

## **IMPROVING THE QUALITY OF CONTROLS AND REDUCING COSTS FOR ON-SITE ADJUSTMENTS WITH EMULATION: AN EXAMPLE OF EMULATION IN BAGGAGE HANDLING**

William Rengelink

TBA Nederland  
Vulcanusweg 259A  
2624 AV  
Delft, THE NETHERLANDS

Yvo A. Saanen

Delft University of Technology  
Jaffalaan 5  
2628 BX  
Delft, THE NETHERLANDS

### **ABSTRACT**

Extending or changing systems that are in operation 24/7 should disturb the operation as little as possible. Therefore, the extensions or changes should have been thoroughly tested in advance under conditions comparable to operational conditions. Baggage handling systems is typically an area to which these requirements apply. Control software is one of the main components for baggage handling systems, at high and low level. Software is often the reason for system failure. Therefore, a need exists for tools that support testing without involving the real equipment on site. By using a simulation environment for emulating baggage handling equipment it was possible to test the low level control software without involving any real baggage handling equipment. The simulation environment enabled detailed tests and provided insight into the behaviour of the PLC for our client. This approach saved significant lead time in the project and reduced the required time for testing on-site.

### **1 INTRODUCTION**

Baggage handling has become one of the major issues in competition between airports. In addition, the events of the September 11 have increased the pressure on how baggage handling systems (BHS) perform. Now more than ever, airports require systems with a high level of redundancy, fast baggage transport and high security levels. Therefore, expansion is occurring within an operational environment, in which there is a high demand for the quality of the newly implemented systems and its controls.

For the manufacturer, this means that it is important to be able to test the control software in house. However, only limited testing abilities have existed up to now for control systems. In practice, this led to controls which needed to be tested excessively and often still contained a large number of problems. Therefore, the manufacturer often had to do a lot of on-site adjustments to the software because the site-specific details could not be easily taken into account

without appropriate tools. This resulted in high costs and long lead times.

To be able to improve the testing possibilities for the controls, emulation environments exist. Emulation is a virtual depiction of the controlled equipment. It reacts almost the same as the real system to the controls and also generates a similar input to the control system as the real system does.

The main objective of emulation for baggage handling is to improve the testing abilities in the factory and to reduce the cost of tests and installation of the new control system. The objectives for the manufacturer are to be able to test the software in an earlier phase, and to be able to install the equipment so that it functions more or less instantly (Wysk, 2001).

In this paper we will first present a short introduction to the usage of emulation within baggage handling projects. Then, the applied methodology for the emulation will be described. The results of the emulation will be discussed as well as the benefits and disadvantages of using a simulation environment. Finally, we will discuss the future role of simulation and emulation as it might be influenced by recent experiences.

### **2 BACKGROUND**

The problem within the industrial field of software development for control systems is often the lack of testing possibilities, especially in early phases (Saanen et al, 2000a ). This adversely affects the quality of the control software as well as the lead time to deliver a proper functioning control system. Furthermore, the time and money spent with on-site testing and adjustments is extensive. This is especially the case regarding adjustments of control software in operational environments. One of the reasons for the adjustments are limitations regarding on-site tests and high costs due to operational disturbances.

So, within a highly competitive market the manufacturer of baggage handling systems is faced with the necessity to smoothly integrate new system extensions with

complicated control functions (i.e. without too many interruptions of the on going process). From this point on, the value of a highly-detailed test environment becomes increasingly important.

In the last 10 years, manufacturers managed to build small emulation models to test specific parts of control systems. The first generation emulators existed out of two parts:

1. A card emulating all local IO cards
2. A PC card emulating the logistic process. This card was linked to the IO-card emulator via FO communication and connected to a second display for visualisation and pre-programming the emulator.

The flexibility for programming the logistic process was insufficient. This means that the processes to be depicted were limited. Moreover, the configuration for the emulator was complex (Rengelink, 2002).

The next generation of emulation environments was often directly linked to the PLC and also required several changes to PLC programming. Furthermore, the actual timing of signals was not displayed correctly. This form of emulation had two major disadvantages:

- In real-time models the actual timing of signals was sometimes critical and might therefore effect the control system.
- Disabling the emulation mode within the PLC could lead to mistakes and also disabled the possibility to emulate on-site.

Therefore, large parts of the actual final adjustments still had to be done on-site. This required a large number of personnel on-site, which of course influenced the project costs.

Nowadays, simulation environments are easier to access from other computer programs. Furthermore, the support of object-oriented (OO) modelling and programming allows for a one-to-one representation of a real system. Processes on any level within logistic or equivalent systems can be depicted, including the control programs. This allows users to answer increasingly complex questions. Furthermore, the combination of open software and OO-methodology allows for the plug-in of external system components, replacing the simulated parts (Verbraeck et al, 2000 and 2002 ).

Summarising, a need existed for an environment enabling highly detailed representation of the system and correct timing of processes. Secondly, modern simulation packages have evolved to the extent that technical design processes can be supported. Thus, real-control components can be combined with simulated system components.

### 3 APPLYING SIMULATION AS EMULATION

One of our customers, a manufacturer of baggage handling systems, was faced with having to build an extension to an existing operational BHS. This concerned the realization of a new baggage screening system. To improve the quality of the installed system and to reduce the on-site costs the manufacturer wanted a proper emulation. With the use of a baggage simulation library that had already been developed, we improved the level of detail in the simulation to a one-to-one model of reality.

During the first test stages one PLC was tested. Each PLC controls a single section of the baggage handling system, existing out of at most 70 conveyors. Such a section is linked to an upstream and downstream section respectively, which are in turn controlled by the upstream and downstream PLCs. Via the linked PLCs, information is transmitted regarding the passing baggage elements. Lastly, the PLCs communicate with higher level controls, which are responsible for routing based on planning and baggage-specific attributes. This is shown in Figure 1.

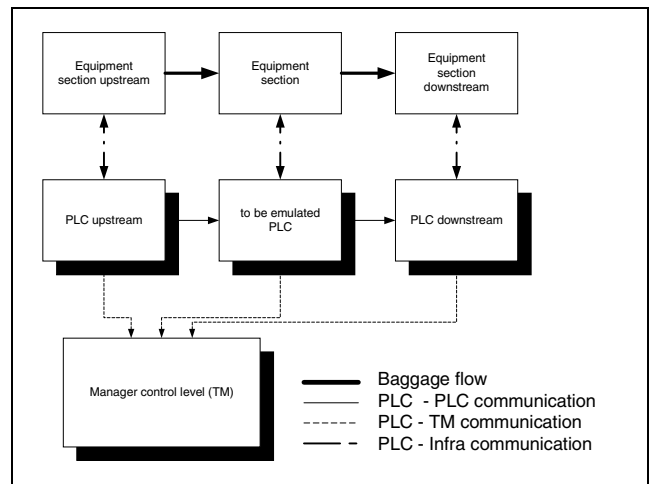


Figure 1: Control System Lay-Out

In a PLC test, where the PLC was not linked to upstream and downstream PLCs (stand alone tests), the emulation environment had to represent the infrastructure of the emulated section (i.e., conveyor belts, communication, and the upstream and downstream PLCs). Whenever real upstream and downstream PLC's were linked, only the infrastructure had to be depicted. Figure 2 represents the two possible lay-outs of the emulator.

The emulator was linked to the PLC by a profibus card (profibus is an international, vendor-independent, open fieldbus standard). In manufacturing, industrial process, and the building of automated applications, serial fieldbusses can act as a communication system. Exchanging information between automated systems and distributed field devices (Hilscher GMBH, 2002). Because the PLC and

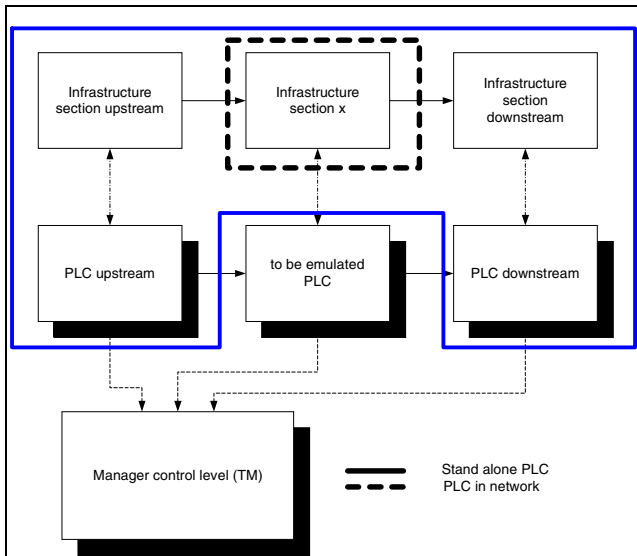


Figure 2: Emulator

profibus card could both act as a servers, it was necessary to have an interface that could act as a client toward both the PLC and the profibus. This was accomplished by means of a decentralized periphery network coupler or DP/DP coupler (Siemens).

During the factory acceptance tests (FAT), the PLC is mostly a stand-alone application. This means that the emulator is linked to the PLC by a profibus network. For higher level controls the PLC is linked via TCP-IP with the transport manager.

To portray the PLC's behavior, the corresponding infrastructure with its material flow had to be displayed in detail. This was done with an accurate two-dimensional depiction of the section. Furthermore, all of the relevant control components on equipment such as local control cabinets and emergency buttons had to be displayed in order to handle all inputs of the PLC and generate the necessary outputs.

Lastly one of the required functions of the emulator was to display the positions of bags as calculated by the PLC or so-called tracking. In a BHS, controls are performed based on this tracking within the PLC. Whenever a discrepancy exists between the calculated position and the actual position of a bag, this will lead to unidentified bags in the PLC. Or in a worst-case scenario, bags with an incorrect destination.

Therefore, it was important to be able to test the tracking function within the PLC. So the emulator receives the following input:

- Control signals from the PLC.
- Information regarding tracking of baggage elements.

And generates the following output:

- Output signals from the infrastructure, cabinets and other equipment
- Visual depiction of emulated baggage flow and the PLC calculated baggage flow (tracking).

The eM-Plant environment is an OO-programming environment (Tecnomatix, 2002), enabling the development of libraries with specific functions for a single type of application. eM-Plant Has been used to create different kinds of simulation models. The software is mainly used in the following fields of industry: automobiles, aerospace, and components. Two main kinds of questions can be answered with this program: capacity analysis and resource planning. For these purposes, specific building blocks exist enabling both easy modelling and sufficient options for simulation.

In the development of an emulation model, a building-block-based approach is used (Verbraeck et al, 2001). The advantage of a such an approach is the reusability of the environment. For the emulation model, the drive of the conveyor is extended with a pulse generator, indicating the motion of the conveyor. The PLC uses the output of the pulse generator to calculate the position of the bags, as well as to identify if the conveyor is functioning properly. Also aspects such as hardwired signals and the status of safety switches at local controls can be easily implemented this way.

Figure 3 shows a detail of the emulated section; each conveyor was equipped with a safety switch, depicting the status of the conveyor. Furthermore, this object enabled the option to create errors (to prevent this from happening in the real system) and perform modifications to the conveyor.

Before the emulator could be used as a test case for the PLC, it was important to verify its functioning. To see whether the emulator was functioning properly, all of the separate functions had to be tested with an independent tool. This was to guarantee that the emulator was tested and not the PLC. Therefore, the PLC was replaced by a visual basic application functioning as a graphical display and a signal inverter.

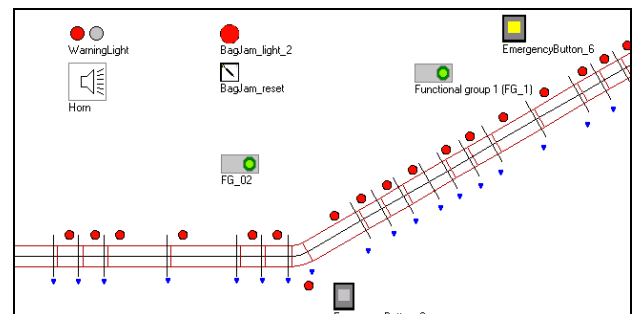


Figure 3: Emulation Visualisation

## 4 RESULTS

The approach to linking a simulation environment to a real PLC proved to be a valuable solution. The manufacturer had a tool enabling visualization of processes (see figure 5, below), easy access to change settings of objects and a large freedom for defining the logistic process.

When reviewing a normal development project within baggage handling the following phases can be noted: design and engineering, production, installation, tests and support. The normal lead times for a small project, are given in Figure 4. The project is part of a larger project at a European airport and concerns one PLC controlling a regular conveyor system. In general, 45% of the lead time is taken by installation and testing activities. On average a reduction of 30% of that time could be achieved with emulation. This could also be translated into a reduction of 25% in project costs (see also figure 4).

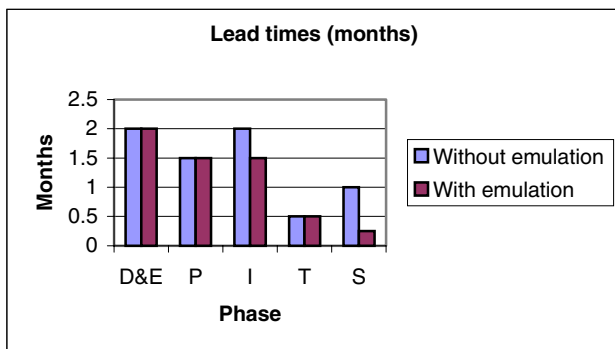


Figure 4: Experienced Lead Times Recent Project

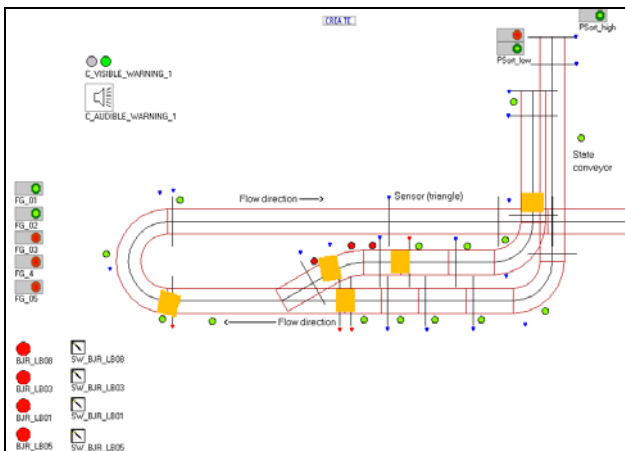


Figure 5: Baggage Flow Visualisation

The graph shows the lead times for the total project. Whenever the lead times for controls are considered, the actual reduction in lead time was even higher. Furthermore, the FAT could be executed easier because of the chosen way of emulation. As can be seen, the required time

on-site was reduced significantly, which is of course related to the improved quality of the installed controls. This already became clear during the site acceptance tests (SAT), the amount of occurring control related problems was reduced significantly.

The emulation environment was able to support the creating of required features for tests, such as specific bag problems (blockage, also known as bag jam). Moreover, by displaying the calculated control flow, errors within the tracking control system could also be discovered easily.

Using eM-Plant proved to be very successful because of the following reasons:

- A large number of basic processes were already depicted within the baggage simulation library.
- The required details for the emulation were able to be easily implemented because of the OO structure.

However, also a number of limitations occurred within the simulation package:

- Real-time and processor loading.
- Pixel-based animation.
- Only one active thread.

Whenever performing actions with poptexts, scrolling, etc., the simulation stopped. The reasons for these limitations could be found in the goals of simulation packages: the minor necessity for real-time simulation and a much lower required level of detail in an average simulation project.

Summarizing, it can be said that emulation with simulation is possible because of a similar approach in depicting processes.

## 5 DISCUSSION

Our customer's objectives could be met by using the emulator. However, the full advantages of emulation have yet not been utilized, nor have all functions for emulation supported by a simulation environment been realized yet.

To increase the added value of the emulator during the testing phase several functions should be added:

- logging of incoming PLC events;
- logging of performed actions within the emulator;
- creating a scenario tool;
- replay possibilities.

This would improve tests because cause-and-effect relationships can be identified easier. The value of this can also be seen in the real environment. Whenever all events are logged at the transport control or TM level, occurring errors can be regenerated.

Emulation is currently used to visualize the flow and analyse the correct functioning of low-level control within

PLCs. However, it can also be used as a tool to improve the lay-out of the system. The emulator can easily be used to define the capacity and travelling times of the emulated section. In the current design trajectory simulation is used in:

- The specification phase: analysis of various alternatives.
- The design phase: proving the validity of the design and improving the design of controls and lay-out.

As of now, by using a simulation environment to emulate real equipment with controls, the application of simulation is being extended to the realisation phase.

In the long-term, simulation and emulation should be used as an integrated part of the design process, analysis, tests and realization. Optimal use should be made of the activities done during all of the former phases. This requires reusability of models, easy adjustability for different lay-outs, and lastly, project management based on the developments. The objective should be to reduce long lead times and improve the quality of both the product and the controls. (Saanen 2002b).

The ultimate use of simulation will be to create high- and low-level controls within the simulation environment. Other projects carried out in another domain (Dobner et al, 2002a and 2002b) show that such an application of simulation is feasible. The input for the PLC is sufficient to test the controls. This also means that controls within the emulation environment should be fast enough to enable real-time control of the equipment.

To enable such a future, the developers of simulation environments (eM-Plant) must undertake the following initiatives.

- Develop a more suitable application for animation, enabling the possibilities of the current state of graphic PC cards. It is important to decrease the processor load due to animation, and furthermore, faster and higher detailed animation is required.
- Enable handling of different threads, in such a way that activities can be performed in the simulation without affecting the simulation clock (e.g., using user interfaces for switching on/off equipment).
- Mode changes in the depicted processes within the emulator should always lead to an event (e.g., change of acceleration to steady running of a conveyor should cause an event).
- Improve the accessibility of eM-Plant from external packages and enable direct access to the simulation clock by using multiple models simultaneously and synchronously.
- Improve the velocity of on-line communication with a running models.

In general, the simulation environment should become more OO and support features required for real-time depiction of systems. The feasibility of these requirements exists considering the developments for games, net applications, etc.

## 6 CONCLUSION

Developing an emulation environment by extending an already existing simulation environment was successful. Due to the structure of the simulation environment the additions could be easily made. The manufacturer was able to finish the project on time, with only limited on-site adjustments. The whole project at the airport was finally done with support of emulation.

The techniques have been used in other projects and show also there a reduction in lead time and cost. These reductions are the highest when is started with using simulation for capacity analysis and finished with using basically the same model for emulation.

Improvements have been made to the emulator to improve factory acceptance tests (FAT) and error analysis. Down the line, benefits of the status of software development might be expected in simulation environments as well. These benefits might include an improved animation of the actual process enabling a better outcome of tests.

To reach an optimal use of simulation, one of the tasks for the developers of simulation software (eM-Plant) is to improve the openness of the environment. Besides that fast online communication with other packages must be guaranteed.

## REFERENCES

- Dobner M., J.R. Rijsenbrij, H.C. Eichner, Y.A. Saanen (2002a), Harwich Feasibility Study Ports of Felixstowe, internal report, Mannesmann-Dematic, Düsseldorf, 2002.
- Dobner M. et al., (2002b), Cost Performance Study APM Terminals, internal report Dematic Mobile Cranes, Düsseldorf, 2002.
- M.C. van der Heijden, Y.A. Saanen, A. van Harten, E.C. Valentin, M.J.R. Ebben, A. Verbraeck. "Safeguarding Schiphol Airports accessibility for freight transport – The design of a fully automated underground transport system with an extensive use of simulation". Paper accepted and to be published in: *Interfaces*.
- Hilscher GMBH, 2002, Website document, <<http://www.Hilscher.com>>, [last accessed on April 4, 2001]
- Saanen Y.A. and U. Franzke (2000a), "Preparing simulations for more advanced purposes: design of an automated container terminal". In: Kai Mertins, Markus Rabe (eds.). *The New Simulation in Production and Logistics – Prospects, Views and Attitudes*. Proceed-

- ings 9. ASIM Fachtagung "Simulation in Produktion und Logistik". Berlin, Germany, 8-9 March 2000. IPK, Berlin. ISBN 3-8167-5537-2. p. 233-244.
- Saanen Y.A., A. Verbraeck, J.C. Rijsenbrij (2000b), "The Application of Advanced Simulations for the Engineering of Logistic Control Systems". In: Kai Mertins, Markus Rabe (eds.). *The New Simulation in Production and Logistics – Prospects, Views and Attitudes*. Proceedings 9. ASIM Fachtagung "Simulation in Produktion und Logistik". Berlin, Germany, 8-9 March 2000. IPK, Berlin. ISBN 3-8167-5537-2. p. 217-231.
- Saanen Y.A. (2001a), "Designing software for robotised terminals", AAPA conference on Automation, Oakland, 2001.
- Tecnomatix, eM-Plant Reference Manual 4.6, Tecnomatix Technologies Ltd., 2002.
- A. Verbraeck, E. Valentin, Y.A. Saanen, "Simulation as a Real-time Logistic Control System: AGV Control with Simple++". In: Kai Mertins, Markus Rabe (eds.). *The New Simulation in Production and Logistics – Prospects, View sand Attitudes*. Proceedings 9. ASIM Fachtagung "Simulation in Produktion und Logistik". Berlin, Germany, 8-9 March 2000. IPK, Berlin. ISBN 3-8167-5537-2. p. 245-255.
- C. Versteegt, A. Verbraeck. "Evaluating the design of fully automated logistic systems using a combination of simulation, emulation, and prototyping". Paper submitted to: *Wintersim 2002*, San Diego, December 2002. Currently in review.
- Wysk R., 2001, Website document, Rapid Prototyping and Development of FMS Control Software for Computer Integrated Manufacturing, <<http://www.engr.psu.edu/cim/rc>> [last accessed on June,21 2001]
- plied in a number of projects at different container terminals, and has been successfully validated by the terminals themselves. Besides he is finishing his PhD (planned 2002) on a design approach for robotised maritime container terminals. The PhD is performed at Delft University of Technology, faculty of Technology, Policy and Management, department of Systems Engineering. Lastly he is also a lecturer at Delft University of Technology, faculty of Technology, giving master courses in simulation and logistics. He gives MBI courses as well at the Rotterdam School of Management. His email addresses are <Y.A.Saanen@tbm.tudelft.nl> and <yvo@tba.nl>.

## AUTHOR BIOGRAPHIES

**WILLIAM RENGELINK** is consultant at TBA Nederland. He received his MSc. (Aerospace Engineering) from the Technical University of Delft, the Netherlands. His work is in the fields of simulation and emulation for logistic systems within airports. He is involved in various international airport projects. His email address is <william@tba.nl>.

**YVO A. SAANEN** (MSc in Systems Engineering) works as principal consultant at TBA, a leading simulation consultancy company in the Netherlands and active for ports all over Europe. TBA is involved in a number of state-of-the-art automation projects concerning underground transportation at Amsterdam Airport Schiphol and the design of the Second Maasvlakte in the Port of Rotterdam. He is the main architect behind the TBA port simulation suite that enables terminal operators, shipping lines and integrators to design, and optimise their terminal and plan their operation in a more efficient way. This port suite has been ap-