ANALYSIS, MODELING AND SIMULATION OF THE INCORPORATION OF EGOVERNMENT IN ADMINISTRATIVE PROCESSES

Pedro Baquero[a], Yurena García-Hevia [b], Rosa María Aguilar [c]

[a] Department of F.F.&E, Electronics and Systems, University of La Laguna, Spain
[b] Engineer at the Computer Science School, La Laguna University, Spain
[c]Department of Systems Engineering and Automation and Computer Architecture, La Laguna University, Spain

[a] pbaquero@ull.es, [b] ygarcia@isaatc.ull.es, [c]rosi@isaatc.ull.es

ABSTRACT
In this paper we present a general model for analyzing the incorporation of eGovernment in the administrative processes that take place in a public institution. This model is developed in three conceptual visions which allow for advances in administration modeling. We start with an organizational vision and proceed to describe the operational vision (i.e., of the processes), and conclude with a technological vision that conceptualizes the processes in an architecture that is able to be simulated. Lastly, we detail a simple test case which analyzes the results of incorporating electronic processes into the services offered by an administrative agency.

Keywords: egovernment; administrative processes, simulation

1. INTRODUCTION

In recent years, public agencies have experienced significant advances from the impetus of the Information Society. To this end, in Spain, Law 11/2007, dated 22 June, on the electronic access to Public Services by citizens, was enacted for the purpose of laying the policy and legal foundations for allowing the citizen to file his paperwork with various agencies either in person or electronically. Among the requirements of this law is that the application of electronic means to handle procedures, processes and services be preceded by the performance of a functional redesign analysis and a simplification of the procedure, process or service. The following aspects are of particular significance: a) the elimination or reduction of the documentation required of citizens, and its substitution with data, data transmissions or certificates, or the regulation of the processing once the transaction is complete; b) an allowance for means and instruments of participation, transparency and information; c) a reduction in response times; and d), the rationalization of the work load distribution and of internal communications.

Within this context, the Government of the Canary Islands has promoted a project for the Interoperability of its Electronic Services intended to modernize its electronic infrastructure. This entailed an analysis of the organization to come up with a valid eGovernment model that would help achieve the primary objective, which was to gradually obtain an efficient model which would allow the Government to adapt to the number of conceptual, organizational and technological changes that still lie ahead. So those aspects intended to streamline and rationalize the process necessarily imply a reduction in the time required to resolve administrative procedures, which in turn result in reduced effort on the part of both the citizen and the agency. This requires that, as part of the planning process for the implementation of electronic procedures, a forecast be made of the expected improvements not only in the filing process for the citizen, but in the productivity of the agency.

This paper presents the framework used for the modeling of administrative process so as to yield a technological model that can be effected to simulate these processes. Section 2 presents the three conceptual visions of the model implemented. The starting point for the modeling process is a hierarchical scheme that involves three models: organizational, operational and technological. These models are described in Sections 3, 4 and 5 respectively. The organizational model merely considers the functional requirements of the simulation, and contains the description of the organization that is the object of the analysis. The operational model addresses the processes. In this paper we will focus only on those processes involving administrative procedures and will not consider the analysis of other types of non-structured processes. The technological model focuses on constructing an architecture that can be implemented in the simulation. Section 6 describes the simulation run on a test case involving how the Government of the Canary Islands oversees the Gaming Board. Lastly, the conclusions drawn from our research are presented.

2. HIERARCHICAL MODEL OF THE ADMINISTRATIVE PROCESSES

A public agency is an organization consisting of a multitude of units executing processes. We refer to the minimum organizational unit that performs a specific process as an operational unit. Said unit comprises a set of resources. The processes are executed by human resources and/or tools (e.g. computer applications) based on set procedures (generally written).

The structural relationship between operational units defines this organization's process architecture. If we consider the modeling of processes as related to administrative procedures, then the model encompasses three conceptual visions:

1. The organizational vision represents (a) the overall view of the relationships among all the operational units, (b) the strategic vision and the services the operational unit offers its clients, (c) a description of the organizational structure and of its associated resources, functions and procedures, and (d) the standards for evaluating and monitoring the operational results.
2. The operational vision defines how the entire organization's processes are carried out as one, as well as each operational unit's specific processes. The operational model describes: (a) the operations which support the organization's strategic mission; (b) the process necessary to provide each service in an operational unit; (c) the management of resources, both internal and external, used to provide the services.
3. The technological model defines how specific aspects of the organizational and operational model should be handled in the final implementation of the model and simulation of the ICT infrastructure. In the technological model we define mainly the general architecture of the entire organization, as well as the specific architecture of each operational unit.

The organizational and operational models form the theoretical reference model of the ICT infrastructure. The former considers the infrastructure as a static organizational model, and the latter as a dynamic organizational model. The technological model, on the other hand, translates these theoretical model into logical patterns that can be implemented using computational tools. A brief description of each of these models is given next.

3. ORGANIZATIONAL MODEL

At the first level of the organizational vision, the operational units are seen as black boxes such that certain inputs produce certain outputs.

The organizational model contains the description of the set of procedures grouped by services, which in turn are grouped by areas. In general, there are two decision grouping levels, the more specific one being found in the service grouping associated with executive level decisions. The less specific one is found in the area or theme grouping, and corresponds to policy decision levels. In the former, the decisions aim for administrative efficiency, while in the latter they are adapted to the policies of each governing body.

The highest level of precision contains the procedure definitions. These arise from an analysis of the legal guidelines regulating the services to be provided to citizens. In general, all of these procedures are set, there being implicitly or explicitly just one way to resolve a citizen's request for service.

The resource level is also developed within this model as conditioned by the agency's organizational structure, along with the functions they carry out. This model also considers the set of policies and projects necessary to implement the different computer applications which gradually allow for the incorporation of eGovernment. In this model we propose new ways to resolve service requests by using computational tools.

The organizational model also describes the target public, divided by services.

Lastly, the objectives to be met by each of the services are defined in the organizational model. For the case in question, these objectives are oriented at improving interactions with the customer by reducing the response and turn-around times and the rationalization of the work load distribution.

4. OPERATIONAL MODEL

The operational model defines the way the services are be apportioned to the resources being managed within the operational units. It also controls the mutual interaction between the operational units. The operational model translates the description of the elements and procedures of the organizational vision; that is, it introduces the element of time into the static representation of the organizational model. This model is thus better suited for modeling organizational processes.

The policy level is defined at the first level and uses a system for prioritizing certain policies over others, clarifying the prioritization of some services over others. The processes taking place inside the operational units are described at the second level. The process result from applying the procedures defined in the organizational model.

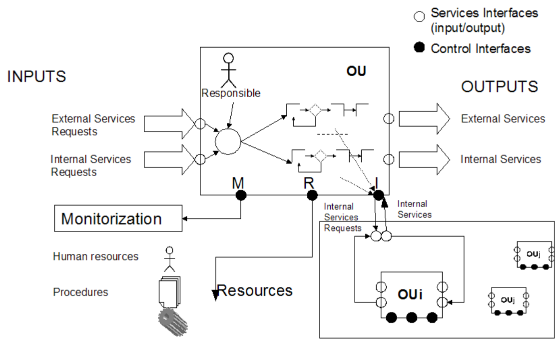


Figure 1: Operational model of an operational unit

Figure 1 shows a general representation of the process model for an operational unit. Basically, it shows how an operational unit (OU) accepts service requests (SR) from other OUs and from the environment (users). These SRs are prioritized internally for subsequent execution of the requests by the resources available to the OU. The resources are of two types, the first ones are human and tools (which execute the processes) and the second one are procedures (which specify how to execute the processes). The processes can, in turn, call other OUs to request services. In other words, an OU can carry out processes obtained from other OUs and for which they are not directly responsible. This means that in certain cases, conflicts may arise when deciding which process to prioritize. The policy level is charged with communicating to each OU which services take priority.

Each OU's resources and services are obtained from the organizational model. In addition, there is an interface which monitors the processes in accordance with the standards defined in the organizational model.

5. TECHNOLOGICAL MODEL

This model features an architecture intended for modeling and simulating processes. There are two architectural levels: one in which the organization's general architecture is represented, and another where the operational unit's specific architecture is represented.

5.1. General architecture

Figure 2 shows the technological model of the organization. Note the presence of an orchestrator (which represents the policy decision levels), responsible for executing the services depending on the organization's priorities, that is, prioritizing some services over others.

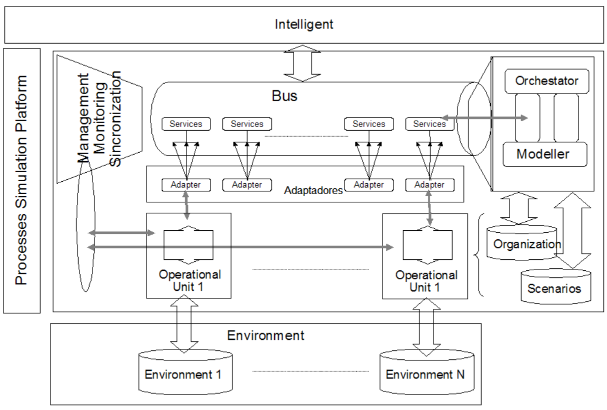


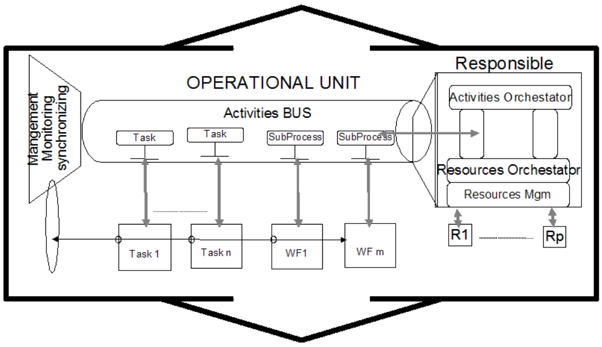
Figure 2: Overall architecture of an organization

The Orchestrator is charged with defining the different scenarios that can exist (for example, intensive vs. scarce use of the ICT by the users). The Orchestrator can also gauge each of the organization's available resources (for example, the introduction of electronic applications, assignment of human resources to the different OUs). The bus, in addition to providing communications between the OUs, is charged with publishing the different services available. A certain algorithm is used to assign the priority each service has to the organization. The modeled setting is also represented in a database, which is the set of citizens that can make use of the agency's services. Lastly, there is a unit charged with managing, monitoring and synchronizing the platform and whose purpose is the proper operation of the simulation platform.

5.2. OU Architecture

Figure 3 shows the technological model of the OU. We can see that there is another orchestrator, representing the manager of the executive level decisions. This orchestrator ensures that the processes are executed according to the priorities established by the policy levels, that is, it prioritizes certain processes over others.

Figure 3: Specific architecture of an operational unit



In this case, the Orchestrator allots, either manually or automatically, the resources to the activities to be executed. It also features its own internal control scheme so as to have a specific platform available for each OU.

The simulator uses different scenarios defined by the policy level.

6. SIMULATION (TEST CASE)

For this paper, we implemented and simulated the services associated with an office that offers services specific to the Gaming Board in the Canary Islands. The Gaming Board manages procedures associated to casinos, games of chance and arcade games.

6.1. Test case

In this subsection the organizational model is described. In this case, there are two OUs: one associated with the Game Management (GM), and the other with the Game Inspection (GI). An additional OU is used for receiving the service requests, the Input Register for the Government of the Canary Islands (IR).

The GM mainly receives service requests from the citizens. These requests are received by the IR. In addition, the GM needs certain services provided by the GI, such as a verification of specific actions to be carried out by the citizens. The GI must also perform inspections, whether they be occupational or resulting from citizen complaints.

The GM prioritizes the service requests according to technical requirements, such as (legally binding)

response times. The GM also prioritizes its inspection processes depending on policy requirements, such as prioritizing those processes associated with GM service requests above those arising from customer complaints, or vice versa.
The set of procedures associated with the GM and GI is fairly broad, the former having 52 and the latter 10 procedures. Each of these procedures has a specific volume of service requests. Some services are in constant demand, while others are seasonal. For example, a procedure called "removal due to substitution" may have 2000 requests spread evenly throughout the year, while the procedure called "exchange of slot machine" may have an isolated demand of 60 requests concentrated in the month of December.

Each procedure has an in-person and an on-line request associated with it. The first assumes the citizen does not make use of the electronic method to present his service requests. Different demand percentages for each of these types of processes are set in each scenario.

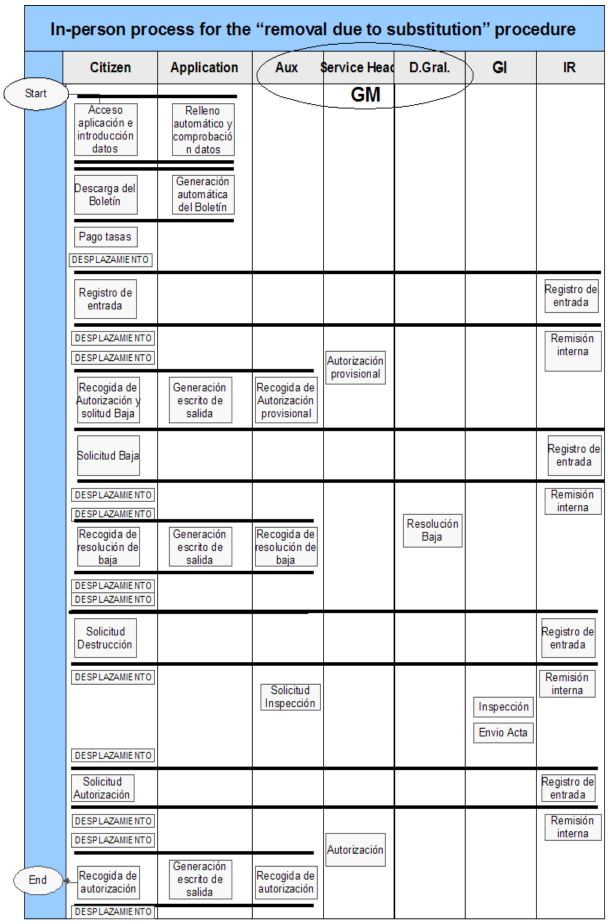


Figure 4: In-person process for the "removal due to substitution" procedure

6.2. Objectives

The objective of the use of simulation is to study, based on different scenarios:

- The average time to resolve the different procedures as the citizen makes use of the electronic method.
- The average decrease in the citizen's effort when the electronic method is implemented.
- The decrease in the human resources of the Government of the Canary Islands when the electronic method is implemented.

6.3. Simulation tool

The above considerations were addressed in this paper so as to yield a modeling and simulation environment for the services of a public agency. In this context, the simulation is presented as a tool to help management with the decision-making process (Hansem 2006). Through the simulation, the question of "what if" can be answered for different scenarios by checking the effectiveness of the possible actions to be applied.

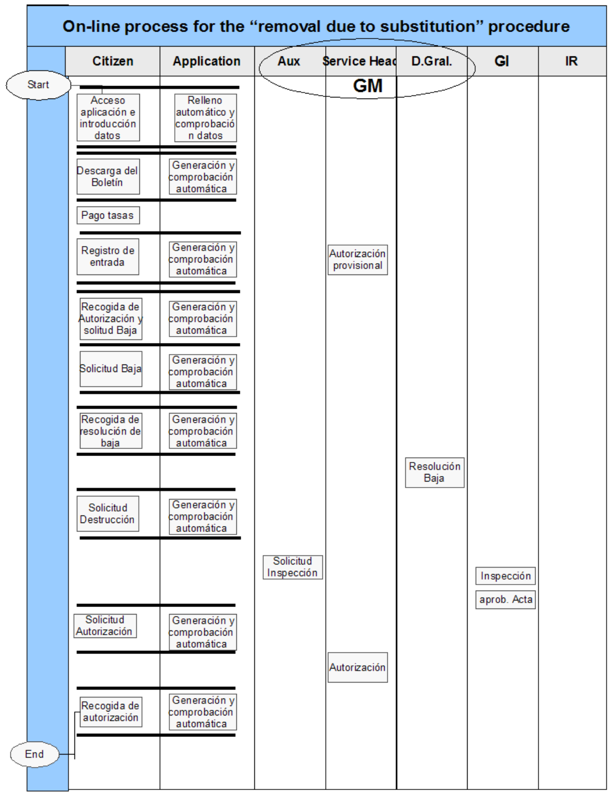


Figure 5: On-line process for the "removal due to substitution" procedure

The tool used was the Java simulation library for discrete event systems named SIGHOS (SIGHOS 2007), and specifically its XML interface (XMLGHOS) which facilitates the use of the library by introducing a layer that abstracts low-level simulation mechanisms (Aguilar et al 2006). Both tools are provided as freeware (http://sourceforge.net/projects/sighos/), and were developed by the Simulation Group at the Systems Engineering and Computer Automation, Architecture and Technology Department of the University of La Laguna.

The library uses a process-oriented methodology to model a system (Lorenzo 2001). In other words, the system is characterized by having the elements flow in stages which can vary depending on the status of the element itself. In one stage, these elements carry out different activities, and in doing so utilize system resources, meaning they have to wait for said resources to become available, at which time they will engage the resource for as long as it takes to perform the activity. This is why it is vital to specify when a resource is available and what functions it can perform.

6.4. Implementation

The first step consists of specifying the processes in accordance with the diagram in the figure. This is done by defining all the processes, along with the priorities and requirements to be met concerning time and importance. Figures 4 and 5 show an example for the same procedure for the two processes associated with in-person and on-line events.

Thanks to the modular design of the SIGHOS library, each module can be mapped to the architecture in Figures 2 and 3. The simulations are described simply by using XGHML (which is the XML SIGHOS Modeling Language).

The system to be simulated is described using two files, one referred to as MODEL, in which the resources, activities and flows followed by the elements are described, and the other as EXPERIMENT, in which the number of simulations desired of a model, the type of information to be displayed and the types of elements flowing in the model are specified.

For example, the simulation of the flow of service requests will allow us to determine how response times evolve. To do so we have to convert the flow diagram to an algorithmic model. The different resource types (resourceType), its characteristics (resource), the activities (activity), the meta-flows (rootFlow) in the system and the types of elements (elementType) that will flow through it are described in the model using XGHML.

The types of resources are level 1 agents (which directly handle the service requests) and level 2 agents (which authorize a petition or provide value services). The IR unit only has level 1 resources. The GM has level 1 and 2 resources at its disposal, while the GI only has level 2 resources.

Defining the resource types is rather simple. The resource need only be assigned a unique key and a brief description.

```
<!-- Resource types -->
<resourceType id="1">
  <description>Level 1 Agent</description>
</resourceType>
```

The resource type descriptions are defined with the resource label, which is used to specify the units to be generated, when each is available, how often this resource is generated and how long the turn lasts. In the following example, a level 1 agent is generated which works every day from 8 am until 4 pm.

```
<!-- Resource description -->
<!-- LEVEL 1 AGENTS -->
<!-- Day shift (8:00 - 16:00) -->
<resource id="1">
  <description>Level 1 Agent</description>
  <units>1</units>
  <timeTable>
    <rt_ref model_id="1" id="1"/>
    <cycle>
                <timeUnit>HOUR</timeUnit>
<!-- Time shift begins -->
      <startTs>8.0</startTs>
      <iterations>0</iterations>
      <period>
        <dist>FIXED</dist>
<!-- Each day -->
        <p1>24.0</p1>
        <p2>0.0</p2>
        <p3>0.0</p3>
      </period>
    </cycle>
          <dur  timeUnit="HOUR">8</dur>
<!-- 8-hour shifts -->
  </timeTable>
</resource>
```

Currently there is no module for representing the orchestrators for both the global architecture and the specific operational unit architecture. The priorities in this case have to be manually assigned to the activity descriptions. The activity label is used to describe the activities, which must have a unique identifier assigned. The label accepts a brief description of the activity and allows priorities to be assigned if several activities will be executed at once (not used in this project). A working group is then assigned in which the resources and the time necessary to complete the activities are specified.

In this case, an “Input Register” is described, and involves an in-person activity which must be executed by a level 1 agent and has a duration of 21 seconds.

```
<activity id="1">
<description>BasicData
Registration</description>
  <priority>0</priority>
  <presential>true</presential>
  <workGroup id="1">
    <description/>
    <role>
      <rt_ref model_id="1" id="1"/>
      <units>1</units>
    </role>
    <priority>0</priority>
    <duration>
      <dist>FIXED</dist>
      <p1>21</p1>
      <p2>0.0</p2>
      <p3>0.0</p3>
    </duration>
  </workGroup>
</activity>
```

The types of elements in this model are the different service requests that may appear depending on the associated procedures.

The element flows through the activities are described in the meta-flows, which are implemented using the tools provided by XGHML. A definition based on meta-flows allows the different operational units to be modeled.

Lastly, a group of listeners is implemented. These listeners indicate what measures are used for the resolution times for each service request, and the utilization times for each citizen and resource.

6.5. Results

The results of the simulation are tabulated in Table 1. Columns show scenarios with different relations between procedures resolved in-person and on-line. These quantitative measures are consistent with the qualitative results expected, as evidenced by the significant reduction in the resolution times in all the scenarios, along with a decreased effort on the part of the citizen, this decrease being particularly noticeable in the third scenario.

The advantage of this modeling environment is that it allows for a quantitative prediction involving different scenarios when a large quantity of procedures are implemented.

Table 1: Results of simulations

	In-person	Online	In-person	Online	In-person	Online
Scenarios	100,00%	0,00%	50,00%	50,00%	20,00%	80,00%
Average Time Procedure (days)	5,2	100,00%	2,63	50,64%	1,09	21,03%
Average Effort/procedure (min):						
Citizen	166,5	100,00%	96,25	57,81%	54,1	32,49%
Level 1 GM	8,7	100,00%	4,9	56,32%	2,62	30,11%
Level 2 GM	3,4	100,00%	3,05	89,71%	2,84	83,53%
Level 2 GI	13,2	100,00%	13,15	99,62%	13,12	99,39%
Level 1 IR	9,2	100,00%	5,1	55,43%	2,64	28,70%

7. CONCLUSIONS

This paper described the implementation of a modeling and simulation environment that allows for quantitative predictions of process improvements when an electronic method is introduced for a specific service. To test this enviroment only 50 procedures for a given service involving three operational units were implemented in this simulation (in the Government of Canary Islands there are about 5000 procedures). As all the procedures for all the services are introduced, a much more global measurement of the total implementation policy for eGovernment will become available.

The modeling environment used attempted to represent the architectures described in this paper. This was done by using the different modules that comprise the SIGHOS library and simulating specific functionalities, such as that of the orchestrator. To date we have not implemented a fixed environment that allows for a global configuration. Work is ongoing to introduce artificial intelligence techniques in the development of the orchestrators.

The implementation of all the procedures of a public agency, such as that associated with the Government of the Canary Islands, poses significant challenges. These challenges result from the large quantity of processes, possibly in excess of 5000, to be modeled. Since modeling the process associated with one procedure can require in the best of cases one full working day, modeling all the processes would be an imposing task indeed, hence the efforts toward making available an environment which can translate the application model directly into a process model. To this end, we are waiting for the Government of the Canary Islands to define or select a standard language for modeling its procedures so that an application code to process code converter can be developed.

ACKNOWLEDGMENTS

This work is being supported by a project (reference DPI2006-01803) from the Ministry of Science and Technology with FEDER funds.

REFERENCES

Aguilar, R. M., Muñoz, R., Castilla, I., Martín, C. A., Muñoz, V., 2006. An interface to integrate workflow diagrams into discrete event simulation. *International Mediterranean Modeling Multiconference*. Barcelona-España..

Hansen, M. H.; Netjes, M., 2006. Business process simulation - a tool survey. *Seventh Workshop and Tutorial on the Practical Use of Coloured Petri Nets and the CPN Tools,* 579, 77-96.

Moreno, L., Aguilar, R.M., Piñeiro, J.D., Estévez, J.I., Sigut, J.F., González, C., 2001. Using KADS methodology in a simulation assisted knowledge based system: application to hospital management, *Expert System with Applications*, vol. 20, 235-249.

SIGHOS Home Page, 2007. Available from http://sourceforge.net/sighos

AUTHORS BIOGRAPHY

PEDRO JUAN BAQUERO PÉREZ, is a Telecommunication Engineer from the Polytechnic University of Madrid. He is an associate professor at University of La Laguna where his research is focused on the modelling and simulation of IT organizations Also, currently he is Head of the Unit for Information Technology of the Ministry of Presidency and Justice of the Canary Island Regional Government. He has worked in different areas: as a freelance consultant in information technology, as an researcher (Interuniversitary Microelectronic Center (Belgium) and Telefónica I+D), as a university professor (University of Carlos III of Madrid), as a civil servant (telecommunication specialist in the Spanish State Telecommunications Office and Head of the Telecommunications Unit in the Government of Canary Islands), and as a managing director (Head of Strategic Planning Area in a cable telecommunications operator and Head of Costumer Engineering Area in a IT company).

YURENA GARCÍA-HEVIA MENDIZÁBAL is a Technical Computing Systems Engineer from the University of La Laguna. She is currently a student of Engineering Computer Science and a research grant holder at the Department of Systems Engineering and Automation at the University of La Laguna. Her fields of interest are the design and implementation of simulation environments for service organizations, modeling and simulation, and business intelligence.

ROSA M. AGUILAR received her MS degree in Computer Science in 1993 from the University of Las Palmas de Gran Canaria and her PhD degree in Computer Science in 1998 from the University of La Laguna. She is an associate professor in the Department of Systems Engineering and Automation at the University of La Laguna, Canary Islands, Spain. Her current research interests are decision making based on discrete event simulation systems and knowledge-based systems, intelligent agents, and intelligent tutorial systems.