1. Introduction
The part of the container sea-borne trade is steadily expanding, especially as far as we are concerned with "handle with care" or high added value products (manufactured products, perishable goods...). This kind of traffic has significantly increased, and is mainly concentrated in the great industrial areas: North America, Europe, and Asia. To respond to the increasing trade and to decrease the working costs, it is necessary to improve the use of existing areas by introducing new storage techniques and new working methods.

The important infrastructure costs in a container terminal imply a simulation-based evaluation (Mastrolilli and al. 1998), (Behera and al. 2000) before considering any modification of one of the three main parameters, which are:

- Geometric configuration of the site,
- Working methods and resource control,
- Equipment investment.

To respond to this requirement, we worked on a simulation project named SGTC (a Container Terminal Management Simulator). This research leads to an object-oriented model with two innovating concepts:

- One concerns the multi levels resource management by means of Resource-Providers and Multi-Resource-Controllers.
- The second is the process oriented simulation model that uses macro-processes and complementary processes. The general operating principle of the prototype SGTC is based on the Customer/Server model.

We developed a simulation prototype to help a decision-maker with his strategic choices: to test hypothesis and to experiment innovative solutions before putting them into practice. The objective of the prototype is to validate specifications for a final simulator. The simulator calibration and validation process is a cooperation with the INTRASEAS European Project (INTRASEAS). In the following paper we present highlights on the model, the validation process and some of the simulation exercises. The model is currently in a re-engineering process (Itmi and al. 2000).

2. Main characteristics of the SGTC

2.1 The SGTC Object Model
The Object Model (Booch 91) in the SGTC is composed of Basic Objects, Resource-Providers (i.e. Objects dedicated to the allocation of Resource) and Multi-Resource-Controllers (to gather different types of Resources). With more than seventy sub-classes, the class of Basic Objects is the heaviest one. Basically, this class is made of static objects and moving objects. Roughly speaking, the object classes are organized in two kinds: Physical classes (such as a container, a place...) and Concept classes (such as a set of rows, a set of transport means, a workers team...).

The second particularity of SGTC is the type of simulation, which is a simulation by processes and complementary processes. These concern the work allocation policies:

- The process-oriented server-customer relationship including tasks and services,
- The process-oriented simulation diagrams.

The behavior of each entity is described through the process approach (Birtwistle and Tofts 1993), which is widely considered as the most natural one. The interactions between entities are obtained through a customer/server model. For modularity, the process of a customer is described in terms of sub-processes.
2.2 Resource management
A Resource represents an elementary object with a physical existence on the site (for example: crane, quay, ship etc.) The team notion is seen as a resource too and objects are organized by team (for example, a loading team can be made of a straddle carrier and a crane); this team is associated with a ship and a storage area.

The resource allocation (Serin and al. 1995) is managed by a Resource-Provider. This object manages only one type of resource. The sets of managed resources, resource requests and available resources constitute a resource provider. At each time a request is made or more resources are produced (or released), the Resource-Provider checks to see if one or more of its pending requests can be satisfied. Very often the gathering of different kinds of resources like machines, transports, etc... to co-ordinate the functioning is needed. But the strategies for allocation could be different. So, a set of heterogeneous resources cannot be combined in the same resource provider. Creating the Multi-Resource-Controller class (MRC) solves the problem. As MRCs collect a few resource providers, they provide a team notion composed of different kinds of resources. Two typical Multi-Resource-Controllers are the Work-Team and the Ship. The team manages handling-apparatus, containers and storage areas. The ship manages her places and her containers.

2.3 The SGTC Functional and Dynamic Models
The Functional Model and the Dynamic Model developed for SGTC are based on the following concepts:

- The behavior of the entities is described through the Process approach.
- An active object capable of moving is an Actor.
- The interaction between the entities is obtained through a Customer-Server model.

The general operating principle of the prototype SGTC is based on the Customer/Server model: any customer (truck, ship...) requests a set of services to the simulator. This one contains all the resources present on the site and decides, during the simulation time to provide the services required by the customer.

Elementary tasks are used to model the dynamics and the capabilities of the Actors. Each service is characterized by a list of tasks to be executed by their Actor and a Sub-Process to be returned to the Customer. For pairs Customer/Server, the user can define services. If a class of customers or a class of servers share the same service, it is not necessary to redefine it for each subclass of those classes. A Service execution consists in executing the job selected by the Server according to the Customer and returning the Sub-Process to be further executed by the Customer. A service can be locking or not locking. The execution of a locking service hangs on the process until the full completion of the job. A typical example of a locking service is the Wait service. The Sub-Process can be either unconditional if it is the same whatever the context of the process may be or conditional if it depends on the context.

To define the global execution the user describes the services as Job/Sub-Process pairs. The user gives each client a process. The process consists of a list of sequential services to be requested to their own servers. As soon as a client enters the simulation, he starts reading his process. The current service is performed, that is to say that the Actor accomplishes the job, which was selected by the server. The complementary process, i.e. the remaining services, is returned to the client.

2.4 The SGTC prototype
From the user's point of view, the SGTC tool is a discrete event simulator providing the following:

- The SGTC is an Object-Oriented prototype tool developed with the objective of taking into account all the attributes concerning the operated objects, the space where these objects can move around, the rules for the vehicles movements, as well as the statistical laws that can apply to the system;
- It is able to take also into account the organization and the working methods;
- A graphic interface supports the user to interactively define the geometrical configuration of the site, the layout and the resources quantification;
- The description of the simulated environment is as close as possible to reality. Each container is taken into account with its characteristics and may even store its history (busy time, trace of activities...);
- The model is general enough to allow either modeling of a centralized management or, more interesting, division of the operational responsibilities over different locations.

From a more technical point of view, the SGTC prototype has the following characteristics:

the methods and tools used for the implementation of the SGTC are suitable for the development of complex prototypes:

- Object programming with Clos (Common Lisp Object System);
- User interface Clim (Common Lisp Interface Manager);
- Object database Statice to store the sites.

The SGTC prototype has been ported from Unix platforms to a PC/NT. It presently runs in Allegro CL environment (Franz Inc.).

3. Simulation of a terminal activity

3.1 Building a Container Terminal Example
The basic idea for approaching the simulation was that to validate the SGTC there was an obvious need to design, in a
first instance, a “suitable” Terminal prototype to be the input to the simulation tool. The “suitability” requirement called for the prototype to be generic, simple, realistic enough, and with a predictable behaviour to allow comparison between the expected results and the results provided by the simulation. To achieve this a reference Terminal site, tagged ITE - INTRASEAS Terminal Example, has been designed, based on:

- The analysis of the visited terminals, defined as the key sites in the European Project,
- Comparing their physical and functional organization, and
- Identifying both common points and specific points.

Due to operational and economical limitations a “Reduced Model” of the ITE has been used in the SGTC. On that basis the ITE has been built and a number of alternative operational situations have been tested on it. All Simulated Scenarios take place inside the ITE. According to the SGTC terminology, a “Scenario” is a typical case of operation representing an activity of routine of a container terminal. A Scenario can be defined as a set of services the simulation can provide. The loading and the unloading of a transport unit are examples of services.

A good and continuing assessment of the overall efficiency of the Terminal is therefore of basic importance for the optimization process, and the so-called Performance Indicators are used to provide quantitative indications on the terminal performances. Performance Indicators are a particular set of values derived from the combination of measured parameters (observed, not calculated) related to events (ship’s arrival time, ship’s berthing time, cranes operation starting time, container handling time, working time of vehicles etc...). A distinction is made between Parameters and Indicators. Parameters define the simulation while Indicators allow the measurement of performance of the simulated situation.

The choice of parameters and indicators for the simulation varies, depending on the situation studied. Thus, some parameters of one situation (number of containers for example) can become indicators for another simulation (if time becomes a parameter the number of containers becomes an indicator). The main difficulty faced in elaborating the list of the relevant Performance Indicators collected at the visited Terminals resides in the fact that there is no standardization in the measurement of the container terminal productivity. A number of parameters and indicators, have been defined as an “average” list for running the SGTC model:

- Main parameters (concerning the containers to handle, ships, trucks...),
- Handling equipment’s indicators (number of import containers handled, number of kilometers covered with load, utilization rate...)
- Transport units’ indicators (waiting time of the transport units, handling times...)
- Global indicator: The simulation time is the only global indicator that is used for the tests.

The analysis of the domain shows 48 theoretical Scenarios to be tested. They have been defined, resulting from the combination of internal and external parameters (traffic flows). One who needs to know how indicators behave when traffic flows change can consider one single Situation changing only the traffic flow parameters. We give below highlights on one scenario showing the cooperation between two teams.

3.3 Simulated Scenario “Noria”

This Simulated Scenario concerns the transshipment between a ship unloading import containers and another ship embarking export containers (Figure 2).

Each ship arrives in the simulation with her own list of containers to load or unload. Two different ways of operation have been simulated:

- “With Noria” one ship team, composed of one crane and N (=1, 2, 3, 4) straddle carriers is associated with each ship;
- “Without Noria”, as with Noria but the teams allow their straddles to be shared.

![Figure 2: The Noria strategy](image)

The Noria choice allows a better use of a small number of straddle carriers (i.e. with less initial investment and quick depreciation of machines). Services are created for the ships by indicating the sites of storage of the containers in the parts of rows (this is the case either in loading or in unloading process). The straddle carrier changes the team after each container handled and receives alternatively a loading or unloading service request. The resources straddle carriers are then shared between the two teams.

In this Simulated Scenario ship 1 unloads 15 import containers and ship 2 embarks 15 export containers. The principle of random choice of the required services was retained for 20 experiments. It can be noted that, for some indicators, the Noria choice leads to significant results in comparison with the simulation without Noria, giving a gain of about 10% for the straddle carriers time and for the number of containers handled within the working time. In table 1 below, the most different figures have been put in bold character.

<table>
<thead>
<tr>
<th>Global indicators</th>
<th>Results without Noria</th>
<th>Results with Noria</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation time</td>
<td>2h02m47s</td>
<td>2h10m20s</td>
</tr>
<tr>
<td>Transport units to treat</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Rice containers to handle</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>Export containers to handle</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>Ship indicators</td>
<td>Ship 1</td>
<td>Ship 2</td>
</tr>
<tr>
<td>Waiting time</td>
<td>0h00m56s</td>
<td>0h00m56s</td>
</tr>
<tr>
<td>Handling time</td>
<td>1h17m56s</td>
<td>1h37m10s</td>
</tr>
<tr>
<td>Stowage time</td>
<td>1h18m56s</td>
<td>1h41m17s</td>
</tr>
<tr>
<td>Crane indicators</td>
<td>Crane 1</td>
<td>Crane 2</td>
</tr>
<tr>
<td>Import containers handled</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>Export containers handled</td>
<td>15</td>
<td>0</td>
</tr>
<tr>
<td>Total distance when unloaded</td>
<td>84.6</td>
<td>84.3</td>
</tr>
<tr>
<td>Working time</td>
<td>0h38m11s</td>
<td>0h38m16s</td>
</tr>
<tr>
<td>Rate of utilization (%)</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>Cadence (Containers / hour)</td>
<td>24</td>
<td>24</td>
</tr>
<tr>
<td>Straddle carriers indicators</td>
<td>SC1</td>
<td>SC2</td>
</tr>
<tr>
<td>Import containers handled</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>Export containers handled</td>
<td>15</td>
<td>0</td>
</tr>
<tr>
<td>Total distance when loaded</td>
<td>13107</td>
<td>10259</td>
</tr>
<tr>
<td>Total distance when unloaded</td>
<td>9591</td>
<td>10229</td>
</tr>
<tr>
<td>Total distance (loaded + unloaded)</td>
<td>22668</td>
<td>20489</td>
</tr>
<tr>
<td>Working time</td>
<td>1h08m57s</td>
<td>1h09m15s</td>
</tr>
</tbody>
</table>

**Table 1: Results with and without Noria**

4. Conclusion

In this paper we have presented the SGTC model and its prototype. Thanks to the INTRASEAS project their capacity to face on-site constraints was tested, albeit in a limited way. We have given some results of this experiment. They partly reflect the possibilities of SGTC system. In its present version it allows limited studies. As a Prototype designed for specification validation it was not optimized. We are currently working in this direction and above all to enrich its calculation and user interface modules.

Acknowledgments

These studies were conducted by the PSI Laboratory at the INSA of Rouen, France. The simulations were carried out in cooperation with the INTRASEAS European Project and thanks to the support of the French Ministry of Equipment, Transportation and Housing. The SGTC system was developed thanks to the support of the Haute-Normandie Council.

REFERENCES


