Master Thesis
Computer Science
Thesis no: MCS-2008:33
August 2008

A Multi Agent Web Based Simulation Model for Evaluating Container Terminal Management

Syed Sikandar Bakht
Qazi Sohail Ahmad

Department of
Interaction and System Design
School of Engineering
Blekinge Institute of Technology
Box 520
SE – 372 25 Ronneby
Sweden

This thesis is submitted to the Department of Interaction and System Design, School of Engineering at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Computer Science. The thesis is equivalent to 20 weeks of full time studies.

Contact Information:
Author(s):
Qazi Sohail Ahmad
Studentvägen 01, Room 37
Ronneby, 37240
E-mail: qazisohail101@gmail.com

Syed Sikandar Bakht
Folkparksvägen 10, Room 10
Ronneby, 37240
Sikandarbakht101@gmail.com

University advisor(s):
Dr. Lawrence Edward Henesey
Department of Interaction and System Design

Department of
Interaction and System Design
Blekinge Institute of Technology
Box 520
SE – 372 25 Ronneby
Sweden

Internet : www.bth.se/tek
Phone : +46 457 38 50 00
Fax : + 46 457 102 45

ABSTRACT

This thesis provides a software prototype of Container Terminal Management system with the help of a Multi Agent systems technology. The goal that has been tried to achieve during this research work was to solve the management issues residing in a CT. The software prototype can be implemented as simulation software that will help the Terminal Managers to take necessary decisions for the better productivity of CT.

The CTs are struggling with taking proper management decisions. There are many policies implemented but the use of a certain policy at a proper time is the main issue. It is possible with simulation software to visualize the affects of decisions taken by the implementation of a policy and see the expected output. This can really improve the performance of a CT.

The management decision problem is solved by modeling the whole CT in a computer modeling language. The prototype shows all the actors appearing in a CT in the form of Agents and these agents are responsible for carrying out certain tasks. The prototype is the final contribution along with partial implementation. The model is proposed to be a web based system which removes the platform dependability problem and provide availability online.

Keywords: Container Terminal, Multi Agent Simulation, Container Terminal Performance,

ACKNOWLEDGMENT

First of all we are thankful to Almighty Allah who is the creator of the whole universe and gave us the strength to do this thesis work in time.

We are indeed thankful to our supervisor Dr Henesey for his priceless guidance and support throughout this thesis work. He always had been there to provide us valuable feedback in time and innovative ideas to show up a good work. We struggled many times during this research work but each time, we found him ready to help us whenever we needed. In short we feel that with his supervision we have learned a lot and have gained knowledge.

Our deepest gratitude goes to our family for their unflagging love and support throughout our life; this dissertation is simply impossible without them. We feel indebted to our fathers, for their care and love. As typical fathers, they worked hard to support the family and spare no effort to provide the best possible environment for their kids to grow up and attend school. They had never complained in spite of all the hardships in their lives. Our mothers, throughout their lives, they forgot about their own enjoyment and kept on praying for us to have a success in life. We believe, no one else in this world would be that much proud of us as much as our mothers will be. Mother and Father, we will never be able to pay you back, but at least we can thank you at every stage of life where we are rewarded a success. Thank you very much for your precious love.

We would also like to thank our colleagues who encouraged us to do this thesis work. They all were there to support us whenever we were facing problems and struggling to accomplish our goals.

Last but not least, we are thankful to our brothers, sisters, friends and all those who love us. It’s just their love that kept on moving us ahead. Each time, we lost our way and each time their love showed us a right path. It was like a driving force for us which kept pushing us forward, to do something special for them. We worked day and night but never felt tired. Their love was always there to motivate us. We wanted to make them proud of us and we are sure, they will certainly be proud. Hence, we would like to dedicate this thesis to all those who loved us and who prayed for us.

Table of Contents

ABSTRACT **1**

ACKNOWLEDGMENT **2**

LIST OF FIGURES.......... **5**

LIST OF ACRONYMS.......... **5**

1 INTRODUCTION **6**

2 BACKGROUND OF THE THESIS **8**

2.1 PROBLEM IDENTIFICATION 9
2.2 RESEARCH QUESTIONS 10
2.3 RESEARCH METHODOLOGY 10
2.4 EXPECTED OUTCOME 10
2.5 LAYOUT OF THE THESIS 11

3 PROBLEM DESCRIPTION **12**

3.1 CONTAINER TERMINAL MANAGEMENT 12
3.2 RESOURCES FOR CT MANAGEMENT 13
3.3 PLANNING IN CONTAINER TERMINAL 13

4 RESEARCH METHODOLOGY **15**

4.1 RESEARCH STRUCTURE 15
4.1.1 Data Collection *16*
4.1.2 Meetings *17*
4.1.3 Literature Review *17*
4.1.4 Design and Implementation *17*

5 CT MANAGEMENT **19**

5.1 SHIP ARRIVAL 19
5.2 LOADING/UNLOADING 19
5.3 HORIZONTAL TRANSPORT 20
5.4 YARD STACK / STACK ON QUAY 20
5.5 CONTAINER TERMINAL OVERVIEW 20
5.5.1 Entities at the container terminal *21*
5.5.2 Container Terminal Policies *23*
5.5.3 Berth Allocation Problem *23*
5.5.4 Crane Assignment *24*
5.5.5 Container Sequencing and planning *24*
5.5.6 Container Stacking and Storage policy *24*

6 SIMULATION MODEL, A SOFTWARE ARCHITECTURE **25**

6.1 AGENTS AND THEIR BEHAVIORS USING UML DIAGRAMS 25
6.1.1 Terminal Manager Agent *26*
6.1.2 Ship Agent *27*
6.1.3 Stevedore Agent *28*
6.1.4 Crane Agent *29*
6.1.5 Straddle Carrier Agent *30*
6.2 CT STRUCTURE DIAGRAM 31
6.3 CONTAINER TERMINAL ENTITY RELATIONSHIP DIAGRAM 33

7 SIMULATION MODEL, A TECHNICAL ARCHITECTURE **37**

7.1 MULTI AGENT PLATFORM 37
7.2 SYSTEM FUNCTIONALITY 38

7.3 DESIGN ISSUES 39
8 DISCUSSION OF THE MODEL 40
8.1 SCENARIO BASED EVALUATION OF THE MODEL 40
8.1.1 Favorable Condition 40
8.1.2 Worst Condition 42
9 DISCUSSION 44
10 CONCLUSION 46
REFERENCES 47
APPENDIX A 49

List of Figures:

Figure 1: port of Rotterdam Holland [17]......7
Figure 2: Research Methodology......15
Figure 3: RM Phase......16
Figure 4: Classification of Ships......21
Figure 5: Quay Crane......22
Figure 6: Stack Crane......22
Figure 7: Straddle Carrier......22
Figure 8: Operations Diagram......26
Figure 9: Activity Diagram for TMA......27
Figure 10: Ship Agent......28
Figure 11: Stevedore Agent......29
Figure 12: Crane Agent......30
Figure 13: Straddle Carrier Agent......31
Figure 14: Class Diagram for Simulation Process......33
Figure 15: CT ER Diagram......35
Figure 16: Web Based System Implementation Details......38
Figure 17: Execution of Scenario......42

List of Acronyms:

MAS: Multi Agent System
CT: Container Terminal
TM: Terminal Manager
QC: Quay Crane
SD: Straddle Carrier
SB: Ship Berth
SC: Straddle Carrier
FCFS: First Come Fist Serve
OTS: Overall Time Shortening
BCS: Berth Closest to Stack
ERD: Entity Relationship Diagram

1 INTRODUCTION

Containers are huge metallic rectangular boxes that are used to ship loads from one destination to another [11]. These containers are mostly transported through ships from one continent to another. The Containers vary in size and types but two of the sizes are now standardized i.e. a Twenty-Foot equivalent-Unit (TEU) and a Forty-Foot sized Container called FEU [11]. The sea port portions where these containers are loaded/unloaded to/from ships are called Container Terminals (CT). The CT involves many entities which depend on each other in order to perform that loading/unloading of Containers or to perform any other specific task. For example to unload a ship, it takes Ship Captain, a Stevedore, Terminal Manager, Cranes and Straddle Carrier Operators to co-ordinate the activities effectively to handle a ship. At the same time, a yard stacking is a complex system by itself. It tries to make sure containers can be stacked in such a way that they can be retrieved later without disturbing the sequence. The decisions at yard stacking affect the ship berthing process as well. In this context the operations performed at a CT are much more complex than they can be expected.

The CT are getting more and more importance in international business today as maximum portion of international cargo is moving between ports which make it the busiest nodes in a business world. Each day the world is expanding as a global market and each day it loads the CT with more and more heavy vessels [12]. According to Dr Henesey et.al, from year 1984 to 2004, TEU containers shipment has increased from a range of 39 million to 356 million. With this increase in work load, the CT management team is forced to operate as fast as possible within a certain amount of time. It is getting a challenge for the CT management team to keep the docking time (the time of stay of a ship at a CT) of ships as minimum as possible to provide them with the required service they demand, in a desired time.

With the development of international business, productivity at CT is increasingly becoming important. Apart from expanding the current infrastructure and resources, it has been pointed out that better results can be achieved even with a more efficient use of current available resources [14]. One way to achieve this goal is by visualizing the affects of management decisions first before they are implemented which is proposed by Dr Henesey et.al, in [14]. According to Dr Henesey et.al, a Multi Agent Based System technology implanted in a computer simulation can benefit an industry like Sea Ports, which involves a variety of inputs, outputs, actors and intrinsic characteristics. The management decisions can be tested and the results can be analyzed with the help of Multi-Agent Based Simulation (MABS). This gives a luxury to the Terminal Management to plan for the best possible solution and prepare for the worst conditions when they occur.

A computer simulation is one such area that can really help in improving the performance of a CT without extending the number of resources but using the same available equipments with proper planning and decision making. Multi Agent System (MAS) technologies have been proposed in different articles by many researchers who intend to solve the issue of low productivity by implementing intelligent algorithms from Artificial Intelligence in a CT domain. One such effort is BAMS (Berth Allocation Management System) which is for the purpose of deciding berth schedules for the arriving ships [15]. A similar MABS concept is proposed by Vicente Botti et.al, in [16] that also addresses the issues regarding Port Terminal Container management.

The purpose of this thesis is also to model the CT in a Multi Agent based technology context. The proposals, ideas, solutions and implementations will be based on a Web based environment. The advantage of having such a system available on a world wide web is that, it can be accessed any time without any sort of restrictions to platform or specialized software. The problems in a CT are identified first which are the main cause of performance downfall in this business. The research is progressed in such a way that research questions are prepared and then a goal is specified which can be achieved by answering the research questions. The goal of this thesis is to create a model for Multi Agent Web Based Simulation Model for Evaluating CT Management. The model is developed using UML diagrams and then an optional phase of implementation is followed which depends on the time factor.

The Web Based Simulation Model presented in this thesis focuses on all the entities of a CT. There are many actors involved in this environment and each actor is modeled as an Agent in this thesis which is responsible to fulfill a specialized task. The result of one Agent affects the performance of other, so they all work in collaboration to meet a final goal in a best way possible.

This documentation starts by introducing the background and domain. After, a separate section is dedicated to have a clear understanding of the problem, existing in the area. Then there is a description of the research methodology progressed towards the goal of the thesis, and hence the achievements are discussed in Chapter 5 and Chapter 6 of this documentation.

Figure 1: port of Rotterdam Holland [17]

2 BACKGROUND OF THE THESIS

This chapter provides a brief background of the domain that is selected as a research area and why this specific domain is selected to carry out research. Further, it discusses a layout of the thesis i.e. what are the problems, what are the research questions, which research methodology is employed and of course what is the expected outcome.

Computer simulation is the process execution of a theoretical physical system on a computer and analyzing the output produced [1]. Combining the web and simulation can achieve an integration that will be the success key for the future simulation research, yet there are several directions that can create relationship between web and simulation [1]. Web based simulation was introduced for conducting simulation experiments online. Due to Java, web based simulation is reusable, portable and independent of the platform. Most of the existing web-based simulation tools have focused on mainly two parts which are development of the runtime simulation libraries and development of the distributed simulation environment [2]. A Multi Agent System is currently gaining attention and is a part of many sub-fields of computer science and artificial intelligence. MAS technology is being used in an increasingly wide variety of applications ranging from simplicity to complexity e.g. Agent online help desk to Air traffic control. It may be felt that each agent-based system is different from the other agent-based system depending on the real environment but the basic logic behind each is the same and they have one thing common i.e. agent [3]. According to Yi et.al, in MAS, business entities are represented as agents and the information flows with proposed coordination method for collaborating among agents. The collaboration is performed by continuously keeping the Agents in contact to each other by passing messages, signals and about the status of Agents. Multi agent system is a collection of mixed/diverse computational entities, which have the ability to solve a problem and are able to interact with each other to reach a goal/result.

Organizations such as Container Terminals (CT) involve many entities and become more complex and difficult to provide co-ordination among teams i.e. public authorities, maritime authorities, terminal operators, shipping companies, transportation companies, railways, labor unions etc[5]. According to Henesey et.al, the main functions of the CT management are the *planning* of operations and *controlling* the movements of containers. As their growth increases many problems for the ports also arise like higher requirements of resources on terminals and faster quick transportation systems in cities and communities. Physical expansion of CTs or proper utilization of resources and increase in number of resources are solutions to this problem but it still is a difficult problem to easily solve. CT requires better management decision making systems. These decisions will be based on computer based systems rather than hardware to utilize the available resources [4]. However, there is much commercial simulation that is focusing on the above mentioned problem but are limited and provide computational support like calculation of total ship time based on planning of operation. In a real system, the interactions are the critical issues. Some of the principal aspects involve interference among yard cranes, interference among dock cranes, interference among connection units, interference among containers, access problems, etc. [5].

We need a management system that makes decisions, identifies policies that use available resources efficiently. The policies used in the simulation should be able to be compared in monetary terms for both ship and terminal [6].

CT is a very fast growing industry and continuously changing, attention is required for different issues in a very short time. Also, from organizational point of view it involves many entities and becomes complex to manage. Complex in terms of finding ways to get good profit based on smart management. The use of simulation software nowadays is a very good way to manage it but still they are limited by operation. Some of the software provides only time calculation, loading/unloading and interference among other ships etc [5]. The real focus should be on improving the utilization of the available resources through simulation for the management to make decisions. The agent based simulation system is argued provide alternative solution for CT performance.

This thesis aims at providing one a Web-Based simulation model, which to our knowledge is novel for managing the resources effectively and efficiently in a CT. There are several software available already for such purpose but there is no web based system available which has been implemented using MAS technology. According to Dr Henesey et.al, the big advantage of implementing such a system with an Agent based technology is that, as there are several entities involved in a Container Terminal and each depends on the other to take a proper decision, so it is rather a good idea to implement the whole system with an Agent based technology.

2.1 Problem Identification

The CTs are getting more and more importance in international business. According to [8], the number of container handling at ports has increased to a ratio of 55% between the years 1998 and 2005. The annual growth rate of Container trade is 6% for US and 9.5% for the rest of the world [9]. Further, to come up with even more statistics, it has been pointed out in [10] that, over a period of 2010-2015, the transshipment demands are going to increase by a ratio of 31-42% in Northern Europe and 41-55% in South Europe. This shows how busy the CTs are going to be in future.

With this increase in workload, the current infrastructure of CTs has to be more flexible in such a way that it can accommodate the future demands. There are few solutions provided in [10] which are:

a) Hiring of new technology and equipments for maximum output.
b) An increase of working time of a CT.
c) Improvement in resource allocation
d) “To increase the length or number of berths”.
e) “Improvements in management policies and decisions”.

Looking at the above given statistics, it indicates that the CTs are getting more and more busy, and at the same time they need to cope with the increasing demands per day. The above given solutions provided in [10] are really the only way to deal with these problems. In this thesis, two of the above solutions were tried to be implemented to play a part towards improving the conditions for a CT. These are “Improvement in resource allocation” and “Improvements in management Policies and decisions”. Both of these solutions are possible to be tested with the help of a computer simulation tool. A computer simulation could solve the issues by allocating resources properly based on predefined criteria and demands in a more quick and efficient manner. A computer simulation is not only helpful in resource allocation but also it can help in decision making process for TM (Terminal Managers). It is helpful in calculating different outputs at different circumstances that a TM can analyze and select a best suitable selection.

Even though many computer simulation software purchases are available, but more and more the need of the day is to propose a Web Based Multi Agent system that can work irrespective of a platform and which can be available online any time. This thesis propose a Web Based system simulation tool for evaluating the operations and management in a CT, also a partial implementation has been deployed.

2.2 Research Questions

The following questions are going to be addressed throughout this research work and answers will be provided with the help of a Research Methodology.

A-How a Multi Agent Web Based Simulation System can be modeled that can be implemented for a better use of resource allocation?

B-How a basic Multi Agent Web based System can be developed on the basis of above proposed model?

In order to meet the above given tasks, small goals have been established that will help later to integrate the solutions and propose the model.

a-What are the entities involved in a CT and how do they function?

b-What is the flow of communication and how much dependencies are involved among entities?

c-What are the necessary decisions that can affect the overall performance of a CT?

D-How does the resource allocation process work?

e-Identify the necessary constraints that are necessary for a CT Manager to take decision on?

The purpose of these goals is to break down the process of ship handling in a CT, understand how does it function and then model it later on with the help of a computer modeling language.

2.3 Research Methodology

In order to meet the expected outcome of the thesis, a literature review was carried out to gain primarily the domain knowledge. The research technique involves collection of articles, magazines and resource journals that can provide some useful information about the Ports, Terminals and Containers. The process of ship handling, berth allocations and management of the resources was thoroughly studied and analyzed. As entry to Ports is strictly prohibited for visitors, so online videos will be observed for how a particular process is performed. The gained knowledge will be later used to model the whole system for a computer based program porting an Agent technology to a web service.

2.4 Expected Outcome

The Research work will yield a Multi Agent Web Based Model that will have the ability to test management policies under different scenarios and will also be able to allocate resources in an efficient manner. The model will be presented with the help of a computer modeling language i-e UML. The entities further will be modeled in the form of Agents and their functioning will be explained based on a practical implementation. The model and partial implementation of the system will be a good ground to work on it further in future and present a more professional version of the system.

2.5 Layout of the Thesis

The second chapter of this thesis provides an overview of the thesis background and the domain to give the reader a better understanding of the research area. It also discusses the problem existing and then few research questions to find a solution for that problem. A research methodology section is provided in this chapter which demonstrates how to move towards a solution in a more standard way. The second last section ‘Expected Outcomes’ puts some light on what a reader can expect at the end of the research work.

The third chapter is all about the problem description. It gives an overview of the Container Terminal Management first and then discusses the resource availability problem, which is a critical factor as it can really put ships waiting in queues, if improperly handled. The result is a loss to the CT Management. It further discusses key issues related to resources and provides a good ground to be researched and taken into considerations.

The forth chapter discuses the Research Outline in more detail. It explains how step by step the research work is progressed in order to meet the required goals.

The fifth chapter provides a detail of CT management and the decisions that play an important role during ship handling process at a Container Terminal. These policies are the core of the management system of a CT.

The sixth and seventh chapters are the heart of the thesis, which actually presents the findings in different sections. The model of the system with the help of operations diagram, activity diagram, class diagram and then the relationship with the help of Entity Relationship Diagram is presented along with the technical details. It explains each and every section by keeping in mind the intended level of the reader and it has been tried to explain the complex terms in a simpler manner.

The seventh chapter is a detail about the partial implementation of the model and how can it be extended to make it more and more usable for Shipment and Logistics Industry.

The final two chapters are actually the Discussion about the findings and the conclusion presented in the end.

3 PROBLEM DESCRIPTION

This chapter gives a brief overview of the problem existing in the domain, and the way this thesis is progressed to find a solution for that problem.

3.1 Container Terminal Management

In early decades cargos were the main medium of transport, but their number reduced with the advent of container shipping. The majority of the cargo handling is now done by containers and according to [18], 90% of the cargos will be shifted to containers at the end of 2010. Also the number of current containers are about 15 million and it will continue to increase in the next coming 10 year at the percent of 8.5 [14]. The container business is increasing day by day because of business demand and of course replacement of cargos.

The management of container terminal (CT) is very complex as it includes decision to be made upon arrival/departure of ships, scheduling of ships and profit made after each decision. CTs are required to serve the ship as soon as possible in order to set resources free i.e. straddle carrier (SC), cranes and other equipments/resources involved. According to [18], serving customer quickly and better use of resources for maximum profit is the main challenge for management. The Terminal managers use policies when serving a ship or allocating resources. In most of the cases these policies aren't effective which result in poor administration and lack of decision system.

The current system of container terminal is centralized system. This isn't flexible to the rapid changes to the system, it's not that much dynamic so those requirements of the system are adopted easily. Traditionally the container terminal is controlled by centralized software which uses top to bottom approaches. Using this approach in this system will chain all the sub-system with each other and within sub-group. This has two effects on the system. First one is if a failure occur at a point than whole system will be shutdown, second one is all are dependent on each other so one system has to wait for the other to finish or reply about the status.

Container terminal involves sub-system that is interconnected with each other and has dependencies on each other. To solve this problem we have to decompose into several entities and define their own tasks and actions to be performed. These entities will be working independently to achieve their task and dependently to achieve the overall common goal. Multi-Agent system technology (MAS) will be a good approach to represent each unit in-term of agent. As we know that MAS technology is used where a unit/entity has to take a decision on its behalf to achieve its goal. Similarly using MAS in container terminal will help us to resolve these issues easily, will help us to represent each entity, its actions and response and etc.

Using computer simulation for the terminal manager is a good choice because TMs will be able to see what will be the result of his decision. According to [19], a decision support system can improve the management of container terminal. Moreover Simulation can be used to evaluate the alternative management policies. The computer generated results from the simulation will be more closed to the fact and figures as compared to the human decision (human can't understand the situation some time or they are not always making perfection decision or they may make incorrect decision).

It's very much important to understand the whole system before doing any sort of computer simulation because all the operations of the systems are linked together by some mean.

Computer simulation result for the terminal manager can be then compared to the real world decision as these simulations are based on assumptions/modal of the decision making terminal manager experience and can be easily validated and can be improved. The everyday increasing problem of CT can be overcome by using Multi Agent System (MAS), when performing simulation. According to [14], MAS will help CT to use its resources effectively and will also show performance without spending large amount on expansion of terminal and on equipments.

3.2 Resources for CT Management

CT is complex and wide spread in nature. Its complex because many external entities (Ship, Trucks, Cranes, Straddle Carrier)are involved in it operating and communicating with each other fulfilling its own/each other request. Each entity involved in the CT work independently, but it is linked with others because one take action based on results of the other entity. For example the crane work independently of terminal manager, but can only work when the ship arrives to the berth or leaving during load or unload process. According to Rida et.al, container enters to container port terminal by ships and leave by different means i.e. trucks, trains etc. Containers reached to the terminal are stored in terminal yard. The container leaves by the same way when leaving the terminal.

CT system can be divided in sub-systems for efficient processing of material handling and best decision making. According to Kim et.al, CT can be distinguished in four planning and control level for efficient making of decision and obtaining better results. These are:

- Strategic Level:
  At this level decision is made, at what position the CT will be built. This decision takes from 1 to several years.
- Tactical Level:
  At this level the decision for equipments is made. What kind of equipments will be used, what position will be chosen for them? This decision takes from months to years.
- Operational Level:
  At this level decision for number of vehicles to be used for efficient transport process, number of Quay Cranes (QC) to be used for serving a ship and Straddle Carrier (SC).
- Real Time Level:
  In real time level decision are made after each minutes e.g. number of vehicles needed for transportation, berthing of a ship, number of cranes needed for serving a ship etc.

3.3 Planning in Container Terminal

Most CTs are a facing lack of planning, which affects valuable customers, increase costs and increases time. This end nowhere but losing their grip on the market as well as affecting the overall economy. Most of CTs follow a simple plan i.e. allocating resources to individual ships upon their requests and treating them as a single entity. According to Gambardella et.al, quay cranes, yard cranes, lifters and man power are provided to serve a ship, although there are other ships waiting to be served, but the lack of planning and co-operation of resource allocation isn’t handled in parallel. This leads to separation of management activities differently.

There are many simulation tools available for the planning of CT but they are focusing on only issues like ship berthing or ship turn-around-time, total cranes needed to serve a ship etc or these simulation just provide some rough/estimated simple calculation not the real time result as the terminal manager needed. According to Bruzzone et.al, the traditional simulation packages support one or few entities in CT, e.g. simulation may provide estimated time for the ship loading/unloaded without considering interaction with other ship at the quay. In real system, interaction is critical issues and some of them are:

- Interference among yard cranes (i.e. transtainers)
- Interference among dock cranes (i.e. portainers)
- Interference among connections units (i.e. straddle carriers)
- Interference among containers
- Containers blocked by others on the ship
- Containers blocked by others on the yard
- Access problems (i.e. containers vs. constackers)

Simulation process can be improved by verification, validation and accreditation and at the same time interoperability issue can downsize the development time and cost. Verification and validation are very much important as the final user will see the accurate result during operation management. However verification and validation of simulation is very much complex and in any case the final user need to be involved in designing models, result of simulation and to receive completed and detailed documentation including:

- Simulation Effectiveness Boundary Ranges
- Confidence of Simulation Results
- Statistical Analysis of Stochastic Factors
- Independent Variable Description
- Target function Description
- User Manual
- Formal Description of Conceptual Model
- Statistical Validation & Verification
- Case Study Testing Report

The purpose of simulation in CT is to be able to allocate the resources usefully and to make all the entities having interoperability and stand work in single as well. Also it will help the CT manager to see what could be the best possible ways to use resources effectively, to make sure that the ship is served with in desire time, to save cost and make some good profit.

4 RESEARCH METHODOLOGY

This chapter illustrates the Research Methodology followed in this thesis. The technique used throughout this research work enabled the students to create a sound ground in order to implement the knowledge gained.

4.1 Research Structure

This section describes the methodology of how a research work is carried out in order to answer the research questions. The following general architecture shows how the methodology is followed. This model diagram shows that three things lead to the definition of research questions i.e. identification of the Research Domain, Context Knowledge and then Data Collection. The research problems are then studied experimented or applied with any other research methodology technique and then a proposed solution is presented for a given specified problem. It is possible that a proposed solution could be a model or design or any solution based on theories, so it can be implemented depending on the type and nature of solution.

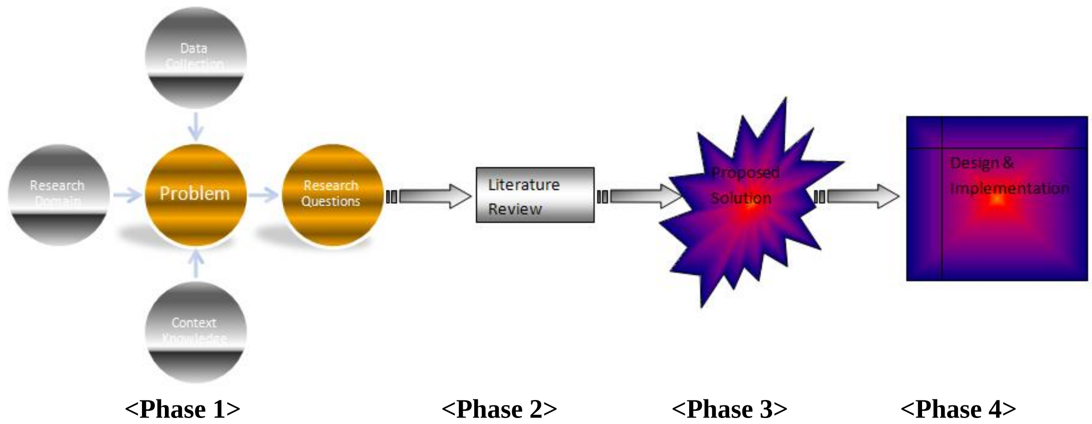


Figure 2: Research Methodology

The research structure of thesis is divided into four phases as shown in Fig.2. The first phase is actually the selection of research domain and evaluation of the context knowledge about that domain. Further collection of some data is also a part of this phase. This leads to the identification of problem that exists in the research domain. The problems are then analyzed and research questions are created in order to settle few aims that are to be achieved at the end of the thesis. The Research Questions are broken down further by setting a small objective which helps to achieve the main goal of the research. In order to reach to a specific objective, strategies are created that leads to the accomplishment of small objectives. According to H.P.Degeling et.al, a strategy is something that defines what and how an objective can be achieved. Fig 3 is presented to show the same with the help of a diagram. The Goal is achieved with the help of small objectives and objectives are met with the help of implementing and following strategies.

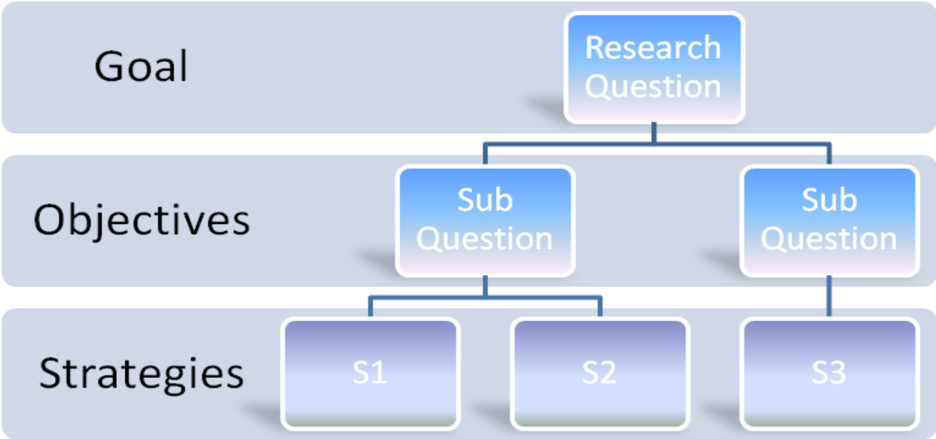


Figure 3: RM Phase

Each and every strategy is designed to meet an objective which eventually leads to a mutual goal. This is how a research methodology is progressed systematically. The key processes involved in a research methodology are discussed below which are followed to meet a goal in this Master Thesis.

4.1.1 Data Collection

Data collection is the very first step of researching in a specific domain. The data to be collected is all about the area where there is a need to carry out a research and improvement is expected. The data here refers to a general term which involves definitions, attributes, entities, processes, methods, procedures and e.t.c. It also contains information, latest developments and updates along with problem identification and description material that can help in narrowing the research scope. The next step to perform is analysis of collected data. There are methods defined to analyze and interpret the collected data. The process of data collection also needs to investigate through a proper method that can result in gathering the required data according to the need and specification of research domain.

For this research the data is collected mostly from our supervisor Dr Henesey. Being a supervisor as well as the Domain Expert, his experience, articles, information and suggestions were the main source of knowledge for our thesis. Apart from him, many more articles written by researchers about the same domain and related problems were searched and collected for the purpose of gathering the required knowledge. Many more researchers were tried to be contacted by email that had made some contribution towards the problem domain or were working on it lately. The main purpose of it was to collect their recent work and contributions. Apart from this, many journals and magazines were reviewed for the latest news and solutions to the research area.

As the domain is more of an industry focused? therefore a thorough observation was required to have an understanding of the work flow and to see how different operations are performed. Because of restriction on entry at these non-public places, many web sites were searched through search engine that could show the work flow visually. This gave us an advantage of observing things right from a computer system. Further, in order to design a computerized model of a system, it takes to have a complete requirement specification of that system, which fortunately was presented by Dr Henesey, hence made a major contribution to our task. The data collection phase took three weeks to complete during which all the necessary data was collected which could serve the purpose as efficiently as possible.

4.1.2 Meetings

Many meetings were held with our supervisor and Domain Expert which purely was for the purpose of brain storming. Further it was also tried to clear the confusions and puzzles that we had in our mind regarding this thesis. The meetings ended with fruitful results for us and the knowledge gained was later used to be applied in our model.

4.1.3 Literature Review

This is the second phase of our research work where after collecting the necessary data, it was time to process it and analyze it. A thorough document review methodology was adopted. The review helped us clearing our minds about the operations performed, the work flow, the entities involved in a system and the methods/procedures for certain tasks. The whole research field was studied during this phase. The critical point which was to be addressed was being targeted the most as it was the heart of thesis. Normally a literature review is carried out to combine the material published by several renowned researchers and scholars. This material is then used to publish a single research paper or can be used to implement any theoretical work that has been proposed to date.

A literature review is actually an evaluative final document of information found in the literature related to a selected area of study. The document should summarize this literature in a clear manner. The purpose of a literature review is to provide a proper context to the research and of course to enable the researcher to learn from the previous work done by other researchers. This process helps to refine the thinking and allows a researcher to outline the flaws left in previous research. The methodology is followed for this thesis because there is a need to identify the previous gaps, if left, and then use the current researched material to design the model of a system.

The document review process helped to learn the system and its functioning. It also helped to identify the areas where there is a need for further research; most of the controversial areas were also identified where an implementation of technology is more needed than expected. It is the same aim of this thesis to solve problems with technology that are so far handled annually or with a minimum use of technology.

Apart from having advantages, the literature review methodology had some disadvantages as well that could affect the overall outcome of thesis in the end, First and by far the most important issue is, the whole review and design phase depends upon the quality of the collected data. The reliability, the validity and the relativity of the collected data has to be ensured before it can be reviewed which itself is a time consuming job.

4.1.4 Design and Implementation

A Model is diagrammatic representation of a job/work. This model shows how a certain work is performed and which actors are responsible to carry out specific tasks. It also represents the entities along with their attributes which are actually their properties, and the functionalities that they are supposed to perform. A real world system is modeled for the purpose of transferring the knowledge to a computer based system. There are many computer languages that are used to model a real world system to a computerized system.

In this phase, the system is going to be modeled with the help of a modeling language. The process is to be shown with the help of work flow diagrams and the actors involved along with their inter communication is to be shown with the help of entity relationship diagram.

Further as there are many entities that can work alone as well as in a group to achieve a certain task, so each and every entity is going to be treated as an Agent that can perform a specific task in order to achieve a mutual goal.

The CT has been proposed to be implemented with an Agent based technology in a computerized system by Dr Henesey in many of his papers. The outcome of this thesis depends upon the research carried out by these researchers, as most of the data is collected from these resources. The design is going to show each and every entity involved in a CT system and its jobs listed with the help of a behavior diagram. The activity diagram displays how these activities are performed by each Agent and how the results are passed at which specific time.

The implementation is the last phase of this research work which aims to implement the whole system programmatically. This phase is somewhat optional depends upon the time factor as it is not possible to model and implement a system like CT within a period of six months.

5 CT MANAGEMENT

This chapter gives a brief overview of the operations performed at a Container Terminal and how different management issues are handled. Different actors involved in decision taken are discussed and of course the roles of each actor with respect to a common job are defined.

We have read in the earlier chapter about the difficulties and numbers of entities involved and working in CT, about the interconnectivity of these entities and the dependency involved on each other. Overall it becomes very difficult to understand the flow of such a complex system. In CT mainly container handling is done, where container are stored, moved to their desire destination. Containers can be divided into mainly 3 categories.

Reefer Container: type of container that need electricity to maintain its internal temperature.
Hazard Container: type of container that contain some hazardous material.
Standard Container: type of container that contains general things and require no special care.

Containers arrive to terminal and are cleared within the provided truck and train operation areas. Then they are distributed to different stocks full-filling the requirements. Majority of container are positioned in pre-reserved area from where they are shipped next. Generally container transshipment and container movement to and from are associated with waterside transshipment process (WTP). Landside transshipment process (LTP) is carried out by special vehicles called gantry cranes [23]. According to Dr Henesey et.al, CT can be divided into four sub parts:

- Ship Arrival
- Loading/Unloading
- Horizontal Transport
- Yard Stacking

5.1 Ship Arrival

Before the ship arrival, CT manager has to decide when to serve the ship? How much time will be required to serve the ship? And what berthing policy will be used after ship arrives? As the berthing policy affect the other decision later on. The Policies implemented for ships are Ship Sequencing Policy and Ship Berthing Policy.

5.2 Loading/Unloading

After arrival of ship, CT manager has to decide which resources will be allocated to work on the ship. This is an important decision for CT manager as there are also other ships and they needed to be served as well, so the CT manager will allocate resources in such a way that current ships on the berth are also serving. CT management involves allocating Quay Cranes (QC) and transport equipment such as Straddle Carriers (SC) or trucks and labor.

5.3 Horizontal Transport

While serving a ship, CT manager try to avoid leaving any resource idle and allocate it to another job as soon as it's free. CT manager avoid Quay Crane (QC) from being idle or anything interrupting QC. For this purpose terminal transport is made efficient and available so that QC can be more productive. However there are still some problems related to the horizontal transport which are, routing, pickup sequencing and co-ordination with QC.

5.4 Yard Stack / Stack on Quay

At ports, containers are stored in yards. They are stored according to the type of container, size of container or by the ship line which own them. As explained in the reference [24], "Ideally, in transshipment operations the ship that is unloading the containers to be loaded by another ship will be serviced at the same time with the other ship in order to avoid problems of stacking containers. This scenario offers a faster service." But in reality it's not like this, the container must stay for some time on the yard stack before loaded to another ship. "The problem/decision affecting this process is: Stacking Density, Yard Stack Configuration, Container Allocation to a Stack According to Rules and Dwell Time."

5.5 Container Terminal Overview

CT is an open place with two different interfaces for material going in and out. One is quayside which is doing loading and unloading container to/from ship, the other is loading and unloading container on/from the trucks/trains. From both the interface the containers are stored temporary in stack. When ship arrives to the port, terminal manager assign berth to the ship, the berth having cranes/straddle carrier to load/unload containers from the ship. The cranes unload the containers and place them in the place where they will be transshipped next. Containers arriving to the terminal by train/truck are handled by the terminal internal equipments and are placed in the respective stock in the yard. Some additional moves are performed if stock/empty depots exist in the terminal [25].

Quayside operation or container transshipment and container movement to and from the wharf is sometime refers to waterside transshipment process (WTP). There is also landside transshipment process (LTP) or hinterland transshipment process. Terminal manager serve different types of ships at the terminal, however most important of them are deep-sea vessels, having loading capacity up-to 8.000 containers unit (TEU). These vessels different ports of different countries and continents. These vessels are 320 meter long, 43 meter of breadth and draught of 13 meter. The container can be stocked up-to 8 tiers and 17 rows wide. For this sort of vessels the terminal manager calls the appropriate number of cranes/straddle carriers. Loading of about 2.000 containers is common and same is for unloading. Feeder's vessels are used to deliver containers to the deep sea vessels, capacity of feeders vessel is normally 100 to 1.200 TEU. Inland barges are used to deliver container to hinterland on rivers and channels.

Storage of container is divided into mainly 3 blocks as described in the start of this chapter. Stack can be divided into separated areas like export, import, special and empty containers. Beside stacks, terminal can be also divided according to the operations. If a railway track doesn't exist then containers have to be moved by trucks or by some inland transportation means between external and internal station of terminal. These results in extra resources allocation on the terminal and other differences may be if shed exist inside the terminal area. At shed container are stripped and stuffed and goods are stored. The additional movement in terminal is connection of yard and shed and it's also for empty depots where empty containers are stored according to the requirement of shipping lines.

5.5.1 Entities at the container terminal

The entities involved in a CT are as follows.

5.5.1.1 Ships

Ships are used for the transportation of containers from one port to another. These ships vary in design, shapes, size, weight, life time (i.e. from 20-30 years) and overall functionality. Classification of the ship shown in the figure 4.1 [26].

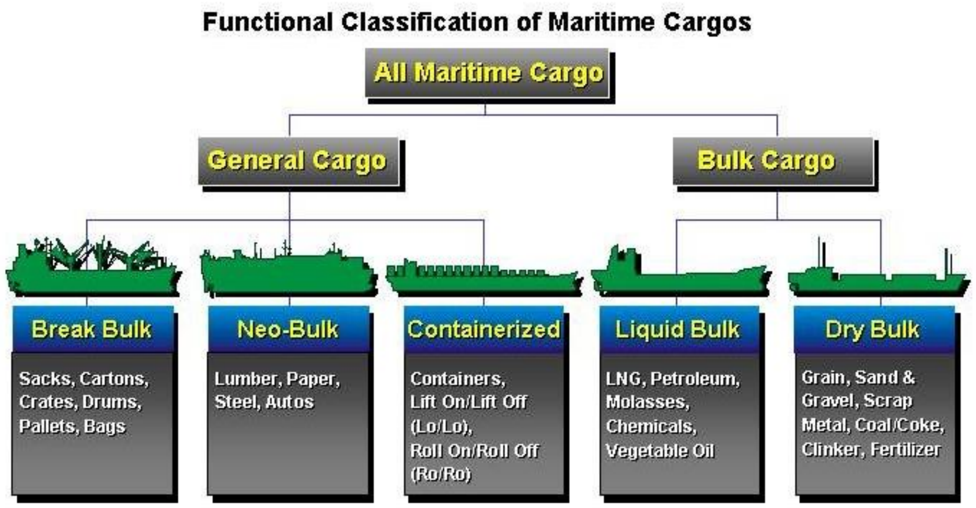


Figure 4: Classification of Ships

5.5.1.2 Cranes

There are different types of container as we have discussed in earlier section of this chapter. According to Steenken et.al in [25], so for this purpose there are also two different types of cranes to handles containers.

Cranes are used for loading and unloading to/from ship. There are two types of quay cranes.

- Single-trolley cranes
- Double-trolley Crane

These trolleys move with the cranes and are equipped with spreaders which are used to pick up a container. New spreaders allow two containers to be picked up at a time. Usually single trolley are used at the CT, use to move the container to/from ship and these are man-driven. Dual-trolley is advanced and available at few terminals. The main trolley moves the container from ship to platform while secondary trolley moves the container from platform to shore (secondary trolley is automatic).

Performance of quay crane depends on type of crane. The technical performance of quay cranes is 56-60 boxes/hr and in operation the performance is between 22-30 boxes/h. Secondary category of cranes is applied to stack where there are three types of cranes:

- Rail Mounted Gantry Cranes (RMG)
- Overhead Bridge Crane (OBC)
- Rubber Tiered Gantries (RTG)

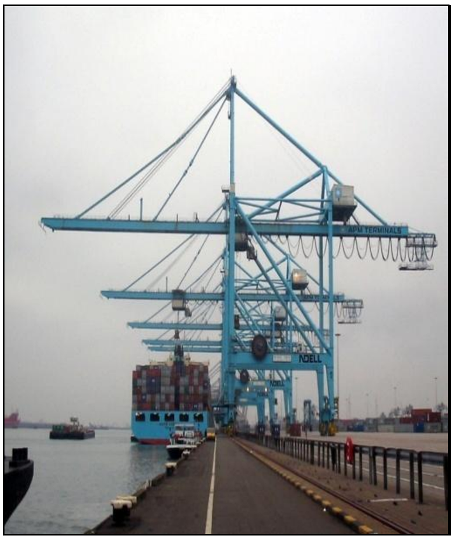


Figure 5: Quay Crane

Figure 6: Stack Crane

5.5.1.3 Straddle Carriers

It's special type of vehicle used for stacking container in yard and in port. They can also stack the container horizontally and vertically. These are very powerful machines lifting the container up-to 4 high. There speed is about 30 km/h.

Figure 7: Straddle Carrier

Max Handling Capacity: 95.000 kg
Width: 6.000mm
Overall length: 11.250 mm

Max Height incl cabin:	9.170 mm
Max Lifting Stroke:	1.800 mm
Max Ground Pressure:	10 bar
Driving System	Hydrostatic
Braking System:	Hydrostatic + Disc Brakes
Parking Brake:	Mechanical

5.5.2 Container Terminal Policies

CT is a very complex business process. There are special business units working on specific tasks. Terminal manager make decision for each of unit in terminal. Therefore CT manager needs real time decision support system for planning of terminal and for individual unit. There are special decision support system specially designed for CT to help in taking and planning for the terminal. Scheduling of berth in efficient way is important and good for customer satisfaction and allocation/usage of resources efficiently. Ship arrival and departure at berth should be kept in mind when scheduling of ships is made. It is important because any delay in the departure of ship from one port will affect the arrival timing of the ship at the other port. After repeated delays the ship suffers from substantial delay at the final port. This overall process effect containers that are to be transshipped at other vessels, moreover the reserve place for the container can be also disturbed which will affect the over movement in the terminal and extra time needed to manage the late arrival of container at the terminal. To overcome this critical situation, terminal manager need such system which will plan for a ship arriving to berth, time spent at berth, berth allocation problem, crane assignment and etc.

In order to perform operations effectively, there are policies implemented at various levels to treat ships equally. To plan a sequence of ships i.e. which ship to serve first and which ship to serve last, there is a set of policies that are called Ship Sequencing Policies. To decide a ship sequencing policy, it takes many things to consider about before a ship is sequenced ahead of the other. Those things include the number of containers a ship is carrying, the ship arrival and departure time etc. The ship sequencing policy includes FCFS (First Come First Serve), SJF (Shortest Job First) and MCF (Most Containers First).

Similarly to place a ship at a right berthing point, there are Ship Berthing Policies. Many factors are considered during these policies to select a better berthing point which can result in maximum productivity in a short interval of time. These policies include BCS (Berth Closest to Stack) and OTS (Overall Time Shortening). There are also policies to place containers at right places in a storage yard. This is the job of Yard Manager to decide a proper Yard Stacking Policy which can result in fast loading/unloading from the storage without any barriers and problems.

5.5.3 Berth Allocation Problem

Berth is a place where ship arrives from other ports. Every berth provides the appropriate berthing place for a ship. Time spent by ship and departure of ship, total cost on ship, resources to be used to serve ship. Terminal manager is responsible for this planning and usually all this planning and resource allocation is done before arrival of the ship. So that ship is served according to its given time frame, any delay in allocation of resources, serving the ship or allocating the inappropriate berth to a ship may affect the overall progress and activity of ship. According to [27], there are two types of berths problem:

Static Berth Allocation Problem How to minimize and exchange of container between vessel and yard and to moor the vessels on designated location along the terminal?

Dynamic Berth Allocation Problem How to use real time planning system at berth to allocate berthing spaces to vessels without effecting long term throughput?

5.5.4 Crane Assignment

Assignment of crane is very critical and important decision made by terminal manager. First of all terminal manager has to look how many crane are required to serve a ship within its given time. Then he has to see how many cranes are available to serve it. Also he has to look the accessibility of crane to ship at berth and is it possible to exchange crane between berth at the terminal or not?

5.5.5 Container Sequencing and planning

Upon arrival of ship at berth, the CT manager sees what will be the best shipping line to be assigned to desire ship. This decision affects the overall progress of container placement and stowage plan. According to stowage policy, loading and unloading of container, inbound and outbound of container are determined. Container attribute such as type, weight, and destination are determined first, based on these attributes CT manager decide which container is to be place in which slot. If decision of container placement is wrong then it will heavily affect loading and unloading sequence of container at terminal. If containers are placed in incorrect sequence it will heavily affect allocation of resources such as yard crane and vehicles scheduling for loading/unloading container [27].

5.5.6 Container Stacking and Storage policy

Placing of container at storage places is done according to container stacking policy. In this policy containers are placed according to their types and retrieval. Also it’s important that containers leaving early should be such placed, so it’s easy to retrieve them without wasting time on reshuffling of containers. Dedicated and reserved areas aren’t mixed and also import and export container storage area. According to [27], in Europe large CT stores total of 10,000 containers for 3-5 average days and 10-20000 turn over daily containers.

6 Simulation Model, a Software Architecture

This chapter describes the main results i.e. a computerized CT model, with the help of computer modeling language. A real scenario is explained with the help of diagrams that can be implemented by any programming language.

A simulation model is a diagrammatic representation of software which can perform simulation based on some criteria in order to collect results for few questions. The model can be implemented with the help of programming languages and results can be achieved by executing the software. The purpose of this thesis is to create one such model of the same kind which is based on a multi agent technology. The MAS technology enables to divide software in pieces in such a way that each piece is represented as standalone software which is responsible for a specific task. In this way many small modules work together in order to achieve one common goal. Each module is named as an Agent and they are designed independently. The Agents and their roles are explained below with the help of diagrams.

6.1 Agents and their Behaviors using UML Diagrams

As discussed in earlier chapters, a CT is composed of many actors that keep exchanging messages between one another, provide updates about the resources and hence performing the tasks. In a computer world, these actors can be called as Agents. Every Agent is a computerized module that is responsible for few resources to be handled efficiently and effectively. These modules are designed in UML and their roles are defined using different diagrams. The work flow in a CT can be presented with the help of an Operations Diagram. The Operations Diagram shows all the activities along with the Actors (Agents) that are discussed below.

The figure 8 shows the Agents that work at a CT and the way they communicate by passing messages to each other. The communication flow has been labeled with numbers that shows how a communication originates and where does it end. The total number of Agents involved at a CT is six and they are represented with different icons in the diagram. The step by step messaging according to diagram is as follows.

1-Ship Agent sends its information and properties to Stevedore Agent. The information Include Ship Type, Ship Name, Ship Company, Length and the Number of Containers to be loaded/Unloaded etc.

2-The Stevedore Agent transmits this information to the Terminal Manager Agent. The Terminal Manager Agent performs few steps to identify a best sequencing policy for the ship (if there are already ships in a queue) along with a berthing policy based on its demands. The demands are like Desired Service Time and Average number of Cranes Needed etc.

3- The resources are allocated at the same time for the Ship. The response is sent to the Stevedore Agent by Terminal Manager Agent.

4-The Crane Agent is then ordered to Load/Unload the Ship.

5-The Crane Agent asks its allotted Straddle Carrier Agents to start shifting the Containers.
6-The Straddle Carrier sends and receives messages to CA to decide a best route for its destination.

7-Keep on exchanging the status i.e. Busy, Idle or Done etc.

8-SCA works as an attached entity to CA.

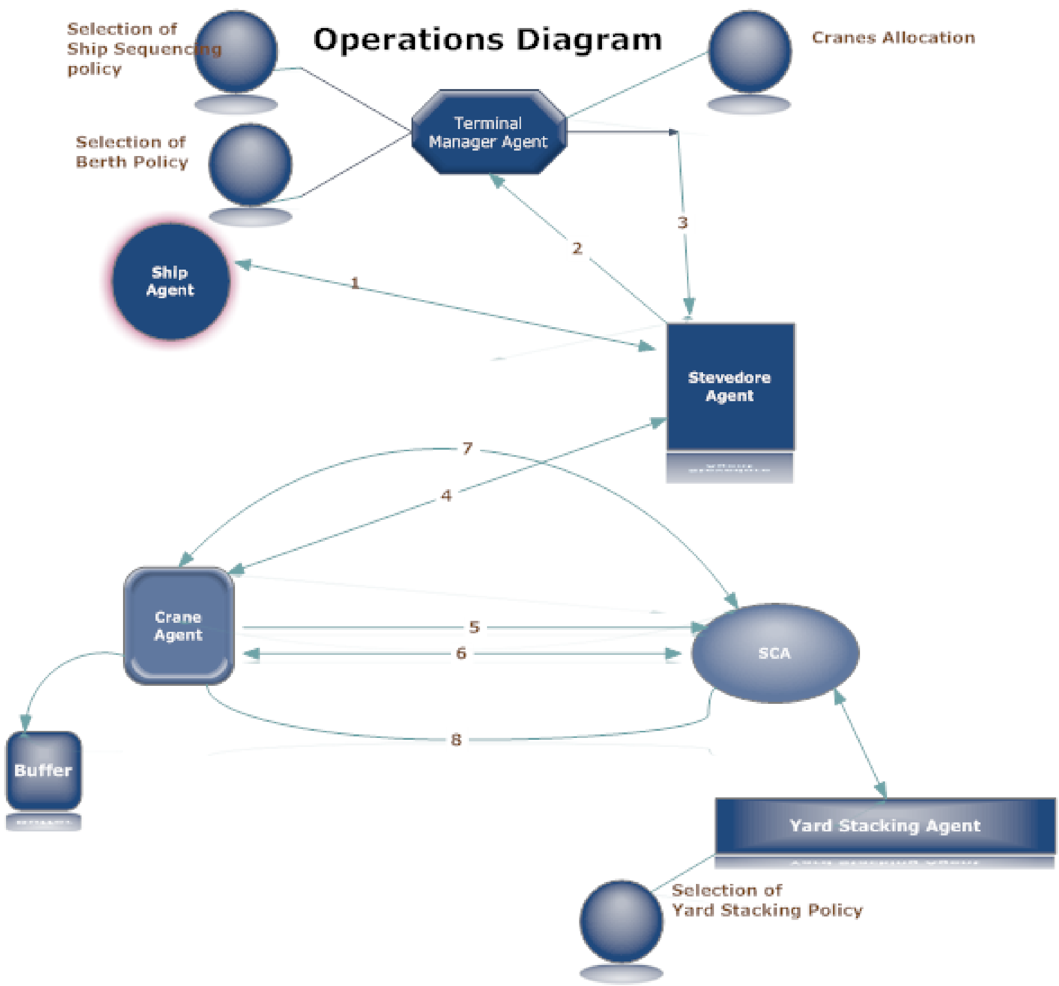


Figure 8: Operations Diagram

6.1.1 Terminal Manager Agent

The Terminal Manager (TM) Agent is responsible for the main decision making of the Terminal Container. As discussed in above section, the TM Agent is responsible for deciding a Ship Sequencing Policy, a Ship Berthing Policy and then allocation of resources. The TM Agent Communicates with Stevedore Agent to fulfill demands of Ships as well as take decisions in the interest of Port that can earn profit and produce capital.

Figure 9 shows an Activity Diagram for the TM Agent that represents different activities, entities and actions with different icons and different colors. The TM Agent starts by selecting a Port and identifying the Port resources that are available. The policies are identified that are to be tested. The execution continues with a receipt of signal from Stevedore Agent which announces the arrival of a ship and presents the demands of that ship to the TM Agent. The TM Agent starts to inquiry the ships already in the queue, finds the free resources available as well as the berth spots allocated and those that are free. After getting the necessary information, the TM Agent continues to process the Ship Sequencing Policy as well as the Ship Berthing Policy for that specific ship. The resources allocated are marked in simulation process and finally a profit made per ship is calculated for that ship.

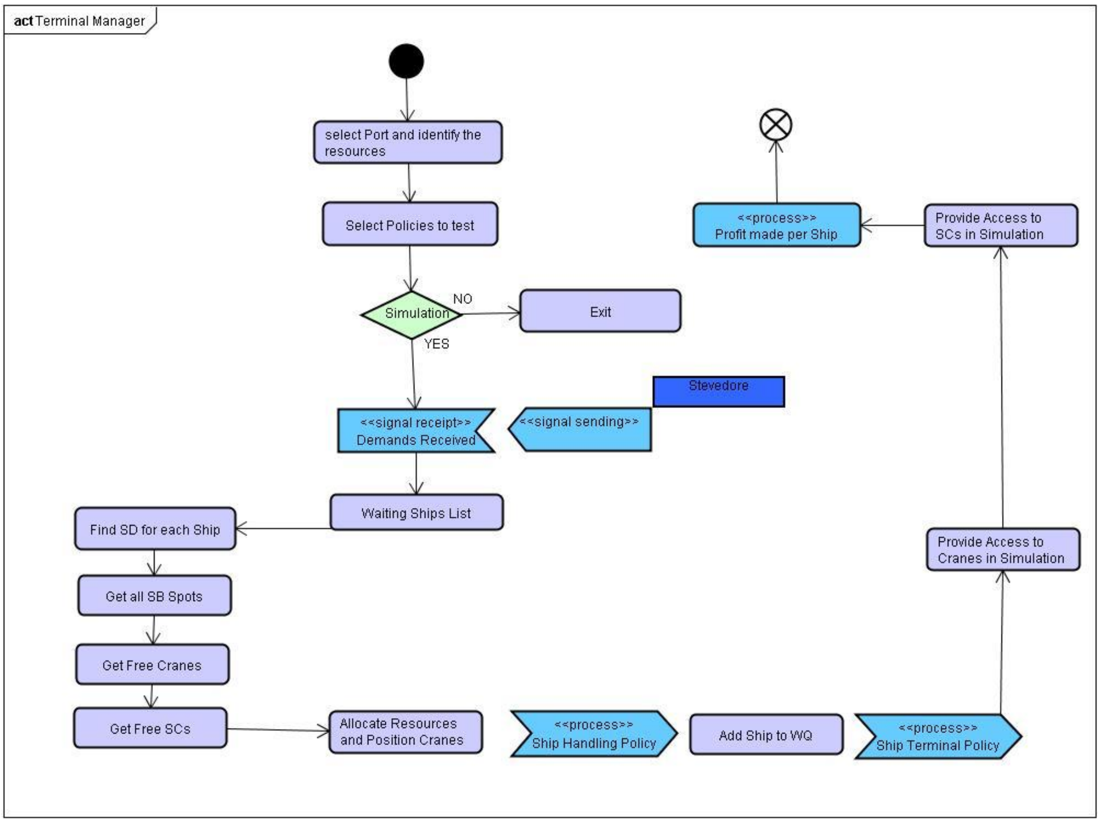


Figure 9: Activity Diagram for TMA

6.1.2 Ship Agent

A Ship arrives or leaves a Terminal loaded with Containers. The arrival or leaving of ships is happening throughout the day, that's why this module needs to be active in a computerized model. Hence, a Ship Agent in the simulation model is going to wait forever and constantly looks for the arrival of ships. If the arrival of a ship is noticed, an object of a Ship Agent is created, it is scheduled first and then a Stevedore Agent is assigned to it. The Stevedore Agent then communicates with TM Agent on behalf of the Ship and demands for the resources for Ship. After the demands are received, the Crane Agent starts to perform its task.

The Waiting Process is be programmed in such a way that it should not eat up all the computer resources. Further in order to schedule the ships, the previous waiting queue should be available to better place the ships in right order. Further the demands requested from the Stevedore Agent will be accompanied by some information that will be needed by TM Agent to process the ship. This information is described in the explanation diagram for Operations Diagram in Figure 10.

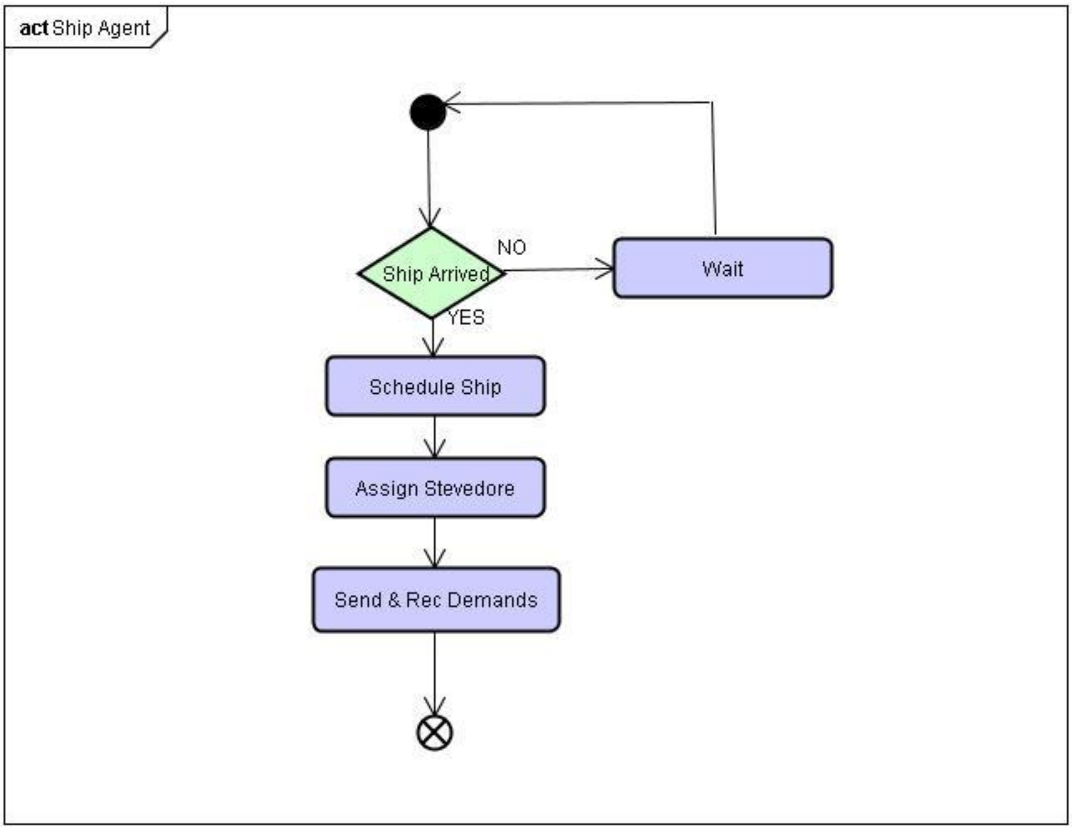


Figure 10: Ship Agent

6.1.3 Stevedore Agent

The Stevedore Agent is responsible for communicating between a Ship and a Terminal Manager Agent. It must be kept in mind that it is a Stevedore Agent that represents a ship to a Terminal Manager Agent. Whenever a Ship arrives, it sends its demands (minimum service time and minimum waiting time) and information (ship type, company, number and type of containers etc) to its allotted Stevedore. The Stevedore receives the demands and calculates the resources needed to fulfill those demands. The calculation considers many things including the speed of Cranes in a CT. This way a Stevedore Agent is in a better position to calculate the accurate number of Cranes needed to serve the ship for a specific number of containers. A desired service time (time to load/unload the containers from a ship) can be achieved in this way. The lower the service time of a ship is, the maxim is the productivity (handling of ships) of a CT and hence a good profit can be earned.

In Figure 5.4, the same operations have been explained with the help of an activity diagram. A ship arrives and Stevedore Agent receives the demands (and necessary information). A calculation is performed which results in finding out the required number of Cranes to service a ship. The achieved results are then sent to the Terminal Agent which allocates the resources to the ship. The Stevedore notifies the allotted Cranes and their assigned Straddle Carriers to start moving the Containers.

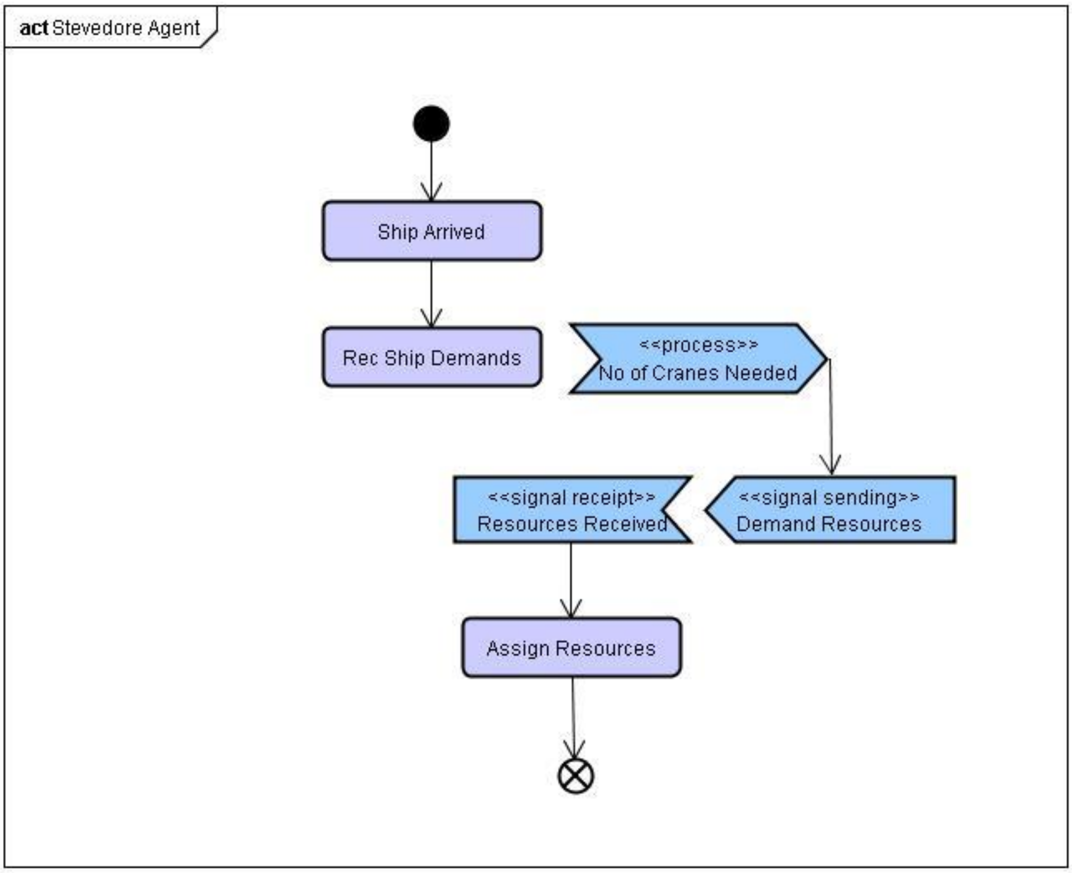


Figure 11: Stevedore Agent

6.1.4 Crane Agent

The Cranes at a CT are used to lift the containers to and from the ships. The Cranes are of different types and different speeds of moving. Hence it affects the decision making process of TM Agent when it allocates the resources to a ship. Each Crane Agent is provided with a number of transporter equipment, e.g. Straddle Carriers that are ordered to serve the Crane. One important task of a Crane Agent is defining a destination route for Straddle Carriers so that they have to cover the least distance possible to the destination yard and hence a desired service time for a ship can be achieved. In case of a longer and improper route, a Crane can be left with a waiting status which wastes the overall time of Cranes and hence resources can be miss-used. The route definition is decided in collaboration with the Stevedore Agent.

According to Figure 12, a Crane Agent receives a list of containers from Stevedore Agent after when it is assigned a task. The Crane Agent allocates its Straddle Carriers to start shifting the containers to the back yard storage place. The Crane Agent load/unload the containers as well as keeps looking at the status of its Straddle Carriers. After when the work is done, it sends a signal to the Stevedore Agent about the accomplishment of the task and waits for another task.

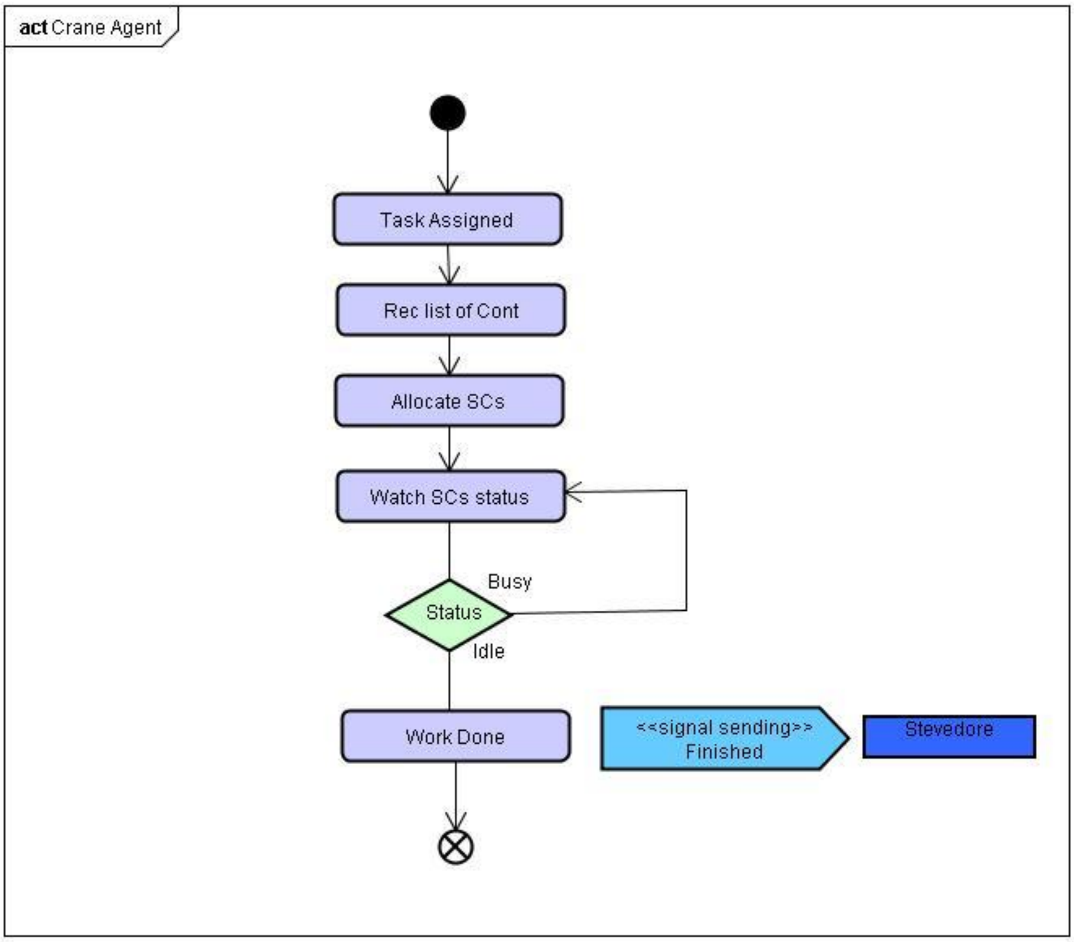


Figure 12: Crane Agent

6.1.5 Straddle Carrier Agent

A Straddle Carrier Agent works under the supervision of Crane Agent. The tasks are normally assigned to it by Crane Agents. The job of Straddle Carrier Agent is to pick containers from Crane and place it in the storage yard according to a defined policy by Yard Manager. The movement of SC really affects the overall performance of a CT, as unnecessary movements may waste the time of Cranes that can cost the Terminal Container at the end of the day.

A Straddle Carrier is explained with the help of an Activity Diagram in Figure 13. It receives a signal from its Crane Agent to start operating. The signal is received, with the help of predefined map; the stack is located where the containers are to be moved. It calculates the overall distance to find out the minimum distant route and starts moving the containers. It keeps on updating the Crane Agent about its status. After when the work is done, the signal is sent to Crane Agent and resource is marked as idle to be available for the next coming task.

While reviewing the activity diagrams, it is necessary to differentiate few symbols based on their working. A *signal sending* and *process* have the same colors and icon except with the label which declares it as a Process or a Signal. Further a *signal receipt* is represented with a symbol bending inside that states a signal is received. A conditional loop is represented with a *diamond* sign and activities have been denoted by a *rectangle*.

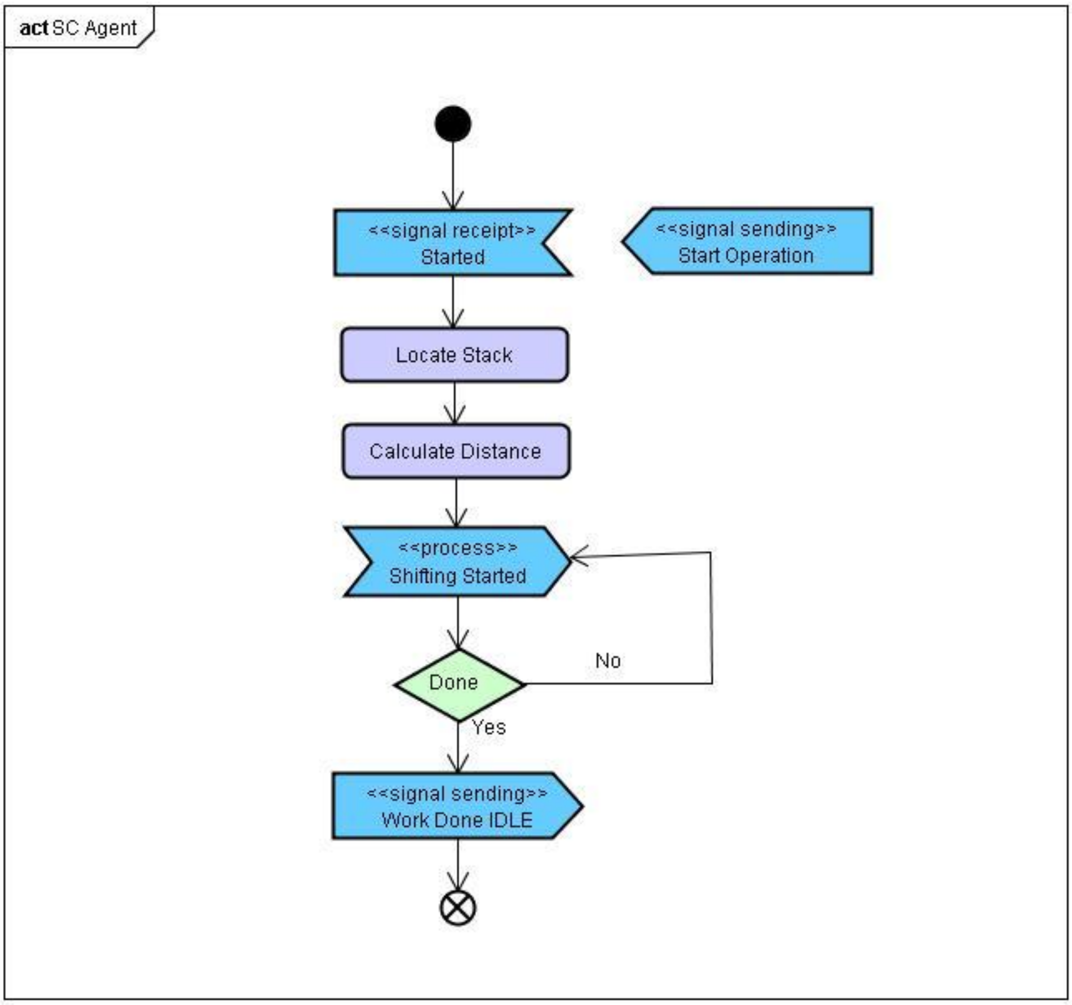


Figure 13: Straddle Carrier Agent

6.2 CT Structure Diagram

A Structure Diagram also known as a class diagram is used to present an object oriented hierarchy between the classes of a model system. We can call that as the relationships between the objects as well. Further it is also used to display the types of elements and their instances which can be called as methods and attributes in a more computer specific terms. Normally these types of diagrams are required when there is a need to model the current structure of a real life business, along with its assets and resources.

Figure 14 represents the same class diagram that describes the entities along with its attributes and methods. The diagram also shows the relationships among the classes. The diagram will be later implemented by java. The diagram shows the earlier discussed things in more of a technical way. As we know a Ship class is used to create ship agent when a ship arrives. Each ship having its own schedule (arrival, time, departure time and loading/unloading of containers). Ship Agent created for every ship communicate with each other in order to use it own allocated resources (cranes) and to avoid any clash between assigned cranes to individual ships. The communication is done with a signaling method that is possible in a java threading technology. Further, a Stevedore Agent must also know that which gantry Crane will be used for loading/unloading and also the sequence of the container in terminal. This agent communicates with other agents in order to minimize the idle time during operation and works on the ship. A Terminal Agent creates a slot for ship when a ship arrives. Terminal Agent decides the shipping handling policy and sequencing policy. Terminal Agent is provided with all the basic info of the ship, upon which all the decision and placement for the container in/out are being made. A Crane Agent is used to load/unload container from bay. Crane agent receive list from stevedore agent, upon which crane agent look for the appropriate straddle carrier agent to move it to the desire place. Crane agent moves containers in an efficient way and straddle carrier move to/from stack. Every crane is assigned a Straddle carrier, so every straddle carrier agent has a crane agent. This straddle carrier agent moves by communicating with the crane agent. When it receives instruction from crane to move from one to another place it moves. It communicates when moving to a location and communicates back when moving ahead to another position. These working have been implemented in the form of methods (or procedures) for each agent.

For example a Stevedore Agent has implemented a method named "*AllocateResources()*" which actually transfers all the resources received from a TM Agent to a Ship Agent so that it can be served. Similarly a Crane Agent is assigned with a method named "*Work_Assigned_to_SCs ()*" which actually assigns a work to its allocated Straddle Carriers just like a real environment. The Classes are also populated with the required properties (attributes) that are also considered in a real environment. For example one important property of a Ship is its *Length* and of course its *Type*. The length and type of ship enables the TM to decide a proper berthing spot for a ship that is available at the spot otherwise to make a ship wait until a proper shipping berth is available. Another important attribute could be a Cranes *Speed* which can be found in a Crane Agent Class. This property enables the TM Agent to properly allocate a Crane at its best available place.

This Class Diagram represents the simulation portion of the software where Agents collaborate with each other based on few message passing techniques. The signals are sent in order to notify each Agent of an activity e.g. if it is done or pending? Further each Agent has extended a main parent Agent Class that holds the common behavior for them. The above relationships are mostly implemented with the interfaces of java which gives the opportunity to a programmer to implement all the methods provided in an interface in a class to make a satisfying object of it.

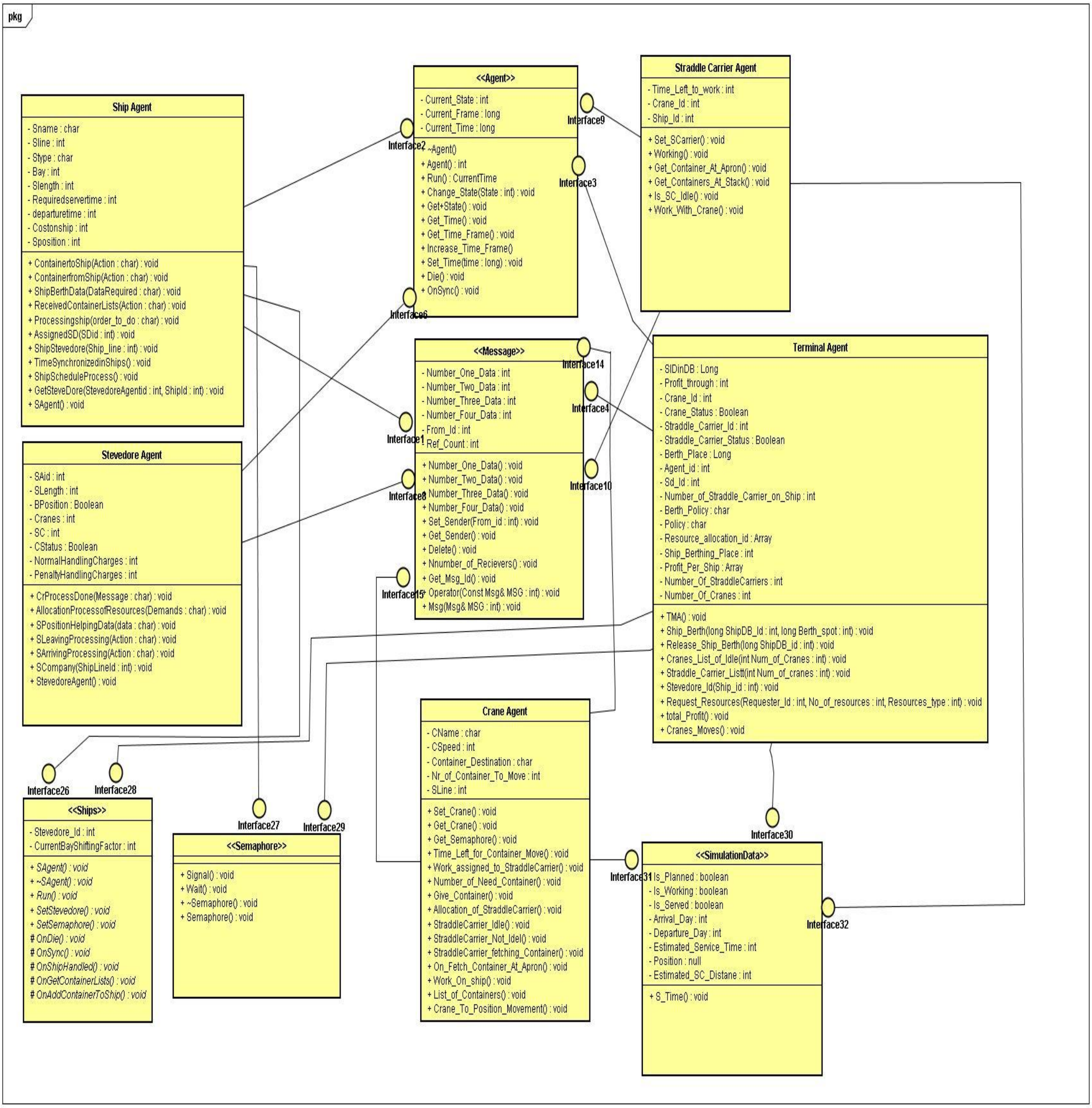


Figure 14: Class Diagram for Simulation Process

6.3 Container Terminal Entity Relationship Diagram

An Entity Relationship Diagram (ERD) is a model (or diagram) that identifies the concepts (or entities) that exist in a real environment business (or system) and the relationships between those entities. An ERD is often used as a tool to visualize a database. Each entity represents a table, and the relationship lines represent the keys in one table that point to specific records in related tables which we call a relationship of Primary and Foreign key. ERDs may also be more abstract, which means that it is not necessary to represent each and every table (or entity) in a database but only those that are vital in order to show the relationships and concepts are enough. This ERD is of the same type, even though most of the entities are identified and presented in the model along with their relationships.

The CT ERD is a combination of graphical symbols that identifies the relationship between the entities of a database. It shows the properties as well as the entities to be presented in a specific manner. The whole diagrams of this thesis have been designed in a UML modeling software named *Jude Professional.* Due to a technical limitation in the software, the relationships in the ERD could not be named using a special *Diamond* symbol but it is still worth, serving the purpose of understanding, efficiently and effectively. The boxes in this diagram represent the entities. Each box has the attributes in it. These attributes can be separately represented with the help of an Oval symbol which again was unfortunately not provided in the Software, hence used a much simpler form of the Diagram.

The Diagram provides the resources and objects involved in a CT in the form of entities e.g. Ship, Terminal, Crane, Company, Stack etc each having some properties. The entities are linked with a primary and foreign key relation in between having different type of cardinalities involved. Having relation between entities at different level bounds them to a certain limit creating dependencies among them. These dependencies enable the computer developer to work under some boundaries. This can be explained with a very simple example like there is a primary and foreign key relation between a Ship and a Company (both are entities) which links a specific Ship object to a specific Company hence the dependency involved bounds the user (or developer) to some predefined criteria.

The figure 15 provided is a CT ERD which was later implemented in MySQL database. The Figure represents almost all of the resources that are available in a CT. There are sub-entities also provided to a main entity e.g. a *Bay* is a sub entity of a *Ship*, as you can have many bays in a ship. Further another example of a sub entity defined here is a *Ship Type* for a *Ship*, which is used to identify a type of Ship.

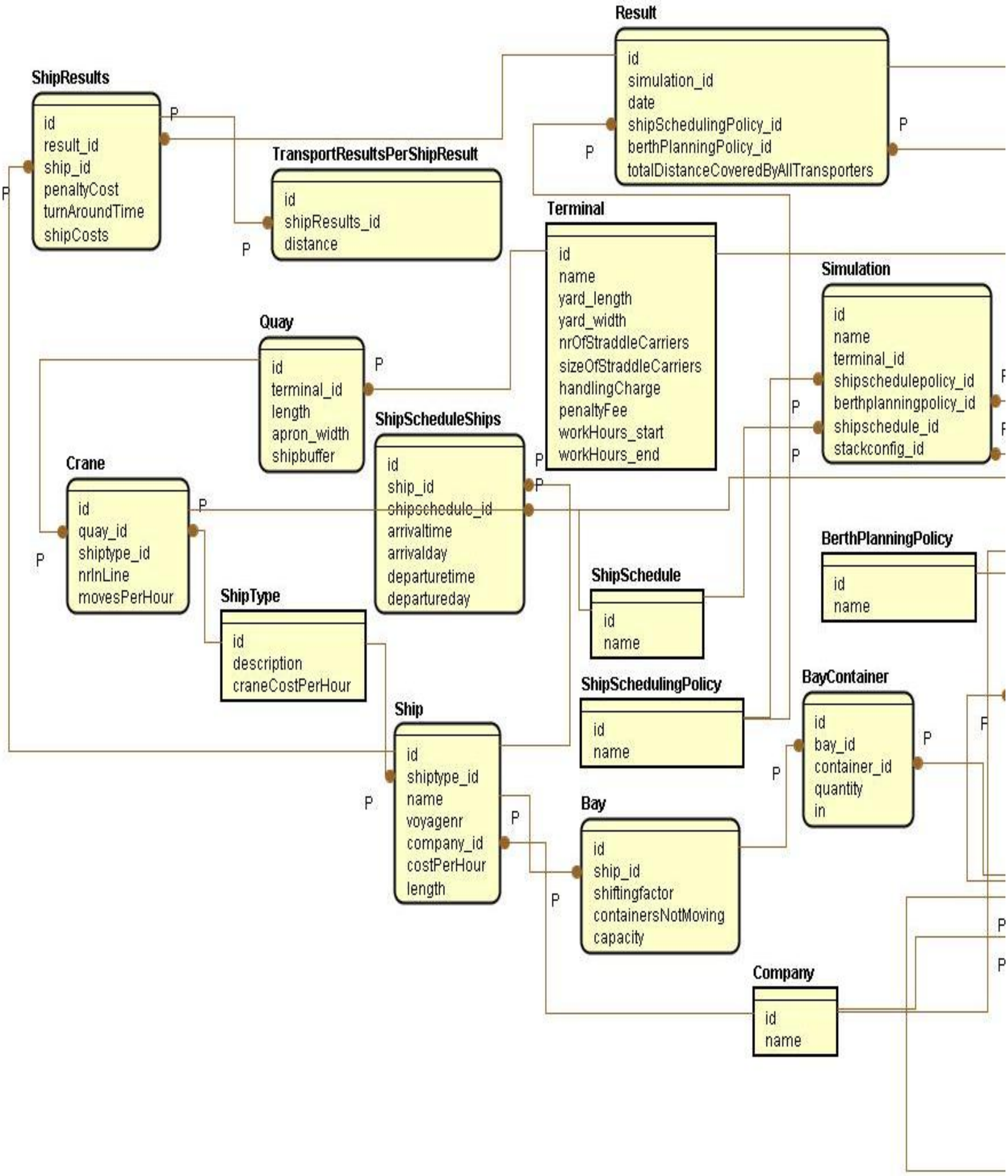
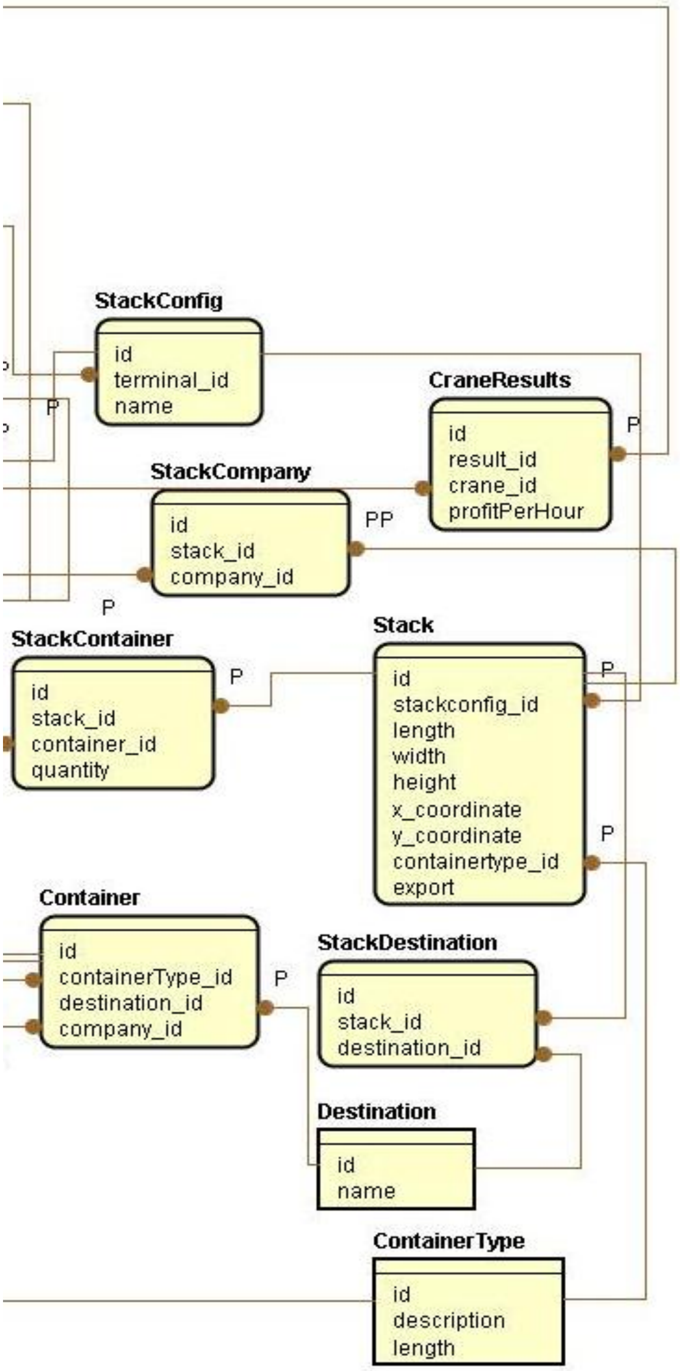


Figure 15: CT ER Diagram


StackConfig
id
terminal_id
name
CraneResults
id
result_id
crane_id
profitPerHour
StackCompany
id
stack_id
company_id
StackContainer
id
stack_id
container_id
quantity
Stack
id
stackconfig_id
length
width
height
x_coordinate
y_coordinate
containertype_id
export
Container
id
containerType_id
destination_id
company_id
StackDestination
id
stack_id
destination_id
Destination
id
name
ContainerType
id
description
length
P
PP

7 SIMULATION MODEL, A TECHNICAL ARCHITECTURE

This chapter gives an overview of what has been done to implement a Web Based simulation model for management of CTs. Its gives a detail of how the software model presented in chapter 5 was actually implemented using different technologies. The technical findings are all addressed in this chapter.

Figure 4.1 in chapter 4 illustrates a Research Methodology followed throughout in this thesis. The 3rd phase i.e. proposed solution of a model was presented in chapter 5, which was a result of thorough literature review and then, finally the work headed to a practical implementation of the proposed model for the CT system. It is necessary to point out that an implementation to full extent could not be achieved due to the time factor. The screen shots of partial implemented system can be founded in Appendix.

7.1 Multi Agent Platform

The Multi Agent platform selected for the development of this model was J2EE which provides a 3-tier solution i.e. Client, Server and Database. As this was proposed to be a web based system so the client side implementation consists of HTML, DHTML, CSS and JavaScript. In order to be able to communicate with the java machine, the JSP technology should also be considered as the Client side technology here. It must be kept in mind that JSP being referred as Client side does not mean it is a normal client side technology but for simplicity the Agents are termed as Clients in this thesis and a main TM Agent is termed as Server which is responsible for actually decision making. Hence the Agents implemented with HTML, DHTML, CSS, JavaScript, JSP and java beans can be called as Clients here and the main Terminal Manager Agent machine as a Server.

In order to be treated as Agents, the java gives a decent technique of java Threading. The threading allows a programmer to implement stand alone modules that run as a separate program and can communicate with other instances or objects running at the same time. The other instances may communicate with the help of messaging technique or signaling passing depending upon the need and skill of developer. A technique called thread signaling is more suitable for this purpose which enables the threads to wait for any other signals that may come from the other threads already running. In a simulation process of a CT, it can be explained in a way that a Crane Agent is in a *wait ()* status unless it receives a *notify()* after taking the control over its *lock* in a *synchronized()* method. In order to call all the Straddle Carriers associated with that Crane, it then calls with a *notify all()* to make them start functioning all at the same time. Similarly in order to start functioning, a class has to implement (extend) the *runnable* interface which has to implement the method *run()*. So in the context of a CT, a Stevedore Agent implements a *runnable interface* with a *run ()* that receives all the demands from a ship Agent using *Receive_Demands()*. The Agent starts functioning by calling its *new Thread(new StevedoreAgent()).start()*.

The backend storage is implemented with a database named MySql. The database was created using the Entity Relationship model developed earlier and presented in chapter 5. The number of total entities was 28 and most of the tables were linked according to the relationships defined in the model. The biggest advantage of using MySql over any other database was due to the fact that it is freely available and hence can be used for small projects without any problems.

Apart from the above stated technologies, there were few more software and utilities used to catch up with the task. Hence the *Apache web server* was installed to run the web based platform on the local machine. Further the developing editor used was the *Eclipse* that is provided with all sort of plug-in to support any working environment.

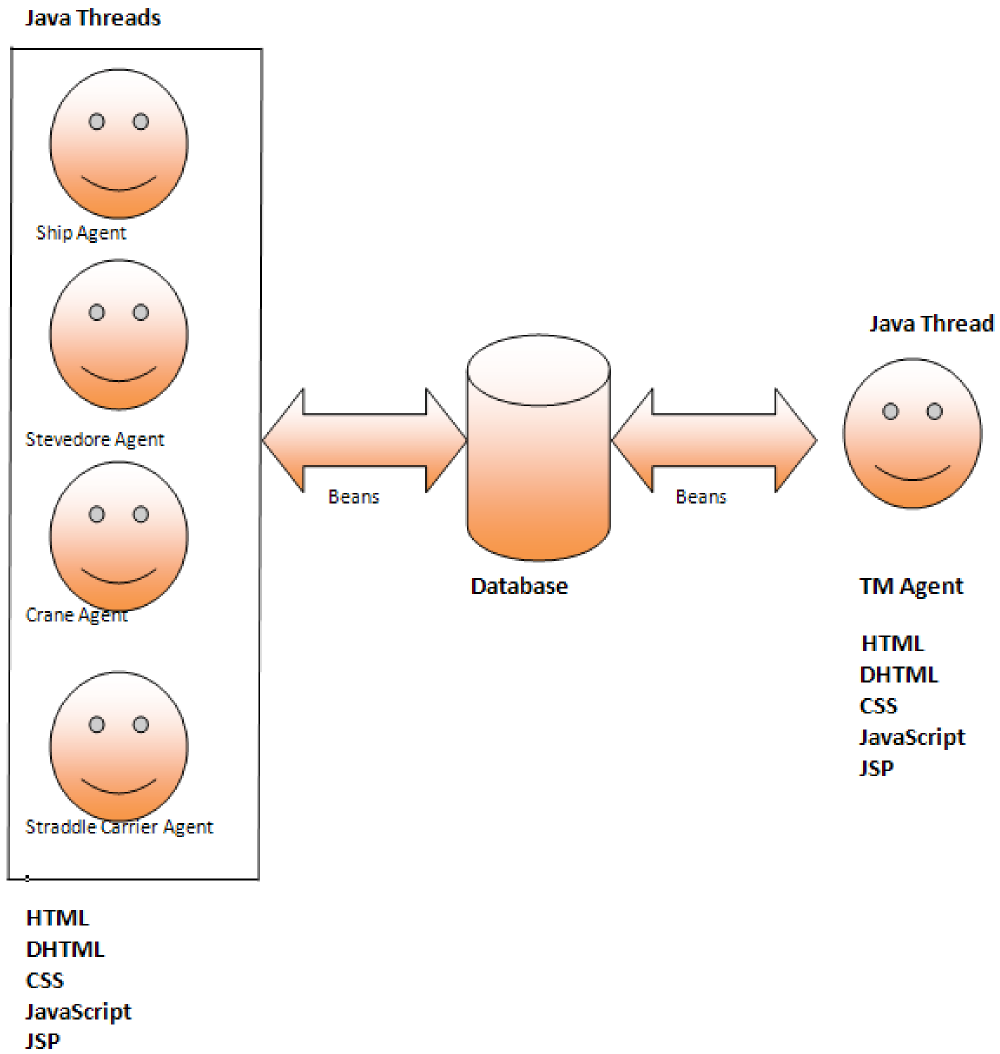


Figure 16: Web Based System Implementation Details

7.2 System Functionality

The Multi Agent Web Based Simulation Model is not fully functional as there was not enough time unfortunately to complete the implementation phase. Hence the main goal cannot be achieved but it still laid down a solid background for being implemented in a very short period of time. This section discusses how the system is functioning and how the system is expected to function under different circumstances.

There are different options provided to the TM i.e. to create a Terminal, to create a Ship or create a simulation etc. A Terminal creation process involves the creation of further small entities that this entity depends on. Those are the quays, berths and cranes. Similarly in order to create a Ship, it takes to create the further sub entities that the ship entity depends on.

Examples of such entities are Bays. The Bays have further to be classified in such an order that which bays are to be loaded on a CT and which are not to be loaded. They are also distinguished with the help of their destination address. The destination is retrieved from the destination table in database. The main task of the software is to perform a simulation which in real retrieves data from many tables to set up and start up a simulation. The simulation process is provided with a set of inputs along with the policies to test. The result achieved after performing this test is the total profit that can be earned by using a specific policy with a specific number of inputs.

7.3 Design Issues

The design issues deals with the initial mockups design for the system and the decision of what inputs to provide a user to and what results are to be shown to a user? How much information is there necessary to be shown to a user so that it makes the system easy to understand and easy to use.

The designed system offers many input screens to the user which allows the user to input the desired data for the simulation to carry out. For example, along with the information needed, the *Create Ship* option requires to create *Bays* as well in the ship so that containers can be stored in those bays. During creation of these bays, each container is marked with its *type*, the *destination*, the *owner* and of course the *quantity*. Except from the last attribute i.e. quantity, the others are predefined and stored already in the database. So the data is retrieved from database, shown in combo boxes to the user to select the desired data. This way a developer reduces the amount of input writing for the user as well as lowering the factor of error that may occur due to a wrong data input.

The Design flow of the system is not that much complex for an average Port Terminal Manager. In order to run a simulation, the first task is to create a Terminal by clicking on the *Create Terminal* Option. At this stage, the name of the Terminal, the number and size of straddle carriers, the yard length/width, along with some more data is entered. The next task is then to create a *Quay*, as a Terminal must have at least one quay. During the creation of quays, the berths can also be created and identified with ids. The Cranes are also created at that stage. The next task is to create a *Ship* and create *Bays*. The *Ship Schedule* is created along with the yard configuration. The data entry is completed at this stage and hence a simulation can be created and run to see the results. It requires selecting few policies during the simulation process for which the simulation is going to be tested.

The front end designed is a lot more simple and easy to work with. The different sections have been separated by frames which allow the user to differentiate between different sections. It was tried to implement the 7 HCI (Human Computer Interaction) principals to make the front end of the software good looking and attractive. The code for the front end designed is a lot more complex because most of the advance level JavaScript techniques are used to create dynamic tables and make the data appear or delete at the run time. This makes the front end more attractive because of its dynamic working way. Further the pages have been mostly programmed in JSP with implementation of beans so that it can interact with database effectively and efficiently.

Few screen shots of the software is provided in the Appendix section, along with the pseudo code. The software still needs to be worked on and it can really benefit a port industry if given some importance. The prototype developed can be implemented in a computer programming language and a simulation tool can be developed which will help the CT manager to take important decision. Moreover partial implementation of the prototype is provided which helps the CT manager to creates/manage (Stack, bays, companies, yard, terminal, quays and e.t.c.).

8 DISCUSSION OF THE MODEL

This chapter discusses the working of the model. A scenario is evaluated based on the concepts expectable in real environment which is explained in detail in this chapter.

As discussed in earlier chapters, the model is presented with the help of different UML diagrams. Those include the Activity Diagrams for Agents involved, the Class Diagram for the simulation process and the ER Diagram which shows all the entities present in the system. The activities are then implemented in a computer language based on the diagrams and terminologies presented in the model. The purpose of this chapter is to evaluate the model by considering a main scenario and present its working by breaking up it in small parts.

The scenario that has been selected for this purpose is the simulation process. This is the main process of the model and hence explaining this scenario will help the developers in future to implement it easily with little effort according to the real scenario. This can be considered as a major contribution of students towards their master's thesis.

8.1 Scenario Based Evaluation of the Model

According to the above section, the main simulation process has been selected as a scenario to be discussed as it is the heart of the thesis. The whole model revolves around this process as the purpose of the model itself is to present an environment for a simulation process. The discussion of the scenario is divided in two sections i.e. a favorable condition and a worst condition. The favorable condition discusses the scenario in normal conditions without any problems or hurdles. The worst condition discusses the scenario in cases where there are chances for error occurrences or where there is a chance of a problem that can affect the simulation process.

8.1.1 Favorable Condition

The favorable condition discusses the scenario in normal state where there is no threat of problems or error occurrences. It gives a brief over view of what things are going to be processed and what are the inputs and what outputs are expected after the results of this process. It also describes the dependencies involved in the system along with the storage mechanism of the data. The process comprises the implementation detail of the entire Agent models along with the ER diagram and Class diagram, which makes it even more important to understand.

8.1.1.1 Input to the Scenario

The simulation process takes its input from the database. All the necessary entries are already gathered in the database using an affective user interface from the user. These entries are then processed according to a process. These entries include data regarding the ship, bays, stacks and policies etc. The user interfaces are provided in the appendix section of this thesis which shows many input places where the user is requested to insert necessary data.

8.1.1.2 Dependencies involved in the Scenario

As this is the main simulation scenario which is going to be evaluated in this section, so there are many dependencies involved in it. The simulation environment is dependent on the Agents to provide the required result in order to proceed with the simulation. Each Agent is allocated a specific task. The Agent performs this task according to the process defined in the model.

8.1.1.3 Process Performed in the Scenario

The process of simulation starts with setting up the policies that are going to be tested after deciding a Port. Deciding a Port first is necessary in the sense that each Port carries its own classification or resources and berths which are necessary to be identified first before the simulation actually starts. Then the policies that a user wants to test against a certain criteria are brought up to the queue. It should be kept in mind that the Ship Agent is just used to notify the Stevedore Agent about its arrival and to arrange regions for it to berth according to a certain policy, so this is the main task of Ship Agent in the simulation process. It does not interfere further which is a necessary point to be noted. The other four agents that are actually responsible to conduct the whole simulation process are i.e. Terminal Manager Agent, Stevedore Agent, Crane Agent and Straddle Carrier Agent. The objects of the Agents are then created. As we all know the whole system deals with proper time management of Ship, Crane and Straddle Carriers, so there should be a time routine class that can keep track of the times and manage it among the Agents. This routine could be a TimeSchedular which is to be used in the simulation process. After creating the Agents, the next task is to identify the resources at the Port i.e. by calling the Crane Agent to receive all the free Cranes or by calling the Straddle Carrier Agent to receive a list of all free Straddle Carriers etc. A similar process can be repeated for the Stevedore Agent by identifying each stevedore Agent for a certain company. After performing these necessary steps, the agents start working by passing messaging to each other, sending and receiving the necessary data through signals and performing the tasks that they are made for. As an output of an agent goes as an input to another agent, that's why it keep stimulating the agents itself by carrying out the required jobs according to the model. Once each agent is done with the execution, the next task is to identify all the process about the completion of the job. All the Agents are sent the task finish signals and each of them is deleted to free the computer resources that it allocated for itself. The different policies that can be tested during the simulation process are BCS (Berth Closest to the Stack), OTS (Overall Time Shortening), FCFS (First Come First Served) and MCF (Most Containers First).

The whole process is explained in figure 17, where the Agents are shown to be all connected together as well as connected to the database. Their execution is performed and output expected is displayed.

8.1.1.4 Output of the Scenario

After executing the scenario, the required results are achieved for which the scenario was executed. It results are the total cost that has been spent on a ship and the profit made on that ship. Further it also provides the total distance covered by the Straddle Carriers and the profit made by Cranes based on those travels.

8.1.1.5 Storage media for Scenario

The data for the scenario is shifted to and from the database. The final results are saved in the database but most of the data received to the scenario comes from the database which was previously received during the data input section for what the user want to test based on his/her demands.

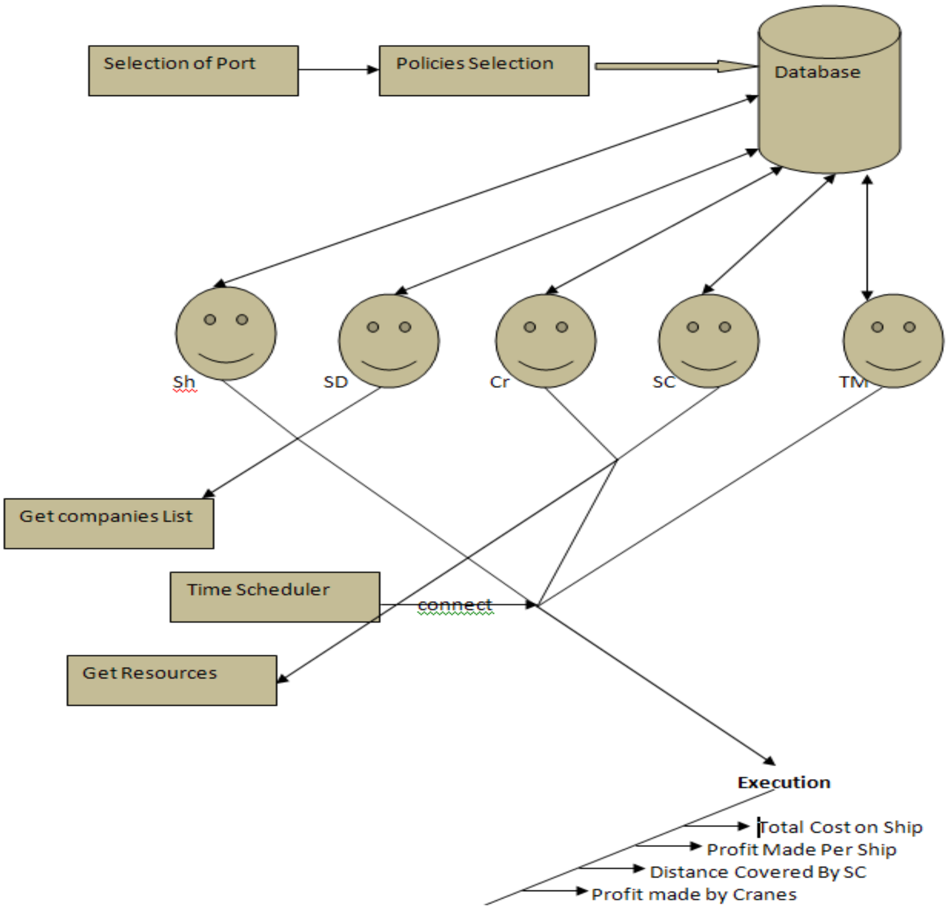


Figure 17: Execution of Scenario

8.1.2 Worst Condition

In this section, those exceptions will be discussed that can be expected during the simulation process and what could be the results if those unfortunately happened.

8.1.2.1 Problems expected during the Scenario

The scenario can be affected by badly programmed algorithms that may results in poor performance of the simulation process and hence can be considered as useless because of loose programming stuff. The scenario may also suffer from the dependency problem. As the agents depends upon each other for one output is processed as input for the other, so a wrong results of one agent may collectively produce a wrong result for the simulation process which is a danger and need to be looked carefully. Further the scenario involves a lot of processing i.e. five agents executing and processing all at once, so it is necessary to have a computer system with enough resources to be able to complete the process safely and efficiently otherwise it is a time consuming job on slow processors and take a lot of time to complete the processing.

8.1.2.2 Identification and Rectification of the Problems

Poor results of the simulation process may identify the lack of accuracy of the algorithms implemented which need to be resolved along with the previously stated wrong input leading to wrong output problem. A simple solution of this is to design and test the modules separately before they can be integrated to the main simulation process. This technique is actually a standard adopted by most of the software developers to identify and rectify the problems in a software product and it can be considered efficient for this module as well.

8.1.2.3 Effects of Problems on Dependent Systems

There are many ways a dependable system can be affected by these wrong modules. One of the issues discussed above is the overall wrong results of the simulation process. The other effects could be time consumption of the process, one wrong result may lead to a wrong input processing for another agent which can be frustrated for the developers as the process take a lot of time to complete its execution. Debugging at this stage again is another major issue as the wrong output provided by an agent could be due to a wrong input which is at times confusing and hard for the developers to identify.

9 DISCUSSION

In practice, modeling of a very complex system such as a CT requires some very great effort. Large amount of information about its current state, historical work been done and what valuable ideas/contribution can be added to it. CT is a huge and complex system so it's possible that in our proposed model there will be number of limitation and flaws. Some of them will be due to thesis limited time and other will be software/model normal problem that always remain open for other to work further in this field. A web-based simulation model was developed to assist the CT manager in planning and resource allocation. This model includes almost all the entities involved in the CT (Ship, Crane, Straddle Carrier e.t.c). Entities and agents involved in CT are designed at the best level and according to the requirements so that there is no chance that an entity/agent can produce wrong result, exceptions are always there. These entities and agents are modeled in such a way that they can easily accommodate any change in the existence system due to the dynamic nature of CT.

Requirement specification and data collection for this model wasn't an easy task, because normally companies don't give out any information because it has commercial value. Therefore our advisor Dr Henesey helped us a lot in this regard and he provides us all the basic information which we need to design the model. During this a thorough literature review was carried out some of which was suggested by Dr Henesey to understand the problem and to provide a better solution in these thesis. After all the things were done, then designing the model was somehow a very delicate and complicated task. The reason is that entities work independently in CT but are inter-dependent on each other activities. So to model this CT every unit was design independently but keeping in mind that has relation with other unit so when combined, it should work accurately.

The model we have design shows that CT capability can be expanded and its through put can also be increased by utilizing the resources efficiently. According to our design model, primary resources of CT like crane and Straddle Carrier are quiet sufficient for handling containers at the terminal, if assignment and co-ordination between agents is done properly. This at the end of operations will increase terminal efficiency and will reduce operational cost. Secondly in the model we are using MAS which in CT reduces the travelling time for cranes and straddle carriers by proper communication with each other. The web model will improve important decision taken by terminal manager for various task and activities or for future planning e.g. a schedule can be made for a ship arriving next day at the berth, which will include how much time will ship spent, arrival time, departure time, container in, container out allocation and e.t.c. Similarly container at the yard can also be placed according to its type, container moving first, container not moving, and container moving last. Container bottle neck situation can also be avoided using yard capacity calculator.

In this design model, most of the initial and important entities are programmed and deployed on web server, which the terminal manager can browse the desire page and can see the over-all activity of a specific entity. For example, if a ship is arriving to port, terminal manager can create the ship and enters it all attributes. Other examples are ship scheduling, quay, berthing, company and e.t.c. Apart of this, special agents are also created for important entities, which work independently and send it work result in a form of message to other agents or agent server. These agents are working in a form of thread and are basically an algorithm which runs separately for every thread. The overall performance and functionality will be much more, because threading doesn't use much of the resources of the system. Another good way to increase the efficiency of the system is to deploy each agent on a separate system; this system will be working under one main system.

Also they can communicate with each other if they needed it. Running simulation on a system requires much time to see the result. So if we deploy the whole model on single system this will make the system much heavier and the terminal manager will need to see the result of the system. Deploying each agent on a single system may help the system to increase its efficiency. Which will ease the terminal to see the result and will serve a ship or allocate a resource quickly rather than they are idle or waiting for the task to be assigned to them. Implementing policies (code is given in appendix), will be used for planning by the TM. TM will use these policies (First Come First Serve, Ship Scheduling and e.t.c.), simulation will start and at the end of simulation result will be produced, from which the TM will decide what the best decision in the current situation is.

Not every system in software industry is ever completed same is the case here. As we has said in the above paragraph that we have provided partial implementation of the system not the complete due to time constraint. Although we have tried, at our level the best to understand the complex system of CT, its functionality, information sharing and communication. We have gather as much information as much we could and documented them in UML(Unified Module Language), Database (MYSQL) and present it as a GUI(Graphical User Interface) using Eclipse. But still this isn't completed. To complete this system policies of CT are to be implemented. We have discussed these policies at the end of chapter 4. The pseudo-code for these policies is given in appendix.

10 CONCLUSION

This thesis was all about modeling a real business environment of a CT in to a computer system. A problem was identified, on the basis of which a thorough study was made to gain the initial knowledge. A research work was then carried out after the initial phase of data collection and the identification of goals and achievements. The research work resulted with a computerized model of a CT which was then decided to be implemented actually to see the results. The model was named as a Multi Agent Web Based Simulation Model for Evaluating a CT Performance. It was mostly based on the work done by researcher Dr Henesey and much of the technical details reflect his ideas and solutions.

Even though many Multi Agent Systems have already been proposed to solve the issues present in a CT but there was no Web Based system as yet that could provide solutions without having a dependency on platform. The contribution of students towards this thesis is the development and implementation of a Web Based simulation model implied with a Multi Agent technology which was proposed by our Advisor Dr Henesey and hence advised to start working on it as our Master thesis.

In this thesis, our main goal was to model the CT management system and we successfully identify those entities, which help the CT management system to increase its performance and outcomes. The flow of communication between every unit/entity, requesting or allocating resources, communication between each other and limitation of these units were modeled. Many of these units (ships, terminal, quays, berths and e.t.c.) modeled were deployed on the web-server, so the CT manager can see the details of all entities e.g. ship schedule can be re-arranged according to the requirements.

The UML model along with partial implementation of the system is the final contribution of the thesis which needs some future work to make it an industry acceptable product for solving the performance problems in a CT. The UML model itself can be followed to build a new implementation of the software from scratch which may result in good productivity of a CT with limited capacity.

In software development time is a critical factor, due to time constraint in our thesis we haven’t completely deployed all the functionality of the system. Although much of the work is done and it will be very easy for others to extends the current work. Policies section of CT can be deployed to extend the functionality of the current model.

References

[1] P.A. Fishwick, "*Web-Based Simulation*", Proceedings of the 1997 Winter Simulation Conference ed. S. Andradóttir, K. J. Healy, D. H. Withers, and B. L. Nelson.

[2] Y. Nam, K. Kim, D. Dong, K. Sans, "*A Gui-Based Web-Based Bimulation Environment*", IEEE 1999.

[3] N. R. Jennings, M. Wooldridge, "*Applications of Intelligent Agents*", Queen Mary & Westfield College, University of London

[4] L.E. Henesey, "*Enhancing Container Terminal Performance: A Multi Agent Systems Approach*", Blekinge Tekniska Hogskola2004.

[5] Agostino G. Bruzzone, Pietro Giribone Roberto Revetria, "*Operative Requirements And Advances For The New Generation Simulators In Multimodal Container Terminals"* Proceedings of the 1999 Winter Simulation Conference

[6] L. Henesey, "*A Simulation Model for Analysing Terminal Management Operations*", Blekinge Institute of Technology.

[7] D.W Yi, S.H. Kim, N.H. Kim "*Combined Modeling with Multi-Agent Systems and Simulation: It's Application to Harbor Supply Chain management*", Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS-35.02), IEEE 2002.

[8] Kang, S. et al., *Optimal operations of transportation fleet for unloading activities at Container Port*, Transportation Research Part B (2008), doi:10.1016/j.trb.2008.02.003

[9] C.I. Liu, H. Jula and P.A. Ioannou, *Design, simulation, and evaluation of automated container terminals.* IEEE Transactions on Intelligent Transportation Systems **3** 1 (2002), pp. 12–26.

[10] Henesey, L. (2006).*Overview of Transshipment Operations and Simulation.* Lead Roundtable Discussion Group on Improving Transshipment Operations and Simulation, 4th MedTrade Conference, Malta.

[11] I.F.A. Vis and R. De Koster, *Transshipment Of Containers At A Container Terminal: An Overview.* European Journal of Operational Research 147 (2003) 1-16.

[12] Henesey, L. (2004). *A Multi Agent Based Simulator for Managing a Container Terminal*, Proceedings of the 2nd European Workshop on Multi-Agent Systems (EUMAS 2004), Barcelona, Spain. December 16 &17, 2004, pages 291-302.

[13] Larry Edward Henesey – *Comparison and Evaluation of Two Automated Guided Vehicle Systems in the Transshipment of Containers at a Container Terminal*, Blekinge Institute of Technology, 2006 http://ipd.bth.se/lhe/PhdOutline last visited 2006-10-26.

[14] Henesey, L., Wernstedt, F., and Davidsson, P. (2003). *Market-Driven Control in Container Terminal Management*" Proceedings of the 2nd International Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT´03), Hamburg, Germany.

[15] Henesey, L., Davidsson, P., and Persson, J.A. (2004). *Using Simulation in Berth Planning at a Container Terminal*, Proceedings of the 3rd International Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT´04), Parador Siguënza, Siguënza, Spain.

[16] Rebollo, M.; Julian, V.; Carrascosa, C.; and Botti, V. (2000). "*A Multi-Agent System for the Automation of a Port Container Terminal.* Autonomous Agents 2000 workshop on Agents in Industry. Barcelona, Spain

[17] www.zpmc.com/newtech_list.asp?column_id=18

[18] S.P. Sgouridis, D.C.Angelides, *Simulation-Based Analysis Of Handling Inbound Containers In A Terminal*, Proceedings of the 2002 Winter Simulation Conference

[19] M. Rida, A. Boulmakoul, R. Laurioi, *Calibration And Validation Of Container Terminal Simulation List*, Computer Sciences Department Mohammedia Faculty of Sciences and Technology (FSTM)

[20] H.O.Günther, K.H.Kim, *Container Terminals and Automated Transport Systems: Logistics Control*, Business & Economics - 2005

[21] L. M. Gambardella, M. M.O. Lilli, A . Izzoli And M. Z. Idsiað, *An optimization methodology for intermodal terminal management* , Academic Publishers.

[22] Hawe, P, Degeling D and Hall J. 1990. *Evaluating Health Promotion A Health Workers Guide*. Maclennan and Petty, Artamon NSW

[23] J¨urgen B¨ose1, Torsten Reiners1, Dirk Steenken2, and Stefan Voß, *Vehicle Dispatching at Seaport Container Terminals Using Evolutionary Algorithms* Proceedings of the 33rd Hawaii International Conference on System Sciences - 2000

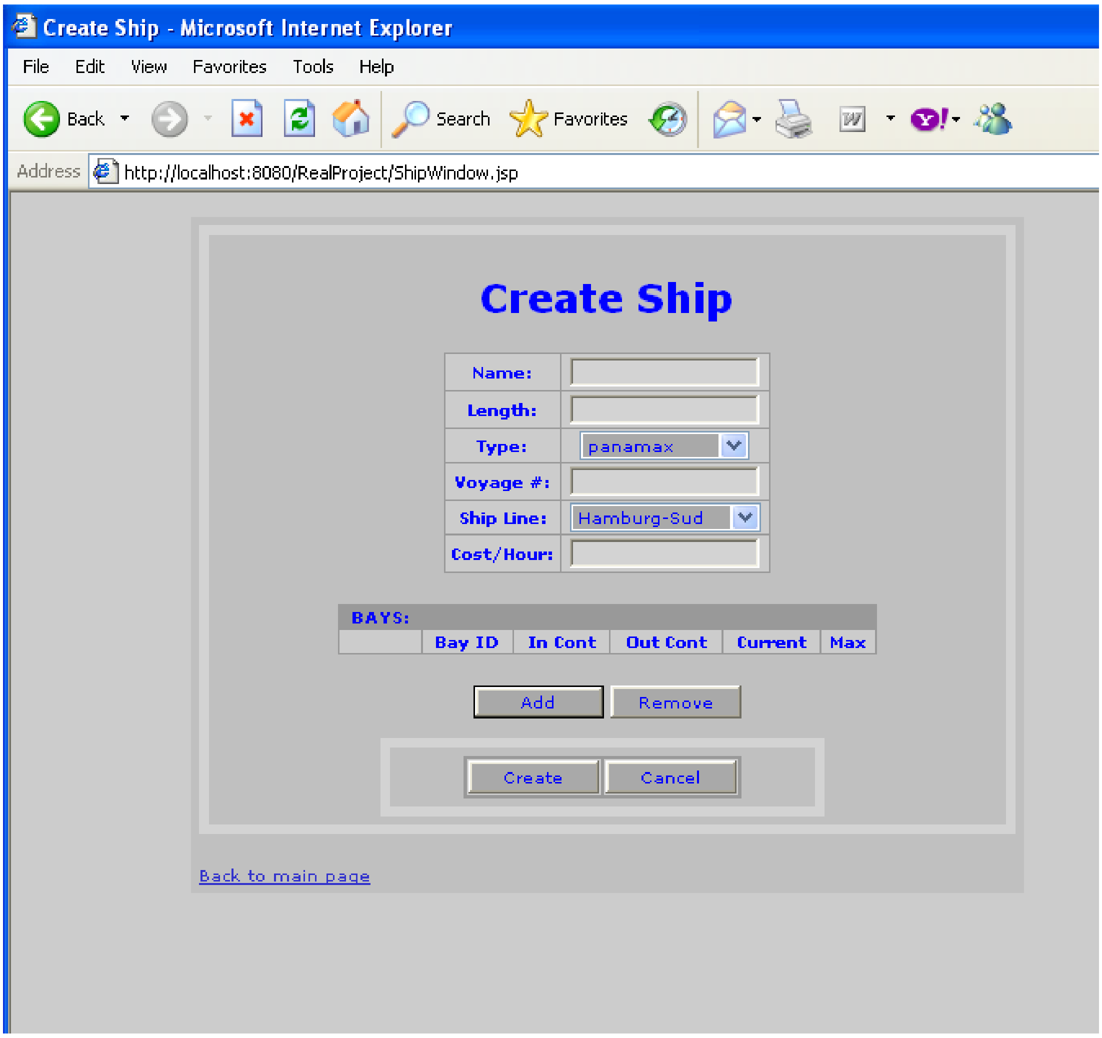
[24] Lawrence Henesey, Paul Davidsson, and Jan A. Persson *Agent Based Simulation Architecture for Evaluating Operational Policies in Transshipping Containers*

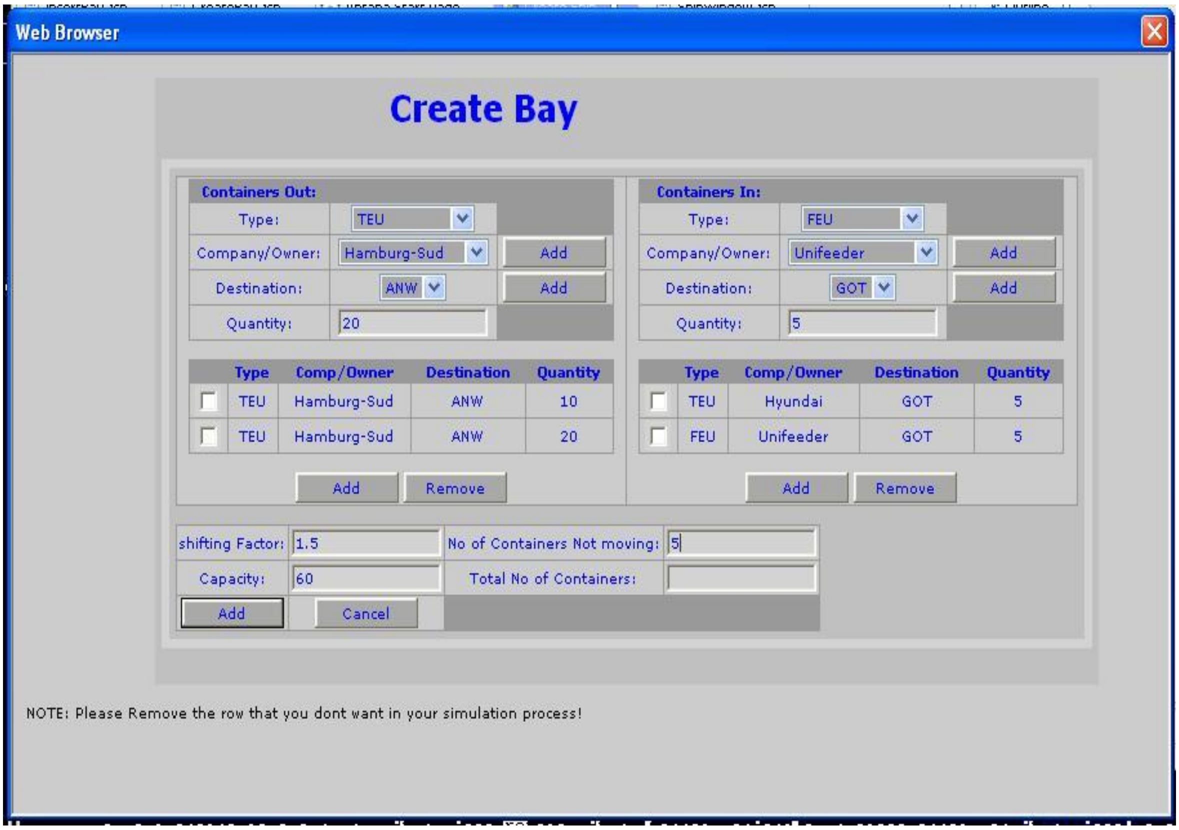
[25] Dirk Steenken1, Stefan Voß2, and Robert *Stahlbock2 Container terminal operation and operations research – a classification and literature review* Springer-Verlag 2004

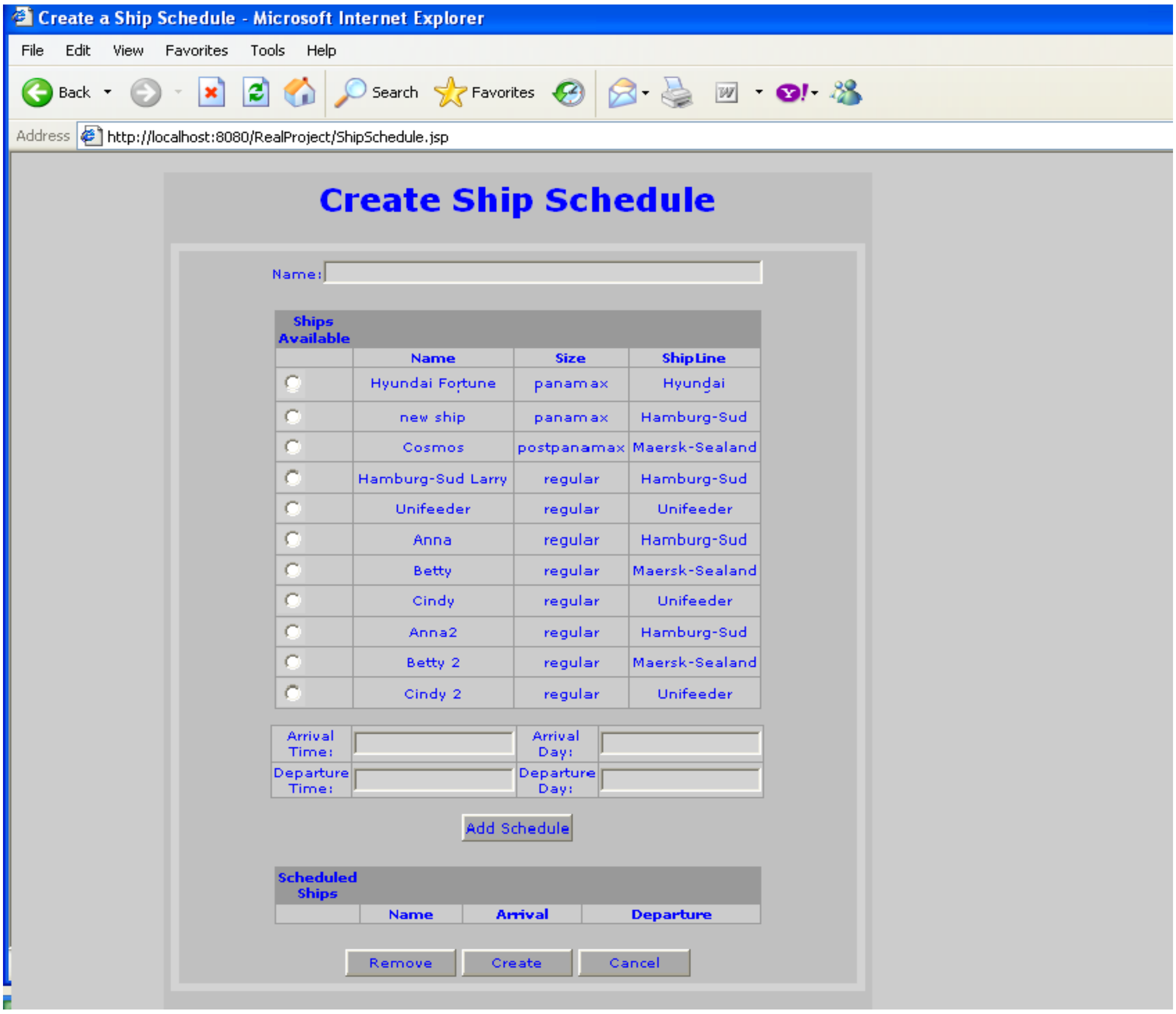
[26] http://www.globalsecurity.org/military/systems/ship/images/sealift-ships.jpg

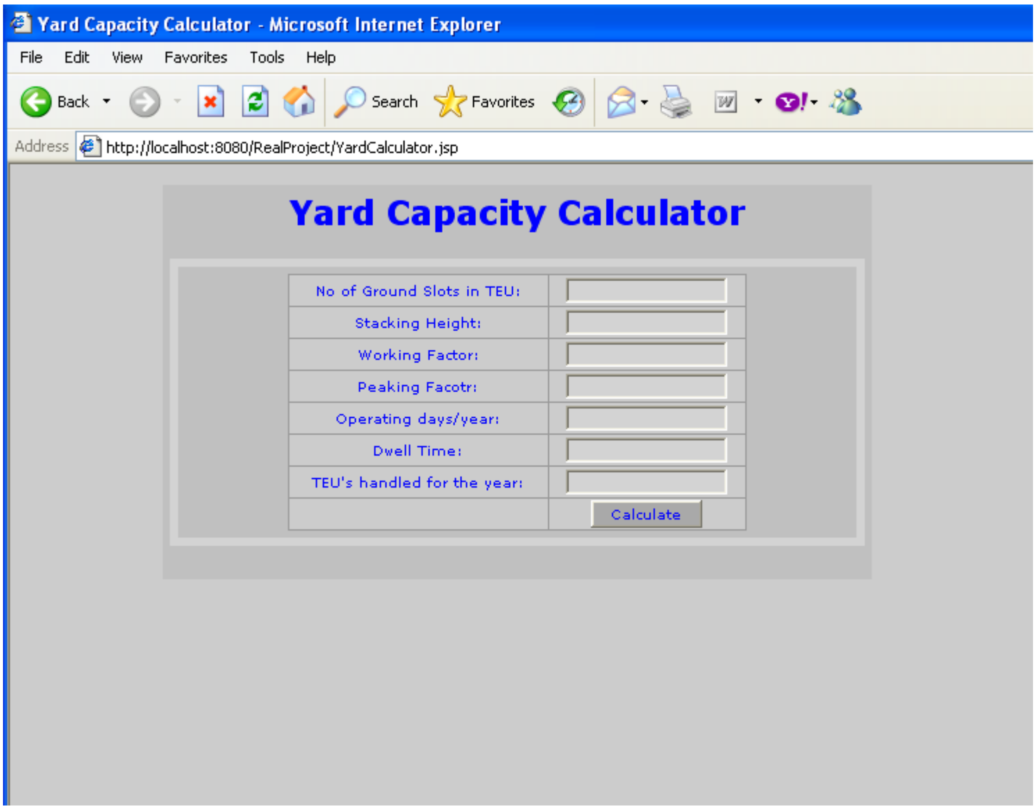
[27] Hans-Otto Günther. Kap-Hwan Kim *Container terminals and terminal operations* Published online: 13 July 2006 © Springer-Verlag 2006

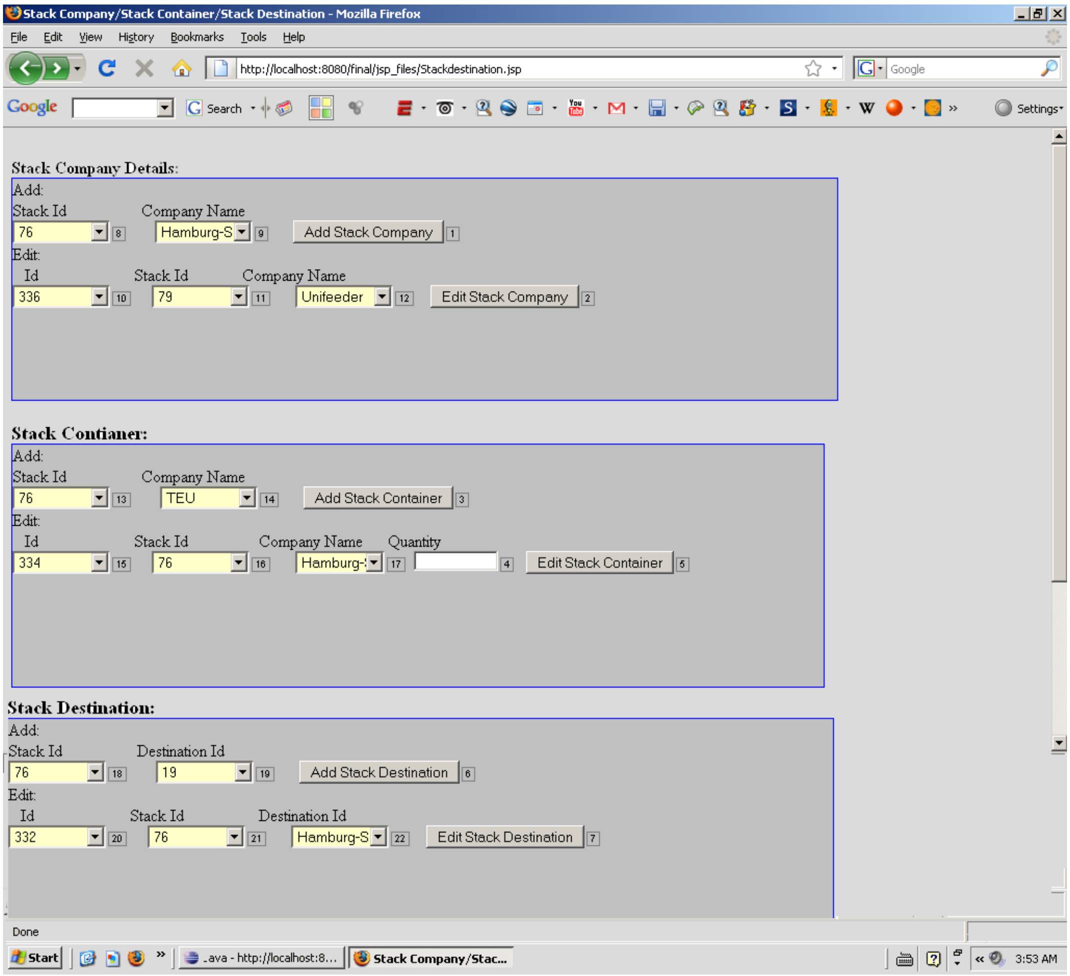
APPENDIX A


Create Ship - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/RealProject/ShipWindow.jsp
Create Ship
Name:
Length:
Type: panamax
Voyage #:
Ship Line: Hamburg-Sud
Cost/Hour:
BAYS:
Bay ID In Cont Out Cont Current Max
Add Remove
Create Cancel
Back to main page


Web Browser
Create Bay
Containers Out:
Type:
TEU
Company/Owner:
Hamburg-Sud
Add
Destination:
ANW
Add
Quantity:
20
Type
Comp/Owner
Destination
Quantity
TEU
Hamburg-Sud
ANW
10
TEU
Hamburg-Sud
ANW
20
Add
Remove
Containers In:
Type:
FEU
Company/Owner:
Unifeeder
Add
Destination:
GOT
Add
Quantity:
5
Type
Comp/Owner
Destination
Quantity
TEU
Hyundai
GOT
5
FEU
Unifeeder
GOT
5
Add
Remove
shifting Factor:
1.5
No of Containers Not moving:
5
Capacity:
60
Total No of Containers:
Add
Cancel
NOTE: Please Remove the row that you dont want in your simulation process!


Create a Ship Schedule - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Back
Search
Favorites
Address
http://localhost:8080/RealProject/ShipSchedule.jsp
Create Ship Schedule
Name:
Ships Available
Name
Size
ShipLine
Hyundai Fortune
panamax
Hyundai
new ship
panamax
Hamburg-Sud
Cosmos
postpanamax
Maersk-Sealand
Hamburg-Sud Larry
regular
Hamburg-Sud
Unifeeder
regular
Unifeeder
Anna
regular
Hamburg-Sud
Betty
regular
Maersk-Sealand
Cindy
regular
Unifeeder
Anna2
regular
Hamburg-Sud
Betty 2
regular
Maersk-Sealand
Cindy 2
regular
Unifeeder
Arrival Time:
Arrival Day:
Departure Time:
Departure Day:
Add Schedule
Scheduled Ships
Name
Arrival
Departure
Remove
Create
Cancel


Yard Capacity Calculator - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://localhost:8080/RealProject/YardCalculator.jsp
Yard Capacity Calculator
No of Ground Slots in TEU:
Stacking Height:
Working Factor:
Peaking Facotr:
Operating days/year:
Dwell Time:
TEU's handled for the year:
Calculate


Stack Company/Stack Container/Stack Destination - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:8080/final/jsp_files/Stackdestination.jsp
Stack Company Details:
Add:
Stack Id Company Name
76 Hamburg-S Add Stack Company
Edit:
Id Stack Id Company Name
336 79 Unifeeder Edit Stack Company
Stack Contianer:
Add:
Stack Id Company Name
76 TEU Add Stack Container
Edit:
Id Stack Id Company Name Quantity
334 76 Edit Stack Container
Stack Destination:
Add:
Stack Id Destination Id
76 19 Add Stack Destination
Edit:
Id Stack Id Destination Id
332 76 Hamburg-S Edit Stack Destination
Done
Start
3:53 AM