

Understanding the Dynamics of Process Models: A Unified Approach

Entendiendo la Dinámica de los Modelos de Proceso: Un Enfoque Unificado

Ana I. Martínez-García¹ and Brian C. Warboys²

¹Ciencias de la Computación, CICESF, Ensenada B.C., México

²Department of Computer Science, University of Manchester
Manchester, UK

e-mail : martinea@cicese.mx, brian@cs.man.ac.uk

Article received on May 16, 2000; accepted on June 25, 2001

Abstract

In this work, a technique to integrate the relevant aspects of process modeling and a simulation methodology is introduced providing with this a more complete guide for the study of organizational processes. Our approach is reinforced with the description of its steps and the development of a support tool. This tool allows the automatic mapping of process models, captured using Role Activity Diagrams (RADs), to the input to a Discrete Event Simulator, trying to avoid the extra effort in building the simulation models from scratch, and allowing the evaluation of proposed process model changes to be influenced by the results of a simulation study. The approach is illustrated with the software process example ISPW-6

Key Words: Process modeling, simulation, software process.

Resumen

En este trabajo se introduce una técnica que integra los aspectos relevantes del modelado de procesos con una metodología de simulación, con lo cual se provee una guía mas completa para el estudio de procesos organizacionales. Nuestro enfoque es reforzado con la descripción de sus pasos y del desarrollo de una herramienta de soporte. Esta herramienta permite el mapeo automático de modelos de procesos, capturados con Diagramas Rol Actividad (RADs), a la entrada de un Simulador de Eventos Discretos con esto tratando de evitar el esfuerzo extra que se lleva a cabo en la construcción de los modelos de simulación desde nada y permitiendo con esto que los resultados del estudio de simulación influyeran la evaluación de los cambios propuestos a los modelos de procesos. El enfoque se ilustra con el ejemplo del proceso de desarrollo de software ISPW-6.

Palabras Clave: Modelado de procesos, simulación, proceso de software.

1 Introduction

Nowadays managers are making critical technical and business decisions to improve their organizational processes. These decisions are based on their understanding of the processes in the organizations but assume that these processes are efficient and well established and that they are, in practice, being executed. However this decision making can be unreliable if these assumptions are not correct.

Process models are used to document and support procedures in organizations in a consistent and uniform manner. Such models aim both to support the analysis of an organization's processes and to aid in the detection of fundamental problems in achieving the aims of the process. A good process model aims to address the three important aspects in processes: Information Technology (IT) support of the process, social issues such as staff training, culture, etc, and the process itself (Kawalek, 1997; Warboys et al, 1999). Here a process is considered as a group of related tasks, which are performed by people and an IT system interacting together to achieve the goals of an organization. The purpose of process models is to aid the task of addressing the problems found in one or more of these aspects in organizational processes. This could be through support for both making simple changes or some radical redesign of the processes (the two basic forms of process improvement). Unfortunately, the models currently available need to address specific dynamic properties (the behavioral aspects, when things are done) if they are going to be useful in helping managers to predict and understand the implications of change and therefore aid in the making of better decisions.

Hammer and Champy (1994) describe how over 50 % of the companies involved in re-engineering projects (in-

novation of organizational processes) fail to provide the benefit expected. One of the main reasons for these failures, given by several publications (Paul et al, 1998; and its references), is the lack of tools to evaluate the (re-designed) process performance before its implementation. Otherwise, any mistakes arising from the re-engineering will only appear once the redesigned processes are implemented, making them very costly and difficult to correct. Thus, it is necessary to understand and assess how the process will behave when changes have been made before actually implementing them.

An integrated methodology or technique for the analysis and improvement of processes should therefore include a phase to address the advantages or disadvantages of a new design or improved process. A simulation study is thus a key element that has to be considered.

Simulation techniques are being widely used to address many different problems in such diverse areas as science, engineering, economics, etc. Simulation is in practice the only means of studying situations where it is dangerous, risky or expensive to experiment with the real system. It is also used to simulate systems that do not yet exist, to check if it is feasible to build them or to try to improve their design before they are built.

Simulation can also be considered as a tool for understanding the impact of change in organizations. It enables a more dynamic approach to the study of organizational processes. It can be used to model a current, a redesigned or a not yet existing (for process design) process. In this manner the behavior of the process can be predicted and analyzed. Simulation of such complex systems is a way of promoting the understanding of current processes, and of any proposed changes to improve their performance. Because of its usefulness, simulation is usually considered as an integral part of the decision making process (Tumay, 1996), assisting in the prediction of the behavior of these processes by investigating "what if" questions. Thereby facilitating the understanding of possible outcomes produced by change and verifying the implementation of the system with a simulation model.

Process modeling is growing in importance as an application area for simulation, in particular in the evaluation of the design and redesign of processes (process improvement). Even though simulation alone could be used to capture a process and perform a redesign analysis, a more detailed analysis of different aspects of an organizational process has to be carried out initially. Features such as what activities are being performed (functional view - activities well defined), when and how are being developed (behavioral view - rules of the process), where and whom in the organization are executing them (organiza-

tional view - responsibilities), and the entities produced and/or manipulated by the process, their structure and relations (informational view) (Curtis et al, 1992) have to be analyzed to provide a complete process analysis. Some information corresponding to the views is provided by the process model (activities, responsibilities, rules, some entities, etc.) and are not given by the simulation model (responsibilities, interactions between agents, etc.). On the other hand with the simulation model some details such as the rules (times) and entities are very well defined. Therefore, the information provided by both models complements each other providing the details needed for a good analysis of the process.

There is a clear increase in the application of simulation to the analysis and design of organizational processes and "we will continue to see the development of specialized tools that are easy to use and focused on this application domain" (Pegden, 1997).

Process modeling can be used to analyzed static aspects of the process such as: duplication of activities and documents, interactions between agents (communication and coordination problems), responsibilities not defined, analysis of the IT which gives support to the process, etc. On the other hand simulation can be used to address questions such as: What is the total process cycle time?, How long do customers have to wait before being served?, What is the best way to schedule personnel?, Bottlenecks analysis (location and timing for processing them), etc., (Gladwin and Tumay, 1994).

Thus, the application of simulation to organizational processes is growing. However, although simulation software has been improved by simplifying the modeling process, it is still perceived as a complicated technology by most managers and engineers (Harrel, 1996) and its potential has not yet been widely recognized in organizations (Hlupic, 1998). A Benchmarking study performed by ProSci (1997)(Hlupic, 1998), revealed from the analysis of 60 large international organizations, who went through process re-engineering, that less than 10 % of the companies used any form of simulation tools. Some possible explanations of why simulation is not being widely used as a tool for simulating process models are (Gladwin and Tumay, 1994; Lingineni et al, 1995): a) There are few capabilities for interfacing or connecting simulation software and other process modeling tools, b) The required expertise necessary to build models with most simulation languages or software is very scarce, c) There is little support for analysis and the model design task, and d) The need to use both, the domain expert and the simulation analyst in the modeling process.

Over the past few years, new tools have been developed

for modeling organizational processes but mostly without addressing, specifically, their dynamic aspects. Most of the tools capture the description of the process with a graphical representation (boxes, arrows, lines, etc.), which shows the activities and goals of the organizational process. Many of these tools provide some kind of analysis of the process depending on the technique (RADs, IDEF, Workflow, etc.) used for the process representation. Most process modeling tools provide the following characteristics. a) *Process capture*, b) *On-screen navigation of complex process*, c) *Multiple views of the process*, d) *Flexible flow models* (Bruce Silver Associates, 1997). These features and the fact that most of these kinds of tools are easy to use and comprehensible, have made them proliferate.

However, few of them have been integrated with any form of simulation software (Kettinger et al, 1997; Lingineni et al, 1995). As a consequence, if an integrated analysis of the organizational processes is desired, a double effort is needed in the modeling process. First, the process has to be modeled in a process modeling tool. Then the analyst or decision maker has to start all over again from scratch and model the process using simulation software, even though most of the process definition has already been captured with the process modeling technology.

To date, little has been done in exploring the relation, both conceptual and technical (technique and tool), between process modeling and simulation to facilitate the analysis, understanding and improvement of processes. The purpose of this work is to explore this issue. We are interested in providing a more integral approach to the analysis of processes. That is, an approach which considers both the static and dynamic aspects of a process.

Our main aim is to provide an approach to a more integrated study of processes by means of unifying the process modeling and simulation approaches. We need to be able to extend the understanding of processes, adding to the usefulness of process models and simplifying the creation of simulation models.

2 Motivation for our approach

Process modeling can be used to study different aspects of a process such as how many discrete processes exist, how they relate to each other, how each process contributes to the overall process objectives, who are the processes owners, etc. (Manganelly and Klein, 1996). It is also useful as a means for communication and for the establishment of the process baseline. It helps the documenting of processes and in finding inconsistencies in

them. Process modeling techniques have frequently been used to analyze, improve and change (re-engineer) organizational processes (Kawalek, 1997; Wastell et al, 1994). However there is a strong need to enhance the analysis of process models by introducing a technique such as simulation as an important part in the analysis and improvement of processes. As Kettinger et al (1997) point out there is a high level of participation of non technical staff in the process change process (capture, analysis, improvement, redesign of processes and support), therefore there is also the need for more user-friendly and better integrated process capture and simulation tools. These packages should be designed to allow team members easy participation in process modeling and improvement, supported by the visualization of the process by means of simulation and animation. As a first approach to this kind of analysis, where process modeling and simulation are involved, the study could be performed using independent tools for each aspect. Nevertheless it would be preferable if the two types of techniques are interfaced together in a tool facilitating, with this, the generation of the simulation models from the process models.

Simulation is a technique that provides different features in support of a dynamic analysis of processes. Hence, this is a technique that facilitates the decision making process by carrying out different "what if" types of analysis, a tool for decision making in the management of processes and in facilitating their analysis.

A workshop was held to present the results of both process modeling and simulating of a case study in the UK health (Martinez, 1999). This workshop asked the health practitioners to compare separate Process and Simulation Models. The overall comments were (Kay et al, 1998): "Most felt that process modeling was effective for communicating and that the documented process (model) can be accessible to more people", "Most found simulation easier to understand because of its Visual representation (process animation) and data collection results (more intuitive)", "Most felt that both techniques complement each other", "Simulation gives a good overview of things happening simultaneously", "Process models are clear and give extra detail", "Simplicity plus detail and focus". Thus, once again the need for an approach and a support tool which provides both, the process modeling activity and simulation modeling was clearly evident.

Hence, this motivated to further our research into the integration of the two approaches to the study of organizational processes. We took the main steps of a process modeling study, and complement it in the phase of analysis with the key steps from a simulation study.

3 Steps in a Simulation Study

In order to perform a correct simulation study it is necessary to follow a series of steps, the basics of which are the development of a model, implementation, validation, verification and experimentation with the model (Pollacia,1989).

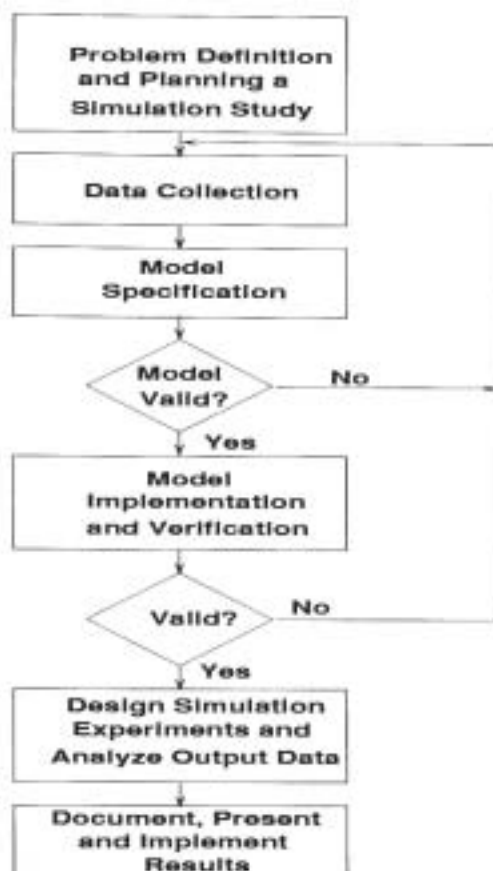


Figure 1: A guide for a simulation study.

A guide of the elements, that have to be considered in a simulation study given by Law and Kelton (1991) are illustrated in Figure 1 and described next:

1. *Problem definition and planning of the simulation study.* The problem must be well defined, including a description of the system to be modeled and its boundaries.
2. *Data collection.* In order to determine the values and probability distributions for the system parameters, real data must be collected from the system under study.

3. *Model specification.* In the model definition, it is necessary to represent and describe the special features of the system (objects, their attributes, the activities of these objects and the possible states of the system).
4. *Model validation.* This step is the process of making sure that the model represents the real system to a certain degree of accuracy and it can imitate its behavior reasonably. It must be done during the entire simulation study.
5. *Model implementation and verification.* The language or tool to implement the model in a computer program is selected by the simulation modeler. Then the abstract model is implemented in this language or tool.
6. *Design of simulation experiments and analysis of output data.* Some experiments must be designed to observe the model, changing the initial values, model parameters, inputs to the model, state variables, etc.
7. *Document, present and implement results.* It is important to document the assumptions that were made in the model, as well as in the computer program. Reports on the simulation and recommendations on the results of this must be documented.

Simulation modeling is fundamentally an iterative process, in a simulation study first we model, experiment, run the simulation and observe the results, then we go back to make the necessary adjustments or corrections in our model according to the results of our simulations. Thus, if these steps (simulation study) are taken and used together with a Process Modeling methodology in the process analysis step then a more integrated view of the process can be obtained.

4 Unified Technique for Process Analysis and Improvement (UTPAI)

The Unified Technique for Process Analysis and Improvement (UTPAI), consists of a combination of some process modeling and key simulation steps. The idea is to take further the static analysis provided by process modeling techniques and combine it with the dynamic features introduced by simulation. In this particular case using Discrete Event Simulation (DES).

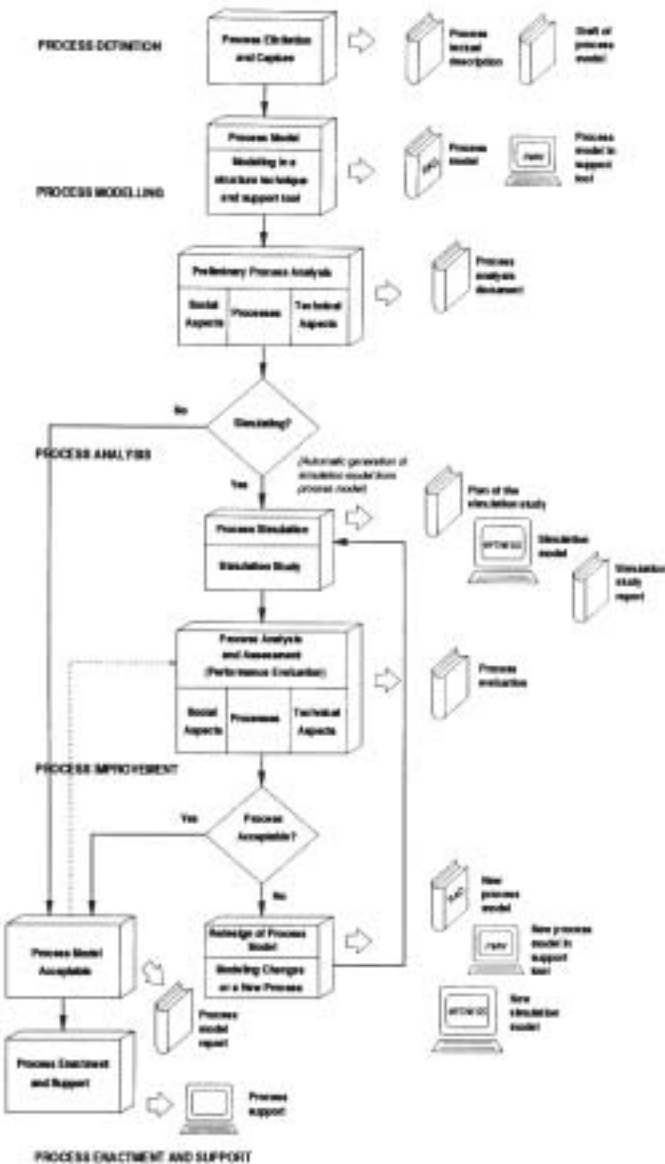


Figure 2: Diagrammatic representation of the Unified Technique for Process Analysis and Improvement (UTPAI). The diagram illustrates the the general and particular steps followed with this technique. General steps such as process definition, modeling, analysis, improvement, enactment and support are represented. More particular steps such as process elicitation and capture, process modeling, preliminary process analysis, process simulation, etc. are detailed. It also shows the corresponding outputs in each step, such as documentation (reports) and implementation of models in a computer system.

The main steps of UTPAI together with the corresponding documentation generated in each phase are illustrated in Figure 2.

The steps are:

- Process Definition.** This step involves the process elicitation and capture. Here the objectives of the process should be established together with the identification of process owners and the interactions with other external processes. The first draft version of a process model (without too much detail) is captured in a process modeling methodology.
- Process Model(ing).** Once the process has been defined. The detailed modeling of the process should be completed. This should include the model using a structure techniques (e.g. RAD, RAD models processes in terms of Roles, the interactions between them and their activities) and their corresponding support tool.
- Process Analysis.** The process analysis involves several sub steps. First a static analysis of the process model is performed. This is an analysis of the processes and, the social and technological aspects using methods or techniques from the Process Modeling area. Once this analysis has been performed, decisions can be made as to whether the process needs to be modified. Based on this decision a more dynamic kind of analysis might be performed. This dynamic analysis is performed by means of a simulation study (Section 3) of the process model. First, a simulation study of the “AS IS” process model is performed. Then, simulation of different scenarios of possible changes to the process “TO BE” (improvements). Finally, comparing the results in order to evaluate the performance of the process model. Thus, experimenting with different conditions and process definitions until a suitable process model is found. In Section 3 a possible approach to a simulation study is introduced. These steps could be used and adapted in this part of the technique. Although this approach is not bound to any specific modeling techniques or tools, the ideal would be to use a combined (process modeling and simulation) tool. A tool which facilitates the construction of the simulation model from a process model would eliminate the double effort made by constructing the simulation models from scratch. Thus, the simulation study introduced in Section 3 could still be used by skipping some of the steps such as model specification.
- Process Improvement.** Based on the previous analysis the process model could go through some

small or radical changes, a redesign. The process model changes also go through the analysis step. This is to determine if the changes have really improved the process performance before proceeding to the following step, the enactment and support.

- **Process Enactment and Support.** When the improvement of the process model is satisfactory, then the enactment and support to the (new or re-designed) process can follow.

From Figure 2 it can be observed that the analysis and improvement of a process is an iterative process. The analysis and simulation has to be performed after any change to the process model in order to ensure that there really is an improvement in the real process. The outputs that result from each process step in UTPAI (Figure 2) are also specified. These outputs are documentation and implementations of some kind. They are briefly described next.

• Process Definition

- *Process textual description.* A description of the process in textual form. This might include specific forms and some other kind of documentation used in the process.
- *Draft of the process model.* The documentation of the first draft of the capture of the process using a given process modeling methodology.

• Process Model(ling)

- *Process model.* Final version of the process model. The process model document should include all the diagrams from the different representations used in the process modeling methodology, including a diagrammatic representation in structured techniques (e.g. RAD (Ould, 1995), ERADes - Extended Role Activity diagrams for Discrete event simulation to (Martinez, 1999)), and the documents and forms used in the process.
- *Process model in support tool.* The process model captured in a support tool for the specific techniques used previously.

• Process Analysis

- *Process analysis document.* A complete and detailed report of the process analysis from the point of view of processes and their social and technological aspects. This will give the relevant information to decide whether the process model needs further analysis to determine if its performance is still satisfactory.

- *Plan of the simulation study.* This document should specify the dynamic aspects (metrics) of the process to be analyzed by means of the study. Some of the metrics that can be analyzed about the transformation of entities are (Manganelly and Klein, 1996): transit time (from one activity to another), waiting time (waiting to be processed), service times etc.; Other type of metrics can be related to the human and technical resources needed for an efficient and effective process, etc. Also, if the study is only concerned with particular parts of the process, this should be established in the plan.

- *Simulation model.* After it has been established from the process analysis report that the process needs further analysis and the simulation study plan has been defined, the simulation model should be generated to perform a study of the different dynamic aspects of the process model such as the timing of activities, resources, etc.

- *Simulation study report.* A report of the dynamic study of the process model. The aim of the study is to verify the process model performance and compare the 'as is' and 'to be' processes. It is in this study that a series of "what if" questions are carried out.

- *Process evaluation.* This is a complete report of the process model performance in general terms. This is taking into account the first kind of analysis (static) together with the simulation study to decide if the process needs changes or if it is already performing satisfactory.

• Process Improvement

- *New process model.* If according to the process analysis it is decided that the process performance is no longer satisfactory, the process model is modified. The changes in the process can be simple or radical. These changes to the model, or the new process model, should be documented and specified using the process modeling methodology used previously to represent the original process.

- *New process model in support tool.* The new process model should be captured in the corresponding support tool.

• Process Enactment and Support

- *Detailed process model report.* Before the process enactment, a report documenting all the

details about the process should be well established together with proposals for process support with IT and the social aspect.

- *Process enactment and support.* The implementation of the support for the process and/or enactment of it.

etc) relate between them and the feasibility of a support tool that connects both.

5 A possible supporting tool

In order to support our approach we developed a tool (Figure 4). Our prototype integrates an existing process modeling tool with a suitable simulation package. In this way, we provide not only a guide but also the elements for a technology which supports it, leading to a position where a more complete and accurate approach to the analysis and management of processes is possible. The approach used to transform Role Activity Diagrams (RADs) (Holt 1983; Ould, 1995) to Discrete Event Simulation (Law and Kelton, 1991) is generic enough that it can be used in a different context with different simulation tools. RAD is a structured technique for modeling processes (its notation is illustrated in Figure 3). RADs focuses the process representation on the concepts of roles, their interactions and their constituents activities, as well as the connection with external events (Ould, 1995). This technique allows for a comprehensive approach to representing most of the features of a process (goals, decisions, etc.) and its notation and concepts are easy to understand with two minutes of instruction, therefore facilitating the communication and understanding of the process to the people involved in it (Miers, 1994). With other techniques one can capture information such as inputs and outputs of the process, mechanisms and controls applied when performing the activities (IDEF0 (Hunt, 1996)) but cannot capture roles, responsibilities, interactions and other aspects relevant for an analysis of the process. The rationale for the use of the RAD notation are the facts just stated and its ability to map to different paradigms without undue difficulty. The RAD models can be easily enacted (implemented) by mapping its elements to the programming language PML (Process Management Language (Warboys, 1989)). In this work the mapping to DES will be explored and developed. A few efforts have been made at interfacing process modeling and simulation in a tool, however they use their own organizational modeling techniques (Cory, 1996) that are not easy to understand or others such as IDEF (Lingini et al, 1995) that do not provide all the information needed in the static process analysis and are not easy to grasp by non technical staff.

In this tool (Martinez and Warboys, 1998), it is assumed that process models have been captured in RADs. From them, the DES models are generated for the WITNESS simulation package (AT&T Intel, 1994; Thompson, 1995) which supports modeling in terms of parts (entities),

ROLE ACTIVITY DIAGRAMS NOTATION

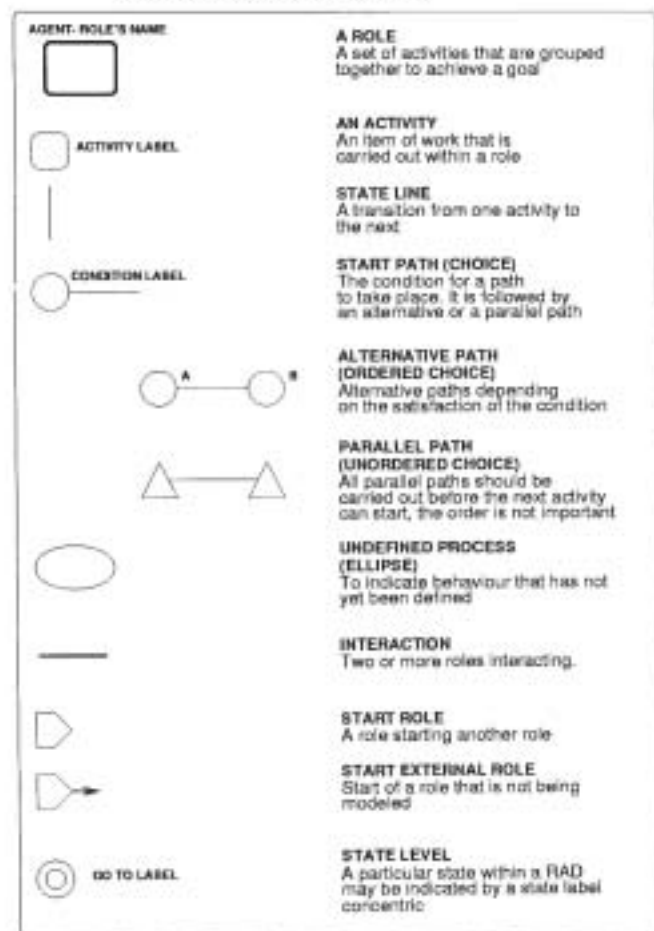


Figure 3: Elements of the Role Activity Diagrams and a brief description of them.

UTPAI introduces simulation for the study of the dynamic aspects of a process in the step of **Process Analysis**. Here the experimentation, statistical analysis (for the initial configuration and final analysis), and outputs will depend on the process being analyzed.

With this technique, according to the type of process study being performed, there can be more emphasis in one or more of the specified steps.

To support UTPAI, we describe how the two conceptual domains (roles, activities, interactions, entities, servers,

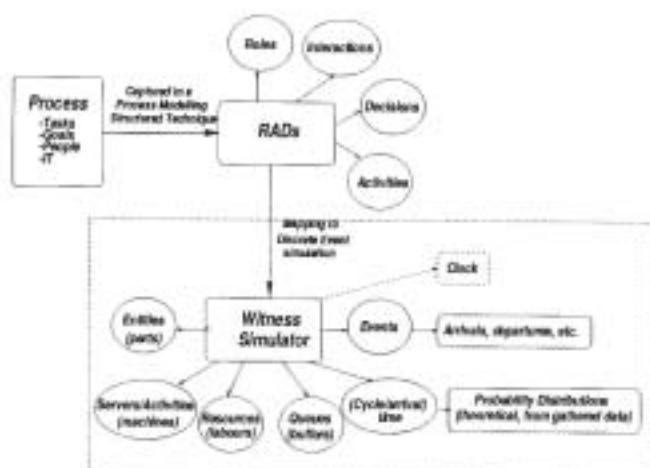


Figure 4: Processes, Role Activity Diagrams and Discrete Event Simulation Models in terms of their main concepts. From an existing process (tasks, goals, people, IT) a model is captured in RADs (roles, interactions, decisions, activities) and then mapped to its corresponding simulation model (entities, servers, resources, queues, etc.).

machines (servers), labours (resources), buffers (queues), etc. In particular, the basic process information is captured using the Process Modeller Workcentre (PMW, developed by the Informatics Process Group of the University of Manchester) in terms of RADs. The first mapping from its elements (activities, roles, decisions, etc.) to the WITNESS simulation elements (machines, buffers, parts, etc.) is performed (Figure 4). Because RADs are not time-based, the technique does not identify the dynamic properties or attributes of the activities that are needed for the simulation. Thus, missing information (some simulation attributes such as cycle times, arrival times, number of instances for a particular element, etc.) for the experimentation is entered directly via the user interface at this stage. However an extension to RADs that contemplates attributes needed for DES (ERADes) is under development and could be used to gather all the information needed from the beginning (Martinez, 1999). With this information the mapping and the final simulation model can be generated in the notation of the WITNESS Input Command Language. The model is generated to maintain consistency with the way in which the process was captured in RADs. The model is automatically loaded into the WITNESS environment and then basic simulation (and animation) of the process can be performed.

The prototype translates a static process model into a dynamic simulation model that can be used for calculating

RAD mapping to WITNESS

RAD element	WITNESS element	Terminology
ROLES	1 PART	Generic Simulation → WITNESS
		Entity → Part
		Queue → Buffer
		Server
		Activity → Machine
		Resource → Labour
ACTIVITY	1 BUFFER	
(when it interacts with another role and the activities continue in the two roles)	1 Production MACHINE	
(when there is an interaction coming from another Role and also a link from a previous activity)	1 Assembly MACHINE	
(in any other condition)	1 Single MACHINE	
THE START OF ALTERNATIVE OR PARALLEL PATHS (CHOICE)	1 BUFFER	
(when followed by alternative paths)	1 Single MACHINE	
(when followed by parallel paths)	1 Production MACHINE	
THE END OF PATHS (END CHOICE)		
(when terminating alternative paths)	1 BUFFER	
	1 Single MACHINE	
(when terminating parallel paths)	N BUFFERS	
	1 Assembly MACHINE	
ALTERNATIVE PATH (ORDE ED CHOICE)	The same as activities	
PARALLEL PATH (UNORDERED CHOICE)	The same as activities	
START ROLE	The same as activities	
START EXTERNAL ROLE	The same as activities	

Figure 5: Mapping from Role Activity Diagrams to the elements of the Discrete Event simulation package WITNESS.

service times, analyzing bottlenecks, etc. Hence the user can then plan the simulation analysis that is required in a particular process context.

The overall structure of the system developed has the following elements: 1. Basic Mapping. A first mapping of the existing attributes in the RAD model to the WITNESS simulator code. 2. User Interface. The different models do not map directly from one to the other (recall that there is a lack of timing related information on the RAD model) and the extra attributes needed for the simulation model need explicit capture. The missing information in the RAD model is directly input so that there is a more complete WITNESS model at the time of the second mapping. 3. Complete Mapping. After adding the missing information we have a complete mapping and hence a complete model. 4. Automatic WITNESS Code Generator. This is the final system.

From the analysis of RADs and DES main concepts, a basic mapping between the elements of both models was established. Essentially the elements of the RAD

model: roles, activities, decisions and interactions, were mapped to entities (parts), resources (labours), servers (machines), and queues (buffers) in the simulation model (Figure 5).

5.1 Evaluation

The evaluation of the prototype was performed initially, with simple models to check the consistency of the mapping and the simulation model. Then further evaluation was made using more complicated models such as the UK health sector case study (Martnez and Warboys, 1998) and the ISPW6 example (Kellner et al., 1990).

The approach adopted here is illustrated using the example process developed at the 6th International Software Process Workshop (ISPW-6). The process is initially represented in RADs, which reflect the real working process of the development of software in an organization. Then from the RADs, a DES model is automatically generated to perform "what if" experiments and the analysis of some metrics for the planning, understanding, control and operational management of the process during a portion of the life cycle and the development project. Some typical kinds of analysis that can be performed using the simulation model are cycle times, staffing requirements, rework effects, coordination (task dependency) of subprocesses and bottlenecks (queue lengths) in particular areas of the process.

The prototype performed an adequate mapping, which has one important feature in this prototype, this is the fact that the simulation model retains the same structure that was established in the process model (Figures 7 and 8). This avoids making the model more complicated by having to make abstractions of a different kind and clearly facilitates the understanding of the simulation model by those already familiar with the process model as represented in RADs. Thus, the modeler can concentrate on the experimentation and the facilitating of the communication process. We have constructed a simulation model oriented to roles.

This prototype does not automate the experimentation process. This is very specific to the application domain and the process owner. The manager or process owner needs to plan a proper simulation study according to the specific domain and activities for investigation (i.e. for process improvement or redesign) and following the UT-PAI steps.

The mapped simulation model reproduces the behavior of the process initially model captured using RADs. When running, the simulation advances through the activities

(events) in a compressed time, animating the RAD process model. This gives a general overview of the important aspects of the process which are occurring simultaneously. In contrast with the static RAD process model, it is possible to view multiple instances of the process acting together, and thus allowing simulation studies, which will reflect more closely the real process, to be performed.

The metrics that can be investigated with the simulation model are: cycle time, staffing requirements, rework effects, waiting times, bottlenecks, and coordination of subprocesses. Some experiments were performed with the ISPW-6 process example, in order to investigate how easy these metrics could be studied given a generated simulation model.

5.1.1 The Example Process ISPW-6

The example process is of a nominally generic process commonly found in organizations concerned with the development and maintenance of software systems (Bruynooghe et al., 1994). The focus is on the activities associated with changes to particular units or modules which compose a single software system. Information related to the participants' skill levels is not provided. The main focus is on the process itself and it is fairly well defined in a textual narrative form. It describes each task to be performed together with their inputs, outputs, responsibilities and constraints.

The roles and interactions of the process were captured in RADs(A brief description of the RADs' main elements was given in Figure 3) The role interactions diagram in Figure 6 illustrates the complexity of the process. In particular it depicts the ISPW-6 core problem process, which is a relatively restricted part of the software change process (Kellner et al., 1990).

The RADs of the ISPW-6 process example are shown in Figure 7 and a brief description of the role activities follows:

- **Schedule and Assign Tasks.** This is the first step carried out in this process. It involves creating a schedule for a software change and assigning the tasks to specific members of the staff. This schedules the Modify Design and Modify Test Plans tasks.
- **Modify Design.** This involves the modification of the design for a particular code unit as stated by the requirements for the change. This can also be modified as defined by the feedback from the review of the design.

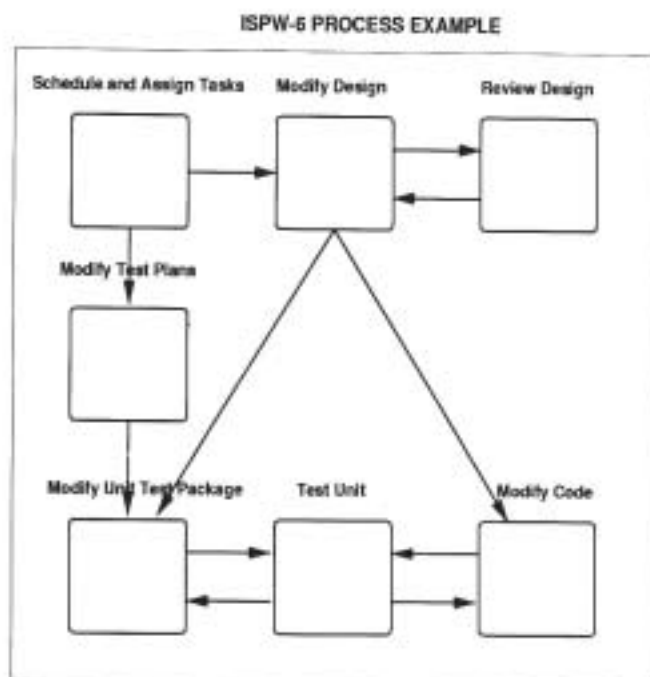


Figure 6: High level Role Interaction Diagram of the software process example ISPW-6.

- **Review Design.** Here the formal review of the modified design is carried out. The review outcome can be: unconditional approval, minor changes or major changes recommended.
- **Modify Code.** This involves the implementation of the design changes in the current code together with the compilation of the modified source code into object code. This can also be modified through to the feedback from testing (if additional modifications are required).
- **Modify Test Plans.** In order to include the testing of the capabilities given by the change of requirements. Modification of the test plans and objectives are carried out.
- **Modify Unit Test Package.** Here the modification of the current unit test package for the affected code unit is carried out according to the new test plans and objectives. This can also be modified through to the feedback from testing (if additional modifications are required).
- **Test Unit.** This involves the application of the unit test package to the modified code and the analysis of the results. If the source code needs more modifications, the unit tests also need to be further modified. Once the modifications have been carried out,

	Time (Units)	Resource Allocation (Labours)
Modify Design	1	3
Review Design	1/4	2
Modify Code	1	3
Test Unit	1/4	2
Modify Unit Test Package	1/2	2
Modify Test Plans	1/4	2
Schedule and Assign Tasks	1/8	1

Table 1: The initial values assumed for the start of the simulation of the ISPW-6 process example. The units of time correspond to each one of the subprocesses and are kept fixed during the simulation. The number of human resources are then changed in order to perform some experiments with the simulation model.

the unit can be re-tested. If the unit passes its tests with an acceptable level, then the example process has been completed.

When interacting with people to develop a model of the process, the staff involved in its execution can identify the activities that have to be performed. Thus, the process can be captured in terms of these activities, the roles and the interactions carried out. However when simple data related to the timing of the process is required, it is not as easy. In order to have a reliable and real simulation model, the data should reflect the actual process as closely as possible. The simulation models were generated with reasonable approximate times. Thus, the analysis of the experiments was not exhaustive in statistical terms. However a global analysis of the sample processes with respect to their critical activities was possible.

Our assumption for an initial process configuration, timing and resource allocation for the simulation, for each one of the roles involved in the process is illustrated in table 1. This table shows the total units of time for each subprocess, in this case the "Modify Design" and "Modify Code" take the longest, 1 unit of time and also these roles are being executed by the bigger number of human resources, 3 labours. The values on table 1 are used to perform the based simulation of the process and then some experiments changing the number of human resources are performed. The experiment concentrated on the analysis of bottlenecks due to the effect of rework (such as in the reviews of the design, the test unit package, etc.), the lack of coordination in some interactions

ISPW-6 Process Example

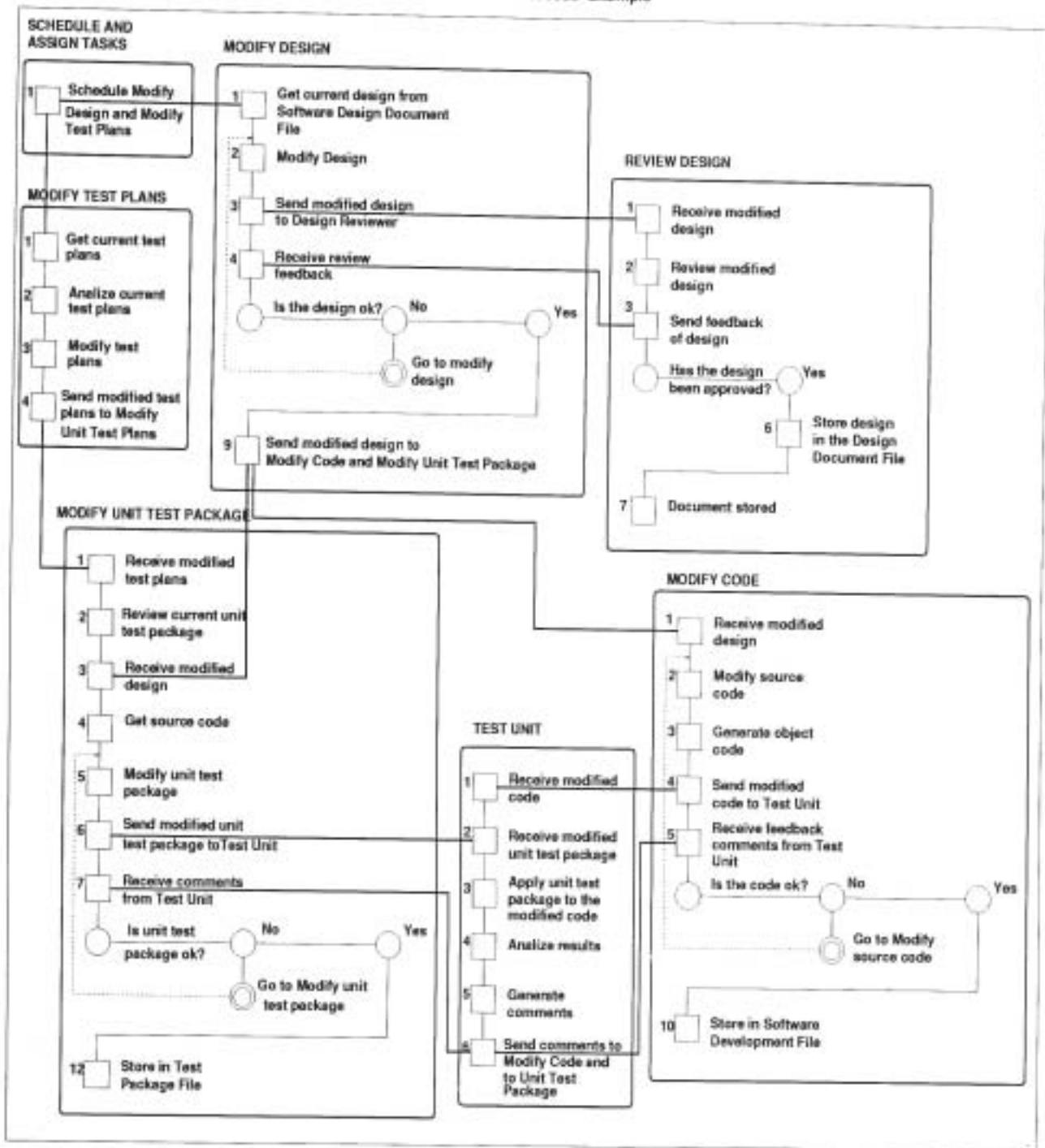


Figure 7: Role Activity Diagrams of the ISPW6 process example, showing its roles, interactions, decisions and main activities.

and the finite number of resources available to perform the process activities. In particular we can observe the queuing effects in the "Modify Design" role, activity 2 (time series graph in Figure 8) due to the rework originated from the interaction with the "Review Design" role. As the number of resources allocated to the "Modify Design" subprocess and particularly to activity 2 is increased, the queue decreases significantly. This same effect occurs when the review policy of acceptance/rejection of designs is changed so that more designs are accepted without corrections. The starting policy is one where the number of designs accepted with minor corrections is bigger than the number of rejected or accepted designs without corrections (this last being the smallest of the three).

An interesting point is the coordination of activities within subprocesses. In this case two activities where bottlenecks are formed because of the lack of coordination between subprocesses are, "Modify Unit Test Package", activity 3 and "Test Unit", activity 2 (time series graph in Figure 8). These two are activities which in order to continue, need two different documents arriving from two different subprocesses. It is evident that one of the two subprocesses feeding each one of these activities is taking longer to be performed their work than the other. In the case of "Modify Unit Test Package", activity 3 the process causing the bottleneck is the interaction coming from the "Modify Design" role. That is, the reception of the modified design. Since the modified test plans arrive first, there is a queue of them waiting for the corresponding design. The "Modify Code" role is also fed by "Modify Design", however the previous effect does not occur as the only document needed in "Modify Code" is the modified design.

The "Test Unit" role interacts with both the "Modify Code" and "Modify Unit Test Package". The experiment carried out with the initial configuration shows a bottleneck caused by the interaction coming from the "Modify Code" role (the reception of the modified code, activity 2). This is due to the fact that the "Modify Code" subprocess takes longer to perform than the "Modify Unit Test Package". As the number of resources (labours) for the "Modify Code" role is increased the queuing effect disappears. Another point highlighted by the simulation is the effect that the "Modify Design" subprocess has on the complete process. Experiments were made increasing the level of resources (labours) assigned to this particular role and the results were very interesting. The labours were increase up to 5, but changing the assignation of them so that there was always one or more resources free to perform other activities when activity 2 of this role was being executed. This, in general, made the process more

efficient since, as the number of resources performing the "Modify Design" role was increased the overall performance of the process increased. However something very peculiar happened in the "Test Unit" role. In the case when there is a free resource (labour) to perform the rest of the activities and three labours assigned to the main activity (modify design, activity 2), then as the simulation established, the queuing effect in activity 2 changes. The queue that was caused by the interaction coming from "Modify Code" disappears and a queue forms as a consequence of the interaction coming from the "Modify Unit Test Package" (the reception of the modified test package).

The resource utilization was consistent in the experiments, the resources assigned to the "Modify Design" role were the busier, followed by the "Modify Code", "Modify Unit Test Package", "Review Design", "Test Unit", "Schedule and Assign Tasks" and the lastly the "Modify Test Plans". With this kind of analysis important areas of the process which are candidates for redesign and improvement can be deduced.

5.2 Overview of the tool

Even though both approaches, process modeling and simulation, are concerned with similar issues, the type of information that is gathered for the analysis and representation of their models is quite different. RADs do not capture some of the basic information needed for a statistical analysis of a process because they are not time based. On the other hand the simulation model does not contain information concerning goals, responsibilities, interactions, etc that should be analyzed to perform process improvement. One advantage of combining process capture and simulation, is to increase the probability of the automatic generation of a simulation model from a process model whilst retaining the structure of the initial process model. This provides the capability for initially generating a RAD model of an abstract view of the process and subsequently adding refinement of the model as more information about the process is being defined or obtained (Bruce Silver Associates, 1997). Thus introducing more, but appropriate, complexity to the process model.

Once this detail hierarchy has been obtained it can be used to examine a large model in terms of specific simulations of behavior at chosen parts of the process. The RAD notation allowing for the development of an easy to understand model and the simulation enabling the detailed inspection of specific parts of the process. The combination of the two allowing for a more structured,

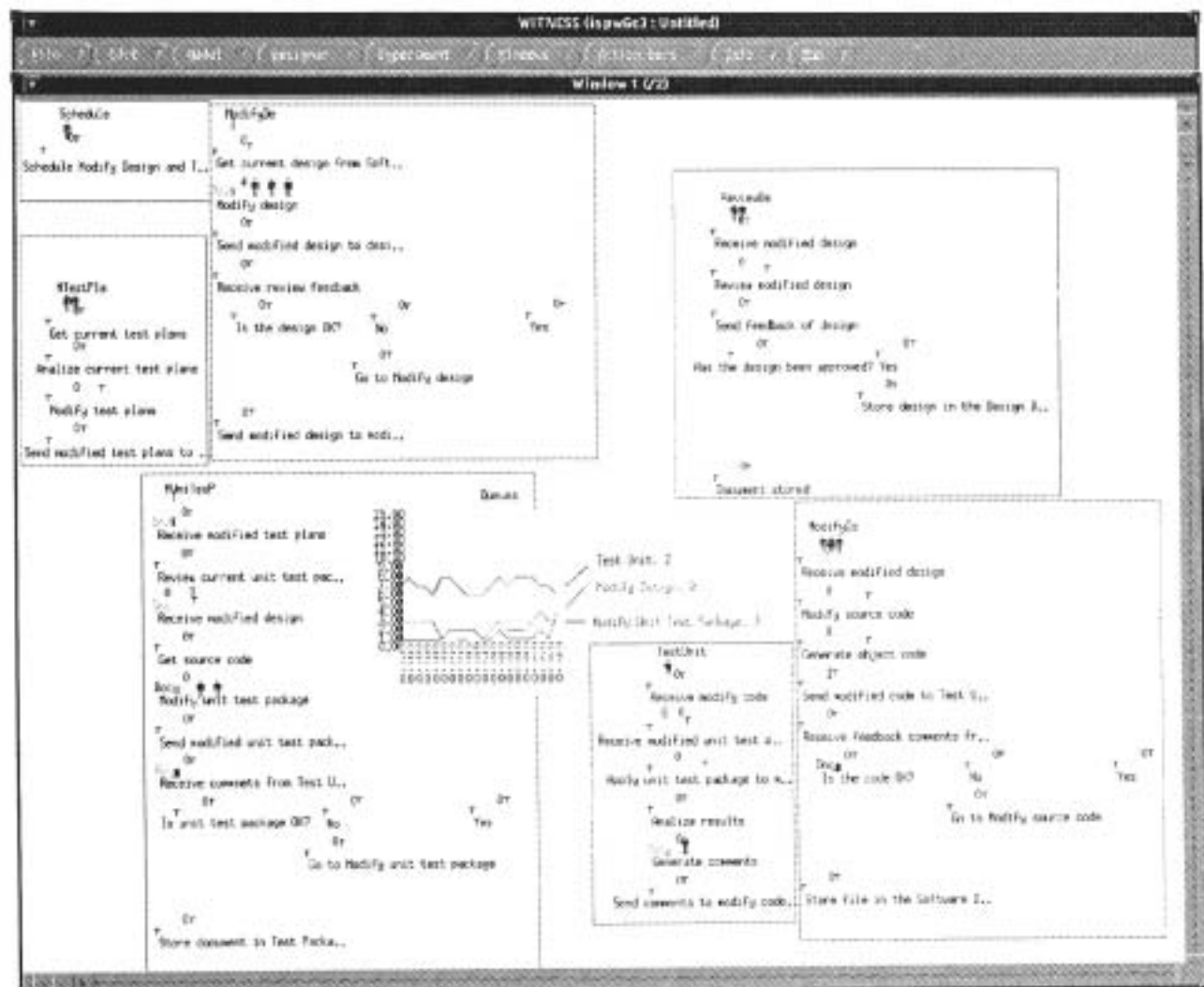


Figure 8: Representation of the ISPW6 process example in the Discrete Event simulation package WITNESS, where it illustrates one of the possible states of the system when it is being simulated. The boxes show how the mapping maintains the same structure of the process represented in Role Activity Diagrams and the time series graph illustrates some of the queues forming during the simulation of the process.

and considerably, cheaper approach to the simulation of very large processes. Rather than produce expensive single level simulations of total processes we perceive the joint approach, as illustrated by our prototype, allows for a refinement approach to systems simulation. Essentially this combines the advantages of abstract models, using the RAD technique, with the advantage of simulation of very concrete models to investigate specific process bottlenecks performance.

From the experience obtained from the development of the tool, we can identify the need to modify the RADs' representation in order to introduce all the necessary dynamic attributes at the time of capturing the process for the generation of a complete simulation model. This extensions have already been made (ERADES, (Martínez, 1999)). The key is clearly to introduce the extra semantics in such a way as to not lose the clarity currently associated with the RAD notation.

6 Conclusions

We have introduced and described a more integrated approach towards the understanding, analysis and improvement of organizational processes.

We defined a technique that unifies the use of process modeling and simulation (UTPAI). UTPAI combines the main steps of a process modeling methodology and adds to the phase of analysis the use of a dynamic approach by means of a simulation study. Thus, enabling a more integrated view and analysis of the process, including the behavioral aspect.

UTPAI is a generic technique for process analysis and improvement. It provides a more complete approach, by adding to the socio-technical and process analysis the dynamic aspects. The last being an important and relevant aspect in the decision process for the redesign and improvement of a process.

UTPAI has not been specifically bound to tools and methodologies. Thus, when performing a process study, an organization could choose their own. However in our particular work, for the modeling of the process the OPM methodology (Warboys et al, 1999) which provides information of *what* interacts (model of the system), *why* (model of the objectives) and the goals are achieved (the RADs (model of the method)) has been used, with our main interest in the RAD notation. OPM has been used in conjunction with DES using a commercial tool that supports this approach (WITNESS). This combination of tools, methodologies and techniques gave very promis-

ing results. We developed a prototype tool that generates simulation models (in WITNESS) automatically from process models in the RAD notation, but the conceptual description of the mapping can be useful for other sort of simulation tools. Thus avoiding, the double effort of building the simulation models from scratch. This is one of the major problems that we seek to address providing the feasibility of building a tool to support UTPAI in the analysis of more complex contexts. Our case study does not use all the steps of UTPAI, nevertheless it establishes the bases of the kind of analysis that can be performed and the idea of a support tool for one of the phases of UTPAI.

Process modeling and simulation grouped together provide more information on the different aspects of a process refinement than any single technique, and in general they complement each other very well.

Most simulation tools cannot capture abstract processes that have not yet been defined in great detail (informational aspects for example). Instead all the process details have to be modeled before implementing the process in the simulation tool. One advantage of combining process capture and simulation is the automatic generation of a simulation model from a process model. This combination provides the capability for initially generating a model of an abstract view of the process and subsequently adding refinement of the model. On the other hand the process model static representation provides us with information to perform the process analysis and improvement such as problems with duplication of activities and information, responsibilities, coordination and communication and the basis to develop process support with IT such as workflow (functional view) and information systems (informational view).

Acknowledgments

AIMG work was supported by CONACYT (México) under the grant I33046-A.

References

- AT&T ISTEEL, WITNESS 6.0 user manual, 1994).
- Bruce Silver Associates, *BPR Tools Add Power to Workflow Automation*, Industry Insider Report
URL: <http://www.metasoftware.com/whiteppr.html> (consulted in September 1997).
- Bruynooghe R. F., Greenwood R. M., Robertson I., Sa, J., and Warboys B. C. in, *Software Pro-*

- cess Modelling and Technology*, edited by Finkelstein A., Kramer J., and Nuseibeh B., Research Studies Press LTD, John Wiley and Sons INC., 1994.
- Cory T., *Business Processes Modelling Tool Products*, Sodan, Second Edition, 1996.
- Curtis B., Kellner M. I., and Over J., *Process Modelling*, Communications of the ACM, Vol. 35, No. 9, 1992, pp. 75-901.
- Gladwin B., and Tumay K., *Modelling Business Processes with Simulation Tools*, Proceedings of the 1994 Winter Simulation Conference, 1994, pp. 114-121, and references within.
- Hammer M., and Champy J., *Reengineering the corporation - a manifesto for business revolution*, Nicholas Brealey Publishing, London, 1994.
- Harrel C. R., and Field K. C., *Integrating Process Mapping and Simulation*, In Proceedings of the 1996 Winter Simulation Conference, December 1996, pp. 1292-1296.
- Holt A. W., Ramsey H. R., and Grimes J. D., *Coordination System Technology as the Basis for a Programming Environment*, Electrical Communication, Vol. 57, No. 4, 1983, pp. 307-314.
- Hlupic V., *Business Process Re-engineering and Simulation: Bridging the Gap*, In Proceedings of the 12th European Simulation Multiconference, June 1998, pp. 641-645",
- Hunt D. V., *Process Mapping*, John Wiley & Sons, Inc., 1996.
- Kawalek P., *A Method for Designing the Software Support for Coordination*, Ph.D. Thesis, The University of Manchester, Computer Science Department, January 1997.
- Kay S., Hardiker N., Aldridge S., Warboys B., Robertson I., Martínez- García A., Deneath M., and Pitt M., *Process Modelling and Simulation for Managing Clinical Care in the Community*, AMIA Annual Fall Symposium, 1998.
- Kellner M., Feiler P., Finkelstein A., Katayama T., Osterweil L., Penedo M., and Rombach H. D., *ISPW-6 Software Process Example*, In Proceedings of the First International Conference on Software Process, (Rocky Mountain Institute of Software Engineering ed.), Washington, IEEE Computer Society Press, (1991).
- Kettinger W. J., Teng J. T. C., and Guha S., *Business Process Change: A Study of Methodologies, Techniques, and tools*, Management Information Systems Quarterly, Vol.21, No.1, 1997, pp. 55-80.
- Law A. M., and Kelton W. D., *Simulation Modeling and Analysis*, McGraw-Hill, Inc. Second edition, 1991.
- Lingineni M., Caraway B., Benjamin P. C., and Mayer R. J., *A tutorial on ProSim: A knowledge-based simulation model design*, In Proceedings of the 1995 Winter Simulation Conference, December 1995, pp. 408-412.
- Manganelli R. L., and Klein M. M., *Como Hacer Reingeniería*, Grupo editorial norma, 1996.
- Martínez-García A. I., and Warboys B. C., *From RADs to DESs: a mapping from Process Models to Discrete Event Simulation*, Software Process Simulation Modeling (ProSim) Workshop'98, June 22-24, 1998.
- Martínez-García A. I., *Understanding the dynamics of process models: a unified approach*, PhD thesis, Computer Science Department, University of Manchester, Manchester UK, 1999.
- Miers D., *Use Of Tools and Technology Within A BPR Initiative*, in Business Processes Reengineering, (Coulson-Thomas Colin ed.), Kogan Page Limited, Pages 142-165, 1996.
- Ould M. A., *Business Processes: Modelling and Analysis for Reengineering and Improvement*, John Wiley and Sons Inc., 1995.
- Paul R. J., Hlupic V., and Giaglis G. M., *Simulation Modelling of Business Processes*, In Proceedings of the UK Academy for Information Systems 1998 Conference, April 1998, pp 311-320.
- Pegden C. D., *Simulation: A Look to the Future*, 11th European Simulation Multiconference (ESM'97), 1997, pp. K17- K21.
- Pollacia L. F., *Survey of Discrete Event Simulation and State-of-the-Art Discrete Event Languages*, Simulation Digest, Vol. 20, Num. 3, 1989, pp. 8-25.
- Thompson W. B., *Introduction to WITNESS and Linking to Process Mapping Tools*, Proceedings of the 1995 Winter Simulation Conference, Pages 462-466, 1995.
- Tumay K., *Business Process Simulation*, Proceedings of the 1996 Winter Simulation Conference, 1996, pp. 93-98.

Warboys B. C., *The IPSE2.5 Project: Process Modelling as the Basis for a Support Environment*, In Proceedings of the First International Conference on System Development Environment and Factories, Berlin, May 1989.

Warboys B., Kawalek P., Robertson I., and Greenwood M., *Business Information Systems: a Process Approach*, McGraw Hill, 1999.

Wastell D. G., White P., and Kawalek P., *A methodology for business process redesign: experiences and issues*, Journal of Strategic Information Systems, Vol.3, No.1, 1994, pp. 23-40.



Ana I. Martinez is a researcher in the Department of Computer Science at CICESE, working in the area of process modeling and simulation. She received her PhD at the University of Manchester in 1999 and her MSc degree from the same university in 1996. She has participated in several workshops on process models and simulation. She is particularly interested in the application of both process modeling and simulation applied to the re-engineering of organizational processes in particular in the software process.



Brian Warboys has been the Professor of Software Engineering at the University of Manchester for the past thirteen years. Previously, since 1963, he worked for ICL, the UK's largest computer manufacturer, and was Chief Designer of the company's mainframe operating system VME during its formative years in the 1970's. This experience led to an ongoing research interest in the problems of designing very large software systems. Focus on the development process for such systems and indeed, business processes more generally, proceeds through the work of the Informatics Process Group (IPG). The IPG was formed in 1979 and is particularly concerned with the general problems of re-aligning systems to co-ordinate and to integrate the wide variety of activity within the organisation, both human and machine, both co-present and distributed.

