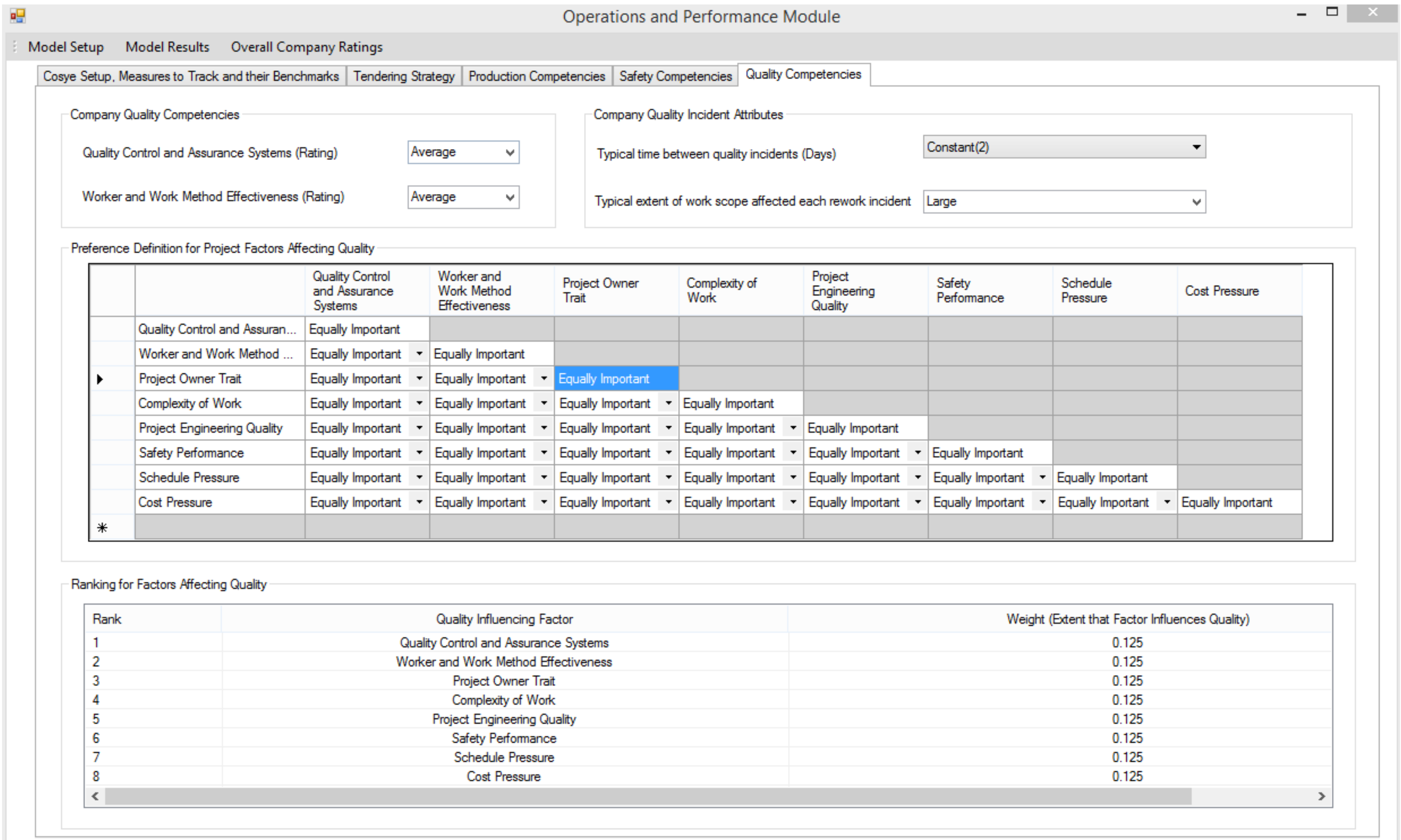


Figure 7.25: Inputs for the Company of Interest – Quality Influencing Factors and Company Quality Systems

7.6.2 Simulation Results from Simphony Federate

The Simulation execution was started and run till it terminated. Fortunately, no exceptions were thrown. This showed that the communication between the two federates was implemented appropriately. Results obtained from the Simulation are summarized in the following Figures/screen shots.

The implementation of the Windows form application dedicated a tab for the display of results of the virtual industry modelled within the AnyLogic federate. Details of these results were sent at the end of the fedartion simulation as interactions via the COSYE RTI. Details sent include:

- Information about company agents that were explicitly modelled within the virtual construction industry
- Details of resource agents modelled at the industry level
- The resource requirements for the different projects created in the course of the simulation execution.

This was done so that the modeller would have most of the output information about their company and the virtual construction industry within which it operated in one place to ease the correlation of results. Figure 7.26 shows a screen shot of the tab that displays those results for the scenario run.

7.6.2.1 Tendering Performance

Since the Company of Interest was the only company operating within the virtual construction industry, it was awarded all three project that were created in the course of the simulation. This is consistent with the results shown in Figure 7.27. The Pie chart shows that an equal number of small, medium size and large project(s) were created during simulation.

The bar chart on the otherhand indicates that the company of interest bid all three projects and was awarded all of them. This translated into it (i.e., the company of interest) holding the entire market share and scoring 100% from a tendering success perspective.

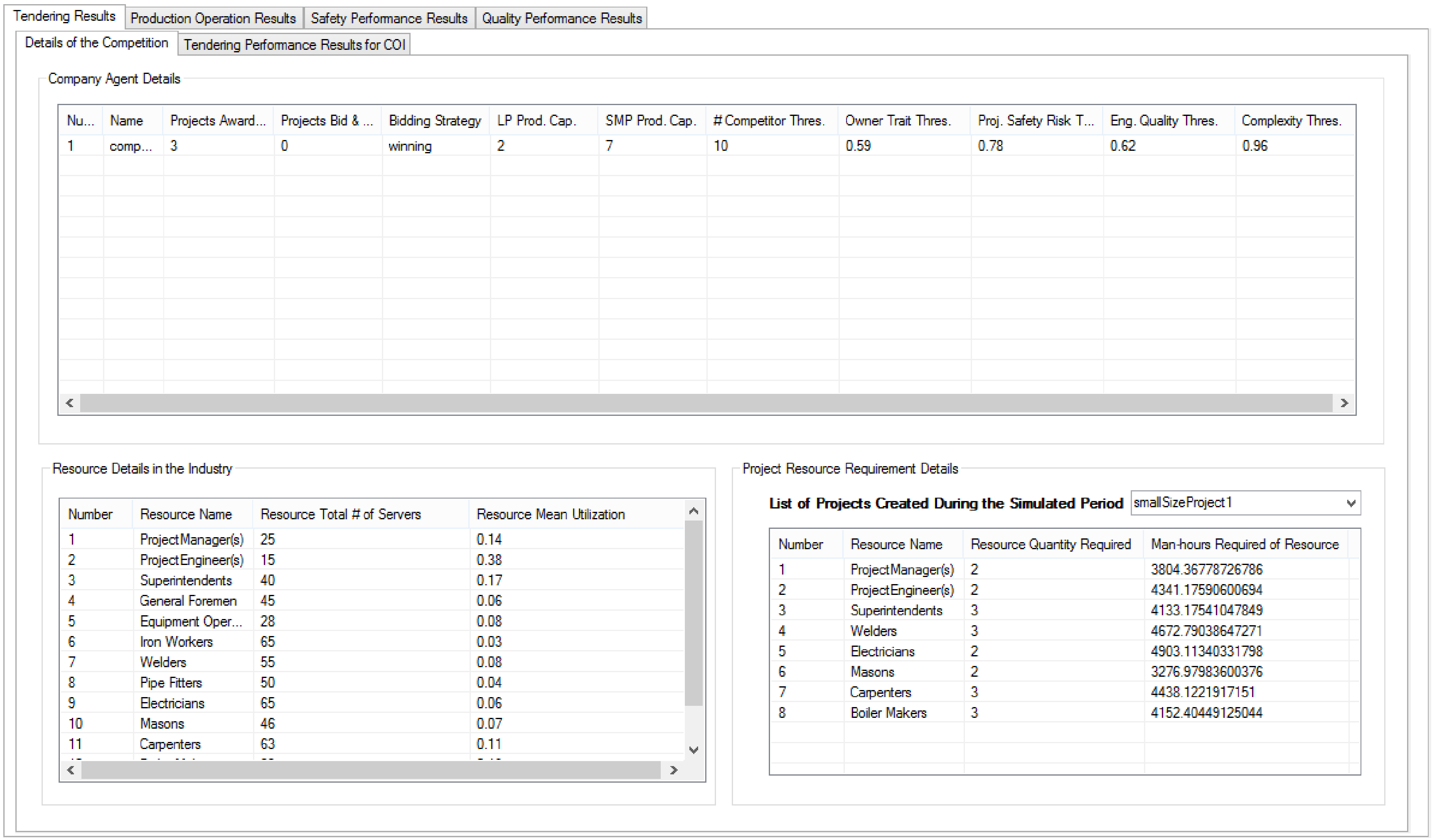


Figure 7.26: Resource Agent Details, Project Resource Requirements and Details of Company Agents in the Virtual Construction Industry

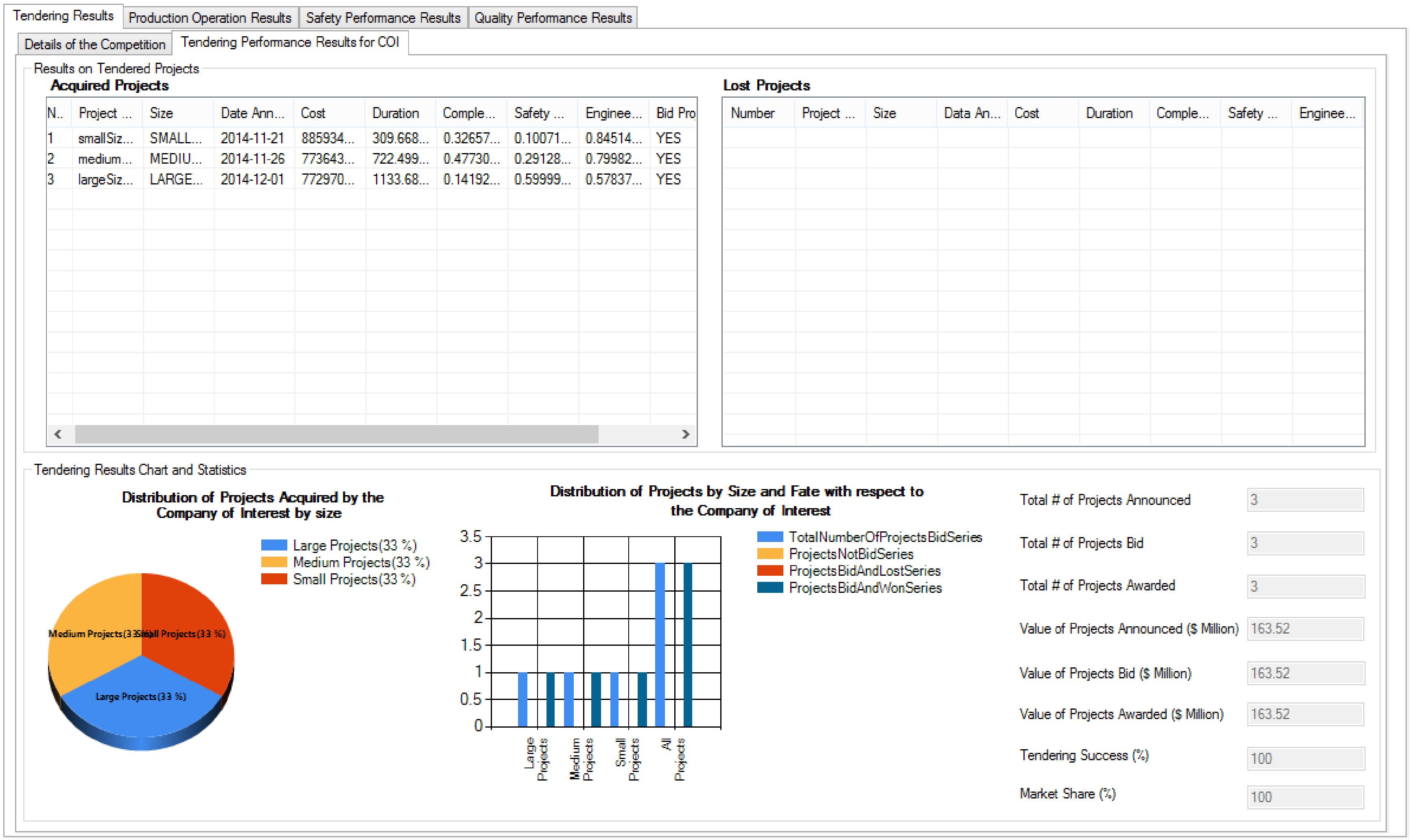


Figure 7.27: Tendering Performance Results for the Company of Interest

7.6.2.2 Operational Details of the Company of Interest

The Simphony federate was developed to track the performance of the company of interest as it processed projects awarded to it. Performance details tracked include safety ratings and incidents, quality ratings and incidents, production efficiency, schedule performance and cost performance. Other information tracked include: awarded projects queued awaiting company resources, projects in process and those completed, company resource (utilization) and file (length and waiting time) details.

Figure 7.28 summarizes results generated for the simulated scenario. Line charts are plotted indicating how the production efficiency, schedule performance and cost performance vary for the different project types as the simulation progressed.

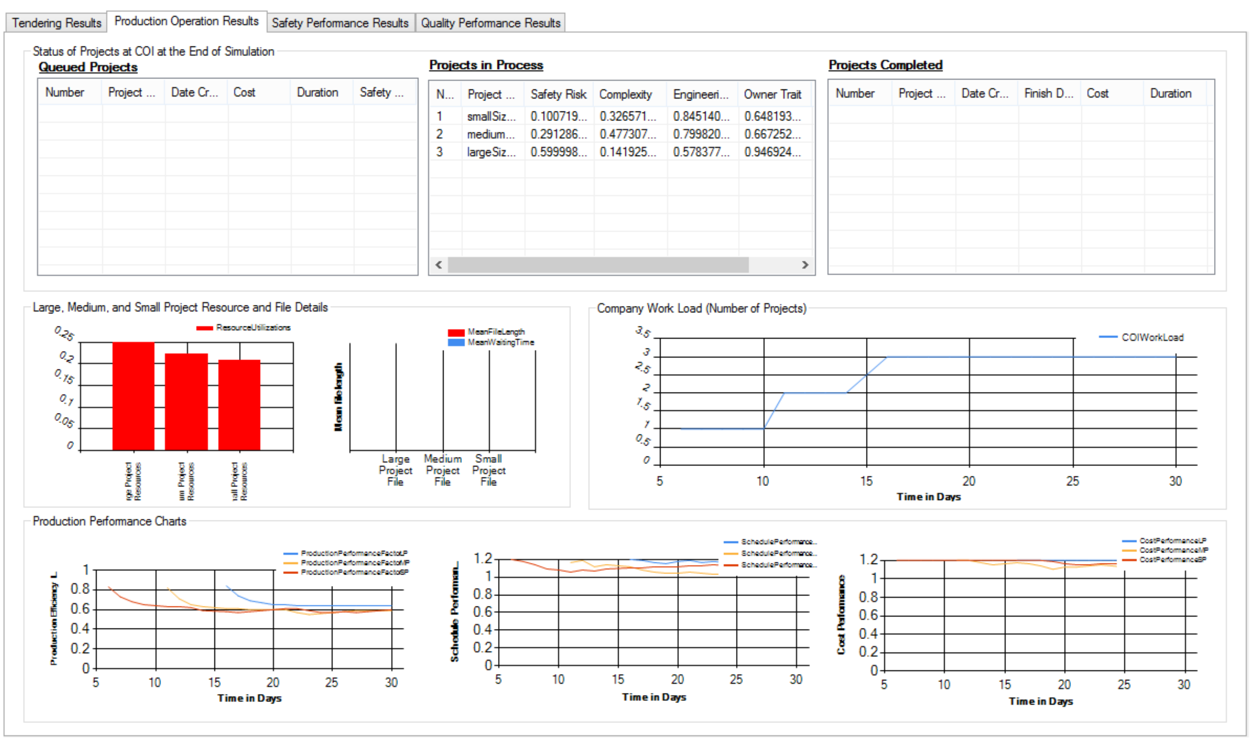


Figure 7.28: Operational Details for the Company of Interest

The chart for the waiting files don't have any visible bars showing values for the company waiting files because the parameters have values of zero. This meant that projects awarded to the company of interest did not have to wait for company resources to start being processed. Details of other operational performance details are discussed in the following sub-sections.

7.6.2.2.1 Safety Performance

Safety performance was modelled using a safety rating metric and the occurrence of safety incidents as work progressed at the company. Results (chart trendlines and pie chart in Figure 7.29) indicate that there are more frequent occurences of missed incidents and first aid cases than medical aid and fatal incidents. This is consistent with the imputs defined prior to the simulation. The trendlines show the number of incidents that occurred on each day for small, medium and large projects processed at the company of interest.

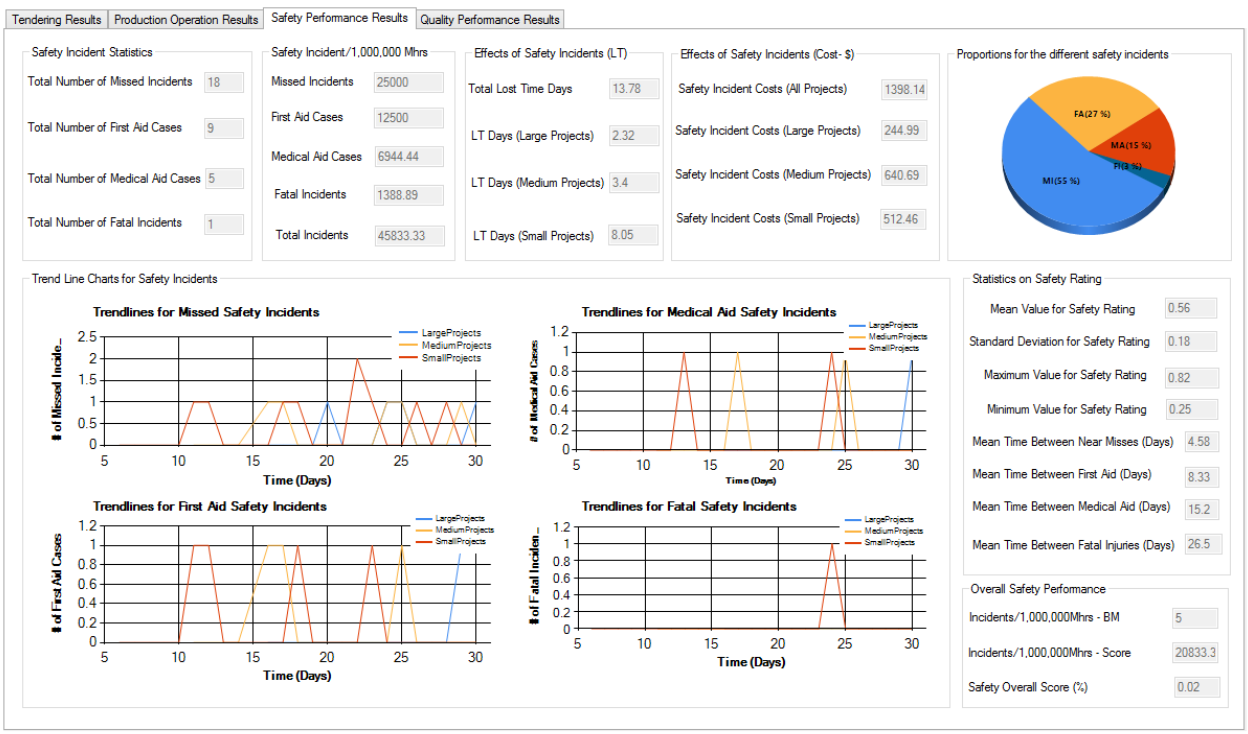


Figure 7.29: Project-Level Details for Safety Performance

There are also other statistics on the safety condition ratings (indicative of how safe projects at the company are), and safety incident occurence reported within text boxes. These results indicate that the developed application was modeling safety performance in an appropriate fashion.

7.6.2.2.2 Quality Performance

Quality performance was modelled using a holistic approach that considered the impacts of quality influencing factors and the occurrence os quality incidents. The Simphony federate was setup to model quality incidents with a nominal time between incidents of 0 days. This implied that a quality event would typicalliy be scheduled to occur each day when the quality rating on the projects and the company as a whole was poor. However, in situations where the quality rating was not extremely bad, these quality incidents would get postponed either by this inter-arrival time between incidents being increased by the simulator of the likelihood of an incident occuring when scheduled being reduced.

This explains the reason why there were no quality incidents reported on each day for the different project types in the trendlines within Figure 7.30.

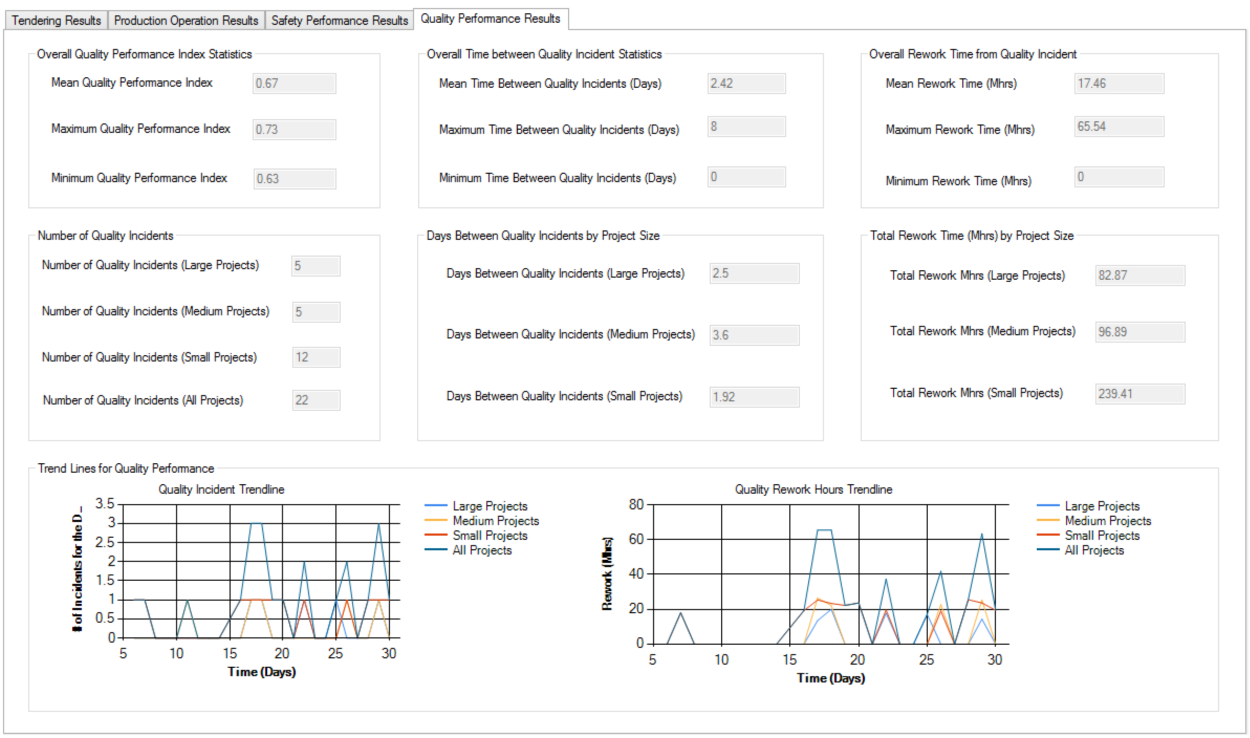


Figure 7.30: Project-Level Details for Quality Performance

The other results presented in Figure 7.30 are statistics about the quality performance of the coompany.

7.6.2.2.3 Overall Company Performance

The overall performance of a company is based on its scores in the different performance measures that were being tracked, the relative importance of these performance measures and the values of the benchmarks against which each of the performance scores are compared.

The plots shown in Figure 7.31 represent an aggregation of the project-level performance.

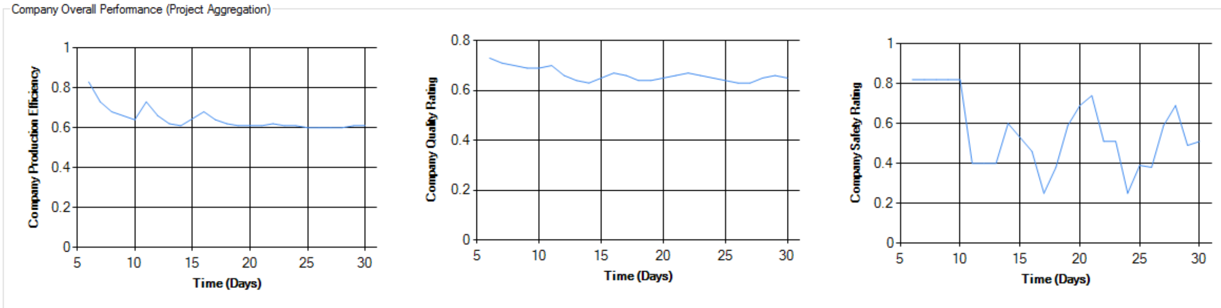
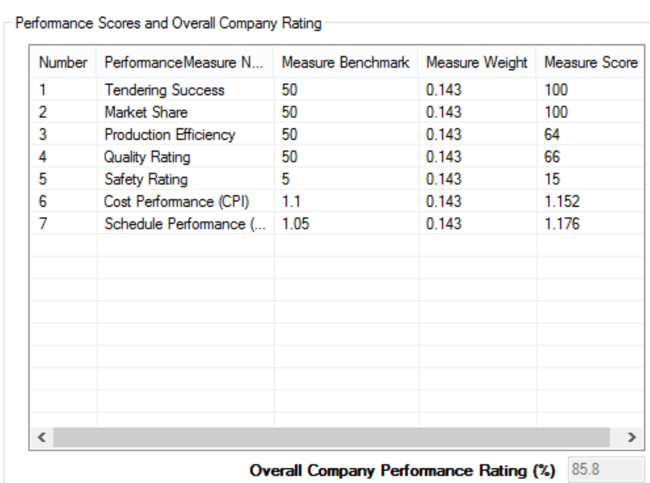


Figure 7.31: Company-Level Performance with respect to Production Efficiency, Quality Rating and Safety Rating

Safety rating (chart to the extreme right) indicates the prevailing conditions at the company with respect to safety. The troughs within the trendline of this chart correlate with the points in time that safety incidents occurred within projects in process at the company.

Similarly, the trendline for the quality rating (chart in the center of Figure 7.31) gently drops off as a result of the quality incidents that are experienced on projects. It averages at about 0.65.On the other hand, quality rating and safety rating represent some of the factors that influence production efficiency. This partly explains why the values of this metric average between those for the quality rating and the safety rating.

At the end of the simulation execution, details of all the performance measures tracked are displayed in a list view control. Figure 7.32 shows a screen shot of the results obtained in the scenario that was just run.



Performance Scores and Overall Company Rating

Number	PerformanceMeasure N...	Measure Benchmark	Measure Weight	Measure Score
1	Tendering Success	50	0.143	100
2	Market Share	50	0.143	100
3	Production Efficiency	50	0.143	64
4	Quality Rating	50	0.143	66
5	Safety Rating	5	0.143	15
6	Cost Performance (CPI)	1.1	0.143	1.152
7	Schedule Performance (...	1.05	0.143	1.176

Overall Company Performance Rating (%) 85.8

Figure 7.32: Overall Performance of the Company

All values are expressed as a percentage except safety performance, schedule performance (SPI) and cost performance (CPI) which are expressed in a fsahion that is identical to that used for their benchmarks. For example, safety performance benchmark in this scenario is 5 incidents/1,000,000 mhrs. In the simulation, a total of 15 safety incidents (don't include missed incidents) were obtained within the period of one month (720 mhrs).

The performance measure scores were each coompared against their benchmarks and then aggregated to obtain a value of 85.8%. At first glance, this seems to be an extremely high performance considering the poor quality and safety performance that was experienced by the company. However, the low values for the benchmarks made this seemingly poor performance seem reasonably good. In addition, all measures were given equal importance and those that scored high out weight those that scored low (in number).

7.7 SIMULATING COMPANY OF INTEREST AND COMPETITORS WITHIN A VIRTUAL INDUSTRY

A scenario was run in which the AnyLogic federate was run concurrently with the Simphony federate within a distributed simulation system in a fashion that the application was intended for use. Unlike prior experiments, this one contained the resource agents, company agents (competitors), bid manager agent, COI Amb. Agent and the construction industry agent (all actively engaged within the AnyLogic federate).

The company agents were setup with different bidding strategies but were also not constrained with respect to the types of projects they could bid. The simulation model was run for 90 calendar days simulating a 3 month period within the construction industry. The default values for modeling project inter-arrivals and their attributes where utilized in this experiment.

The trace log generated in the AnyLogic federate was filtered to determine details of one bidding cycle for a specific project. The details shown in Figure 7.33 indicate that the simulation system subjected projects created in the virtual construction industry to a rigorous competitive bidding process that involved different company agents. The virtual construction industry was comprised of the COI Amb. Agent and nine other competitor company agents. Of these competitor agents, three were small companies, four were medium size and two were large size companies.

```
The Bid manager has been notified of smallSizeProject1 arrival
smallCompanies[0] submitted a: true initial bid decision to buy tender documents for smallSizeProject1
smallCompanies[1] submitted a: true initial bid decision to buy tender documents for smallSizeProject1
smallCompanies[2] submitted a: true initial bid decision to buy tender documents for smallSizeProject1
mediumCompanies[0] submitted a: true initial bid decision to buy tender documents for smallSizeProject1
mediumCompanies[1] submitted a: true initial bid decision to buy tender documents for smallSizeProject1
mediumCompanies[2] submitted a: true initial bid decision to buy tender documents for smallSizeProject1
mediumCompanies[3] submitted a: true initial bid decision to buy tender documents for smallSizeProject1
largeCompanies[0] submitted a: true initial bid decision to buy tender documents for smallSizeProject1
largeCompanies[1] submitted a: true initial bid decision to buy tender documents for smallSizeProject1
companyOfInterestAmbassador[0] submitted a: true initial bid decision to buy tender documents for small
-------------------------------------------------------------------------------------------------------
smallCompanies[0] submitted a : true final bid decision for smallSizeProject1
smallCompanies[1] submitted a : true final bid decision for smallSizeProject1
smallCompanies[2] submitted a : true final bid decision for smallSizeProject1
mediumCompanies[0] submitted a : true final bid decision for smallSizeProject1
mediumCompanies[1] submitted a : true final bid decision for smallSizeProject1
mediumCompanies[2] submitted a : true final bid decision for smallSizeProject1
mediumCompanies[3] submitted a : true final bid decision for smallSizeProject1
largeCompanies[0] submitted a : true final bid decision for smallSizeProject1
largeCompanies[1] submitted a : true final bid decision for smallSizeProject1
companyOfInterestAmbassador[0] submitted a : true final bid decision for smallSizeProject1
-------------------------------------------------------------------------------------------------------
```

Figure 7.33: Details Traced in AnyLogic Federate for the Competitive Bidding Process of a Sample Project

All companies made it through the initial and final bid/no bid decision phases and went ahead to generate and submit their most competitive bid. The Bid manager agent identified the lowest bidder and awarded the project to them. Details of the bid prices submitted by each company agent are summarized in Figure 7.34.

```
smallCompanies[0] has submitted a bid of $1.2647744393787034E7 for smallSizeProject1
smallCompanies[1] has submitted a bid of $1.3241059313962795E7 for smallSizeProject1
smallCompanies[2] has submitted a bid of $1.3122764736491574E7 for smallSizeProject1
mediumCompanies[0] has submitted a bid of $1.2526446808374096E7 for smallSizeProject1
mediumCompanies[1] has submitted a bid of $1.2546670901010998E7 for smallSizeProject1
mediumCompanies[2] has submitted a bid of $1.2527164963662473E7 for smallSizeProject1
mediumCompanies[3] has submitted a bid of $1.2825248233900687E7 for smallSizeProject1
largeCompanies[0] has submitted a bid of $1.3312664727487199E7 for smallSizeProject1
largeCompanies[1] has submitted a bid of $1.3169479184189929E7 for smallSizeProject1
companyOfInterestAmbassador[0] has submitted a bid of $1.2671866454239536E7 for smallSizeProject1

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

mediumCompanies[0] was awarded smallSizeProject1 at $1.2526446808374096E7

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

Figure 7.34: Details of the Prices that each Company Agent carried with their bid

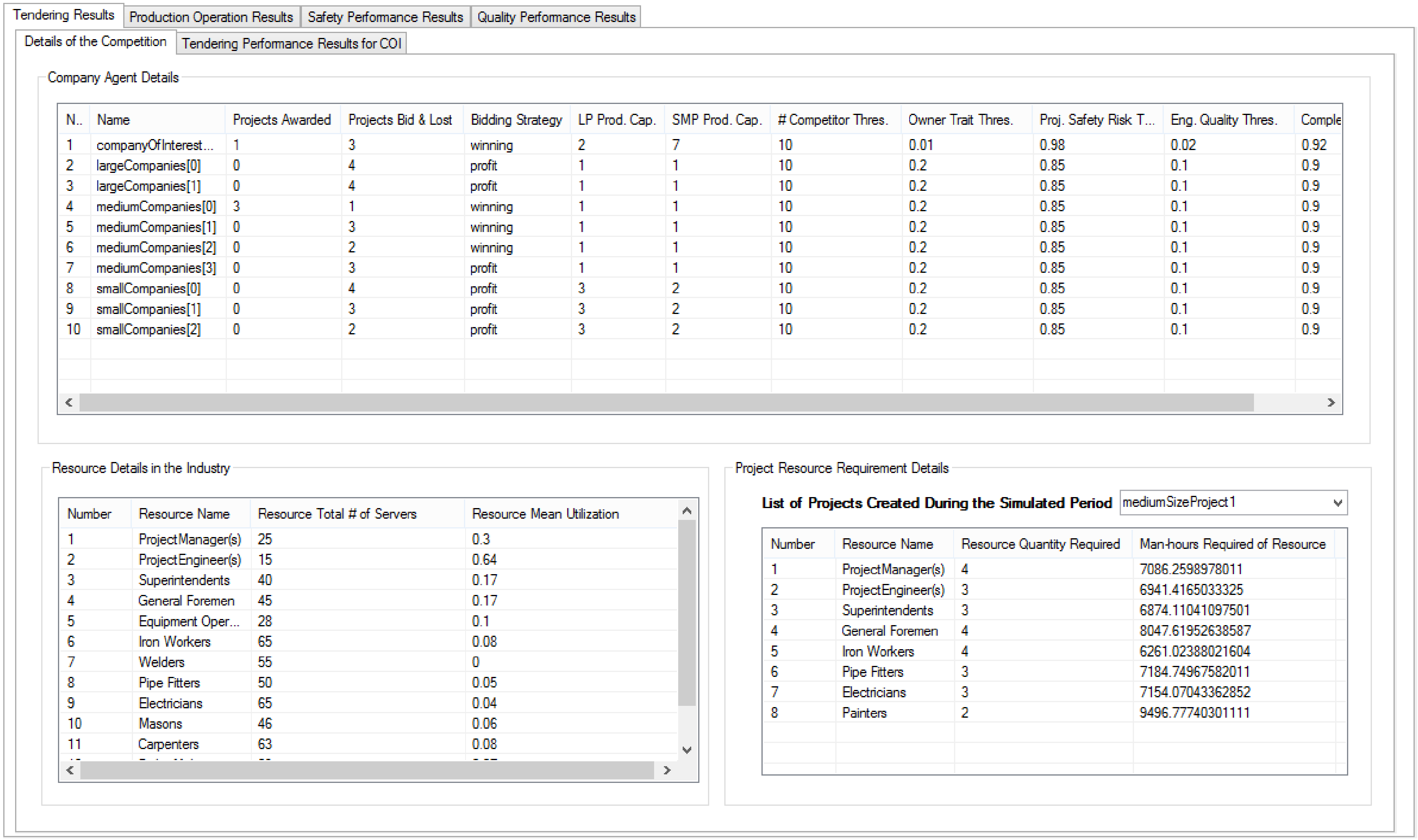
Other details generated at the end of the simulation as outputs were viewed in the user interface of the Windows forma application for the Simphony federate. The first of these is presented in Figure 7.35.

The Figure shows details of the company agents that were operating within the virtual construction industry, their properties i.e., tolerances for different projects, the projects they bid and lost that they lost or won. Details of resource agent utilization and their total quantity are also shown in the Figure 7.35.

Project resource requirements were displayed as proof of the assignment of requirements on creation of new projects. The user of the application can go through all projects created using the combo box and view their corresponding resource requirements in the list view control.

Another set of results are summarized in Figure 7.36.

This output display shows details of the projects that were awarded to the company of interest. It also shows projects lost by the company of interest in the competitive bidding process (three in this case). The bar chart reports details of projects not bid, those bid and awarded and those bid and lost. The text boxes indicate details of tendering performance of the company of interest. The performance of the company of interest dropped in the current scenario because it was exposed to competitor companies that were effective in acquiring projects.



N..	Name	Projects Awarded	Projects Bid & Lost	Bidding Strategy	LP Prod. Cap.	SMP Prod. Cap.	# Competitor Thres.	Owner Trait Thres.	Proj. Safety Risk T...	Eng. Quality Thres.	Comple
1	companyOfInterest...	1	3	winning	2	7	10	0.01	0.98	0.02	0.92
2	largeCompanies[0]	0	4	profit	1	1	10	0.2	0.85	0.1	0.9
3	largeCompanies[1]	0	4	profit	1	1	10	0.2	0.85	0.1	0.9
4	mediumCompanies[0]	3	1	winning	1	1	10	0.2	0.85	0.1	0.9
5	mediumCompanies[1]	0	3	winning	1	1	10	0.2	0.85	0.1	0.9
6	mediumCompanies[2]	0	2	winning	1	1	10	0.2	0.85	0.1	0.9
7	mediumCompanies[3]	0	3	profit	1	1	10	0.2	0.85	0.1	0.9
8	smallCompanies[0]	0	4	profit	3	2	10	0.2	0.85	0.1	0.9
9	smallCompanies[1]	0	3	profit	3	2	10	0.2	0.85	0.1	0.9
10	smallCompanies[2]	0	2	profit	3	2	10	0.2	0.85	0.1	0.9

Number	Resource Name	Resource Total # of Servers	Resource Mean Utilization
1	ProjectManager(s)	25	0.3
2	ProjectEngineer(s)	15	0.64
3	Superintendents	40	0.17
4	General Foremen	45	0.17
5	Equipment Oper...	28	0.1
6	Iron Workers	65	0.08
7	Welders	55	0
8	Pipe Fitters	50	0.05
9	Electricians	65	0.04
10	Masons	46	0.06
11	Carpenters	63	0.08

Number	Resource Name	Resource Quantity Required	Man-hours Required of Resource
1	ProjectManager(s)	4	7086.2598978011
2	ProjectEngineer(s)	3	6941.4165033325
3	Superintendents	3	6874.11041097501
4	General Foremen	4	8047.61952638587
5	Iron Workers	4	6261.02388021604
6	Pipe Fitters	3	7184.74967582011
7	Electricians	3	7154.07043362852
8	Painters	2	9496.77740301111

Figure 7.35: Company Agent, Resource Agent and Project Resource Requirement Details Simulated

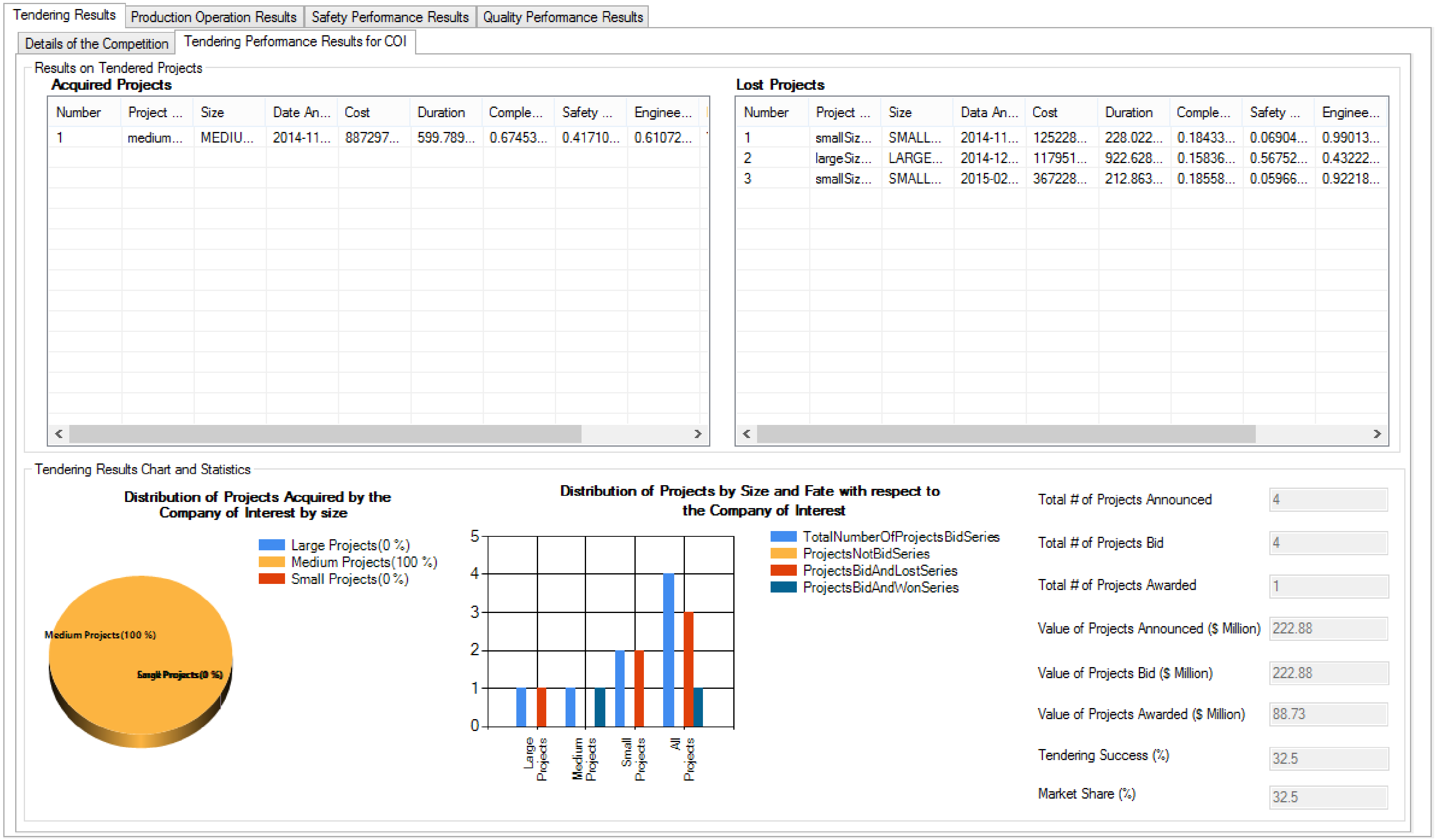
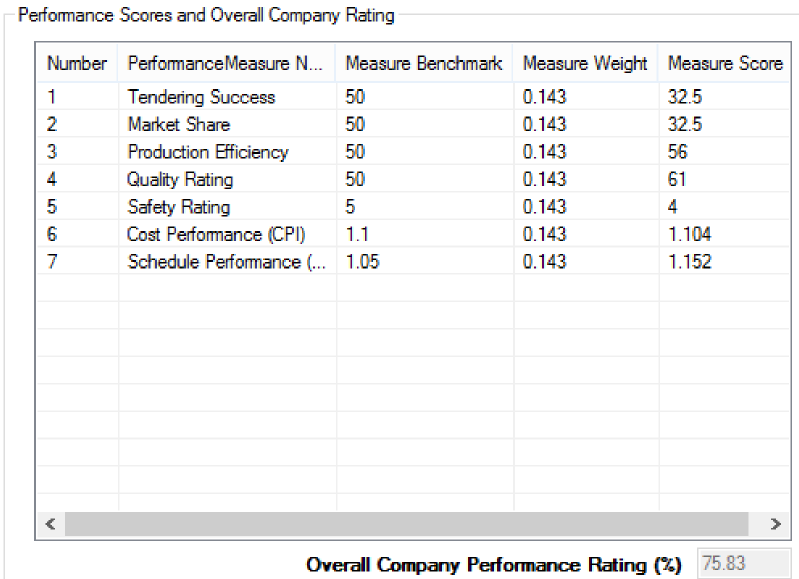


Figure 7.36: Tendering Results for the Simulation Experiment

The last result discussed in this section is that which summarizes the overall performance of the company of interest. This is summarized in Figure 7.37. The overall performance rating of the company dropped compared to that obtained in the scenario that did not contain competitors. The drop in performance could mainly be attributed to poor market share and tendering success scores.

The performance of safety improved in this scenario i.e., less incident count compared to the previous because the base distributions used for defining the inter-arrival times between safety incidents were different.



Number	PerformanceMeasure N...	Measure Benchmark	Measure Weight	Measure Score
1	Tendering Success	50	0.143	32.5
2	Market Share	50	0.143	32.5
3	Production Efficiency	50	0.143	56
4	Quality Rating	50	0.143	61
5	Safety Rating	5	0.143	4
6	Cost Performance (CPI)	1.1	0.143	1.104
7	Schedule Performance (...	1.05	0.143	1.152

Overall Company Performance Rating (%) 75.83

Figure 7.37: Overall Performance Result for the Company of Interest when Operating amongst Competitors

7.8 PROBABILISTIC SENSITIVITY ANALYSIS

Probabilistic sensitivity analysis requires that one or multiple input variables be changed while others are held constant so that the variation in model output(s) can be studied. This cannot always be done for all kinds of simulation models especially those that have a high number of input variables as this would be extremely cumbersome. Such problems arising as a result of high dimensionality in inputs are common amongst large scale complex models such as the one developed in this thesis.

To overcome these issues, high level scenarios were defined for the developed simulation application and experimented with. These scenarios were conveniently setup to mimic the response of the model to *extremely adverse conditions*, *average conditions* and *excellent conditions*. The input variables that exist within the model were used to define these conditions. Details of the combination of variables utilized in experiments are summarized in the following section.

7.8.1 Input Details for Probabilistic Sensitivity Analysis

The simulation application developed in this thesis was setup in such a way that the performance and competitiveness of the company of interest was dependent on the type of conditions that the company operates amidst. These conditions were defined based on two criteria namely:

- The type of work that the company does i.e., the attributes of projects it acquires and processes
- The competencies that exist at the company

For purposes of carrying out the probabilistic sensitivity analysis, project attributes and company competencies were conveniently defined to set up each of the scenarios (*extremely adverse conditions*, *average conditions* and *excellent conditions*). The Tables 7.3 summarize the input variables that were used in these definitions.

Table 7.3: Inputs Used for Defining the Adverse, Average, and Excellent Conditions

Scenario	Type of work	Competencies at the Company of Interest		
		Production Efficiency	Safety Competencies	Quality Competencies
Extremely adverse Conditions	Project owner trait (very poor); Project complexity	Experience rating of Managers (extremely low); Experience rating of supervisors (extremely low); Experience rating of	Does not have certificate of recognition (COR); Soundness of safety policies and practices (extremely low); Effectiveness of	Quality control and assurance system rating (extremely low); Worker and work method

Scenario	Type of work	Competencies at the Company of Interest		
		Production Efficiency	**Safety Competencies**	**Quality Competencies**
	(very low); Project engineering quality (very poor); Project safety risk (very low);	trades (extremely low); Most complex past project completed (extremely low complexity); Past largest project-days (not long); Past largest project-cost (not high); Quality of subcontractors (extremely low); Quality of suppliers (extremely low); Rating of other internal work strategies(extremely low);	safety supervision and audits (extremely low); Worker experience (extremely low); Extent and relevance of worker safety training (extremely low); Worker safety consciousness and caution (extremely low);	effectiveness (extremely low); Typical extent of work scope affected every quality incident (extremely large);
Average Conditions	Project owner trait (somewhat good); Project	Experience rating of Managers (average); Experience rating of supervisors (average); Experience rating of	Does not have certificate of recognition (COR); Soundness of safety policies and practices (average);	Quality control and assurance system rating (average); Worker and

Scenario	Type of work	Competencies at the Company of Interest		
		Production Efficiency	Safety Competencies	Quality Competencies
	complexity (somewhat high); Project engineering quality (somewhat good); Project safety risk (somewhat high);	trades (average); Most complex past project completed (average); Past largest project-days (moderate); Past largest project-cost (moderate); Quality of subcontractors (average); Quality of suppliers (average); Rating of other internal work strategies(average);	Effectiveness of safety supervision and audits (average); Worker experience (average); Extent and relevance of worker safety training (average); Worker safety consciousness and caution (average);	work method effectiveness (average); Typical extent of work scope affected every quality incident (somewhat large);
Extremely favorable Conditions	Project owner trait (very good); Project complexity (very	Experience rating of Managers (extremely high); Experience rating of supervisors (extremely high); Experience rating of	Has a certificate of recognition (COR); Soundness of safety policies and practices (extremely high); Effectiveness of safety supervision and audits (extremely	Quality control and assurance system rating (extremely high); Worker and work method effectiveness

Scenario	Type of work	Competencies at the Company of Interest		
		Production Efficiency	Safety Competencies	Quality Competencies
	high); Project engineering quality (very good); Project safety risk (very high);	trades (extremely high); Most complex past project completed (extremely high complexity); Past largest project-days (long); Past largest project-cost (high); Quality of subcontractors (extremely high); Quality of suppliers (extremely high); Rating of other internal work strategies(extremely high);	high); Worker experience (extremely high); Extent and relevance of worker safety training (extremely high); Worker safety consciousness and caution (extremely high);	(extremely high); Typical extent of work scope affected every quality incident (extremely small);

Most of the input variables used within the developed simulation application were set up as linguistic variables. These input variables were appropriately defined to reflect the targeted conditions as indicated in Table 7.3. At simulation run-time, these inputs were mapped onto statistical distributions that ranged from 0.0 to 1.0. For the definition of *adverse conditions*, input variables were appropriately selected to map to statistical distributions on the range 0.00 to 0.20.

Statistical distributions ranging from 0.38 to 0.58 were mapped to using the inputs that defined *average conditions*. Statistical distributions ranging from 0.70 to 0.95 were utilized in simulation execution when input variables were defined that corresponded to *excellent conditions*.

7.8.2 Results of the Probabilistic Sensitivity Analysis

Each scenario (condition) defined using the variables in the previous section were experimented with. A set of model outputs were tracked in each of these experiments. These only represented a sub-set of the all model outputs and included:

- Money lost each day (positive value for a loss and negative for a profit)
- Time lost each day (positive value for time lost and negative for time saved)
- Mean quality rating
- Mean safety rating
- Production efficiency

10 model runs were performed with each scenario (condition) and results recorded from each. Values from experimenting with the three different conditions were summarized in Table 7.4, 7.5, and 7.6.

Table 7.4: Simulation Results - Extremely Good Conditions

Simulation Experiment	Money Lost each day ($)	Time Lost each Day (Hrs)	Mean Quality Rating	Mean Safety Rating	Production Efficiency
1	-24179.65	-5.35	0.82	0.91	0.90
2	-27143.90	-6.67	0.81	0.90	0.90
3	-21115.83	-7.30	0.81	0.90	0.87
4	-26103.73	-5.43	0.80	0.86	0.88
5	-25423.16	-5.89	0.81	0.90	0.90
6	-25572.54	-5.38	0.79	0.90	0.89
7	-24276.23	-7.03	0.82	0.92	0.89
8	-30782.68	-6.18	0.82	0.93	0.90

Simulation Experiment	Money Lost each day ($)	Time Lost each Day (Hrs)	Mean Quality Rating	Mean Safety Rating	Production Efficiency
9	-27262.33	-7.61	0.80	0.96	0.90
10	-23449.87	-6.60	0.81	0.93	0.90

Table 7.5: Simulation Results - Average Conditions

Simulation Experiment	Money Lost each day ($)	Time Lost each Day (Hrs)	Mean Quality Rating	Mean Safety Rating	Production Efficiency
1	-12773.05	0.17	0.59	0.49	0.60
2	-9373.60	-0.67	0.58	0.43	0.60
3	-8940.13	-0.72	0.59	0.46	0.61
4	-7890.71	0.40	0.58	0.44	0.62
5	-11278.55	0.26	0.58	0.44	0.60
6	-12274.40	-0.65	0.59	0.41	0.62
7	-16352.16	0.59	0.60	0.50	0.62
8	-8207.74	0.66	0.59	0.49	0.63
9	-9212.93	1.29	0.59	0.46	0.62
10	-11030.20	0.10	0.60	0.51	0.63

Table 7.6: Simulation Results - Extremely Bad Conditions

Simulation Experiment	Money Lost each day ($)	Time Lost each Day (Hrs)	Mean Quality Rating	Mean Safety Rating	Production Efficiency
1	19228.53	9.52	0.37	0.21	0.27
2	18956.41	9.00	0.37	0.23	0.27
3	24601.04	10.79	0.37	0.17	0.27
4	18991.76	9.81	0.37	0.21	0.27

Simulation Experiment	Money Lost each day ($)	Time Lost each Day (Hrs)	Mean Quality Rating	Mean Safety Rating	Production Efficiency
5	17605.22	9.61	0.38	0.19	0.28
6	22965.34	8.64	0.37	0.20	0.28
7	18748.46	8.97	0.37	0.21	0.28
8	17092.23	9.10	0.38	0.20	0.28
9	21477.46	10.01	0.37	0.20	0.28
10	16327.73	7.22	0.37	0.21	0.29

Statistics were computed for the results obtained from experimenting with each of the scenarios. These were summarized and presented in Table 7.7.

Table 7.7: Statistics of the Results Generated from the Probabilistic Sensitivity Analysis

Scenario	Statistic	Money Lost each day ($)	Time Lost each Day (Hrs)	Mean Quality Rating	Mean Safety Rating	Production Efficiency
Extremely Good Conditions	Mean	-25681.1	-6.344	0.809	0.911	0.893
	Standard Deviation	2709.171	0.826293	0.009944	0.026437	0.010593
	Confidence Interval	[-27619,-23,743]	[-6.94,-5.75]	[0.80, 0.82]	[0.89, 0.93]	[0.89, 0.90]
Average Conditions	Mean	-10733.3	0.143	0.589	0.463	0.615
	Standard Deviation	2589.018	0.658248	0.007379	0.03335	0.011785
	Confidence Interval	[-12585.3, -8881.4]	[-0.33, 0.61]	[0.58, 0.59]	[0.44, 0.49]	[0.61, 0.62]
Extremely Bad Conditions	Mean	19599.42	9.267	0.372	0.203	0.277
	Standard Deviation	2634.891	0.949784	0.004216	0.01567	0.006749
	Confidenc	[17714.7,	[8.59,	[0.37,	[0.19,	[0.27, 0.28]

Scenario	Statistic	Money Lost each day ($)	Time Lost each Day (Hrs)	Mean Quality Rating	Mean Safety Rating	Production Efficiency
	e Interval	21484.2]	9.95]	0.38]	0.21]	

7.8.3 Discussion of Simulation Results

In order to facilitate the interpretation and discussion of the model results, three result sets were conveniently selected and plotted on graphs. These included results for production efficiency, safety rating, and quality rating. Charts generated are presented in Figures 7.38, 7.39, and 7.40.

The distribution of data points within each graph and between graphs were then used to make inferences about the validity of the developed simulation application. The simulation application would be considered valid if values obtained from different simulation runs for the same performance metric were close to each other i.e., no significant vertical scatter. Also, the simulation application would be considered valid and reliable if results that were indicative of poor performance were obtained for model inputs used to define *extremely bad conditions*. Observations of average results from experiments run with inputs corresponding to *average conditions* along with excellent performance results for inputs corresponding to *excellent conditions*, would confirm the validity of the simulation application from the face of it.

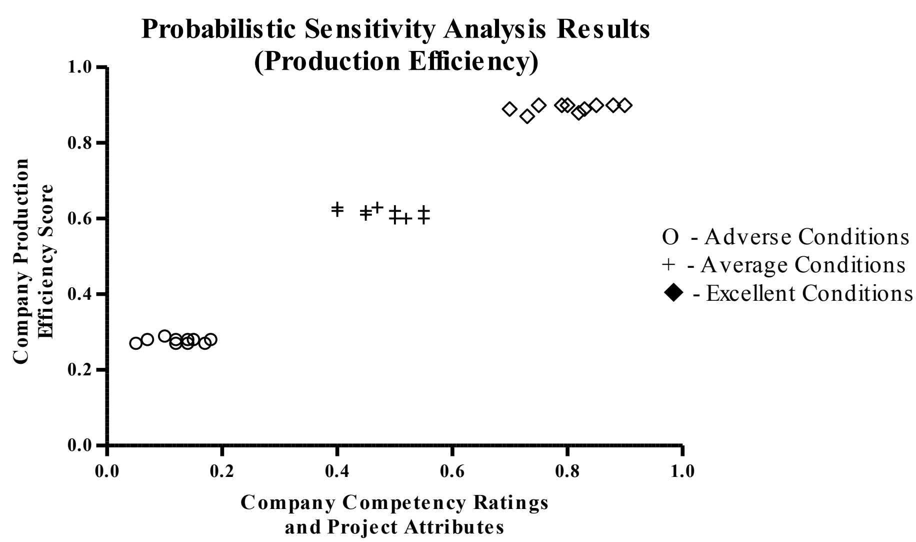


Figure 7.38: Probabilistic Sensitivity Analysis Results for Production Efficiency for Adverse, Average, and Excellent Conditions

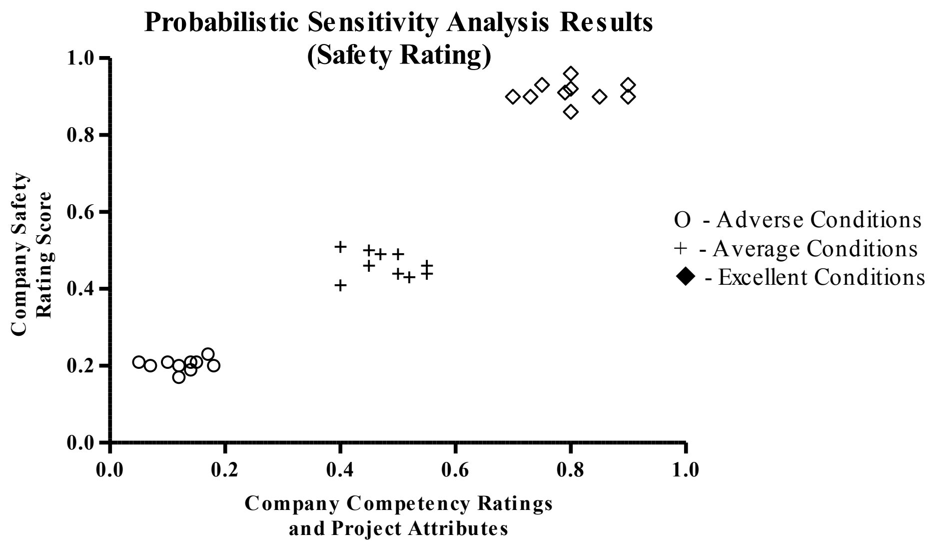


Figure 7.39: Probabilistic Sensitivity Analysis Results for Safety Rating for Adverse, Average, and Excellent Conditions

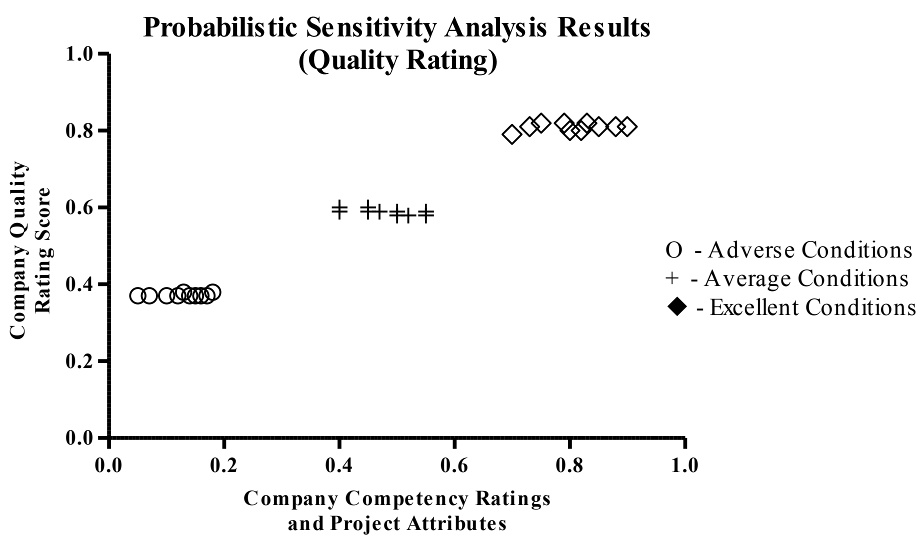


Figure 7.40: Probabilistic Sensitivity Analysis Results for Quality Rating for Adverse, Average, and Excellent Conditions

The results plotted in Figures 7.38, 7.39, and 7.40 indicate that the simulation application was responsive to the different operating conditions that the company of interest was immersed into. *Extremely bad* conditions yielded poor performance (values in the vicinity of 0.2 – 0.4) while *average conditions* yielded an average performance (values in the vicinity of 0.4 - 0.8), and *excellent conditions* yielded a good performance (values in the vicinity of 0.8 - 1.0).

In addition, results for all performance metrics obtained for the defined conditions were distributed along a thin horizontal strip. This implied that there was no significant variation in the output for inputs that defined the same condition. The variations existed were due to the randomness in samples drawn from statistical distributions during each simulation experiment.

The trends of these results indicate that the developed simulation application is sensitive to changes in inputs. This confirmed that the simulation application generates valid and reliable results.

7.9 SUMMARY FOR CHAPTER SEVEN

This chapter has successfully discussed the systematic steps followed in developing the performance management simulation application. Aspects of the verification and validation of this application are also discussed. The components of the application were executed independently and shown to run successfully and generate reasonable results. The entire application was also executed and shown to also run without exceptions and generated a reasonable result.

A high-level probabilistic sensitivity analysis was carried out in the chapter for purposes of demonstrating that the developed simulation model generates a valid result. Findings from these experiments revealed that the model is valid and generates reliable results.

CHAPTER EIGHT – CONCLUSIONS AND RECOMMENDATIONS

8.0 INTRODUCTION TO CHAPTER EIGHT

Academic research studies will typically have a number of limitations that are a result of the assumptions made or the methods used in solving the problem. This thesis was no exception because it had a number of limitations. These are summarized in the first section of this chapter.

This was followed by a discussion of the conclusions that were realized in the course of carrying out this thesis work. Some of the conclusions arise from the formalism of concepts that relate to company competitiveness while others became evident in the development and deployment of the model/application. The rest were learned from published literature on the subject of competitiveness within construction and other domains. The most relevant ideas are presented in this chapter.

Also, recommendations are proposed based on a number of simplifying assumptions and limitations of the study. Aspects that could not be covered as a result of time constrains and other constrains such as lack of relevant data, have also been used as a basis for recommendations.

8.1 LIMITATIONS OF THE DEVELOPED SIMULATION APPLICATION

Several simplifications were made in the course of developing the simulation application in this study. The majority of these manifested as limitation in the functional capabilities of the model. These are summarized in the following paragraphs and served as a basis for the discussion presented in the recommendations section of this thesis.

Construction companies typically have work in process at any given point in time. Assessing their performance by emulating their work execution process would require that the work-in-process be considered and modelled explicitly. However, in the application developed in this thesis, this aspect was assumed not to exist and was not modelled.

Most construction industries transition through cycles of economic boom and bursts. These affect the rate at which projects are commissioned and for extreme adverse cases could result in on-going projects being suspended. The developed application in this thesis does not provide robust features for explicitly modeling such cycles. Moreover, temporary or indefinite suspension of on-going projects and their possible resumption at a future date (for works temporarily suspended) were not explicitly modelled in the application.

The production capacity that exists at each construction company was explicitly modelled using resources blocks. Resource numbers were defined to represent the total number of small, medium size and large projects that a company could concurrently execute. This production capacity was defined in such a way that it was not interchangeable i.e., small projects could only utilize their own production capacity and not that of large or small companies. In a real life setting, production capacity can be utilized indiscriminately by any project provided it is available. This flexibility in the utilization of production capacity was not provided for in the developed application.

The competitive acquisition of work by companies was modelled by representing all companies that operate within the specified construction industry. In the real world, companies enter or leave any construction industry from time to time. In the developed model, companies operating within an industry that were defined prior to simulation persisted till the end of simulation. This implied that the dynamic of varying company numbers was not represented within the model that was developed.

Likewise resource pools, from which companies draw to perform their operations, get depleted and replenished from time to time as a result of workers leaving and joining the construction industry respectively. For the case of equipment, the depletion and replenishment could arise from the disposal of old equipment and procurement of new ones respectively. The developed model was limited in this respect because it did not represent this dynamic explicitly.

8.2 CONCLUSIONS

An extensive literature review of competitiveness within different domains revealed that it is a very broad and complex topic that cannot be exhaustively studied and reported especially within time bound studies like this thesis. It is multi-faced (can be perceived differently depending on who is assessing it), hierarchical (can be assessed at different levels i.e., national, industry, company or individual levels) and of different types.

This thesis set out to study competitiveness at a company level from a strategic and operational stand point. This was successfully achieved and resulted in the development of a simulation-based application for contractor performance management within the construction domain. A number of challenges were encountered due to the complex and ill-defined nature of the phenomenon of competitiveness. Some of these included:

- A lack of standardized and formal approach for expressing and quantifying (measuring) certain aspects of competitiveness i.e., performance measures
- A lack of formal numerical or mathematical relations that represent constructs that relate to competiveness.
- A lack of formal knowledge or cases that one could learn from or replicate in the abstraction of ill-defined or ill-structured concepts that closely relate to the subject of company competitiveness which could not be ignored and left out of the modeling process.

It was also noted that competitiveness of companies varies with the size of the company. Variations in competitiveness of companies of different size stem from the availability and management of resources. For example, small companies face competitiveness challenges as a result of scarcity or a lack of resources while large companies encounter problems that stem from inadequately managing their resources. For companies of the same size, variations arise from uniqueness of internal systems and processes. Also, variations in the attributes of the resources they possesses and the nature of their interaction with the environment, explain differences in competitiveness.

Efforts have been directed towards the development of methods and tools to measure performance and competitiveness in a quantitative fashion. Some of these tools are in use within large organizations but possess a number of pitfalls as pointed out in earlier chapters. This thesis adopted a simulation-based approach to address these issues while advancing the state-of-the-art. Simulation was used because it has the ability to cope well with dynamic and uncertain phenomenon. In addition, applications developed using simulation have the potential to generate knowledge or information that is not obvious (i.e., was not known beforehand and could not be estimated as accurately from projections) hence the need to develop dynamics performance management systems that are superior to static systems. With this approach, an application was successfully developed. The process of developing this application resulted in a realization of effective ways to formalize certain competitiveness concepts that were previously not well understood, never abstracted and represented on computer in an efficient fashion. An example of this is the use of statistical distributions and basic probability theories to model the inter-arrival of safety incidents on construction projects. The likelihoods and impacts of these safety incidents were also modelled using probability and statistical distributions.

Other lessons were learned about when to and not to use a distributed simulation approach for developing a simulation model. It was established that the following situations warrant the use of distributed simulation:

- In situations where there are features or functionality that are required to develop, deploy and execute the simulation system which do not all exist within a single application, federates can be developed around each respective application and subsequently integrated into a simulation federation that represented the desired system.
- In addition, situations in which the deployment of the simulation system dictates the spatial distribution of its components. Simulation-based games for education purposes are a very good example of such systems.

Distributing the components of a simulation system should never be done without an appropriate justification otherwise it would become overly complicated, difficult to develop, deploy, and maintain.

A systematic process for developing and documenting large scale complex simulation models was presented in this thesis. The practice of designing constructs to be modelled and their interaction prior to model development proved to be helpful. Standard design tools such as state charts, activity diagrams, sequence diagrams etc. were extensively used to document design specifications. It was realized that the practice of developing simulation model design specifications was not quite prevalent amongst researchers and practitioners in the simulation domain. It is hoped that this will change.

It was pointed out in Chapter 7 that there are different techniques for validating models and data-driven approaches are usually at the top of the hierarchy. It was not possible to carry-out this type of validation in this thesis because of the difficultly in accessing a good data set to utilize. This challenge could not be attributed to a single reason but rather could be related to the complexity of the type of data that would be required and the inadequacies in data tracking techniques currently used by contractor companies especially within the Alberta construction industry. Consequently, theoretical approaches were adopted which involved subjecting all design specifications (i.e., sequence diagrams, activity diagrams, flow charts, other concept models, numeric and mathematical models) to rigorous scrutiny by different groups of people. The first group of people included academia (my professor/supervisor, his technical support staff, colleagues, and other professors within the construction research group at the Hole School of Construction Engineering at the University of Alberta) while the other group included experienced practitioners from industry (mainly from partner companies to the Industrial chair held by my professor).

Tests were carried out to verify that the designs were precisely translated into simulation models. Details of these were discussed in chapter 7. Tests involved generating trace logs, using message boxes, creating break points and stepping through the code snippets in debug mode. These tests were successful hence demonstrating that the model is reliable. It was also shown that the application runs without exceptions being thrown and generates a reasonable result. This confirms that there were no syntax and logical errors in implementations of the developed application.

8.3 RECOMMENDATIONS

A number of ideas were generated in the course of doing this thesis study. Most of these were intended to make the model emulate reality in a more accurate fashion. They can be pursued in an independent follow-up study in a formal structured or informal fashion. These recommendations were based off of some of the assumptions that were made in the model development phase of the thesis while others were realized at the point of model experimentation and deployment. Each of these is discussed in the following paragraphs.

It was evident that there is an apparent need to come up with consistent ways of measuring and expressing benchmarks for performance measures within the construction industry. For example, it was not clear in the literature which unit of measure is used for tracking quality performance.

Various options seem to be in use such as the percentage of hours worked that need to be reworked, the number of quality incidents experienced for a given work period etc. Performance measures and their benchmarks seem to be defined and measured according to the convenience of the individuals involved in tracking performance, something that should be addressed in future studies.

Utilizing simulation-based approaches for modeling performance issues within the construction industry required for the explicit modeling of projects. Simulating projects within the construction domain can be accomplished in one of two ways. One of these involves a high level of abstraction and representation of the work scope and progress, while the other is low level in the fashion that work scope is represented and processed. The latter approach would require special or general purpose simulation templates to be developed (if they don't exist), embedded within the application and used in the course of the simulation. The latter approach involves reducing the man-hours of effort required in the project as the simulation engine clock advances its time. These are explained in more detail.

In the first approach, man-hours are used to express the effort (scope of what needs to be performed) work in a project. This high-level approach does not require the abstraction and modeling of the activities within the project and their logic relationships. This was found to be convenient as a way around not being able to represent project details using process interaction modeling methods.

The second approach entails modeling construction projects at an activity level which explicitly represent the process flow sequence logic for the entire project and other relevant constructs that relate to this and the execution of each task can be explicitly modelled. In order to achieve this, process interaction models that emulate the execution of the project at a task level would need to be created, stored behind the scenes, instantiated, attributes set and run. These process models could be general purpose simulation models or special purpose simulation models. A special purpose template approach would be preferred over a general purpose template approach because the former is more generic and allows for more flexibility which implies that construction projects of with different features (sizes, forms and work scope), can be instantiated and executed. Modeling a project at the activity level can be referred to as a *"plug and play"* approach because whenever a project gets awarded to a company, it gets deserialized and instantiated from a generic model, attributes set along with the work scope, to match the specific project; and simulated. This *"plug and play"* approach was discussed in Ekyalimpa, AbouRizk, Mohamed, and Saba (2014).

The simulation model developed in this thesis utilized an approach that involved constraining the processing of project by the availability of resources. These resources were all pooled at an industry level and requested by companies whenever they required them. In the developed simulation model, the resource quantities are static and don't change over time. This was an assumption made to simplify the modeling process. This assumption presents a lucrative opportunity for further development work in this area. This is because in practice, the resource quantities fluctuate over time as new trades, craftsmen and professionals enter the industry and others abandon or leave the industry for numerous reasons. One way of accomplishing this embellishment to the model could be through the use of a System Dynamics approach that involves modeling the resource pool as a stock and the rates of entry and exit of workers into and from the industry, as flow rates. These flow rates could also be made to be affected by other variables that emulate the different conditions within the industry.

In a similar way, the number of construction companies operating within the virtual industry created in the model is static. This does not accurately represent reality. The model can be embellished to represent a dynamic number of companies operating within the construction industry. A System Dynamics approach, similar to that suggested for modeling resource variations at an industry level, is also proposed for modeling this phenomenon.

Another recommendation is in-line with the resource definitions for companies. In real practice, construction companies temporarily hire their workers on a project by project basis from hiring halls. These workers in the hiring halls are available for hire to any company operating within that construction industry. There are some exceptions to this hiring practice. Some companies will possess two streams of workers – those that they permanently employ and those that they employ from the hiring halls. Competitors were assumed not to have access to the former category of resources for the time that they are engaged by a company. The categories of workers continuously retained at a company were not provided for in the model that was developed in this thesis. This concept needs to be considered in follow-up studies.

Also, the resources available at the industry level for the developed simulation application were defined by the analyst prior to simulation. During simulation execution, these quantities don't change. This assumption was made to simplify the modeling process and confine the scope of the study. In reality, resources, especially construction workers, exhibit mobility. They move from trade to trade or may decide to leave the industry. Also, there are new entrants into the industry and specific trades. These dynamics can be included as embellishments in follow-up studies. A few insights are given here on how to implement such an embellishment using a combined system dynamics – agent-based modeling approach. First, information about the rates of new worker arrivals and worker departures should be gathered. These arrivals and departures could then be mapped directly onto flow rates that are linked to a stock which represents the resource pool. The resource quantities could be modified on the fly to make their numbers match the values in the stock by either adding to or removing from the resource agent population.

There are prospects of the developed application being embellished for use as a training tool i.e., as a simulation game. Once this is done it can be used for training practitioners in industry on effective way of managing performance related issues at their companies. The game could also be used in Universities for training students about performance issues that relate to contractor companies operating in the construction industry.

The pursuit of data-driven validation of the different components of the developed application is also recommended in follow-up studies. Some aspects of this validation process would require the commitment of several contractor companies (for the collection of a relevant data set) operating in a select construction industry or sub-sector of the industry. This commitment would have to be sustained over a great length of time that could easily turn out to be several years.

REFERENCES

AbouRizk, H. (2013). Understanding and Improving Input for Quantitative Risk Analysis in the Construction Industry. *MSc Thesis Report*. Alberta, Canada: University of Alberta.

AbouRizk, S. M. (1993). Stochastic Simulation of Construction Bidding and Project Management. *Microcomputers in Civil Engineering, 8*(2), 343-353.

AbouRizk, S. M. (2011). Enhancing the competitiveness of the construction industry in Alberta. *IRC renewal application term 4 (2012-2016)*.

AbouRizk, S. M., & Hague, S. (2009). An Overview of the COSYE Environment for Construction Simulation. *Proceedings of the 2009 Winter Simulation Conference* (pp. 2624-2634). Austin, Texas: IEEE.

AbouRizk, S., Hague, S., Mohamed, Y., & Robinson, F. A. (2010). An HLA-based Bidding Game with Intelligent Virtual Player. *Proceedings of the Winter Simulation Conference* (pp. 3056-3068). Baltimore, Maryland, USA: IEEE.

AbouRizk, S., Mohamed, Y., Taghaddos, H., Saba, F., & Hague, S. (2010). Developing Complex Distributed Simulation For Industrial Plant Construction Using High Level Architecture. *Winter Simulation Conference*, (pp. 3177-3188).

AbouRizk, S., Mohamed, Y., Taghaddos, H., Saba, F., & Hague, S. (2010). Developing Complex Distributed Simulation For Industrial Plant Construction Using High Level Architecture. *Winter Simulation Conference*, (pp. 3177-3188).

Aguinis, H. (2011). Performance Management. Retrieved September 19, 2013: https://www.ebsglobal.net/documents/course-tasters/english/pdf/h17pe-bk-taster.pdf

Ahn, S., & Lee, S. (2014). Methodology for Creating Empirically Supported Agent-Based Simulation with Survey Data for Studying Group Behavior of Construction Workers. *Journal of Construction Engineering and Management, 0*(0).

Ahn, S., Lee, S., & Steel, R. (2013). Construction Workers' Perception and Attitudes Towards Social Norms as Predictors of Their Absence Behavior. *Journal of Construction Engineering and Management, 140*(5).

Ahn, S., Lee, S., & Steel, R. (2013). Effects of Worker's Social Learning:Focusing on Absence Behavior. *Journal of Construction Engineering and Management*, 1015-1025.

Alberta Finance and Enterprise. (2010). Alberta Economic Quick Facts. *Government of Alberta: Alberta Finance and Enterprise, Edmonton, Alberta*.

Alberta Government. (2011). An Outstanding Place for Tourism Investment, Albert, Canada. Accessed on 3rd September 2012: http://tpr.alberta.ca/tourism/programs-and-services/investment/international/pdf/AlbertaOverview.pdf

Aldington Report (1985). Report from the Select Committee of the House of Lords on Overseas Trade. *London: HMSO*.

Al-Hussein, M., Niaz, M. A., Yu, H., & Kim, H. (2006). Integrating 3D Visualization and Simulation for Tower Crane Operations on Construction Sites. *Automation in Construction, 15*, 554-562.

Al-Qirem, R. M., & Yaseen, S. G. (2010). Modeling a Small Firm in Jordan Using System Dynamics. In R. M. Al-Qirem, & S. G. Yaseen, *Discrete Event Simulation Environments: Technologies and Applications* (pp. 484-508).

Al-Qirem, R., & Yaseen, S. G. (2010). Modeling a Small Firm in Jordan Using System Dynamics. *IGI Global*.

Anastasi, A., & Urbina, S. (1997). *Psychological Testing*. Upper Saddle River, New Jersey: Prentice Hall.

Anderson, E. C. (1999). Lecture Notes for Stat 578C - Statictical Genetics. Retrieved from Monte Carlo Methods and Importance Sampling: http://ib.berkeley.edu/labs/slatkin/eriq/classes/guest_lect/mc_lecture_notes.pdf.

Anderson, W. J. (1991). Continuous-time Markov chains: An applications-oriented approach. *Springer Series in Statistics: Probability and its Applications. Springer-Verlag, New York*.

Arditi, D., Koksal, A., & Kale, S. (2000). Business Failures in the Construction Industry. *Engineering, Construction and Architectural Management, 7*(2), 120-132.

Armstrong, M. (2006). *Performance Management: Key Strategies and Practical Guidelines.* London: Kogan and Page.

Athanassopoulos, A., & Curram, S. (1996). A Comparison of Data Envelopment Analysis and Artificial Neural Networks as Tools for Assessing the Efficiency of Decision Making Units. *Journal of the Operational Research Soceity, 47*, 1000-1016.

Aubrey, D. (2004). *Performance Management:Changing Behavior that Drives Organizational Effectiveness.*

Awad, A., & Fayek, A. R. (2010). Developing a Framework for Construction Contractor Qualification for Surety Bonding. *ASCE, Construction Research Congress* (pp. 899-908). Banff, Alberta, Canada.

Axtell, R. L., & Epstein, J. L. (1994). Agent-Based Modeling:Understanding Our Creations. *The Bulletin of the Santa Fe Institute*, pp. 28-32.

Azimi, R. (2011). An Integrated Real-Time Control System for Structural Steel Fabrication Projects. *PhD Thesis Report*. Edmonton, Alberta: University of Alberta Libraries.

Azimi, R., Lee, S., & AbouRizk, S. (2011). Intelligent Look-Ahead Scheduling for Structural Steel Fabrication Projects. *4*(1), 207-217.

Azimi, R., Lee, S., AbouRizk, S., & Alvanchi, A. (2011). A Framework for Automated and Integrated Project Monitoring and Control System for Steel Fabrication Projects. *Automation in Construction, 20*(1), 88-97.

Babulak, E., & Wang, M. (2010). Discrete Event Simulation: State of the Art, Discrete Event Simulations. *Rijeka, Croatia: InTech*.

Banker, R., Charnes, A., & Cooper, W. (1984). Some Models for Estimating Technical and Scale Inefficiencies in Data Envelopment Analysis. *Management Science, 30*, 1078-1092.

Banks, J., Carson, J. S., Barry, N. L., Nicol, D. M. (2001). Discrete-Event System Simulation. *3rd Edition, New Jersey: Prentice* Hall.

Barros, C., & Alves, C. (2003). Hypermarket Retail Store Efficiency in Portugal. *International Journal of Retail and Distribution Management, 31*, 549-560.

Bassioni, H. A., Price, A. D., & Hassan, T. M. (2004). Performance Management in Construction. *Journal of Management in Construction, 20*(2), 42-50.

Behdani, B., Lukszo, Z., Adhitya, A., and Srinivasan, R. (2009). "Agent-Based Modeling to Support Operations Management in A Multi-Plant Enterprise." In *International Conference on Networking, Sensing and Control, ICNSC '09*, 323 – 328, Okayama, 10.1109/ICNSC.2009.4919295. Accessed August 12, 2011: http://www.nextgenerationinfrastructures.eu/download.php?field=document&itemID=515584.

Blunck, F. (2006). What is Competitiveness? Retrieved June 3, 2012: http://www.caps.am/data.php/877.pdf

Bonabeau, E. (2001). Agent-Based Modeling: Methods and Techniques for Simulating Human Systems. *National Academy of Sciences, 99*(3), 7280-7287.

Bond, A. H., & Gasser, L. (1988). Readings in Distributed Artificial Intelligence. San Mateo, California: Morgan Kaufmann.

Brown, J. D. (1996). Testing in Language Programs. Upper Saddle River, New Jersey: Prentice Hall Regents.

Carver, N., & Lesser, V. (1995). The DRESUN Testbed for Researcg in FA/C Distribution Situation Assessment: Extension to the Model of External Evidence. *1st International Conference on Multi-Agents Systems (ICMAS-95).* San Francisco, USA.

Carver, N., Cvetanovic, Z., & Lesser, V. (1991). Sophisticated Cooperation in Distributed Problem Solving. *9th National Conference on Artificial Intelligence*, (pp. 191-198). Anaheim.

Casti, J. (1997). Would be worlds: how simulation is changing the world of science, New York: Wiley.

Chaib-draa, B., Moulin, B., Mandiau, R., & Millot, P. (1992). Trends in Distributed Artificial Intelligence. *Artificial Intelligence Review 6*, pp. 35-66.

Charnes, A., Cooper, W., & Rhodes, E. (1978). Measuring the Efficiency of Decision Making Units. *European Journal of Operations Research, 2*, 429-444.

Chinneck, J. W. (2000). *Practical Optimization: A Gentle Introduction.* Ottawa, Canada: Carleton University.

Cho, B. G. (2003). The Application of the M&S in the KMSAM. *Proceedings of the 2nd Conference of the Weapon System Modeling and Simulation.* Korea.

Cho, B.-G., Kim, S.-H., & Youn, C. (2005). A Design and Application of HLA-Based Air Defense Simulation Framework. *Journal of Korea Information Science Society, 12D*(5), 709-718.

Clark, P., Ryan, P., and Zalcman, L., (2000). Advanced Distributed Simulation for the Australian Defense Force. *A Report published by Air Operations Division, Aeronautical and Maritime Research Laboratory*. Accessed August 15th 2014: http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA385312

Constructing Excellence in the Built Environment. (2012). KPI Management Tool: A Simple to Use On-Line Performance Measurement Collection and Reporting System. *Key Performance Indicators*. London, U.K.: Constructing Excellence in the Built Environment. Retrieved September 04, 2014, from http://www.constructingexcellence.org.uk/pdf/kpizone/KPI_Management_Tool.pdf

Construction Excellence in the Built Environment. (2012). KPI Management Tool: A Simple to Use On-Line Performance Measurement Collection and Reporting System. *Key Performance Indicators*. London, U.K.: Construction Excellence in the Built Environment. Retrieved September 04, 2014, from http://www.constructingexcellence.org.uk/pdf/kpizone/KPI_Management_Tool.pdf.

Construction Industry Institute. (2003). *Benchmarking and Metrics Value of Best Practices Report.* Austin, Texas.

Construction Users Roundtable. (2005). Construction Measures: Key Performance Indicators. Accessed on September 04, 2014: http://www.nwoglca.org/PDF_Files/Construction%20Measures_Key%20Performance%20Indicators.pdf.

Conway, S. R. (2006). An Agent Based Model for Analyzing Control Policies and the Dynamic Service Time Performance of a Capacity Constrained Air Traffic Management Facility, ICAS 2006. *25th Congress of the International Council of the Aeronautical Sciences* Hamburg, Germany, 3-8 Sep. 2006, Available online at: http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20060048296_2006250468.pdf.

Cooke, D. (2001). The Real Success Factors on Projects. *Internation Journal of Project Management, 20*(3), 185-190.

Cox, K. (1998). A Framewok-Based Approach to HLA Federate Development. *A Report from a Simulation Interoperability Workshop*.

Cox, R. F., Issa, R. A., & Ahrens, D. (2003). Management's Perception of Key Performance Indicators for Construction. *Journal of Construction Engineering and Management, 129*(2), 142-151.

Cristobal, J. R. (2012). Contractor Selection Using Multi-Criteria Decision-Making Methods. *Journal of Construction Engineering and Management, 138*(6), 751-758.

Cronbach, L. J., & Meehl, P. E. (1955). Construct Validity in Psychological Tests. *Psychological Bulletin 52*, pp. 281-302.

Cullinane, K., Wang, T., Song, D., & Ji, P. (2006). The Technical Efficiency of Container Ports: Comparing Data Envelopment Analysis and Stochastic Frontier Analysis. *Transportation Research Part A*, 354-374.

Cummins, J., & Rubio-Misas, M. (2006). Deregulation, Consolidation, and Efficiency: Evidence from the Spanish Insurance Industry. *Journal of Money, Credit and Banking, 38*, 323-355.

Dahmann, S. J., Fujimoto, M. R., & Weatherly, M. R. (1997). The Department of Defense High Level Architecture. *Proceedings of the 1997 Winter Simulation Conference.* Dec. 7–10, Atlanta, Georgia, USA.

Davis, R., & Smith, R. G. (1983). Negotiation as a Metaphor for Distributed Problem Solving. *Artificial Intelligence, 20*, 63-109.

D'Cruz, J., & Rugman, A. (1992). *New Compacts for Canadian Competitiveness.* Kodak Canada.

Decker, K. S. (1995). Distributed Artificial Intelligence Testbeds:Foundations of Distributed Artificial Intelligence. London: Wiley.

Decker, K. S., & Lesser, V. R. (1993). Designing a Family of Coordination Algorithms. *11th National Conference on Artificial Intelligence*, (pp. 217-224). Washington.

Department of Trade and Industry (DTI). (1998). The 1998 Competitiveness White Paper. Accessed 15 March 2010: http: //www.dti.gov.uk/comp/competitive/main.htm.

Dess, G. G., & Robinson, R. B. (1984). Measuring Organizational Performance in the Absence of Objective Measures: The Case of the Privately-held Firm and Conglomerate Business Unit. *Strategic Management Journal, 5*(3), 265-273.

Donella, M. (2008). Thinking in Systems:A Primer. London, U.K.: Earthscan.

Doran, J., Carvajal, H., Choo, Y., & Li, Y. (1991). The MCS Multi-agent Testbed: Developments and Experiments. *Springer-Verlag: Cooperate Knowledge based Systems*.

Dozzi, S. P., AbouRizk, S. M., & Schroeder, S. L. (1996). Utility-Theory Model for Bid Markup Decisions. *Journal of Construction Engineering and Management-ASCE, 122*(2), 119-124.

Du, J., & El-Gafy, M. (2012). Virtual Organization Imitation for Construction Enterprises: Agent-Based Simulation Framework for Exploring Human and Organizational Implications in Construction Management. *Journal of Computing in Civil Engineering, 26*(3), 282-297.

Dun and Bradstreet. (1989-93). U.S. Business Failure Records. New York: Dun and Bradstreet Corporation.

Durfee, E. H., & Montogomery, T. A. (1989). MICE: A Flexible Testbed for Intelligent Coordination Experiments. *1989 Distributed Artificial Intelligence Workshop*, pp. 25-40.

EFQM. (2012). An Overview of the EFQM Excellence Model. Accessed on July 18, 2014: http://www.efqm.org/sites/default/files/overview_efqm_2013_v1.pdf

EFQM. (2014). The EFQM Excellence Model. Accessed on July 15, 2014: http://www.efqm.org/the-efqm-excellence-model

Egan, J. (1998). Rethinking Construction. *London, U.K.: Department of Environment, Transportation and the Regions*.

Ekyalimpa, R., & AbouRizk, S. (2014). A Multi-Agent Model that Emulates the Typical Competitive Bidding Process in the Construction Industry. *The 13th International Conference on Modeling and Applied Simulation.* Bordeaux, France.

Ekyalimpa, R., & Fayyad, S. (2010). Administrator Federate Module in the Crane Lift Planning Game. *CIVE 709 Advanced Simulation Course Project Report*. Edmonon, Alberta, Canada.

Ekyalimpa, R., AbouRizk, S., & Farrar, J. (2012). Effective Strategies for Simulating One-Of-A-Kind Construction Projects. *Proceedings of the 2012 Winter Simulation Conference* (pp. 1-13). Dec. 9-12, Berlin, Germany: IEEE.

Ekyalimpa, R., AbouRizk, S., Mohamed, Y., & Saba, F. (2014). A Prototype for Project Management Game Development Using High Level Architecture. *International Journal of Simulation and Process Modeling, 9(3),* 131-145.

Ekyalimpa, R., Al-Jibouri, S., Mohamed, Y., & AbouRizk, S. (2011). Design of A Tunnel Simulation Game for Teaching Project Control in Construction. *Proceedings of Construction Research Congress Conference* (pp. 2118-2128). Ottawa, Ontario, Canada: CSCE.

Elliot, W., & Moscardini. (1994). Using a System Dynamics Approach as a Tool for Enhancing Company Performance through Developing Imagination Models for Managers. *1994 International System Dynamics Conference*, (pp. 205-213).

Evans, E. G. (2004). Performance Management and Appraisal:A How To Do It Manual for Librarians. New York: Neal-Schuman.

Flanagan, L. R., Shen, L., & Jewell, C. (2007). Competitiveness in Construction: A Critical Review of Research. *Construction Management and Economics, 25*(9), 989-1000.

Flegg, A., Allen, D., Field, K., & Thurlow, T. (2004). Measuring the Efficiencies of British Universities: A Multi-Period Data Envelopment Analysis. *Education Economics, 12*, 231-249.

Folcik, V.A., & Orosz, C. G. (2007). The Basic Immune Simulator: An Agent Based Model to Study the Interactions between Innate and Adaptive Immunity. *Theoretical Biology and Medical Modelling,* 4(1):39-56.

Forrester, J. (1971). Counterintuitive Behavior of Social Systems. *Technology Review, 73*(3), 52-68.

Gardner, M. (1970). Mathematics Games - The Fantastic Combinations of John Conway's New Solitaire Game "Life". *Sceintific American 223*, pp. 120-123.

Gasser, L. (1991). Social Conceptions of Knowledge and Action:DAI Foundations and Open Systems. *Artificial Intelligence, 47*(1), 107-138.

Gasser, L., & Huhns, M. (1989). Distributed Artificial Intelligence2. San Mateo, California: Morgan Kaufman.

Gasser, L., Braganza, C., & Herman, N. (1987). "MACE: A Flexible Testbed for Distributed AI Research". London: Pitman.

Ghaffari, A., & Jain, R. K. (2013). Developing a Model for Profit Sharing amongst Partners in Consulting Engineering Companies of Construction Projects. *Advances in Civil Engineering and Building Materials*.

Gonzales, C., Hu, D., & Mogadam, M. (2010). Operations federate module in the Crane Lift Planning Game. *An internal report for Advanced Simulation course (Cive. 709), advanced topics in Construction Engineering and Management, University of Alberta, Canada*.

GPS USA Division. (2014). Contractors: Contribution to the U.S. GDP. Buffalo Grove, Illinois, USA. Accessed on June 2, 2014: http://www.gps-business.net/industries/contractors-contribution-to-the-u-s-gdp/

Griffin, A. F. & Stanish, C. (2007). An Agent Based Model of Prehistoric Settlement Patterns and Political Consolidation in the Lake Titicaca Basin of Peru and Bolivia, Structure and Dynamics. *e-Journal of Anthropological and Related Sciences*, 2(2). Available online at: http://repositories.cdlib.org/imbs/socdyn/sdeas/vol2/iss2/art2

Grosman, P. D., Jaeger, J. A. G., Biron, P. M., Dussault, D., & Ouellet, J. P. (2011). "Trade-off between Road Avoidance and Attraction by Roadside Salt Pools in Moose: An Agent-Based Model to Assess Measures for Reducing Moose-Vehicle Collisions." *Ecological Modelling* 222(8):1423-1435.

Grünberg , T. (2004). Performance Improvement: Towards a Method for Finding and Prioritizing Potential Performance Improvement Areas In Manufacturing Operations. *International Journal of Productivity Management, 53*(1), 52-71.

Haas, D., Kocher, M., & Sutter, M. (2004). Measuring the Efficiency of German Football Teams by Data Envelopment Analysis. *Central European Journal of Operational Research, 12*, 251-268.

Hajjar, D., & AbouRizk, S. M. (1999). Simphony:An Environment for Building Special Purpose Construction Simulation Tools. *Proceedings of the 1999 Winter Simulation Conference* (pp. 998-1006). Phoenix: IEEE.

Hajjar, D., & AbouRizk, S. M. (2002). Unified Modeling Methodology for Construction Simulation. *Journal of Construction Engineering and Management, 128*(2), 174-185.

Herzberg, F. (1959). The Motivation to Work. New York: John Wiley and Sons.

Hewitt, C. (1977). Viewing Control Structures as Patterns of Passing Messages. *Artificial Intelligence, 8*(3), 323-364.

Hoffman, P. (1998). The Man who Loved Only Numbers:The Story of Paul Erdos and the Search for Mathematical Truth. New York, USA: Hyperion.

Huprich, J. (2008). A Brief Introduction to Performance Management. Accessed on June 21, 2014: http://ala-apa.org/newsletter/2008/06/15/a-brief-introduction-to-performance-management/

Hursman, A. (2010). Measure What Matters. Accessed on June 26, 2014: http://web.ebscohost.com/ehost/pdfviewer/pdfviewer?vid=48&hid=11&sid=a294ed98

Investopedia. (2014). Key Performance Indicators - KPI. Accessed on June 26, 2014: http://www.investopedia.com/terms/k/kpi.asp

Iozzi F., Trusiano, F. Chinazzi, Billari, M., Zagheni, F. C. E., Merler, S., Ajelli, M., Del Fava, E., & Manfredi, P. (2010). "Little Italy: An Agent-Based Approach to the Estimation of Contact Patterns – Fitting Predicted Matrices to Serological Data." *PLoS Comput. Biol.* 6(12): e1001021. doi:10.1371/journal.pcbi.1001021. Accessed on August 12, 2011: http://openaire.cern.ch/record/773/files/journal.pcbi.1001021.pdf.

Iveta, G. (2012). Human Resources Key Performance Indicators. *Journal of Competitiveness, 4*(1), 117-128.

Jangmi, H., Zhang, H., & Saba, F. (2010). Player Federate Module in the Crane Lift Planning Game. *An internal report for Advanced Simulation course (Cive. 709), advanced topics in Construction Engineering and Management, University of Alberta, Canada*.

Jaselskis, E. J., & Ashley, D. B. (1991). Optimial Allocation of Project Management Resources for Achieving Success. *Journal of Construction Engineering and Management, 117*(2), 321-340.

Jennings, N. R. (2000). On agent based software engineering, *Artificial Intelligence, 117:277-296.*

Jennings, N., Corera, J. M., Laresgoiti, L., Mamdani, E., Perriollat, F., Skarek, P., & Varga, L. (1995). Using ARCHON to Develop Real-World DAI Applications for Electricity Transportation and Particle Accelerator Control. *IEEE Expert, Special Issue on Real World Applications of DAI Systems*.

Kagioglou, M., Cooper, R., & Aouad, G. (2001). Performance Management in Construction: A Conceptual Framework. *Construction Management and Economics, 19*(1), 85-95.

Kale, S., & Arditi, D. (2002). Competitive Positioning in United States Construction Industry. *Journal of Construction Engineering and Management, 128*(3), 238-247.

Kale, S., & Arditi, D. (2003). Differentiation, Conformity, and Construction Firm Performance. *Journal of Management in Engineering, 19*(2), 52-59.

Kamat, V. (2000). Enabling 3D Visualization of Simulated Construction. *An MSc. Thesis Report, University of Virginia, Virginia, USA.*

Kaplan, R. S., & Norton, D. P. (2007). Using the Balanced Scorecard as a Strategic Management System. *Harvard Business Review*. Accessed on 20th August 2014: https://hbr.org/2007/07/using-the-balanced-scorecard-as-a-strategic-management-system

Kim, D. S. (2002). A Design and Implementation of ROM Framework for Developing HLA Federates. *Journal of Korea Information Science Society*, 1137-1144.

Kim, J.-H., Hong, S.-Y., & Kim, T. G. (2006). Design and implementation of simulators interoperation layer for DEVS simulator. *Proceedings of the 2006 SCS International Conference on Modeling and Simulation - Methodology, Tools, Software Applications*, pp. 95-100.

Kuhl, F., Weatherly, R., & Dahmann, J. (1999). Creating Computer Simulation Systems: An Introduction to the High Level Architecture. New Jersey: Prentice Hall International.

Kuhn Jr, J. R., Courtney, J. F., Morris, B., & Tatara, E. R., (2010). “Agent-Based Analysis and Simulation of the Consumer Airline Market Share for Frontier Airlines.” *Knowledge-Based Systems,* 23(8): 875-882.

Lansley, P. R. (1985). Putting Organizational Research into Practice. *Construction Management and Economics,* 3: 1-14.

Lansley, P. R. (1986). Modeling Construction Organizations. *Construction Management and Economics,* 4: 19-36.

Latham, M. (1994). Constructing the Team. London, U.K.: HMSO.

Lee, A., Cooper, R., & Aouad, G. (2000). A Methodology for Designing Performance Measures for the UK Construction Industry. *A Paper Presented at Bizarre Fruit Postgraduate Research Conference on the Buillt and Human Environment*. Salford, UK.

Leicht, R. M., Hunter, S. T., Saluja, C., & Messner, J. I. (2009). Implementing Observational Research Methods to Study Team Performance in Construction Management. *Journal of Construction Engineering and Management*, 1943-7862.

Lesser, V., & Corkill, D. (1981). Functionally Accurate, Cooperative Distributed Systems. *IEEE Transactions on Systems, Man and Cybernetics, C-11*(1): 81-96.

Levitt, R., Cohen, P., Kunz, J., Nass, C., Christiansen, T., & Jin, Y. (1994). The Virtual Design Team:Simulating how Organisational Structure and Communication Tools affect Team Performance. San Francisco: Lawrence Erlbaum.

Leyk S, Binder CR, Nuckols JR. (2009). Spatial modeling of personalized exposure dynamics: The case of pesticide use in small-scale agricultural production landscapes of the developing world. *International Journal of Health Geographic*, 8(17): 1-16.

Lin, G. B., & Shen, Q. P. (2007). Measuring the Performance of Value Management Studies in Construction: A Critical Review. *Journal of Management in Engineering, 23*(1), 2-9.

Liu, Y., & Mohamed, Y. (2012). Modeling industrial construction operations using a multi-agent resource allocation framework. *Engineering, Construction and Architectural Management, 19*(4), 406-427.

Liyin, L. S., & Yam, C. H. (2006). Contractor Key Competitiveness Indicators (KCIs): A China Study. *Journal of Construction Engineering and Management, ASCE*, 416-424.

Lu, W. (2006). A System for Assessing and Communicating Contractors' Competitiveness. Hong Kong Polytechnic University, China.

Lu, W., Shen, L., & Yam, M. C. (2008). Critical Success Factors for Competitiveness of Contractors: China Study. *Journal of Construction Engineering and Management, 134*(12): 972-982.

Lucko, G., & Rojas, E. M. (2009). Research Validation: Challenges and Opportunities in the Construction Domain. *Journal of Construction Engineering and Management*, 1943-7862.

Luis, R., Serge, S.-D., Pastrana, J., Marin, M., Bhide, S., Joledo, O., & Bardina, J. (2013). Simulation Modeling of Space Missions Using the High Level Architecture. *Modeling and Simulation in Engineering*.

Luo, X., & Donthu, N. (2005). Assessing Advertising Media Spending Inefficiencies in Generating Sales. *Journal of Business Research,* 58: 28-36.

Macal, C. M., & North, M. J. (2005). Tutorial on Agent-based Modeling and Simulation. *Winter Simulation Conference*, Dec. 4-7, Hilton Downtown Disney Lake Buena Vista, Florida, USA: 2-15.

Macal, C. M., & North, M. J. (2011). Introductory Tutorial:Agent-Based Modeling and Simulation. *Winter Simulation Conference*, Dec. 11–14. Grand Arizona Resort Phoenix, AZ, USA: 1456-1469.

Malcolm, D. G., Roseboom, J. H., Clark, C. E., & Fazar, W. (1958). Application of a Technique for Research and Development Program Evaluation. *Operations Research, 7*(5): 646-669.

Malleson, N. (2010). Agent-Based Modelling of Burglary. *PhD Thesis, School of Geography, University of Leeds*.

Marsh, K., & Fayek, A. R. (2009). Development of a Fuzzy Expert System for Surety Underwriting. *ASCE, Construction Research Congress.* Washington, Seattle: ASCE: 219-228.

Martinez, J. C. (2009). Methodology for Conducting Discrete-Event Simulation Studies in Construction Engineering and Management. *Journal of Construction Engineering and Management*: 1943-7862.

Maslow, A.H. (1943). A Theory of Human Motivation. *Psychological Review* 50 (4): 370–96.

Mast, E. H. M., G. A. M. van Kuik, and G. J. W. Van Bussel. (2007). Agent-Based Modelling for Scenario Development of Offshore Wind Energy. Delft University of Technology, the Netherlands.

McCabe, B., Tran, V., & Ramani, J. (2005). Construction Prequalification Using Data Envelopment Analysis. *Canadian Journal of Civil Engineering, 32*(1): 183-193.

McCabe, B., Tran, V., & Ramani, J. (2005). Construction Prequalification Using Data Envelopment Analysis. *Canadian Journal of Civil Engineering, 32*(1): 183-193.

Menges, F., Mishra, B., & Narzisi, G. (2008). “Modeling and Simulation of E-Mail Social Networks: A New Stochastic Agent-Based Approach.” In *Proceedings of the 2008 Winter Simulation Conference*. Dec. 7-10, Miami, Florida, USA.

MESA International, & Cambashi Inc. (2012). Pursuit of Performance Excellence: Business Success through Effective Plant Operations Metrics. Chandler, Arizona, USA.

Miller, S., & Noulas, A. (1996). The Technical Efficiency of Large Bank Production. *Journal of Banking and Finance,* 20: 495-509.

Mock, K. J., & Testa, J. W. (2007). An Agent-Based Model of Predator-Prey Relationships between Transient Killer Whales and other Marine Mammals. University of Alaska Anchorage, Anchorage, AK, May 31, 2007.

Moghani, E. (2013). A Simulation-Based Framework for As-Built Documentation of Construction Process and Product Information. *PhD Thesis, University of Alberta*.

Moller, B., Antelius, F., Johansson, M., Lofstrand, B., & Wihlborg, A. (2010). Processes and Tools for Management and Reuse of FOM Modules. *Joint SISO/SCS European Multi-Conference.* Istanbul, Turkey.

Momaya, K. (1996). International Competitiveness of the Canadian Construction Industry: A Comparison with Japan and the US. Toronto, Canada: University of Toronto.

Momaya, K. (1998). Evaluating International Competitiveness at the Industry Level. *Vikalpa, 23*(2): 39-46.

Moscardini, A., Loutfi, M., & Al-Qirem, R. (2005). The Use of System Dynamics Models to Evaluate the Credit Worthiness of Firms. *The 23rd International Conference of the System Dynamics Society.* Boston, US.

Mostafa, M. M, (2007). "Evaluating the competitive market efficiency of top listed companies in Egypt"; *Journal of Economic Studies,* 34(5): 430 – 452.

Mostafa, M. M. (2009). Modeling the Competitiveness Market Efficiency of Egyptian Companies: A Probabilistic Neural Network Analysis. *Expert Systems with Applications, 36*: 8839-8848.

Mulva, S. P. (2011). CII's Benchmarking and Metrics Program: Measuring the Impact of Effective Construction and Program Management on Large-Scale Capital Infrastructure Projects. Orlando, Florida, USA. Accessed on October 6, 2013: http://www.tcm-conference.org/presentations/2011/Tuesday/Controlling_Constructing_Projects_Mulva.pdf

Murphy, K., & Cleveland, J. N. (1995). Understanding Performance Appraisal: Social, Organization, and Goal-Based Perspective. New York: Sage Publications.

Namho, K., Hyun-Soo, L., Moonseo, P., & Seungjun, R. (2007). Performance Management Method for Construction Companies. *24th International Symposium on Automation & Robotics in Construction (ISARC 2007); September 19-21* (pp. 523-528). IIT Madras: Construction Automation Group. Accessed on: August 2, 2014: http://www.iaarc.org/publications/fulltext/isarc2007-5.1_7_124.pdf

Nance, E. R. (1993). A History of Discrete Event Simulation Programming Languages. *ACM SIGPLAN Notices, 28*(3).

Neely, A. (1999). The Performance Revolution: Why Now and What Next? *International Journal of Operations & Production Management,* 19(2): 205-228.

Neely, A. (2002). *Business Performance Measurement:Theory and Practice.* Cambridge, UK: Cambridge University Press.

Neely, A., Gregory, M., & Platts, K. (1995). Performance Measurement System Design: A Literature Review and Research Agenda. *International Journal of Operations & Production Management,* 15(4): 80-116.

Niven, P. R. (2002). Balanced Scorecard Step-by-Step. New York: Wiley.

North, M., Macal, C., Aubin, J. St., Thimmapuram, P., Bragen, M., Hahn, J., Karr, J., Brigham, N., Lacy, M., and Hampton, D., (2009). "Multi-scale Agent-based Consumer Market Modeling." *Complexity* 15(5): 37-47.

Norton, D., & Kaplan, R. (1992). The Balanced Score Card: Measures that Drive Performance. *Harvard Business Review:* 70-79.

Nwana, H. (1996). Software Agents:An Overview. *Knowledge Engineering Review, 11*(3): 1-40.

OECD. (1997). Industrial Competitiveness: Benchmarking Business Environments. Paris, Francis: OECD.

Ofori, G. (1993). Managing Construction Industry Development. Singapore: Singapore University Press.

Ogunlana, O. S., Li, H., & Sukhera, F. A. (2003). System Dynamics Approach to Exploring Performance Enhancement in a Construction Organization. *Journal of Construction Engineering and Management,* 129(5): 528-536.

Ogunlana, S. O., Promkuntong, K., & Jearkjirm, V. (1996). Construction Delays in a Fast-growing Economy: Comparing Thailand with other Economies. *International Journal of Project Management,* 14(1): 37-45.

Ogunlana, S., Li, H., & Sukhera, F. (2003). System Dynamics Approach to Exploring Performance Enhancement in a Construction Organization. *Journal of Construction Engineering and Management,* 129(5): 528-536.

Opaluch, J. J., Anderson, J. L. and Schnier, K. (2005). "A Risk-Based Approach to Managing the Intentional Introduction of Non-Native Species." *Paper presented at the AERE Workshop Natural Resources at Risk*, June 12-14, Grand Teton National Park, Wyoming.

Organizational Heart Beats. (2014). Performance Management - A Historical Overview. Accessed on July 10, 2014: http://www.organisationalheartbeats.com/topicinfo/pm_info.aspx.

Orozco, F., Serpell, A., Molenaar, K., & Forcael, E. (2014). Modeling Competitiveness Factors and Indexes for Construction Companies: Findings of Chile. *Journal of Construction Engineering and Management, 140*(4): B4013002-1 - B4013002-13.

Page, E. H., and Roger, S. (1998). Introduction to Military Training Simulation: A Guide for Discrete Event Simulationists. *Proceedings of the 1998 Winter Simulation Conference;* Dec. 11-16, Washington, DC.

Phelps, A. F., & Horman, M. J. (2009). Ethnographic Theory-Building Research in Construction. *Journal of Construction Engineering and Management*, 1943-7862.

Phua, F. T. (2007). "Does Senior Executive's Perception of Environmental Uncertainity Affect the Strategic Functions of Construction Firms?". *International Journal of Project Management, 25*(8): 753-761.

Pilateris, P., & McCabe, B. (2003). “Contractor financial evaluation model (CFEM)” *Canadian Journal of Civil Engineering*, 30(3): 487-499.

Polit, D. F., & Beck, C. T. (2012). Nursing Research: Generating and Assessing Evidence for Nursing Practice. Philadelphia, USA: Wolters Klower Health, Lippincott Williams & Wilkins.

Porter, M. (1986). Competition in Global Industries. *Harvard Business School Press*.

Porter, M. (1990). The Competitive Advantage of Nations. *New York: The Free Press*.

Public Record Office Victoria. (2010). Guideline 3: Key Performance Indicators. Victoria: Public Record Office Victoria.

Puckett, R. R., (2009). Multi-Agent Crowd Behavior Simulation for Tsunami Evacuation. Master’s Thesis, Department of Information and Computer Sciences, University of Hawaii, Accessed on August 12, 2011: http://www2.hawaii.edu/~puckett/research/past/tsunami.pdf.

Raab, R., & Lichty, R. (2002). Identifying Sub-areas that Comprise a Greater Metropolitan Area: The Criterion of County Relative Efficiency. *Journal of Regional Science, 42*: 579-594.

Radzicki, M. J., & Taylor, R. A. (2008). Origin of System Dynamics: Jay W. Forrester and the History of System Dynamics. Washington DC: U.S. Department of Energy.

Rajive, J., & Gerardo-Pardo, C. (2006). A Comparison and Mapping of Data Distribution Service and High-Level Architecture. Real-Time Innovations, Inc.

Rohani, M., Fan, M., & Yu, C. (2013). Advanced Visualization and Simulation Techniques for Modern Construction Management. *Indoor and Built Environment, 0*(0): 1-10.

Rosenschein, J. S. (1985). Rational Interaction: Cooperation Among Intelligent Agents. *A PhD Thesis Report, Stanford University*.

Rosenschein, J. S., & Zlotkin, G. (1994). Rules of Encounter: Designing Conventions for Automated Negotiation among Computers. Cambridge: MIT Press.

Saaty, T. L. (1980). *The Analytic Hierarchy Process.* New York: McGraw-Hill.

Sargent, R. G. (1998). Verification and Validation of Simulation Models. *Proceedings of the 30th conference on winter simulation,* Dec. 11-16, Washington DC, USA: 121 – 130.

Sarkis, J. (200). An Analysis of the Operational Efficiency of Major Airports in the United States. *Journal of Operations Management, 18*: 335-351.

Scott, B. R., & Lodge, G. C. (1985). US Competitiveness in the World Economy. *Boston: Harvard Business School Press.*

Scott, B. R. (1985). *National Strategies: Key to International Competition: US Competitiveness in the World Economy.* Boston: Havard Business School Press.

Serrano-Cinca, C., Fuertes-Callen, Y., & Mar-Molinero, C. (2005). Measuring DEA Efficiency in Internet Companies. *Decision Support Systems*:557-573.

Shen, L. Y., Lu, W., Shen, Q., & Li, H. (2003). A Computer-aided Decision Support System for Assessing a Contractor's Competitivenes. *Automation in Construction, 12*(5): 577-587.

Sigala, M., Jones, P., Lockwood, A., & Airey, D. (2005). Productivity in Hotels: A Stepwise Data Envelopment Analysis of Hotels' Rooms Division Process. *The Service Industries Journal, 25:* 61-81.

Smith, M. (2001). Getting Construction Back on Track: Beyond the Bottom Line, the Industry in Developing Countries. *Proceedings of the First International Conference of CIB.* CIB.

Smith, M. (2002). Getting Construction Back on Track: Beyond the Bottom Line, The Industry in Developing Countries. *Proceedings of the First International Conference of CIB.*

Smith, P. (1997). Model Misspecification in Data Envelopment Analysis. *Annals of Operations Research,* 73: 233-252.

Smith, R. G. (1980). The Contract Net Protocol:High-Level Communication and Control in a Distributed Problem Solver. *IEEE Transactions on Computers, C29*(12).

Sopha, B. M., Klockner, C. A. and Hertwich, E. G. (2011). "Exploring Policy Options for a Transition to Sustainable Heating System Diffusion Using an Agent-Based Simulation." *Energy Policy* (in press).

Statistics Canada. (2014). Gross Domestic Product at Basic Prices, by Industry. Accessed on June 2, 2014: http://www.statcan.gc.ca/tables-tableaux/sum-som/l01/cst01/econ41-eng.htm

Sterman, J. D. (2000). Business Dynamics:Systems Thinking and Modeling for a Complex World. New York, U.S.: McGraw Hill.

Sterman, J. D. (2001). System Dynamics Modeling: Tools for Learning in a Complex World. *California Management Review, 43*(4): 8-25.

Taghaddos, H. (2010). Developing a Generic Resource Allocation Framework for Construction Simulation. *PhD Thesis Report, University of Alberta, Edmonton, Canada*.

Taghaddos, H., Hermann, U., AbouRizk, S., & Mohamed, Y. (2010). Simulation-Based Scheduling of Modular Construction Using Multi-agent Resource Allocation. *Second International Conference on Advances in System Simulation*, Nice, France: 115-120.

Takim, R., & Akintoye, A. (2002). Performance Indicators for Successful Construction Project Performance. *18th Annual ARCOM Conference.* Retrieved September 04, 2014, from http://www.arcom.ac.uk/-docs/proceedings/ar2002-545-555_Takim_and_Akintoye.pdf

Tan, Y., Shen, L., & Langston, C. (2012). Competition Environment, Strategy, and Performance in the Hong Kong Construction Industry. *Journal of Construction Engineering and Management, 138*(3): 352-360.

Tan, Y.-t., Shen, L.-Y., Yam, M. C., & Lo, A. A. (2007). Contractor Key Competitiveness Indicators (KCIs): A Hong Kong Study. *Surveying and Built Environment, 18*(2): 33-46.

Taylor, M. (2009, 04 25). What is Sensitivity Analysis? What is ...? series: Health Economics. London, UK: Hayward Group Ltd. Accessed on May 24, 2014: http://www.medicine.ox.ac.uk/bandolier/painres/download/whatis/What_is_sens_analy.pdf

The Construction Users Roundtable. (2005). Construction Measures: Key Performance Indicators. Accessed on September 04, 2014: http://www.nwoglca.org/PDF_Files/Construction%20Measures_Key%20Performance%20Indicators.pdf

The Institute of Electrical and Electronic Engineers, T. I. (2000). IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)-Federate Interface Specification. *HLA Standard*.

Tsola, I. E. (2011). Modeling Profitability and Effectiveness of Greek-Listed Construction Firms:An Integrated DEA and Ratio Analysis. *Construction Management and Economics, 29*(8): 795-807.

Tyson, L. D. (1992). Who's Bashing Whom? Trade Conflicts in High-Technology Industries. Washington D C: The Institute for International Economics.

U.S. Bureau of Economic Analysis. (2014). Gross-Domestic-Product (GDP) by Industry, Data. Accessed on June 2, 2014: http://bea.gov/industry/gdpbyind_data.htm

U.S. Defense Modeling and Simulation Office. (2001). RTI 1.3 - Next Generation Programmer's Guide Version 4. U.S. Department of Defense.

U.S. Department of Defense. (1994). Department of Defense Directive on Modeling and Simulation Management. Washington D.C.

U.S. Office of Personnel Management. (2008). Performance Management Overview & History. Accessed on June 20, 2014: http://www.opm.gov/policy-data-oversight/performance-management/overview-history

Usman, A. M., Mueller, W., Elsheikh, A., Palensky, P., & Widl, E. (2013). Using the HLA for Distributed Continuous Simulations. *8th EUROSIM Congress on Modelling and Simulation*. Cardiff, Wales, United Kingdom, Sept. 10-12 2013.

Van Dam, K. H., Lukszo, Z., Ferreira, L. & Sirikijpanichkul, A., (2007). Planning the Location of Intermodal Freight Hubs: An Agent Based Approach." In *Proceedings of the 2007 IEEE International Conference on Networking, Sensing and Contro*l, 15-17 April, London, UK: 187-192.

Velasco, H., Manuel, J., Pleguezuelo, H., & Van Dorp, J. R. (2011). Revisiting the PERT mean and Variance. *European Journal of Operations Research*: 448-451.

Wang, J., Gwebu, K., Shanker, M., and Troutt, M. D., (2009). An Application of Agent-Based Simulation to Knowledge Sharing. *Decision Support Systems* 46:532–541. Accessed on August 12, 2011: http://www.personal.kent.edu/~mshanker/personal/Zip_files/DECSUP11534.pdf.

Wang, M., & Sun, G. (1993). Manufacturing Simulation: An Effective Tool for Productivity Improvement. *3rd International Microelectronics & System '93 Conference.* Malaysia.

Wang, S. (2003). Adaptive Non-Parametric Efficiency Frontier Analysis: A Neural Network-Based Model. *Computers and Operations Research, 30*: 279-295.

Weisstein, E. (2014). MathWorld. Accessed September 02 2014: http://mathworld.wolfram.com/topics/CellularAutomata.html

Winston, W. L. (2000). *Simulation Modeling Using @Risk.* Cengage Learning.

Wittig, T. (1992). *An Architecture for Multi-Agent Systems.* London: Ellis Horwood.

Wong, J., & Thomas, S. (2010). Business Failure in the Construction Industry: A Critical Review and a Future Research Agenda. *FIG Congress 2010.* Sydney, Australia.

Wooldridge, M., & Jennings, N. R. (1995). Intelligent Agents: Theory and Practice. *Knowledge Engineering Review, 10*(2): 115 - 152.

World Economic Forum. (2009). The Global Competitiveness Report 2009-2010. Geneva, Switzerland.

WRC. (1989). The World Competitiveness Report 1989. Switzerland: IMEDE and the World Economic Forum.

WRC. (1991). The World Competitiveness Report 1991. Switzerland: IMEDE and the World Economic Forum.

Xie, H. (2011). Improving Dynamic Project Control in Tunnel Construction. *PhD Thesis, University of Alberta, Edmonton.*

XJ Technologies. (2013). Simulation Modeling with AnyLogic: Agent Based, Discrete Event and System Dynamics Methods.

Yang, H., Yeung, J. F., Chan, A. P., Chiang, Y. H., & Chan, D. W. (2010). A Critical Review of Performance Measurement in Construction. *Journal of Facilities Management, 8*(4): 269-284.

Yates, J. K. (1994). Construction Competition and Competitive Strategies. *Journal of Management in Engineering, 10*(1): 58-69.

Zlotkin, G., & Rosenschein, J. S. (1989). Negotiation and Task Sharing among Autonomous Agents in Cooperative Domains. *11th IJCAI*, Detroit, Michigan: 912-917.

APPENDIX

APPENDIX A1 – Questionnaire for the Performance Management Best Practices' Survey

SURVEY QUESTIONNAIRE

An Investigation into Performance Management Best Practices at Select General Contractors within the Alberta Construction Industry

COMPANY GENERAL INFORMATION

1.0 **Company Experience:** *In which year was your company founded?*

2.0 **Business Operations:** *Indicate the domain in which your company operates. (Please check beside applicable.)*

Industrial	
Heavy Civil	
Other (please specify)	

3.0 **Annual Workload:** *Indicate your typical annual workload volume.*

Work Load Range (Million Dollars)	Please Check Beside Applicable
0 – 50	
50 - 250	
250 - 1000	
Over 1000	

PERFORMANCE MANAGEMENT SYSTEMS

4.0 **Benchmarking Frequency Rate:** *How often do you appraise your performance? (Please check beside applicable.)*

Not Applicable	
Monthly	
Quarterly	
Bi-annually	
Annually	
Other (please specify)	

5.0 **Type of Benchmarking:** *What type of benchmarking does your company engage in? (Please check beside applicable.)*

Company level (internal) benchmarking	
Industry level (competitive) benchmarking	
Inter-Industry (generic) benchmarking	
None	
Other(s) (please list)	

6.0 **Involvement in Benchmarking:** *State your involvement in benchmarking groups/clubs. (Please check beside applicable.)*

Provide data used for benchmarking	
Subscribe to have access to benchmarking data	
None	
Other(s) (please list)	

7.0 **Benchmarking Clubs:** *Which benchmarking clubs are you involved in? (Please check beside applicable.)*

Construction Industry Institute (CII)	
Construction Owners Association of Alberta (COAA)	
Center for Construction Innovation - UK (CCI)	
Independent Project Analysis (IPA)	
Other(s) (please list)	

8.0 **Other Performance Management Systems:** *Indicate other performance management systems used in your company. (Please check beside applicable.)*

Balanced Score Card	
Excellence model(s)	
Simulation-based systems	
Key Performance Indicators (KPIs)	
None	
Other(s) (please list)	

9.0 Pitfalls of Performance Management Systems: *Indicate problems encountered with using systems in 6.0 and 8.0. (Please check beside applicable.)*

Not holistic (Don't capture all key performance drivers in a homogenous fashion)	☐
Difficult to automate (hence slow, tedious & error prone)	☐
Static (not dynamic)	☐
Lagging & don't provide warning of likely problems to be encountered ahead	☐
Don't support what-if experimentation of possible improvement strategies to arrive at the best option	☐
Other(s) (please list)	

PERFORMANCE MEASURES

10.0 Performance Measurement: *Indicate how performance measures are tracked in your company.*

Performance Measure	Unit of Measure	Please indicate if used: Yes/No	Other method of Measurement (please list)
Tendering Success/Market Share	% projects acquired of those tendered	○ ○	
	% $ value of projects acquired of those tendered	○ ○	
Productivity	$/Unit	○ ○	
	Mhrs/Unit	○ ○	
	Performance Factor	○ ○	
Cost Slippage	% overrun or % under run	○ ○	
Schedule Slippage	% overrun or % under run	○ ○	
Quality	% Conformance Rating	○ ○	
	Costs associated with rework	○ ○	
Safety	Insurance Premiums	○ ○	
	Excellence awards e.g. Certificate of Recognition (COR)	○ ○	
	Lost Time Injury Frequency Rate (LTIFR)	○ ○	
	Lost Time Injury Incident Rate (LTIIR)	○ ○	
	Severity Rates	○ ○	
Others (1) (please list)	(please list)		
Others (2) (please list)	(please list)		
Others (3) (please list)	(please list)		

APPENDIX A1 – Information and Consent Letter for the Performance Management Best Practices' Survey

UNIVERSITY OF ALBERTA

Department of Civil and Environmental Engineering
School of Mining and Petroleum Engineering

3-133 Markin/CNRL Natural Resources Engineering Facility
Edmonton, Alberta, Canada T6G 2W2
www.engineering.ualberta.ca/civil
Tel: 780.492.4235
Fax: 780.492.0249

December, 13, 2013

RE: Invitation to Participate in a Research Study on Performance Management Best Practices of Heavy Civil and Industrial Construction Contractors in Alberta, Canada

Conducted as part of: A PhD Thesis Work and The University of Alberta's NSERC Industrial Research Chair in Construction Engineering & Management - Theme One Research Activities

Chair Holder: Dr. Simaan AbouRizk, Hole School of Construction Engineering
Department of Civil & Environmental Engineering

Dear Participant,

You are invited to participate in a research study conducted by a representative of Dr. Simaan AbouRizk, Construction Engineering and Management Department at the University of Alberta, Ronald Ekyalimpa, PhD Candidate and Research Analyst.

Your participation in this study is entirely voluntary. You should read the information below and ask questions about anything you do not understand, before deciding whether or not to participate. You are being asked to participate in this study because your company has been identified as one of those that operate within the Heavy Civil and Industrial Construction sectors in Alberta, Canada.

Purpose of the Study:
The Canadian heavy civil and Industrial construction sectors are facing an increase in international competition. It is therefore necessary for companies in this sector to achieve and maintain high levels of operational efficiencies in order to sustain a competitive edge. This requires a clear understanding and implementation of sound performance management practices and development of tools that aid with the implementation of these practices. The objective of this study is to establish the performance management best practices for companies within this sector in Alberta. This knowledge will serve as a basis for identifying gaps in the practice and a means of standardizing best practices for performance management within the sector.

Procedures:
We are conducting a survey to determine the performance management best practices of heavy civil and industrial construction companies in Alberta, Canada. Therefore, we request your participation in this research study by providing us with accurate and current data about your company. Completing the survey indicates your agreement to participate in the study. Submitted surveys can be withdrawn by participants that opt to, within three weeks from the date of receipt of the questionnaire and related documents. Withdrawal after this period will not be possible. Data for participants that withdraw will be disregarded and safely destroyed.

Ronald Ekyalimpa will receive your completed questionnaire, then will analyze the responses. Please note that identifiers for participants (company name and email) will only be retained during the data collection phase of the study for purposes of follow-up on questionnaire responses. After data collection, all identifiers will be destroyed and responses aggregated and analyzed independent of participant identities (the questionnaire results will be grouped in a generic and non-company-specific format).

Please take the time to complete the attached questionnaire and email, or mail it to Ronald Ekyalimpa. For those that opt to fax their responses, please address the fax to Brenda Penner and Ronald Ekyalimpa. Use the contact information provided below.

Participants who opt to fax/mail their responses should indicate their company name, contact person and email address on their questionnaire responses for easy follow-up. It should take about 3-5 minutes to complete the survey. Your timely response will help to make this study successful; we ask that you return your completed survey by January 10th, 2014.

Anticipated benefits resulting from this study:
The results of this study can be made available upon request. The study will help to increase understanding of the performance management best practices of companies operating in the Heavy Civil and Industrial Construction sectors of the Alberta Construction Industry. This knowledge will then facilitate with follow-up endeavours that involve standardizing performance management practices for the region, suggesting improves to the sector where necessary and development of software tools using these standards, that can be deployed at companies for managing performance effectively.

Confidentiality:
Any information that is obtained in connection with this study and that can be identified with your company will remain confidential. Confidentiality will be maintained by storing all electronic materials on a file server hosted in the Construction research group at the University of Alberta. Access to this file will be secured and restricted to Ronald Ekyalimpa, and data/results will be restricted to the Chair. Email threads with study responses will be deleted after downloading the responses. Paper document submissions (through fax) will be stored in a locked file cabinet during the data collection phase, then translated into electronic format and stored on the secured file server, and the paper copies shredded at a designated shredding area for confidential materials within the Civil Engineering Department.

We will not use your company name in any of the information we get from this study or in any of the research reports. Information that can identify your company individually, or direct company survey responses will not be released to anyone outside the study. However, we may use results from this study for professional reports, and/or for academic conferences and publications (thesis).

Contact Information:
If you have any questions concerning this study, please do not hesitate to contact me.

Thank you in advance for your participation.

Regards,

Ronald Ekyalimpa
Research Analyst/PhD Candidate
NSERC Industrial Research Chair in Construction Engineering and Management
Department of Civil and Environmental Engineering
3-133 Markin/CNRL Natural Resources Engineering Facility
University of Alberta, Edmonton, Alberta, Canada, T6G 2W2
Office Fax: 780-492-0249, Email: rekyalimpa@ualberta.ca

Brenda Penner
Assistant to Dr. Simaan AbouRizk
NSERC Industrial Research Chair in Construction Engineering and Management
Department of Civil and Environmental Engineering
3-133 Markin/CNRL Natural Resources Engineering Facility
University of Alberta, Edmonton, Alberta, Canada, T6G 2W2
Office Fax: 780-492-0249, Ph: 780.492.5120

APPENDIX B1 – Definition of Linguistic Variables Utilized within the Model

Parameter	Linguistic Variable	Representation
Project Owner Trait	Very Good	95% - 100% chance of owner interrupting contractor in the course of project execution
	Good	95% - 100% chance of owner interrupting contractor in the course of project execution
	Somewhat Good	95% - 100% chance of owner interrupting contractor in the course of project execution
	Poor	95% - 100% chance of owner interrupting contractor in the course of project execution
	Very Poor	95% - 100% chance of owner interrupting contractor in the course of project execution
Project Engineering Quality	Very Good	95% - 100% of the scope has been fully designed and specified
	Good	75% - 95% of the scope has been fully designed and specified
	Somewhat Good	50% - 75% of the scope has been fully designed and specified
	Poor	25% - 50% of the scope has been fully designed and specified
	Very Poor	0% - 25% of the scope has been fully

		designed and specified
Project Safety Risk	Very High	95%-100% chance of safety incidents during project execution
	High	75%-95% chance of safety incidents during project execution
	Somewhat High	50%-75% chance of safety incidents during project execution
	Low	25%-50% chance of safety incidents during project execution
	Very Low	0%-25% chance of safety incidents during project execution
Project Complexity	Very High	It is not obvious to the contractor how they will perform 95% - 100% of the scope
	High	It is not obvious to the contractor how they will perform 75% - 95% of the scope
	Somewhat High	It is not obvious to the contractor how they will perform 50% - 75% of the scope
	Low	It is not obvious to the contractor how they will perform 25% - 50% of the scope
	Very Low	It is not obvious to the contractor how they will perform 0% - 25% of the scope