

CONCEPTUAL FRAMEWORK FOR SIMULATING SEAPORT TERMINALS

**Bernard J. Schroer, Maruf Rahman and
Gregory A. Harris**
University of Alabama in Huntsville
Huntsville, AL

Dietmar P.F. Moeller
University of Hamburg
Hamburg, Germany

ABSTRACT

This paper presents a conceptual framework for the development of seaport simulation models. The framework has been successfully used to model a bulk coal handling terminal, a seaport container terminal and an airport intermodal center. Included in this paper are a description of the modeling framework, descriptions of the application implementations and conclusions.

KEYWORDS

Seaport, container terminal, coal terminal, intermodal center, simulation

1. INTRODUCTION

A number of difficulties exist for the successful development and implementation of simulation models. Quite often these difficulties handicap the wider use of simulation. This is especially true in obtaining management support because of preconceived ideas about the time and cost overruns on past simulation projects.

The time to create, validate and verify a simulation project seems to be the most significant barrier to overcome. In many instances the data needed for a successful simulation do not exist. Or, on the other hand, the data are generally not readily available in a form that can be easily used. Even then the available data are not credible, incomplete or inaccurate. Furthermore, in many instances there is not sufficient time to collect the data because of urgency from management for answers.

Another constraint is model development time. Most models become too detailed and take considerably longer to develop than estimated, thus driving up delays and costs and infuriating management. In addition, complex models are difficult to debug.

Another difficulty is that in many instances model verification and validation (V&V) are minimized or more frequently just ignored. Model verification consists of determining if the

model is correctly represented in the simulation code. Model validation consists of determining if the model is an accurate representation of the real world system (Harris et al, 2008a). V&V generally takes more time than estimated or available.

Another issue is having trained simulation personnel or the ability to recruit the necessary personnel to assure a successful simulation outcome. Many companies (and especially smaller companies) do not have such a simulationist on the payroll. At a minimum this person must have some knowledge of modeling and model construction, know a simulation software package and understand fundamentals of statistics.

As with all domains the modeling of seaports offers unique challenges. The research focus of this paper is on developing a conceptual framework that reduces the impact of many of the previously stated difficulties. Simulation can be used as inexpensive insurance against costly mistakes, especially when significant capital expansions are being considered. This is especially true in the large capital investments at seaports where a crane can cost in the millions.

2. MODELING FRAMEWORK

Figure 1 is a sketch of the conceptual framework for a seaport model development. The model consists of a number of submodels that run independently. Each model has its own data input and entities with specific attributes. For example, the data input can include arrival and service times, storage capacities, and available resources.

In the conceptual framework data are shared between the submodels by global variables. The content of global variables can be altered within any submodel with the new values immediately shared and used by any other submodel. These global variables not only pass data between the

submodels but can also be used in logic statements to control the movement and routing of entities, branching logic, and updating entity attributes.

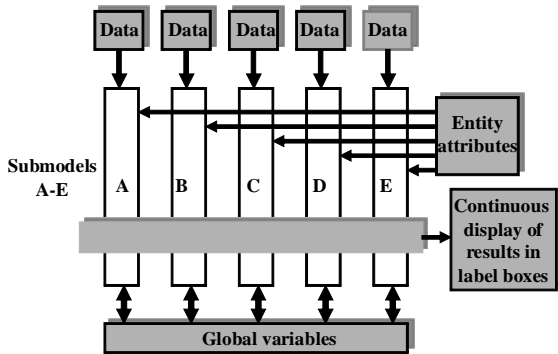


Figure 1. Conceptual framework

To assist in the verification and validation the conceptual framework includes a set of output blocks that display current values from the global variables during the running of the simulation. These values are generally overlaid on top of the simulation model so the user can observe the movement of entities as well as any bottlenecks.

3. MODELING SYSTEM

ProcessModel (1999) was selected to implement the conceptual framework. The building blocks in ProcessModel were ideal for constructing the submodels with each submodel having a name specific entity. ProcessModel has four building blocks: activities, entities, resources and stores. Labels are used to display results from global variables. Within each block and for each routing option (connecting line) there is the capability of adding very complex logic. Global variables and entity attributes can be easily defined within ProcessModel. ProcessModel also has a label block that can be used to display the current content of selected global variables during the simulation.

The basic steps in constructing a ProcessModel following the conceptual framework are:

1. Define and name as many of the global variables, entity attributes, resources and output blocks as possible.
2. Each submodel is constructed, debug and verified and validated separately. The use of constants for all data input greatly reduces the debugging time, as well as model verification. Before starting another

submodel development the arrival of entities is turned off.

3. Once all the submodels have been constructed, the arrival entities are added back into the submodels.
4. All submodel are then combined into one model and again verified and validated this time with distribution data.

4. COAL TERMINAL MODEL

Figure 2 is the model of the McDuffie Coal Terminal at the Alabama State Docks in Mobile, Alabama. Low sulfur coal arrives on ships and leaves on barges and trains. High sulfur coal arrives on barges and trains and leaves on ships (See Table 1).

Table 1. Coal terminal arrivals & departures

Coal Arrives On	Coal Departs On		
	Ship	Barge	Train
Ship	NO	YES	YES
Barge	YES	NO	NO
Train	YES	NO	NO

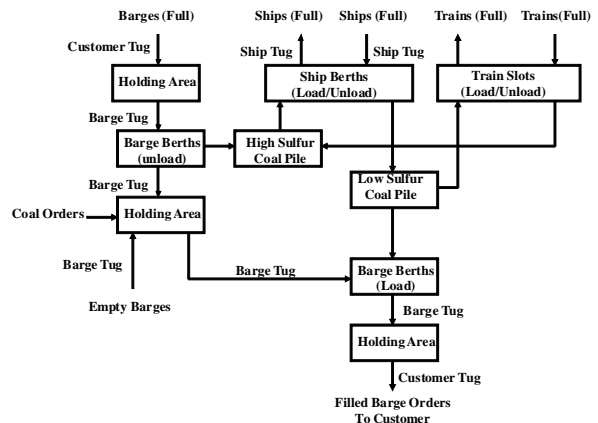


Figure 2. Coal terminal model

Translating this model into the conceptual framework resulted in the following submodels:

- Ships unloading low sulfur coal and loading high sulfur coal
- Barges unloading high sulfur coal loading low sulfur coal
- Trains unloading high sulfur coal and loading low sulfur coal

The labels that display selected global variables during the running of the simulation are:

- Cumulative unloaded high sulfur coal
- Cumulative unloaded low sulfur coal
- Cumulative loaded high sulfur coal
- Cumulative loaded low sulfur coal
- High sulfur coal pile
- Low sulfur coal pile

Harris, et.al. (2007a) have developed and implemented the coal terminal simulation model in Figure 2 using the conceptual framework. A perceived barrier to an increase in capacity at the coal terminal was the number of tugs for moving barges and ships throughout the terminal. This simulation model evaluated the various tug alternatives for improving the velocity of coal through the terminal.

Figure 3 is the ProcessModel for the submodel named Ships Unloading Low Sulfur Coal and Loading High Sulfur Coal (Harris, et.al., 2007a). The comments to the right of each block are the logic that is coded into the action section of each block. The numbers to the left of each block are the capacity of each activity. The entity is named Ship. The coal terminal has the following resources: ship tugs, dock tugs, customer tugs, ship berths, barge unload berths and barge load berths.

A description of the logic that is nested in the Move Ship to Berth activity is:

- Define Load_Ship time = 2,100 minutes
- Define Unload_Ship time = 3,000 minutes
- GET resource Ship_Berth
- Then GET resource Ship_Tug
- TIME(20 min) = time to move ship to berth
- TIME(10 min) = time to position ship in berth
- FREE resource Ship_Tug
- Note that the resource Ship_Berth is still in use

In the above logic there are two global variables: Load_Ship and Unload_Ship and two resources: Ship_Berth and Ship_Tug. A description of the built in logic for the Load Ship Coal activity is:

- WAIT UNTIL the global variable High_Sulfur_Coal_Pile >= 15,000 tons
- If true, then DECREMENT the coal pile by 15,000 tons
- TIME(Load_Ship/5 minutes) = time to load 15,000 tons
- The above logic is repeated four more times for a total of 75,000 tons unloaded

- After loading the ship the global variable High_Sulfur_Out is INCREASED by 75,000 tons

In the above logic there are three global variables: High_Sulfur_Coal_Pile, Load_Ship, and High_Sulfur_Out. The Load Ship Coal activity has a capacity of three, or equivalent to loading three ships simultaneously in three ship berths.

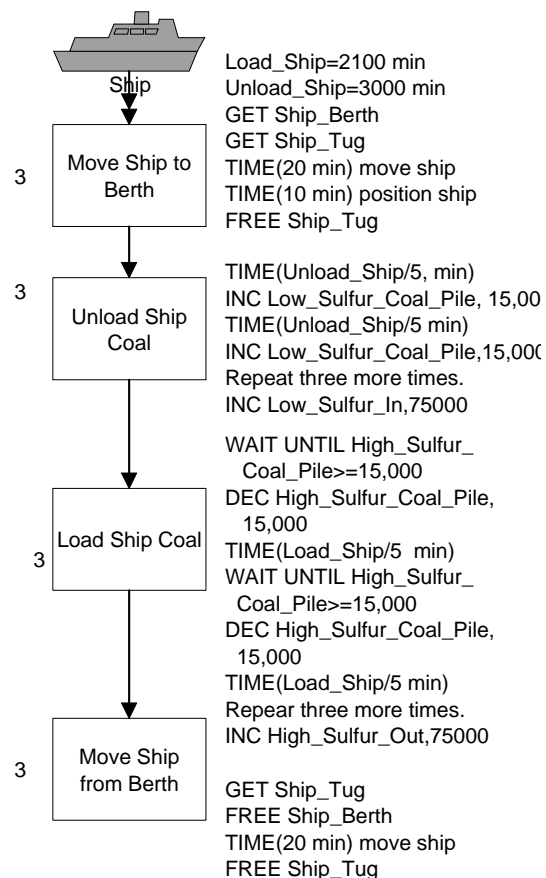


Figure 3. ProcessModel for ship submodel

One of the significant hurdles during the model development was the level of model fidelity. The decision was made to model at the “scoop” level where a scoop is defined as the amount of coal that is moved at a time. Several more detailed ProcessModels have been written for a variable scoop size of as low as 15 tons. A ProcessModel scoop entity was drawn that is displayed and moved on the screen during coal unloading and loading. This definition of scoop and the subsequent development allowed stakeholders and process experts to verify the functionality of the model.

5. CONTAINER TERMINAL MODEL

Figure 4 depicts the model of the container terminal at the Alabama State Docks in Mobile, Alabama. Table 2 gives container arrivals and departures. Containers arriving on ships depart on trains and trucks. Containers arriving on trains and truck depart on ships.

Table 2. Container terminal arrivals and departures

Container Departs On			
Containers Arrive On	Ship	Train	Truck
Ship	NO	YES	YES
Train	YES	NO	NO
Truck	YES	NO	NO

Translating the model into the conceptual framework resulted in the following submodels:

- Ships unloading and loading of containers
- Trains unloading and loading of containers
- Trucks unloading and loading of containers
- Movement of containers from ship dock to container yard
- Movement of containers from container yard to ship dock

The labels that display selected global variables during the running of the simulation are:

- Cumulative containers unloaded from ship
- Cumulative containers unloaded from train
- Cumulative containers unloaded from truck
- Cumulative containers loaded onto ships
- Cumulative containers loaded onto trains
- Cumulative containers loaded onto trucks
- Containers on dock unloaded from ships
- Containers on dock waiting to be loaded onto ships
- Containers in container yard from ships
- Containers in container yard from trains and trucks

The container terminal had the following resources: tugs, ship cranes, stackers, forklifts, carts and ship berths.

Harris, et.al. (2007b) have developed and implemented the container terminal simulation model in Figure 4 using the conceptual framework. The Alabama State Docks is currently spending \$300 million to enhance container operations and of interest is the validation of the design capacities of the

container terminal. Of special interest were the utilization of the berths, cranes and stackers and the maximum container throughput of the terminal.

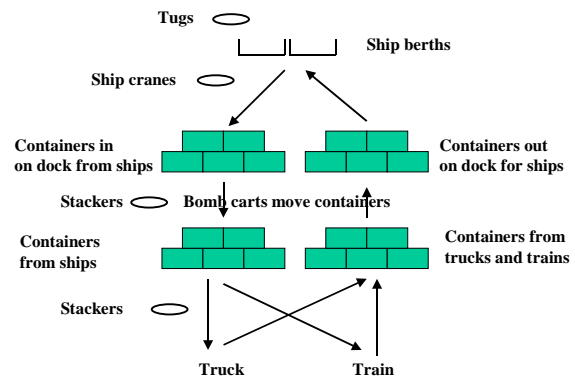


Figure 4. Container terminal model

Figure 5 is the ProcessModel for the submodel named Ships Unloading and Loading of Containers (Harris, et.al., 2007b). The comments to the right of each block are the logic coded into the action section of each block. The entity is named Ship.

A description of the logic that is nested the Wait for Container at Ship activity is:

- WAIT UNTIL global variable Containers_On_Dock_Out >= global variable Containers_Out_Ship
- If true then DECrement Containers_On_Dock_Out by Containers_Out_Ship
- Then INCrement global variable Total_Out_On_Ship by Containers_Out_Ship
- GET resource Ship_Crane

In the above logic there are three global variables: Containers_On_Dock_Out, Containers_Out_Ship and Total_Out_On_Ship. The activity Wait for Container at Ship has capacity of two or the equivalent of loading two ships at a time.

As with the coal model, a hurdle during the model development was the level of model fidelity. The decision was made to model at the container level. A ProcessModel container entity was drawn that is displayed and moved on the screen during any container movement such as unloading and loading.

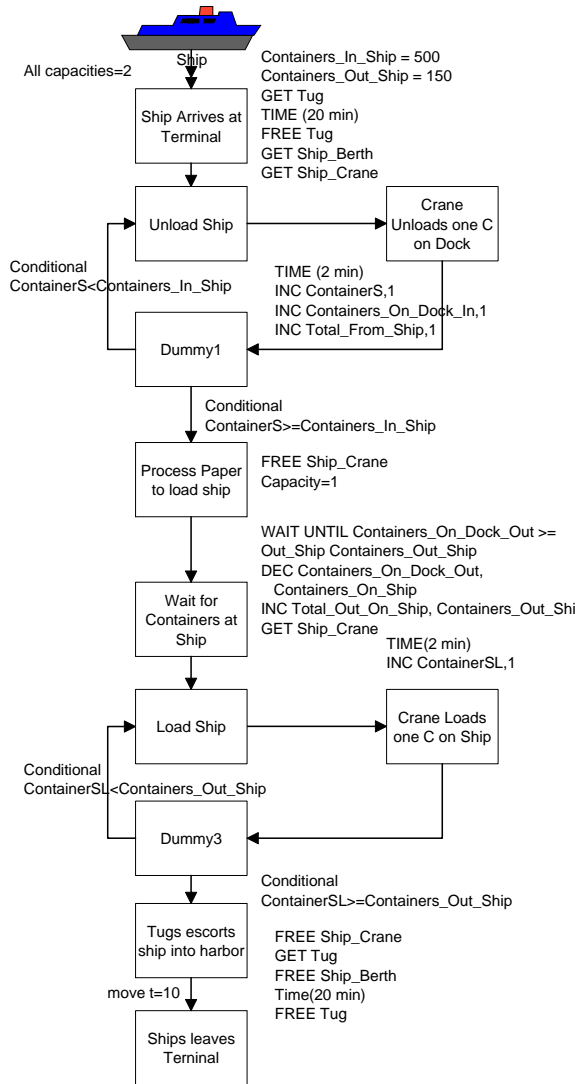


Figure 5. ProcessModel for ship submodel

This model provides the needed insights for relationships that the management of the port desired to see in a decision analysis tool.

6. INTERMODAL CENTER MODEL

Figure 6 is the model of the intermodal terminal center at the Huntsville International Airport in Huntsville, AL. Containers arriving on airplanes depart on trucks. Containers arriving on trains depart on airplanes and trucks. Containers arriving on truck depart on airplanes and trains. The movement options are shown in Table 3.

Translating this model into the conceptual framework resulted in the following submodels:

- Planes unloading and loading of containers
- Trains unloading and loading of containers
- Trucks unloading and loading of containers
- Movement of containers from plane dock to container yard
- Movement of containers from container yard to plane dock
- Movement of containers from train dock to container yard
- Movement of containers from container yard to train dock

Table 3. Intermodal center container arrivals and departures

	Containers Depart On		
Containers Arrive On	Airplane	Train	Truck
Airplane	NO	NO	YES
Train	YES	NO	YES
Truck	YES	YES	NO

The intermodal terminal had the following resources: gantry cranes, stackers, forklifts and carts.

Figure 7 is the ProcessModel for the submodel named Plane Unloading and Loading of Containers. The comments to the right of each block are the logic coded into the action section of each block. This submodel for the plane is almost identical to the submodel Ship Unloading and Loading in Figure 6.

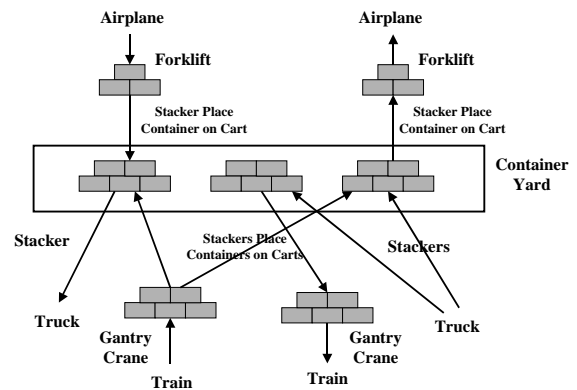


Figure 6. Intermodal center model

The model provides insight to the operations of a typical intermodal center.

7. CONCLUSIONS

Table 4 presents a comparison of the various models developed using the conceptual

framework. The first model developed was the coal model, followed by the container model and finally the intermodal model. A number of model variations have been made to provide a fast response to potential options Port management may have to developing issues. For example, the impact of increased security inspection of containers was added to the container model allowing stakeholders to better understand the effects of supporting different inspection protocols (Harris et al., 2008b). Also, the impact of a number of continuous improvement events was also added to the coal model helping validate the impact of continuous improvement activities by Port personnel.

The model development times were forty hours for the coal model, twenty-four hours for the container model and sixteen hours for the intermodal model. The intermodal model was the most complex model, especially in terms of the logic; however, this model required the least development time, showing a learning effect and the use of previously defined models. The development time for the intermodal model was sixteen hours as compared to forty hours for the coal model.

Table 4. Comparison of various models

	Coal Model	Container Model	Intermodal Model
Submodels	3	5	7
Entities	5	7	9
Blocks	43	50	55
Attributes and Global Variables	10	23	28
Logic Statements	110	99	178
Development Time (hours)	40	24	16
V&V Time (hours)	16	12	12

In summary the following conclusions are made:

- The conceptual framework provides an excellent template in the development of port and terminal activities and processes. This framework greatly reduced development time, model debugging and verification and validation.

- The submodels for the three models were very similar. Consequently, the ProcessModels for each submodel were similar with the exception of the branching logic.
- The use of the global variables was also similar for all three applications. As a result, the use of the ProcessModel Label blocks was similar.

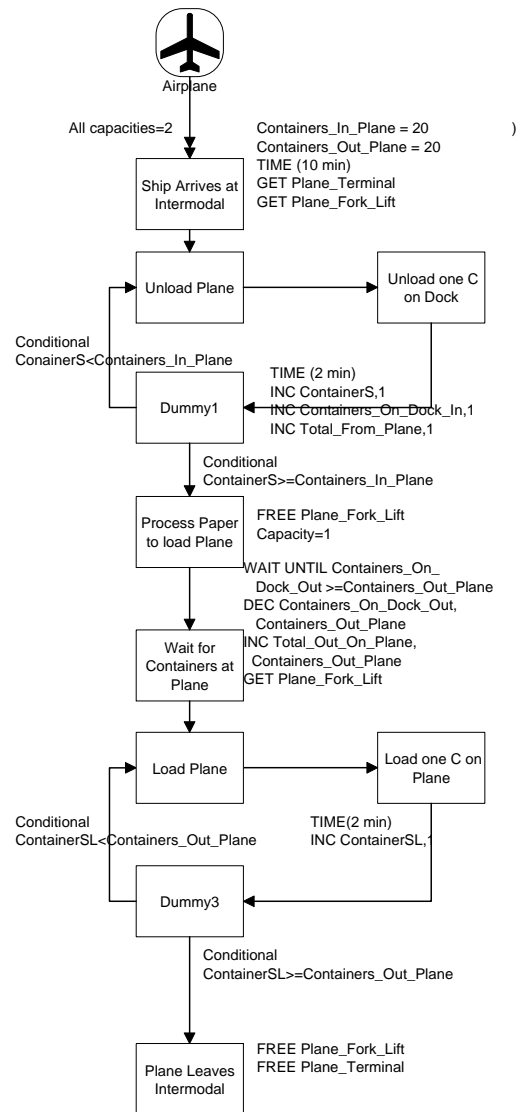


Figure 7. ProcessModel for airplane submodel

- Data collection for the applications was done by interviewing the personnel at the Alabama State Docks and the Huntsville Intermodal Center. By asking the appropriate questions

most of the input data was collected through interviews. For example, a good assumption is that all service times follow triangular distributions. It is rather easy to ask knowledgeable personnel the most frequent time, the smallest time and the largest time to obtain the parameters for the triangular distribution.

- Modifications to the model are simplified because of the submodel framework. Changes made to a submodel could be easily debugged without having to worry about the other submodels.
- Using the conceptual framework it is possible to construct the model in pieces, or one submodel at a time. Consequently, each submodel can be debugged and verified separately, thus reducing development time.

8. ACKNOWLEDGEMENTS

This research was sponsored by the U.S. Department of Transportation, Federal Transit Administration, Project No. AL-26-7262-00.

9. REFERENCES

Harris, G., A. Holden, B. Schroer, and D. Moeller, 2007a: "Coal Terminal Simulation," *Proceedings of 2007 Huntsville Simulation Conference*, Huntsville, AL, October.

Harris, G., L. Jennings, B. Schroer, and D. Moeller, 2007b: "Container Terminal Simulation," *Proceedings of 2007 Huntsville Simulation Conference*, Huntsville, AL, October.

Harris, Gregory A., Anthony Holden, Bernard Schroer, and Dietmar Möeller, 2008a: *A Simulation Approach to Evaluating Productivity Improvement at a Seaport Coal Terminal*; The Transportation Research Record – Journal of the Transportation Research Board, 2008. Paper #08-1263.

Harris, G., M. Rahman, B. Schroer, and D. Moeller, 2008b: "Container Security inspection: Simulation to Evaluate Various Container Sampling Plans on Port Operations," Accepted for the *Proceedings of 2008 Huntsville Simulation Conference*, Huntsville, AL, October, 2008.

NIST, 1998: *Principles of Lean Manufacturing with Live Simulation Users Manual*, NIST

Manufacturing Extension Partnership, Gaithersburg, MD.

ProcessModel, 1999: *Users Manual*, ProcessModel Corp., Provo, UT.

10. AUTHOR BIOGRAPHIES

Bernard Schroer is a Principal Research Engineer at the University of Alabama in Huntsville (UAH). He is a Fellow of IIE, a Fellow of the SME and a member of SCS. He has a PhD in Industrial Engineering from Oklahoma State University and is a registered Professional Engineer.

Maruf Rahman is a Principal Research Engineer at UAH. He has facilitated over 500 Lean Implementation projects (Kaizen events) and has trained approximately 7000 engineers, managers, and other employees in Lean manufacturing and six sigma methodologies. He has a PhD in Industrial and Systems Engineering from UAH, is a certified NIST lean manufacturing trainer and a certified Black Belt.

Gregory Harris is Director of the Alabama Technology Network and the Office for Freight, Logistics & Transportation at the University of Alabama in Huntsville (UAH). Harris is a certified NIST lean manufacturing trainer. Harris has a Ph.D. in Industrial and Systems Engineering from UAH and is a registered Professional Engineer.

Dietmar Moeller is Professor of Computer Science and Computer Engineering at the Mathematics, Computer Science and Science Faculty of the University of Hamburg, Germany. His is also Chair of Computer Engineering. He has a Dr.Ing. in electrical engineering and control theory from the University of Bremen.