

OPTIMISING ADVANCED CONTAINER HANDLING TECHNOLOGIES VIA DISTRIBUTED SIMULATION/EMULATION

Horst-Dieter Kassl^(a), Tobias Puchert^(b), Dr.-Ing. Holger Schütt^(c)

^(a) ^(b) ^(c) Institute of Shipping Economics and Logistics, Barkhausenstrasse 2, D 27568 Bremerhaven, Germany

^(a) puchert@isl.org, ^(b) kassl@isl.org, ^(c) schuett@isl.org

ABSTRACT

Simulation technology is state of the art in planning new terminals and reorganising existing ones. Typically the systems used for simulation are working offline and all information have to be entered into the model before running the scenarios. This effort can be reduced by combining the terminal operation system (TOS) with the simulation model using the parameters which are stored within the TOS. In this combination the material flow as well as the control system including all strategies has to be rebuild within the simulation. In a next step not only some defining parameters but the strategies theirselves are used in the simulation by coupling the whole TOS to the simulation. This kind of simulation, where only the material flow is build in, is called emulation. Splitting the whole simulation into different modules (the TOS and the device emulators) is the base for the distributed system. Typical emulation systems are often developed for just one application and therefore use very restricted communication protocolls. The DeCoNet framework is the base for an open distributed emulation network where each module may be realised using its own development environment and operation systems.

Keywords: Test Bed, Emulation, Container Terminal, Terminal Operation System (TOS), Distributed

1. INTRODUCTION

The rapid growth in worldwide container handling will continue in the near future, as predicted by several forecasting organisations. The terminals have to reorganise to enhance productivity. But: the more complex and automated the operation at the container terminal becomes, the more rises the importance of a high sophisticated IT-system to cope with the new demands.

The system "container terminal" is relatively complex. It is characterised by numerous parameters and interactions between technical, operational and economical components. Furtheron some of the influencing quantities have a random character as e.g. arrival times, daily no. of boxes, loading and discharging times of vessels, container movement time of a crane etc.

With the aid of simulation technology it is possible to reproduce the system "container terminal" as a virtual

system in order to analyse an existing or planned terminal in detail. As a simulation model is a computer-based system the real system "container terminal" has to be represented in such a way that an equivalent mathematical model can be constructed which then reproduces the processes - including fortuitous events - in a realistic way.

Thus a simulation system is a powerful tool with the help of which the user can "play through" and subsequently analyse the processes of a terminal in order to get a transparent basis for the decision-making process. Special simulation models have been developed for each planning level because there are different problems resulting in different requirements.

Additionally to these systems, which are mainly used for strategic planning tasks simulation nowadays is also used to support the day to day operation. The basic principle for this is the emulation of the terminal equipment. Emulation is defined as "a model that accepts the same inputs and produces the same outputs as a given system." (IEEE 1989). The emulation is directly coupled to the TOS. With this structure emulation can be used among others for:

- Evaluation and optimisation of strategies used in the TOS
- Testbed for the real TOS
- Visualisation of new Terminals
- Testbed for acceptance tests of external equipment (figure 1)

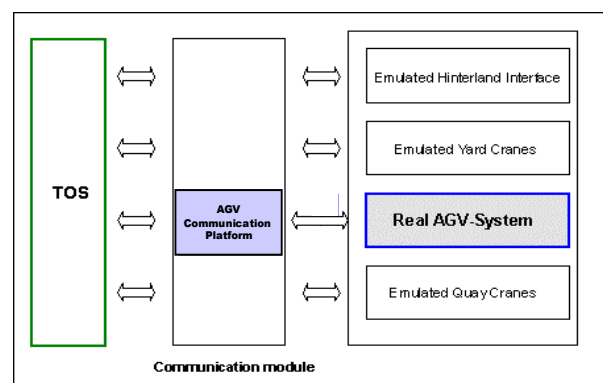


Figure 1: Using Emulation for Acceptance Test of External Equipment

One of the first projects considering all these topics has been realised at the Container Terminal Altenwerder (CTA, Hamburg, Germany). Till this day these emulators are in use to verify new releases of the TOS before installing it at the real terminal. Hartmann and Schütt (2000) describes the whole project beginning with the feasibility check up to an emulation system (IEEE 1989).

Emulation technology is found also in other parts of logistics. Burges et al. describe an application in the automotive branch, Ha (2007) describes approaches of different levels to combine the terminal operation system (TOS) with simulation/emulation modules.

2. FROM SIMULATION TO EMULATION

With the aid of simulation models it is possible to support resp. investigate the planning and design of new container terminals as well as the optimisation of long term strategies of existing ones. The evaluation of the simulated container terminal operation is carried out with regard to economic and technical aspects. The target variables which are output, measured against each other and interpreted are the costs incurred and the handling volumes achieved.

2.1. Terminal simulation

Simulation systems may be classified into low-term, medium-term and very detailed types. The low level system does not reproduce the terminal operation at all, but limits to global processes (e.g. berthing, quay productivity). Detailed systems do reproduce the single process in detail (e.g. equipments acceleration, container tracking within the terminal). Medium-term level tools for the decision-making process for the strategic planning and design level cover the expansion or re-organisation of existing terminals and changes in organisational structures that extend beyond short-term deployment planning (operational level). This kind of model does not simulate the individual containers in isolation but the behaviour of the whole system "container terminal".

With the aid of a medium-term model it is possible to determine

- the bottlenecks of a system
 - terminal layout and/or operation strategies
- the productivity of a system
 - service time for different vessel types/external truck operation/railway operation
 - the required number of transport/stacking equipment and STS cranes
- the performance of equipment used (operation, waiting-, idle-, cycle-time)
 - extensive cost evaluation

Figure 2 shows the structure of an example of a medium-term level simulation system (SCUSY-Simulation of Container Unit Handling Systems)

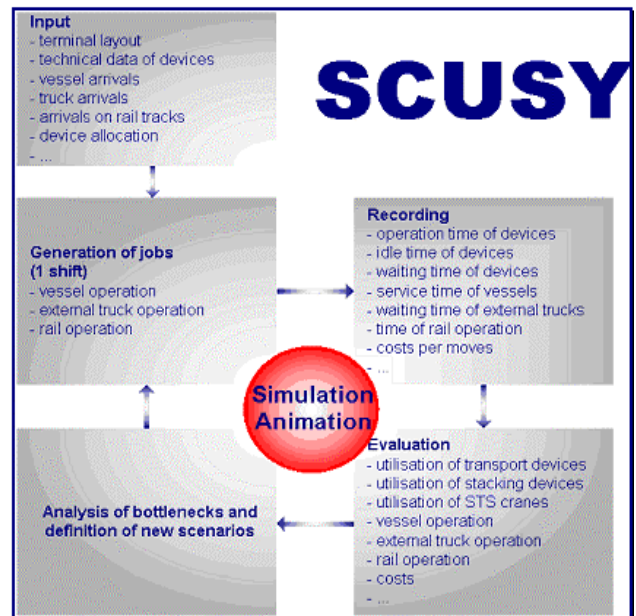


Figure 2: Structure of a Medium-Term Level Simulation System

The output shows e. g.:

- operation time for the individual vessel services
- the performance of STS cranes (moves/hr, operation time, waiting time and idle time)
- service time for external trucks and railway operation
- capacity utilisation of all container stacks
- utilisation of equipment (straddle carriers, terminal chassis, fork lift trucks, RMG, RTG)
- operation time, idle time and
- waiting time for other equipment (e. g. RTG for chassis, or chassis for RTG)

With the results of the simulation it is possible to calculate the operation costs for each terminal operation system. At the end a technical and economical evaluation of all analysed terminal systems will be executed.

2.2. Coupling TOS and simulation

Using standard "offline" simulation systems forces the user to enter all input data into the simulator. A first step to reduce the input effort needed may be done by using defined interfaces between the terminal operating system (TOS) and the simulator. As shown in Figure 3 data concerning layout information, technical data of the devices used and some parameters defining the strategies are transferred to the simulation.

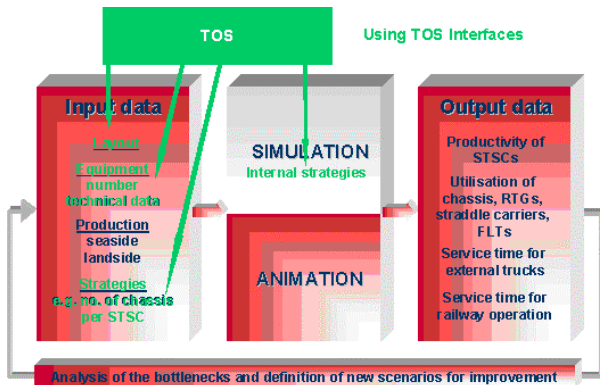


Figure 3: Coupling the TOS with the Input Module of the Simulation

The coupling may be realised via flat ASCII files, which are written by the TOS and is read by the simulator. If the data needed is stored within a (SQL) database, the simulator may get the information directly out of this database.

Besides the reduced input effort this method provides another benefit: Parameter changes made in the TOS are used automatically in the next simulation run.

2.3. Terminal emulation

While in phase 2 (see 2.2) all processes are controlled via the SCUSY strategies (resource scheduling, stack allocation, ...), in this phase the simulation –now called emulation– will only model the container flow, while strategies are directly taken from the TOS via the coupling. (see figure 4)

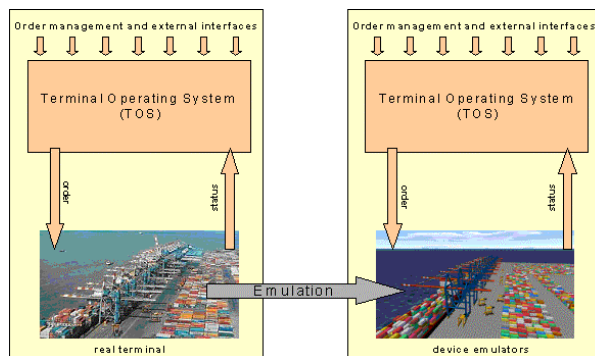


Figure 4: Switching between Real Terminal and Emulation System

Using the emulators of STS cranes, horizontal transport equipment (AGV, terminal truck, straddle carrier, ...) and stacking equipment (RMG, RTG, Fork lift trucks, reachstacker, ...) the real TOS can be used for the control of all terminal operations. In this way the following tasks may be performed:

- Testing new releases of the TOS
New software versions of the TOS may be tested without any side effects to the real terminal operation.

- Fine tuning of TOS strategy parameters
Internal parameters of the TOS may be adjusted to the specific terminal operation using actual throughput data of prior shifts. The implemented strategies of the TOS are used for controlling terminal processes without disturbing the real terminal operation.
- Evaluating new strategies to be used at the terminal

New strategies for resource scheduling, allocating stacking blocks and stacking strategies within one block may be tested before implementation at the real terminal. The proper comparison of different strategies may be done under exactly the same conditions. In real operation it is normally not possible to re-adjust exactly identical conditions

But using emulation will not only provide a test bed for the TOS, but furthermore will allow the terminal operator to fine-tune the strategies by finding the optimal parameter combination for the specific terminal. Moreover historical data may be imported and using them scenarios may be recalculated via the emulation to train the operator to cope with problems better next time.

The approach described is able to split up the material flow into small pieces (device emulators), which can be operated in a distributed simulation using a communication framework (see [3]). This does not only lead to higher performance and therefore to the opportunity to run scenarios in multiple real time. Moreover it is possible to combine emulators implemented using various simulation tools, based on various operation systems and hardware platforms, if they are solely able to dock to the communication framework (see figure 5).

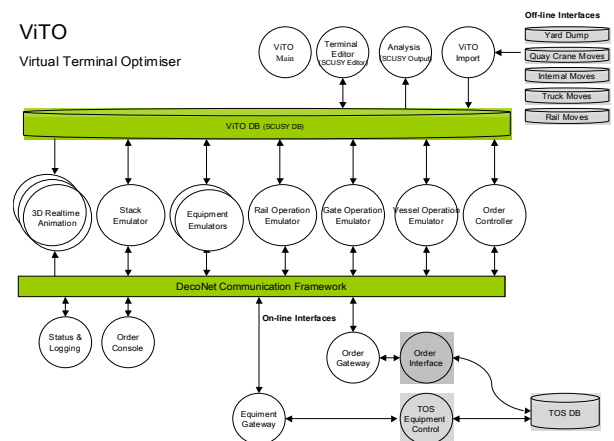


Figure 5: Structure of Distributed Terminal Emulation

As a last step the same segmentation made within the material flow may be done in the information flow, i.e. in the TOS. This is a precondition for using this approach in the field of “the port of the future” where new container handling technologies will be evaluated. E.g. it is not possible to test and optimise an AGV-

system using a TOS created for straddle carriers – due to the coupled handover of the container.

Based on this modularly built TOS and the distributed emulation, the testing and optimisation of advanced container handling technologies may be done in advance for existing terminals without building new simulation models and with regard to the existing TOS with only small changes where necessary. Moreover enhanced TOS functionalities may be fine tuned to optimise new technologies productivity.

3. DISTRIBUTED EMULATION VIA DECONET

The biggest challenge in designing software based emulations is to handle the complexity of corresponding real systems. In most circumstances a real system representation results in a single and highly specialized software implementation. In many times badly maintainable and hardly expandable software applications are resulting. Even simple changes leads in expansible, time-wasting redevelopments for necessary adjustments.

The idea to improve flexibility and efficiency of emulation systems is the use of distributed software techniques. Each part of a real system should also be realized as a separate software module. Each module represents an independent and specialized system which is assigned to exact one task. Possible modules are device emulators (e.g. crane-emulator), reporting systems or also a tool for 3D-visualisation of an emulated environment.

With this modular design principle a simple reusability of each module is warranted and this causes in long-term reduction of development efforts and costs. To achieve this objective a centralized network is needed which manages data and control flows between different emulation modules. This network has to implement the whole functionality which is needed to realize a time-controlled, event-driven emulation environment.

The solution of these requirements results in the Device-emulator Communication Network (DeCoNet). The DeCoNet is designed as a star layout network with client/server architecture based on Microsoft .NET framework techniques for service oriented distributed software-development. These .NET techniques are known under the name of Windows Communication Foundation (WCF) (Justin 2007).

The distributing centre bases on the DeCoNet service application which runs on each .NET 3.0 compatible PC. Each DeCoNet compatible client can connect to this service and is able to use the whole operational environment.

Neither the used development platform nor the target operating system for the implementation of individual DeCoNet clients is predetermined. In additional also the most important communication technologies are implemented to realize the connection between client and service. Every combination is possible. Heterogeneity is in the DeCoNet environment the most important keyword.

The following chart illustrates the fundamental communication techniques provided by DeCoNet

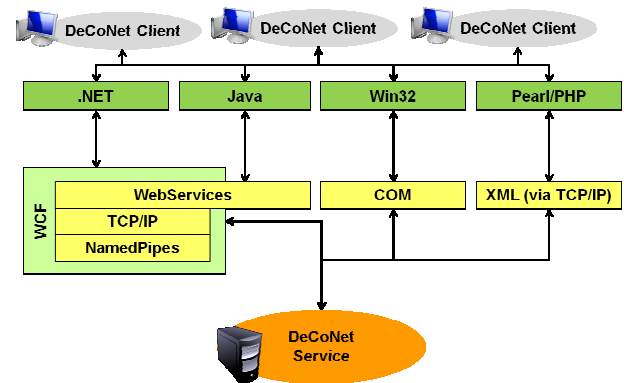


Figure 6: DeCoNet's Communication Techniques

3.1. DeCoNet functionality

In summary DeCoNet emulation systems are made up by separated software modules. Each module is realized as a DeCoNet client and each client is connected to the DeCoNet service. The DeCoNet service provides a lot of features to ensure a smoothly interaction between this modules. An integrated event manager, a free configurable emulated time environment and a complex status management are only some features of the DeCoNet. Each client can define its behavior in miscellaneous ways.

The most important capability represents the communication between separate software modules. To achieve this goal the DeCoNet uses two different communication techniques, the push and the pull principle.

The push principle is realized by the DeCoNet message-system. Each client can register a new message-type which is defined by a unique name and a specified data type (e.g. XML-Documents, floating points). Such registered message-types can be used by every client for sending a message through the DeCoNet system. A client has three possible ways to handle with message-types. First and by default a client cannot process any message types. To enable processing capabilities a client has to specify this behavior explicit. A client can establish a message-type connection and in special cases it can additionally subscribe to it.

A message-type connection permits the direct sending capability. In this case the sending client specifies one or more receiving clients who are connected to the used message-type. This message is transferred to the DeCoNet service which forwards it to each specified client.

Message-type subscriptions are enabling an outstanding feature. Subscriptions are offering the possibility for indirect sending, a very flexible way of message forwarding. The sending client does not have to specify any receiving clients. Instead of it the DeCoNet service looks up to the whole list of registered clients and sends the incoming message to each client with an active subscription to the specified message-type.

The pull principle is implemented by the DeCoNet property-system. Each property is assigned to an individual client but it is globally stored on the running system of the DeCoNet service. The maximum amount of registered properties is unlimited and each property stores data of a predefined type. Properties are equipped with configurable behaviors, like permission management to define its read and/or write capabilities.

Property-subscriptions standing for the special feature of passive data transfer. Each client can register a subscription containing one or more properties. The behavior of a property-subscription is configurable in miscellaneous ways. Possible settings are for example the minimal time-interval of delivery or to trigger only when a value has changed. A reporting system which will log the current positions of each device-emulator on every second, but only if an involved property has changed, would be a thinkable usage.

In conclusion, the DeCoNet stands for an innovative and complex communication framework to realize software based emulations within distributed and heterogeneous system environments.

4. CONCLUSIONS AND OUTLOOK

Emulation technology may be used for testing terminal's control system as well as for fine-tuning its parameters. The splitting into small modules allows the usage of a distributed network.

With the implementation of the DeCoNet framework these modules can be realised in different development environments using standard communication procedures. In this way an open system is created, where modules of different developers may be combined. E.g. the AGV-emulator may be developed by the AGV-supplier, while the crane emulator is developed by terminal's IT department.

If the TOS itself is build up in a module based architecture, the emulation system may use these small modules and via the DeCoNet framework new strategies may be implemented in the emulator and tested before the are implemented in the real TOS.

REFERENCES

- Burges, U., Mayer, G. 2006 Virtuelle Inbetriebnahme von Produktionssystemen in der Automobilindustrie mittels Emulation (german). *Proceedings of Simulation in Produktion und Logistik 2006, asim*, pp 541-550. SCS Publishing House,
- Daniluk, D. 2007 *Emulation mit der High Level Architecture* (german). Saarbrücken, germany:VDM Verlag Dr. Müller.
- Ha, Byung-Hyun 2007 Development of a Simulator of Interacting with terminal Operating Systems *Proceedings of the International Joint Symposium on Logistics Information Technologies*. November 27-28, Pusan National University, Busan, Korea
- Hartmann, S., Schütt, H. 2000 Simulation in Planung, Realisierung und Betrieb am Beispiel des Container-Terminals Altenwerder (german). In:

Möller, D.P.F., eds. *Frontiers in Simulation - Simulationstechnik*. Ghent, Belgium: SCS, pp 425-430

IEEE 1989 Std 610.3-1989, *IEEE Standard Glossary of Modeling and Simulation Terminology*

Smith, Justin 2007 *Inside Windows Communication Foundation*, Redmont, Washington, USA: Microsoft Press.

AUTHORS BIOGRAPHY

Horst-Dieter Kassl studied systems analysis at the University of Applied Science Bremerhaven. Since his diploma in 1989 he works as a project manager for the Institute of shipping Economics and Logistics Bremerhaven. Horst-Dieter Kassl is mainly involved in implementation of software for the maritime container economy, e. g. shipping companies, port operators, shipowners and manufacturer of port devices. This work includes the complete software life cycle, from the first draft conception through the several steps of systems analysis and implementation up to selling, installation and education of the users.

Tobias Puchert studied informations technology at the University of Applied Science Bremerhaven. Since his diploma in 2007 he works as a scientist for the Institute of shipping Economics and Logistics Bremerhaven. Currently he is working on his master thesis in the field of distributed simulation and emulation. Tobias Puchert is mainly involved in implementation of the Device Emulator Communication Network.

Dr.-Ing. Holger Schütt is the head of the section Optimisation and Simulation within ISL's Information Logistics department. The focal point of his work is the simulation and optimisation of business processes as well as material flow in logistic systems. In former times he worked as a computer scientist and project manager at the biggest harbour and logistic provider in Hamburg. He was responsible for the development of various administrative and planning systems (e.g. dangerous goods information system, berth planning system) as well as for the simulation based consultancy of the new fully automated container terminal Hamburg Altenwerder.