A user-centred approach to the design of an expert system for training

Marcos Augusto Francisco Borges and M. Cecília C. Baranauskas

Marcos Augusto Francisco Borges is a graduate student of the Institute of Computing, IC, at the State University of Campinas (Unicamp). Maria Cecília Calani Baranauskas is a staff member of the Institute of Computing, IC, and associate director of the Nucleus of Informatics Applied to Education, at the State University of Campinas (Unicamp). Her research interests have focused on computer-based learning environments, human factors in the development of computational systems, and artificial intelligence applied to education. Address for correspondence: Institute of Computing IC-Unicamp & Nucleus of Informatics Applied to Education, Nied-Unicamp CP 6176-13083 970 Campinas, S.P. Brazil.

Abstract
This paper reports the objectives, design approach and first results of a research project that aims to develop an intelligent computer-based learning environment for industrial applications. Jonas, an expert system, is part of a modelling/simulation environment which enables shopfloor workers to test and to put new philosophies of work into practice in the context of manufacture. Our approach is focused on three main ideas: the Intelligence Augmentation paradigm, user-centred system design and the constructionist theory of learning. The architecture of the modelling environment and the main features of Jonas are presented as an illustration of the design ideas discussed in the paper.

Introduction
Commercially available environments for the creation and simulation of models in companies presuppose a formal education in mathematics, favouring those who work in management areas. Despite recent developments in modelling and simulation environments, not much of this work has addressed training in companies.

Enxuto is a modelling and simulation computational environment designed for the manufacturing training context (Borges et al., 1995). In contrast to intelligent training systems (ITS), it has the different focus of aiming to lead the users to explore concrete models of important things, ideas, and their relationships on their own. In ITSs, the student is controlled by an expert tutor, ignoring the fact that the student, after a little time, must go on in independent exploration (Lawler, 1987; Lawler and Yazdani, 1987).
The computer-based learning environment we describe addresses the use of the computer not as a teaching machine, but as a new educational medium. The aim is to encourage the learner to take the initiative. Cumming and Self (1990) propose the term intelligent educational system (IES) to weaken any association with authoritarian tutoring and to emphasise that a wide variety of types of interaction should occur between learners and IESs. Enxuto addresses the needs of shopfloor workers learning manufacturing concepts. This environment aims to create a distance from their real world by means of introducing an abstraction level required to understand how their work functions. Particular attention is paid to stimulate workers to take risks by engaging them in the process of exposing and testing their own ideas.

This computational system provides entities and interconnections related to manufacturing: machine cells, containers, paths, queues, operation times, assembly and other operational information. Furthermore, the system provides an interface appropriate to non-experts, based on graphics and animation. By interacting with the elements of the system, a learner can build, test and refine his proposed model in a concrete way. Jonas, the system being described, is the expert system supporting Enxuto.

The benefits of using simulation in educational contexts are already well known (De Jong, 1991; Pagano, 1992; Hebenstreit, 1991; Baranauskas, 1994). Such benefits are enhanced if we enable a simulation user to construct the models he wants to “simulate”. We define computational modelling as the act of using the computer to express the model and then to explore its consequences to evaluate not only the constructed model, but the knowledge of the target system/phenomenon as well. In the cycle of activities in which the user gets involved during the modelling process, the analysis of his model simulation is a critical phase which requires special attention. In many situations, the results of model simulation are not “self-explaining”. The efforts Jonas makes are to help the user to understand the results obtained from the simulation of the created model. In the modelling environment, Jonas plays the role of a counsellor agent with whom the user will interact in order to get a better understanding of the situation being modelled.

The main ideas underlying the conception of Jonas are the paradigm of Intelligence Augmentation (IA) (Fischer, 1995); user-centred system design (Norman, 1986) and the constructionist approach to learning (Papert, 1986). In the next sections, we discuss the main ideas underlying Jonas’ design and present a description of its main features.

Theoretical basis
Within the tradition of Artificial Intelligence (AI), intelligent systems for learning have focused on a “mimicking” approach whose basic assumption is to provide systems with the abilities of tutoring, coaching or instructing the student. The paradigm of Intelligence Augmentation (IA) focuses on augmenting and empowering human intelligence rather than emulating it (Fischer, 1995). In this approach, human-computer collaboration exploits the asymmetric abilities of humans and computers and tries to share the responsibility assumed by them. The main issue raised by this approach
is how to organise elements which could make human communication with computational systems easier, by shifting the focus on interaction mechanisms designed for the user, rather than for the system.

The interaction of a user with a computational system in order to perform a task involves a cycle of activities related to both the execution of an action and the evaluation of the effects of this action (Norman, 1986). Executing an action requires forming an intention and specifying a sequence of steps. The continuation of an interaction depends on the user’s interpretation of the system state with regard to his initial goals and intentions. This involves the perception, interpretation and evaluation of the system state after carrying out an action.

Computational systems for modelling are a good example to illustrate Norman’s theory of action. Furthermore, such systems are potentially rich learning environments, since they can involve the learner in the basic cycle of expression, evaluation and reflection of the domain being modelled. The fact that a computer demands formal expression of a model leads the learner to a more precise understanding of his knowledge of the domain. Additionally, the execution of the model by the machine enables an evaluation which in turn can motivate the learner to question the model, to re-evaluate his knowledge and to re-express it, thus, continuing the cycle of actions, in a constructionist style of learning (Papert, 1986; Valente, 1993).

In short, in the human-centred approach to the design of computer-based learning environments, we must consider that the goals stem from human needs, and not from the computational system. Cumming and Self (1990) argue that many types of good learning interaction consist of a task level and a discussion level. They say that at the task level the learner works on the activity itself, while reflection about what is going on occurs at a higher discussion level. The Jonas system is an attempt to develop an advising component that can support a discussion level interaction with the user of a modelling/simulation environment. The system detects potential problems which might have remained unnoticed without it, and provides explanations relevant to the identified situations. The user evaluates Jonas’ criticisms and decides what to do next while keeping control of the interactions. These features set our approach apart from expert systems and intelligent tutoring systems, which are traditionally based on system-driven interactions.

Thus, the human-centred approach to the design of computer-based learning systems raises two important questions:

• How could the system make it easier for the user to form his intention and its change into actions so they could be performed upon the mechanisms available through the interface?
• How could the system make the interpretation of the output easier, so the user perceives it through the interface?

These are the main questions we have been considering during our design approach.
The Enxuto/Jonas environment

The Enxuto/Jonas architecture is composed of three main modules: user interface, modelling/simulation, and an expert system (Jonas), as illustrated by Figure 1.

During the interaction with the environment, the user gets involved in a cycle of activities pretending s/he is a factory manager. In this process the user represents a model of his/her factory, simulates its operation and explores its consequences. Consider, for example, a simplified scenario where a user intends to evaluate the results of a production line composed of a supplier (S), a stock of raw material, a machine (M), a product stock and a client (C). Figure 2 is a snapshot of the model construction in Enxuto. On the left of the machine there are pieces waiting processing and, on the right, the manufactured products, awaiting transport to the client.

The user models the production line by means of the user interface of Enxuto. In this process, s/he not only selects and connects the objects provided by the pallet, but also sets up the attributes of the objects such as the production capacity of the machine, or the velocity of the transport. Then, s/he executes a simulation of the model. The simulation running is represented by an animation of the model shown by the interface. At the end of the simulation, the user can evaluate the model attributes resulting from the simulation process. For example, the user can observe the number of pieces that a machine was asked to produce and how many it really produces; how long the transport and the machine takes, how long they take waiting for something; how many pieces the supplier provides and the client receives, etc.

When the user gets the results, s/he has to identify where the model could be enhanced. It could be considered a model enhancement, for example, to make the production line provide more products using the same time and spending the same amount of money.

According to the constructionist paradigm, the user could learn through this cycle of modelling-simulation-evaluation of the results, and this is the learning process Enxuto intends to explore. There are, however, situations in which the user cannot identify any model attribute that could be changed in order to get a better simulation result. This
could be a very demotivating situation. Jonas was conceived to cope with these situations. The efforts Jonas makes are to help the user to understand the results obtained from the simulation of the created model. This constructivist view to a learning interaction could be described in terms of the task and discussion levels proposed by Cumming and Self (1990). More specifically the learner uses Enxuto to work at tasks of the manufacture domain, while, at a higher level, Jonas may be called to give help and advice.

**The Jonas system**

Jonas aims to propose lean production concepts which could be used for improving the model and for giving explanations of changes in simulation results. The main idea of lean production is to eliminate work, energy, time, money and material waste. The goal is to continuously reduce production cost to a minimum (Mazzone, 1993). In order to do so, Jonas is able to analyse model components, their functional relations and their operational characteristics. In a simplified description, the Jonas system architecture, schematically illustrated by Figure 3, is composed of:

- an expert system: the inference machine and the knowledge base (KB);
- a graphic interface:
connections with the graphic interface, modelling data base and simulation results;
• an administrative subsystem.

The inference machine examines the model base, and identifies modifications performed on the model by the user and attributes that had a significant change after the simulation. Then, the inference machine uses the KB in order to infer how the system could achieve better results or why the simulation results changed after the last modification in the model.

The administrative subsystem was designed to be used by a lean production expert of a real plant. Using this subsystem, the expert must build and operate the KB in order to maintain it according to the plant’s needs. The administrative subsystem is composed of tools for queries and manipulations in the knowledge base.

Special attention was given to the construction of the KB. In model-based diagnosis, the use of abductive inference is common (Boutilier and Becher, 1995). Although unsound, abduction is often essential to this type of problem. Problems in the plant show symptoms, thus, diagnosis must work from the symptoms back to the cause (Luger and Stubblefield, 1989). We designed an abductive KB based on rules and logical operations of the model attributes. This type of KB allows the domain expert to add knowledge to the KB directly, making it possible to build a more complete KB. It was built using definitions of real plant workers in a relational data-base structure.

To simplify, the KB is composed of a list of conclusions. In order to identify a valid conclusion for a specific situation (a hint that should be presented to the user), the system tries to verify a list of conditions. Each condition is related to an object of the model. A condition is true if its constituent elements are true.

The design of the model base uses the object-oriented paradigm, getting data from Enxuto (objects, attributes and connections) and organising it as a general object structure. Jonas was developed using Multimedia ToolBook and Visual C++. Its knowledge base is stored in Microsoft Access.

Figure 3: The Jonas system architecture

The design process of Jonas
We built a system to help users with no experience in computers to acquire knowledge about new manufacturing techniques. Based on the constructionist paradigm, Jonas does not just present explanations for the situations addressed. Jonas’ design intends to help users to build and refine their own knowledge of the domain. This is a difficult task when the user’s mental models are far from the system model presented by the interface design. First of all, we needed to design a “pleasant” tool. Ease of use and attractiveness are crucial qualities for users’ acceptance of the system (Wenger, 1987). Based on Laurel’s theory (1986), the system is pleasant when the interface is close enough to the user’s mental model. User-centred design helped us to direct the project towards this goal.

The first point addressed in the design was to understand the user’s needs and expectations. With this understanding, it was possible to design a system model closer to the users’ mental models (Norman, 1986; Laurel, 1986). Trying to provide what was considered important by these users, we have carried out two experiments based on Gomol (1990) with three groups of two persons in a typical plant environment. Two of these groups of typical users worked on manufacturing scenarios and tried to solve specific problems. The other group simulated the “Jonas” behaviour. We used the experiment’s results to move the original interface closer to the user’s mental model. Based on the results of this first experiment we designed a first version of the system interface. A second experiment conducted led to other adjustments in the interface. After that, the internal functionality of the system was defined and implemented.

Instead of “how can the system solve a problem?”, the main worry of IA is to identify “how will the system and the user work together?”. We applied the IA approach to the design of Jonas in order to build a system that could enhance the potential of the user.

Jonas in action: an example
Consider the simplified typical scenario represented by Figure 2. Evaluating the model based on the lean production theory, it can be verified that the machine could be a bottleneck (it is just after a high stock of material and before a small one). If the user needs help in order to know how to change the model to get a better result, s/he can ask Jonas for help by clicking on the Jonas icon.

When Jonas is called, it uses the KB to identify the model results that could be used to give hints about possible enhancements. At the beginning of the interaction, Jonas presents a list of this kind of result as shown by Figure 4. Analysing these results, the user can think about, and choose, how to make an improvement in the model.

If the user still cannot identify where to make an improvement in the model, he must select a result and, then, click on the button “Thinking about the selected result”. Jonas will try to present some hints related to the selected result. It is important to notice that Jonas will not present the answer directly, pointing out where the user should change attributes, but instead, will give hints to help the user to find this information by
himself/herself. In the case of this example, if the user clicks the “Thinking ...” button, the system will present Figure 5.

Thus, rather than presenting direct answers like “You should increase the capacity attribute of the machine”, which uses an instructionist style of interaction, Jonas gives the hint “There is a plant bottleneck”. It is worth noting that the system does not tell the user where the bottleneck is located, neither does it teach him/her how to solve this problem. If this indirect suggestion is not enough s/he can ask for more hints, continuing the search process for the improvement in the model. Reflection on what is taking place occurs during this process. In this sense, knowledge of the domain is not given, or done, to the learner by Jonas, but is constructed by the learner as the result of performing actions and experiencing their consequences in the environment.

Summary and conclusions
New methods of manufacture have changed industrial production systems fundamentally. As a consequence, more flexible workers capable of using their creative potential are the new force required for industry. Training methods have not measured up to this evolution. In this paper we addressed the design of a learning environment for factory workers in the context of new production systems. In order to build a useful learning environment for these workers, we have supported our design with the user-centred approach, constructionist ideas about learning and IA techniques.
The Enxuto/Jonas system is part of a research project conducted in cooperation with GM in Brazil (Nied, 1995). The design of the proposed environment is a result of continual interactions of the design/development team with the target user (shopfloor workers). Several evaluations reported in Borges and Baranauskas (1997) took place during the design process and their results were used to conduct the development of the interface.

Preliminary results of using the Jonas/Enxuto system at the factory encourages future research in formal assessment of the learning environment. In providing this new style of training, the major concern is effectiveness in the work context: how much do the workers learn? How to measure this learning? Test scores are not necessarily appropriate. They do not necessarily address the depth of understanding or the skills the workers have acquired. Evaluation of the system in a factory context is challenging work to be done in the near future, and the next step in the project.

**Acknowledgements**

We thank the CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) for financial support; Harrison Thermal Systems (GMC) for the opportunities of interacting, collecting data and testing the system in a real plant environment; and Nied (Núcleo de Informática Aplicada à Educação) for making available the environment.
used for research and development. We are also grateful to the editors and to the referees for accurate reading and helpful comments.

References
Fischer G (1995) Rethinking and reinventing artificial intelligence from the perspective of human-centred computational artefacts Proceedings of XII Brazilian Symposium on Artificial Intelligence Campinas, Brazil.
Lawler R W (1987) Learning Environments: now, then, and someday in Lawler R W and Yazdani M (ed) Artificial intelligence and education, volume one, Learning environments and tutoring systems Ablex, Norwood, NJ.
Lawler R W and Yazdani M (1987) Artificial intelligence and education, volume one, Learning environments and tutoring systems Ablex, Norwood, NJ.
Luger G F and Stubblefield W A (1989) Artificial Intelligence and the design of expert systems Benjamin/Cummings, Redwood, CA.
Papert S (1986) Constructionism: a new opportunity for elementary science education. A proposal to the National Science Foundation, Massachusetts Institute of Technology, USA.
Wenger E (1987) Artificial Intelligence and tutoring systems: computational and cognitive approaches to the communication of knowledge Morgan Kaufmann, Los Altos, CA.