Experiences of Simulation Use in Industrial Projects

P. Fonseca i Casas, J. Casanovas, J. Montero and A. Guasch.

Statistics & Operational Research, Campus Nord – B6. Universitat Politècnica de Catalunya, c/ Jordi Girona 1-3, 08034 Barcelona (SPAIN) Tel. +34 934016941
Fax +34 934017040
E-mail: {pau, josepk, monty }@fib.upc.es, toni.guasch@upc.edu

Abstract *– This paper presents experiences obtained from our involvement in the development of industrial simulation projects. Some important, common questions are covered, such as the need to define model behavior using a conceptual model, the problem of choosing the appropriate tool to code the model, and the validation and verification process required. As we will see, the scope of applicability of simulation is broad and the tools are therefore diverse. A clear understanding of the objectives of the simulation, the client's aims and the resources at our disposal are key issues that often determine the success of a simulation project.*

1. INTRODUCTION

Over the last fifteen years our simulation team (affiliated to the Computing Laboratory of the Barcelona School of Informatics, LCFIB) has carried out simulation projects that often have a common characteristic: they are critical or present some technical complication that entails an element of risk (hopefully only in the economic sense).

In this paper we present some of the experiences obtained from the projects in which we are, or have been, involved.

Not all simulation projects have the same objective. Although the main purpose of constructing a simulation model is usually to understand and predict the behavior of a system, this system knowledge can be applied for several purposes. According to (Kellner, Madachy, & Raffo, 1999), the main categories that define the purpose and use of a simulation model are:

1. Strategic Management.
2. Planning.
3. Control and operational management.
4. Process improvement and technology adoption.
5. Understanding.
6. Training and learning.

In this paper we review three examples, related to *planning*, *control and operational management* and *process improvement and technology adoption*.

2. SIMULATION FOR PLANNING

Planning in organizations is the process of creating and maintaining a plan following a defined purpose. Planning entails studying the likely consequences of several alternatives and choosing the most suitable option. Simulation can be used to support the planning process in a number of ways by forecasting outcomes and facilitating decisions regarding resource constraints and resource allocation.

The example presented in this section is the simulation of an industrial plant operated by the pharmaceutical firm (Casanovas, Perez, Montero, & Fonseca, 1999). Its main objective is to represent the industrial processes requiring planning during the merger of two major pharmaceutical companies, each with its own management culture and technical structure, which needed to be aligned to unify production processes. From the initial stages of the merger, all of the actors involved in the project understand the advantages of simulation as a tool for evaluating different plant layouts and alternative production planning schemes.

The project focuses on the simulation of reception, picking, expedition, and production orders at the pharmaceutical plant in question. The baseline situation is as follows: the order expedition process is partially automated, and completed products are stored for subsequent distribution. Since the existing processes are semi-automatic, the merging of several company units could have a major impact on different sections of the plant.

One of the main concerns is to enable increased throughput and to detect the possible bottleneck problems anticipated for intermediate and completed products. The simulation model must also consider the issues associated with the flow of semi-manufactured products between the company's different laboratories and plants, to create a representation of the complete in-factory transportation system

Some of the structural modifications and operational changes envisaged were as follows:

- To maximize the use of automatic processing by installing a robot to carry out the picking of completed boxes.
- To analyze unit-picking data to evaluate the cost and potential suitability of automatic dispensers for the picking of product fractions.
- To introduce new order preparation policies.
- To increase production levels.

On the one hand, these modifications must maintain existing quality and due-time parameters, and alternative shift policies should be evaluated to reach a maximum level of 24-hour customer service. The duration of the order preparation process and existing agreements with clients and transport companies must be respected. Special short-life products have to be considered.

On the other hand, production costs and picking of fractions must be reduced. To achieve this, a tool was created to design the optimum plant layout, including alternative equipment and specialized machinery. Finally, the different alternatives and possible decisions had to be explored and compared.

To achieve these aims, several tasks were considered necessary, some related to data and others related to the processes that govern the system behavior. Consequently, two main model phases were conceived:

1. Macroscopic specification and modeling. Numerical modeling covering the hypotheses related to data and data validation. Here we only consider the main industrial elements, the behavior of which is simple.
2. Detailed microscopic modeling of machines, conveyors, resources and other elements. In this phase we also consider the behavior of the individual elements that make up the system.

Operational validation of the different simulation scenarios was performed during the study of the system. As a result of this validation, different modifications were requested for certain elements in the initial layout. These modifications often do not imply that the initial plant behavior is incorrect but rather include several alternative equipment evaluations. Since modifications of this type are commonly introduced during project development, the prototyping approach (Janson, 1986) was one of the most effective ways of maintaining focus on the client's requirements, as it enabled us to implement these changes to the model quickly and without incurring substantial additional costs.

The R&D team was of medium size (5 members). As is often found in this type of project, the main problems concern the quality and availability of current or historical data and the agenda of critical factory experts.

2.1. Modeling methodology

To develop the model, the LeanSim methodology was used, which consists of the following steps:

1. Identify the objects to be considered in the model.
2. Define the channels for communication between objects.
3. Define an external event list for each object.
4. Define the detailed behavior of each object.
5. Define the external routines.

There is a clear distinction between the model definition, which follows a formal language, and its implementation, in C++, which facilitates model validation, verification and maintenance.

There are several languages for formally representing a simulation model. In our projects we use the Specification and Description Language (Doldi, 2003), (Telecommunication standardization sector of ITU, 1999), Petri Nets (Recalde, Teruel, & Silva, 1999), (Peterson, 1981) or DEVS (Zeigler, Praehofer, & Kim, 2000). The choice of a specific formalism is determined by the nature of the project and the industry figure you are dealing with. It is important for the client to feel confident about the language and to understand it to a sufficient level to be able to assist with the conceptual validation of the model.

At the operational level, the model definition for this project consists of three steeps:

1. Define the block diagram, the system and the outermost agent, following the SDL specification paradigm.
2. Define the state diagrams. Although this representation can be ambiguous, it simplifies communication with the client. It is our responsibility to ensure that this representation is consistent at all times with any other system representation diagrams.
3. Process and procedure diagrams for all the elements, following SDL. This representation is formal, complete and unambiguous and helps us to fully define the behavior of the model elements, simplifying the implementation.

Figure 2 shows the state diagram for one of the model elements. Figure 1 shows a simulation procedure using SDL language.

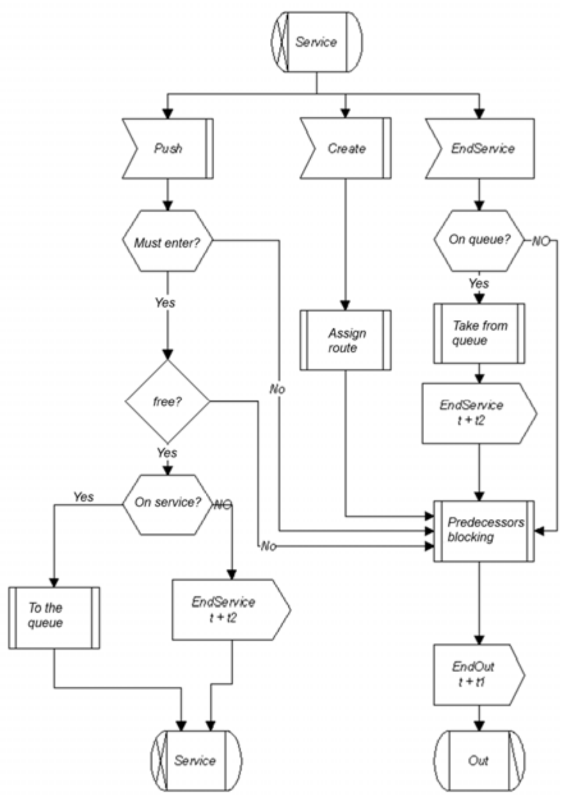


Figure 1. SDL procedure diagram.

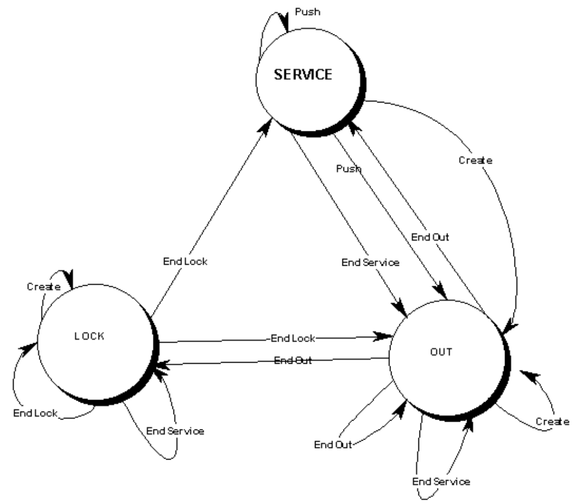


Figure 2. State diagram.

2.2. Model implementation

This project was implemented as a stand-alone application in C++.

Although the graphical representation is not a mandatory element of this project, it is vital for understanding the model behavior and performing the operational validation. Accurate representation of the model has several advantages, including the following:

- The user can view the system modeled in the simulation as a replica of the real system.
- The model is easier to validate, increasing the overall process credibility from the end-user perspective.
- Minor errors or inconsistencies can be debugged during the specification phase, leading to a better understanding of the model.
- The model can be used as a powerful operator training tool.
- The model shows the closest representation to the real image of the studied system.

Of course, if the simulation model is in the *training and learning* category, an accurate representation is a key element.

To produce an accurate representation, two principal alternatives are available: Off-line and Integrated. In **Off-line**, the representation is based on analysis of a list of events stored in a file (or database). This technique does not allow interaction with the model but it is generally easy to implement and consumes less resources that the integrated option. In **Integrated**, the representation of the model is shown during execution. In this case, the visualization is closely bound to the simulation model and there is a dynamic connection between visualization and simulation. The system implements an interactive on-line communication mechanism in both directions throughout the execution of the model. This enables the user to determine the operational state of the model at any moment by observing what appears on in the screen and to introduce changes when desired. This approach can be used for the implementation of training systems. Implementation of the integrated option is usually more complex and more resources are required to execute the model.

The simulation environment developed in this case consists of a simulator with an interactive graphical representation of the model.

Figure 3 show a virtual representation of the robot element.

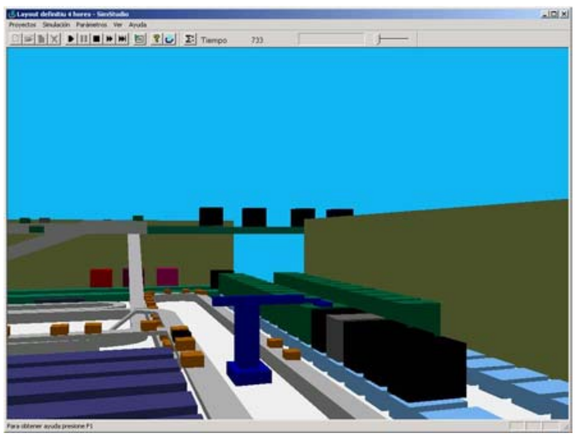


Figure 3. Automatic robot.

Partial results are visible in real time, and several scenarios can be studied and compared. The user can change certain scenario parameters in real time and observes the effects of these changes on the model.

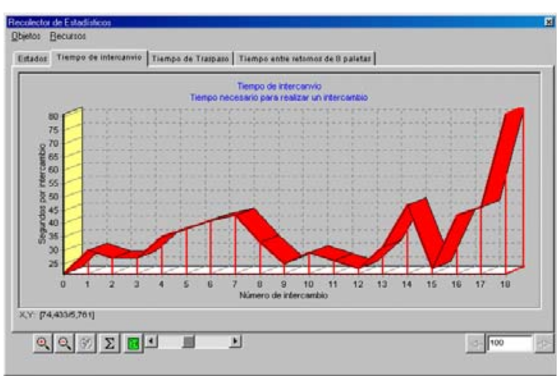


Figure 4. Results obtained during execution of the model.

2.3. Results and validation

The system was designed to be parameterized from a standard text file listing the required characteristics of objects, model parameters and statistical results (Figure 5 shows the parameterization window).

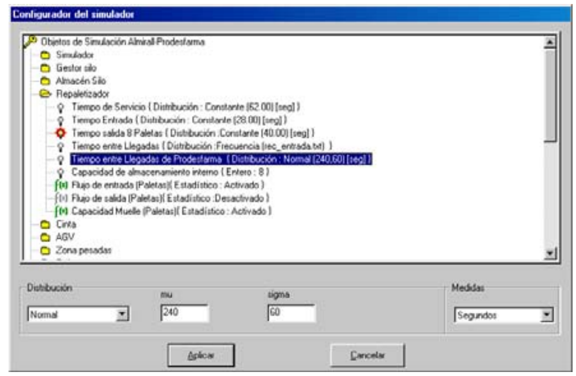


Figure 5. Model parameterization window.

Solutions were obtained for several different alternatives and scenarios, mainly proposed by the client. Here we do not discuss the alternative solutions proposed for the industry, and the Validation, Verification and Accreditation process (Sargent, 2007), but it should be noted that validating the solution is very important for the simulation team. Validation of the solution consists of client analysis of the use of the simulator in the industrial application it models and implementation of the conclusions.

The main consideration in our project is that the project budget and outlay for picking and expedition areas are determined on the basis of the conclusions drawn from the simulation. The revision of the model and final implementation of the solution, one year after the end of the project, indicated that the simulation results were accurate and that the solution successfully tackles problems encountered in the industry in question.

3. SIMULATION FOR CONTROL AND OPERATIONAL MANAGEMENT

In this section we present the simulation of the new Barcelona International Airport Terminal (NAT). In this huge project, the main objective is to provide the architects, civil engineers and aeronautical engineers with the information they need to ensure that the new terminal performs well following the IATA standards. As such, the primary use of the simulators could be for planning, although the resulting models, a set of micro and macro simulators (that allow predicting risky situations in the operational management of the process related to the airport), was subsequently used to execute the operational management of the airport. So in this section we focus on this part of the model.

Airport management is a complex task because many flows converge in different terminal spaces. Each of these flows has its own personality and characteristics, and it is obviously necessary to know the structure and quantity of each flow in order to manage the different airport spaces and prevent bottlenecks. Space requirement calculations for an airport are typically based on differential equations or queuing theory. These approaches, however, are not ideal if we want a precise model of the real environment or a model that can be defined using a large number of different parameters. Our project is based on the need to identify the potential bottlenecks in the operation of the new terminal before its construction. This determines the build requirements for the simulation model, which was developed to reflect the real environment as closely as possible in order to return the most realistic data about the possible collapse of certain critical areas. Check in, ticket control, baggage retrieval and many other resource-limited processes determine the quality level of an airport, and in a completely new design it is very difficult to determine the most suitable dimensions of each area because the behavior of the different flows (such as passengers flows) is hard to predict.

3.1. Modeling methodology

Again, we applied the LeanSim methodology, defining detailed behavior for each of the simulation entities and elements that compose the model.

The micro-simulation of the new terminal at Barcelona International Airport is an extremely large-scale simulation project involving five teams (a team of computer scientists and statisticians who develop the model, an aeronautic and civil engineering team, aeronautical teams, architects, and an external consulting firm responsible for performing the validation of the model). This means that different languages and knowledge must be processed. Again, the formalization of the model helps a great deal. SDL was selected due to the heterogeneity of the team involved.

Although the processes that govern the model behavior were relatively simple to represent, one of the main problems in this project was to define the nature and structure and model the data that govern the behavior of the model.

This project has two objectives: to test the structures, spaces and processes of the new terminal, and to act as become a testing environment to analyze the operations in typical daily use.

The modeling process of big systems like airports, that are intended to be used for management, requires low response times in front of new configurations. To allow a faster modification of the model configurations we develop a Witness model that can be constructed dynamically from a set of files (that allows us the creation of the simulation objects that defines the model). These files also controls the generation of new entities through the Witness part files, the files that represents the generation of entities in witness (Lanner, 2011). All the parameterization was centralized in an excel Spreadsheet based application. This application, that represents all the main elements of the Terminal, allows the generation of the files needed to define a new simulation scenario (the configuration files that define the model elements and the part files that feeds the model.

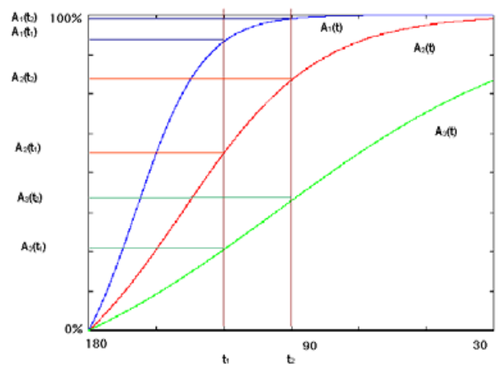


Figure 6. Hypothetical presentation functions for the passengers.

In this project we need to define the probability distributions that govern the behavior of the airport over the coming years (from 2007 to 2025), and since the date are not available we need to use statistical techniques to define the distributions.

It is important to note that six months were spent acquiring, analyzing and structuring the data for its correct use in the simulator. Data acquisition and analysis can be a demanding task and care must be taken to monitor the time required to complete this phase to ensure the success of the project.

3.2. Implementation of the model

The selection of the implementation tool (or tools) for a specific simulation project is a task in itself.

In any simulation project, the system requirements must be analyzed to determine the best option for implementing the model. In this case the implementation tool, for interoperability reasons was selected by the client. However, as always, we analyze potentially suitable alternatives for performing the implementation to identify the strengths and weaknesses of the selected tool. To do this we often use the methodology presented by (Rincon, Alvarez, Perez, & Hernandez, 2005). Notably, the technology constrains the nature of the implementation and the time needed to reach a solution. In our case, the analysis showed that, due to the time constraints of the project and the structure of the team involved, Witness (Lanner, 2011) (the tool used) is a good alternative for implementing this model.

Figure 7 shows the layout of the model representing the jet bridges of the Schengen area of the airport.

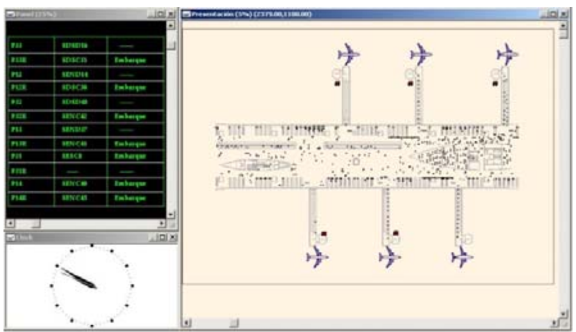


Figure 7. NAT layout.

3.3. Results and validation

Models must be executed to obtain the results. Here, it is important to note that, due to its detail, a single execution of the initial micro simulation model (that encompasses a complete subsystem) takes approximately three hours. This implies that if replications of the model are required (at least eight in our analysis) we need 24 hours to obtain the results, after which we need to analyze the output and write the report. Two comments should be made at this point: first, time is needed to obtain the results, so when defining the project plan it is important to remember that experimentation can be time consuming; second, certain techniques can be used to parallelize the simulation to reduce the time needed. A very simple technique is to use “n” computers that execute the model replications in parallel, although other, more complex techniques can be considered depending on the project, See (Fujimoto, 2001) to review a set of techniques to perform a parallel simulation.

In this project the managers continue to use the simulator on a regular basis for their operations. For example, one of the simulation models implements an algorithm to assign the jet bridges for each of the planes that arrives at the airport. This algorithm can be modified, and since the simulation model represents all of the flows in the airport, the managers can determine the effects of new assignment algorithms over the whole airport.

4. SIMULATION FOR TECHNOLOGY ADOPTION

Testing and debugging logistic control systems in real scenarios can be very difficult when there is a risk of disrupting manufacturing or logistic operations. Moreover, not all possible scenarios can be tested because constraints are introduced by the plant status. Finally, some problems may be difficult to solve due to the complexity of tracing their possible causes and the impossibility of repeating the exact same sequence of events that caused the failure.

In this project we focus on the simulation of a production plant. The main objective of this project is to perform a simulation of the AGVs of an important beer industry in order to evaluate the current management and planning algorithms. Secondary objectives include testing and validating the communication protocols between the different agents involved in the transport operations and analyzing the behavior of the system in response to different situations.

4.1. Modeling methodology

In this project we use Coloured Petri Nets (CPN). This formal, graphical language has proved to be successful a tool for modeling logistic systems thanks to several advantages such as the concision with which it represents both the static structure and the dynamics, the availability of mathematical analysis techniques, and its graphical nature (Recalde, Teruel, & Silva, 1999). Furthermore, CPN is very suitable for modeling and visualizing patterns of behavior comprising concurrency, synchronization and resource sharing, which are key factors when trying to optimize the performance of logistic systems.

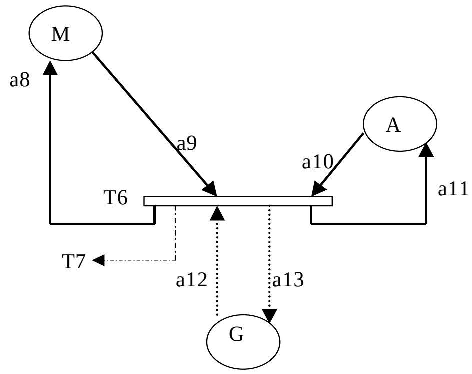


Figure 3. Colored Petri Net of the path generation

However, although CPN models contain the essential information to build the simulation model, they are not widely used in commercial simulators as a mechanism to code and specify simulation models, nor are they commonly used by the optimization community, due to a lack of qualitative analysis tools. Consequently, a generic simulation tool must be used.

4.2. Implementation of the model

The tool selected to implement the simulation model was Arena (Figure 8 shows a detail of the model). Several additional components were also needed.

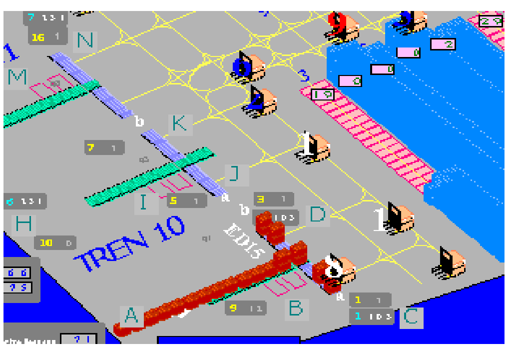


Figure 8. Line 2 simulated section.

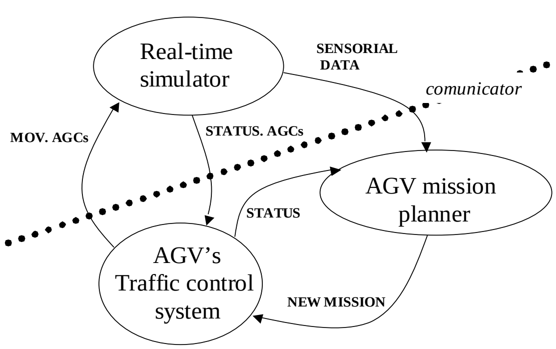


Figure 9. System architecture.

The system architecture is composed of the following modules (see Figure 9):

- AGV mission planner.
- AGV's traffic control system.
- Real-time simulator.
- Experiment design sheet in MS-Excel.
- Communicator (ECI).

In the hardware architecture it is possible to identify four PCs involved in the study—clients as yellow PCs and servers as orange ones—and the message types transmitted between them. The double blue arrows identify a communication via TCP/IP, whilst the single red arrow shows a Remote Procedure Call (RPC).

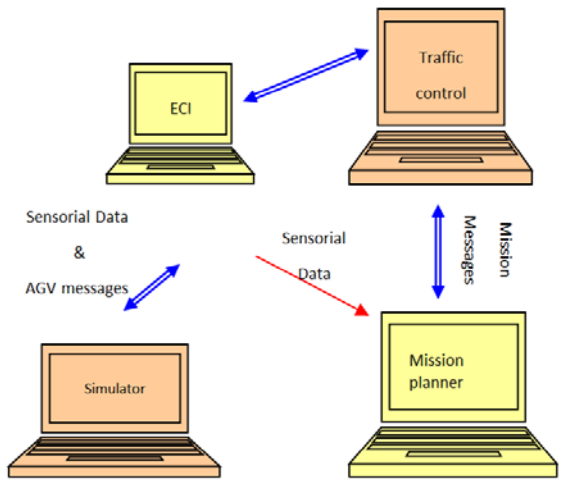


Figure 10. ECI, Architecture & message communication

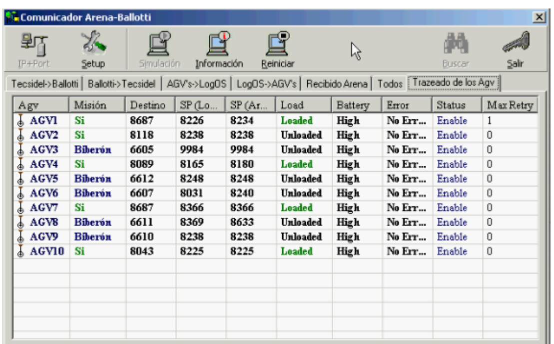
Two techniques were used to integrate the communication services into the hybrid simulation system.

- Transmission and reception of messages using sockets over TCP/IP protocol.
- Remote Procedure Call (RPC), to emulate the sensorial signals in the plant and update distributed databases in other PCs.

These techniques are encapsulated in the External Communication Interface (ECI). The ECI is a communication gateway between the simulator and the AGV's traffic control and mission planner (Figure 11). The main tasks associated with the ECI are shown in the following points: (i) to establish the correct connections between the real-time simulator, the AGV's traffic control, the mission planner and the ECI; (ii) to propagate the messages from the simulator to the AGV's traffic control and vice versa with the codec/decodec needed for comprehensive use; (iii) to update, using RPC, the mission planner's databases of those plant sensorial signal records which are simulated and sent from the Simulator to the ECI; (iv) to report in real time the status of the simulated

AGV's battery level, mission status and load status; (v) to generate log files, which record the movement sequence and messages generated, enabling the debugging process.

We developed the ECI component instead of using an embedded alternative because: (i) it allows a simple analysis and trace of outgoing and incoming messages, simplifying the detection of bugs in the communication ports associated to external software; (ii) it decreases the overall development time, since model building and the communication protocol tasks can proceed in parallel; (iii) the ECI can run on a different PC. Therefore, the speed of the simulator increases. This is an important factor, since a working mode which runs faster than real time is required to reduce the total amount of time needed to perform the simulation experiments. A real-time scale factor of 5 was chosen because the AGV's traffic control and mission planner could not function properly at higher speeds.



Agv	Misión	Destino	SP (Lo...	SP (Ar...	Load	Battery	Error	Status	Max Retry
AGV1	Si	8687	8226	8234	Loaded	High	No Err...	Enable	1
AGV2	Si	8118	8238	8238	Unloaded	High	No Err...	Enable	0
AGV3	Biberón	6605	9984	9984	Unloaded	High	No Err...	Enable	0
AGV4	Si	8089	8165	8180	Loaded	High	No Err...	Enable	0
AGV5	Biberón	6612	8248	8248	Unloaded	High	No Err...	Enable	0
AGV6	Biberón	6607	8031	8240	Unloaded	High	No Err...	Enable	0
AGV7	Si	8687	8366	8366	Loaded	High	No Err...	Enable	0
AGV8	Biberón	6611	8369	8633	Unloaded	High	No Err...	Enable	0
AGV9	Biberón	6610	8238	8238	Unloaded	High	No Err...	Enable	0
AGV10	Si	8043	8225	8225	Loaded	High	No Err...	Enable	0

Figure 11. ECI, detail AGV status.

Additionally, the ECI emulates a third software application which manages all of the missions in the production line. These tasks are:

- To update the newest product references in the scheduler database.
- To initialize the warehouse status, i.e. to set up several experimentation scenarios.
- To acknowledge production line messages to the mission planner.

4.3. Results and validation

The last example is an industrial application developed over a five-month period. The key factors that contributed to the success of the application are: (i) the use of a real-time system (Arena) to model the plant behavior and the AGV fleet; and (ii) the use of the ECI communication gateway. This component simplifies the debugging of the communication protocols. Furthermore, since the ECI can run on a different PC, the speed of the simulator increases.

Again, the formalization of the model—in this case using CPN to represent the complex AGV behavior—simplifies its implementation and the definition of the model behavior.

In this project, success was again achieved by verifying the accuracy of the model, implying that the AGV fleet can be used in the industry in question.

5. DISCUSSION

All of the examples given in this paper contain common elements that we believe contribute to the success of a simulation project.

The first key element is the use of a formal model. This simplifies the understanding of the hypotheses used by all the actors involved, simplifies the verification of the model implementation and simplifies the validation process.

The second element is to hold regular meetings with the client, to revalidate the hypothesis document on a regular basis and keep the formal representation of the model up to date. This is important as it increases the credibility of the model from the clients' perspective; ultimately, the model belongs to the modelers and the client, not to the project team.

The third element is the use of techniques to accelerate the execution of the model, running parallel executions in the case of the Airport or a distributed component simulation in the case of the beer industry. A distributed simulator is not always necessary, but some techniques (not necessarily complicated) can help to accelerate the execution of the models.

The fourth element is the creation of an accurate representation. Although in some cases the client does not require one, the more accurate the representation the easier it is to validate the model. Since the tools currently help on this a good animation (Law, 2005) can be considered a must.

The fifth element is the validation stage, which is vital in establishing trust between the project team and the client. It also increases the level of confidence in the techniques used and, of course, helps to detect errors.

Finally, it is important to bear in mind that each action requires time: to acquire the data, to build the model, to implement the simulator, and of course, to execute the models and write the documentation.

6. REFERENCES

Casanovas, J., Perez, W., Montero, J., & Fonseca, P. (1999). Simulation of reception, expedition and picking areas of a pharmaceutical products plant. Proceedings of the 1999 IEE Industry Applications Society Annual Meeting. IEEE.

Doldi, L. (2003). Validation of Communications Systems with SDL: The Art of SDL Simulation and Reachability Analysis. John Wiley & Sons, Inc.

Fujimoto, R. M. (2001). Parallel simulation: parallel and distributed simulation systems. Winter Simulation Conference, (pp. 147-157).

Janson, M. (1986). Applying a pilot system and prototyping approach to systems development and implementation. Information & Management, 10(4).

Kellner, M. I., Madachy, R. J., & Raffo, D. M. (1999, April 15). Software process simulation modeling: Why? What? How? Journal of Systems and Software, 46(2-3), 91-105.

Lanner. (2011). Witness. Retrieved 02 14, 2011, from Lanner: http://www.lanner.com/en/witness.cfm

Law, A. M. (2005). How to build valid and credible simulation models. In M. E. Kuhl, N. M. Steiger, F. B. Armstrong, & J. A. Joines (Ed.), Proceedings of the 2005 Winter Simulation Conference.

Peterson, J. L. (1981). Petri Net Theory and the Modeling of Systems. Prentice-Hall.

Recalde, L., Teruel, E., & Silva, E. (1999). Autonomous continuous P/T systems. Application and Theory of Petri Nets. Lecture Notes in Computer Science, 107-126.

Rincon, G., Alvarez, M., Perez, M., & Hernandez, S. (2005). A discrete-event simulation and continuous software evaluation on a systemic quality model: An oil industry case. Information & Management, 42, 1051-1066.

Sargent, R. G. (2007). VERIFICATION AND VALIDATION OF SIMULATION MODELS. In S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, & R. R. Barton (Ed.), Proceedings of the 2007 Winter Simulation Conference. IEEE.

Telecommunication standardization sector of ITU. (1999). Specification and Description Language (SDL). Retrieved April 2008, from Series Z: Languages and general software aspects for telecommunication systems.: ttp://www.itu.int/ITU-T/studygroups/com17/languages/index.html

Zeigler, b., Praehofer, h., & Kim, d. (2000). Theory of Modeling and Simulation. Academic Press.

Author Biographies

Dr. Pau Fonseca i Casas obtained his master degree in computer engineer on 1999 and his Ph.D. on 2007 from the Technical University of Catalonia. Now is full time professor of this university, teaching in statistics and simulation area. Also he works in the LCFIB (Barcelona informatics school computing laboratory) developing Simulation projects since 1998, and is member of LOGISIM, Centre of Simulation and Optimization of Logistic Systems. His email is pau@fib.upc.edu and his webpage is: http://www-eio.upc.es/~pau/.

Dr. Josep Casanovas is full professor of the Statistics and Operations Research Department at Technical University of Catalonia, his main research areas are Modeling and Simulation, Internet and Information Systems. Author of different research articles and papers in International Congresses, he is co-director of LOGISIM. He has collaborated in many projects for the EU and different companies and institutions. His email is josepk@fib.upc.edu.

Jordi Montero is a computer science engineer and a researcher member of the LCFIB (Computer Calculus Laboratory of the Barcelona faculty of Computer Science of the Technical University of Catalonia) developing Simulation projects since 1997, also is member of the LOGISIM group. His research interests are discrete simulation, distributed platforms and cell phone application development. His email is monty@fib.upc.edu.

Dr. Antoni Guasch received his Ph.D. from the Technical University of Catalonia (UPC) in 1987. After a postdoctoral period at the State University of California (USA), he becomes Professor of the UPC (www.upc.edu). Currently he is Assistant Professor in the Automatic Control Department (ESAII) in the UPC. Prof. Guasch is also co-director of LOGISIM and contributes to the development of toopath (www.toopath.com), a web server system for the free tracking of mobile devices. His CV and website can be accessed here: http://sac.upc.edu/people/academic-and-research-staff-1/dr.-antoni-guasch-petit-1. His email is toni.guasch@upc.edu.