Online and Distance Learning:
Concepts, Methodologies, Tools, and Applications

Lawrence Tomei
Robert Morris University, USA

Information Science REFERENCE

INFORMATION SCIENCE REFERENCE
Hershey · New York

Assistant Executive Editor:	Meg Stocking
Acquisitions Editor:	Kristin Klinger
Development Editor:	Kristin Roth
Senior Managing Editor:	Jennifer Neidig
Managing Editor:	Sara Reed
Typesetter:	Sharon Berger, Jennifer Neidig, Sara Reed, Laurie Ridge, Jamie Snavely, Michael Brehm, Elizabeth Duke, and Diane Huskinson
Cover Design:	Lisa Tosheff
Printed at:	Yurchak Printing Inc.

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue, Suite 200
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-pub.com
Web site: http://www.igi-pub.com/reference

and in the United Kingdom by
Information Science Reference (an imprint of IGI Global)
3 Henrietta Street
Covent Garden
London WC2E 8LU
Tel: 44 20 7240 0856
Fax: 44 20 7379 0609
Web site: http://www.eurospanonline.com

Library of Congress Cataloging-in-Publication Data

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

Chapter 2.31
Mascaret:
A Pedagogical Multi-Agent System for Virtual Environment for Training

Cédric Buche
CERV/ENIB, France

Ronan Querrec
CERV/ENIB, France

Pierre De Loor
CERV/ENIB, France

Pierre Chevaillier
CERV/ENIB, France

ABSTRACT

This study concerns virtual environments for training in operational conditions. The principal developed idea is that these environments are heterogeneous and open multi-agent systems. The MASCARET model is proposed to organize the interactions between agents and to provide them reactive, cognitive and social abilities to simulate the physical and social environment. The physical environment represents, in a realistic way, the phenomena that learners and teachers have to take into account. The social environment is simulated by agents executing collaborative and adaptive tasks. These agents realize, in team, procedures that they have to adapt to the environment. The users participate to the training environment through their avatar. In this chapter, we explain how we integrated, in MASCARET, models necessary to the creation of intelligent tutoring system. We notably incorporate pedagogical strategies and pedagogical actions. We present pedagogical agents. To validate our model, the SÉCURÉVI application for fire fighters' training is developed.

INTRODUCTION

This study concerns the design of virtual environments for training (VET). We want to immerse learners in their professional environment simulated using virtual reality techniques. This enables them to manipulate the environment so that they can *"learn while doing"*. This idea is driven by the "constructivism" paradigm defined by Piaget (1978) and can find a good implementation in virtual reality techniques as presented by Burdea and Coiffet (1993). Our definition of virtual reality is the one proposed by Tisseau (2001), who proposes to give autonomy to models evolving in the virtual environment by giving them the "triple mediation of senses, decision and action". So, the main developed idea is that virtual environments for training are heterogeneous and open multi-agent systems. Those multi-agent systems (MAS) had been presented by Demazeau (1995) using the *VOWELS* model considering a MAS with four vowels: **A**gent, **E**nvironment, **I**nteraction and **O**rganisation. It also has been use for collaborative work simulation by Clancey (2002). We consider the user of a virtual environment as other autonomous agents because he or she can interact with the environment and with other agents or users in the same way. Then, as Tisseau, we propose to add a last vowel, the letter **U** for **U**ser, in the *VOWELS* model.

Our goal is to provide an agent-based model to create a virtual environment for training (VET) integrating an intelligent tutoring system (ITS) in order to provide students with dedicated tutoring. ITSs are based on four models (Woolf, 1992): domain model (Anderson, 1988), learner model (Self, 1988), pedagogical model (Wenger, 1987) and the interface model (Miller, 1988).

Comparing to STEVE (Rickel & Johnson, 1999), which is a mono-agent system based on a virtual tutor, we propose a multi-agent system where each entity can contribute to the pedagogy. Moreover, pedagogical skills are not imposed by our model; they are viewed as knowledge items manipulated by agents. These elements provide flexibility and adaptability to our VET. In Lourdeaux, Fuchs and Burkhardt (2001), HAL is a pedagogical agent (based on a pedagogical model) helped by environment-agent and scenario-agent responsible to the detection of the learner actions and intentions. The present work goes further; it takes into account all the ITSs models and both the learner environment (physical and social context) and the pedagogical one are considered as a multi-agent system.

A major issue in multi-agent system is the definition of agents' interactions. In our case, interactions have to be flexible and controlled by pedagogy. Therefore, we propose a model centred on the concept of organisation, which permits to structure these interactions. We show that information about organisation can also structure knowledge upon classical ITS models and thus allows agents to make pedagogical decisions.

After an overview on ITS models, we present our *MASCARET* model (multi-agent system for collaborative adaptive and realistic environment for training). We start by defining a generic model, which is then derived to represent the different types of interaction in our VET. Next we explain how the different models of ITSs are taken into account by pedagogical agents, defining therefore the pedagogical organisation. Finally we briefly present the application of *MASCARET* to a firefighter training environment: *SÉCURÉVI*.

INTELLIGENT TUTORING SYSTEMS

Intelligent tutoring systems (ITS) are computer processing systems for training incorporating communication techniques of knowledge and skills. They were conceived from the combination of interactive learning environments (ILE) and artificial intelligence (AI) techniques. Such systems were developed with the objective to adapt speed and level of the knowledge representation to the student's needs. The system uses an internal representation of this knowledge and has possibility of reasoning.

In the last 10 years, ITSs were used within the framework of training and had proved their effectiveness (Shute, 1990). For example, students using LISP tutor (Anderson, 1990) finish their exercises 30% faster than those that receive a traditional training. The final examination shows a difference of 43% in result between the two methods.

Traditionally an ITS is described using four major functions or components (Wenger, 1987). Thus, an ITS is composed of models, each one playing a particular role and contributing to ITS decision-making. The first ITSs were composed of an expertise on the domain, an expertise on what must be learned and a representation of what the student learned or misunderstood. Burns and Capps (1988) identify these components as the three models of an ITS. They correspond to the "expert module" (Anderson, 1988), to the "diagnostic module of the student" (VanLehn, 1988) and to the "instruction and curriculum module" (Halff, 1988). Later, a fourth model has been introduced, augmenting the first three, the "interface module" providing knowledge about environment representation. Although there is no standard, an ITS generally consists of four models, identified in Woolf (1992):

- a *domain model,* representing the expert knowledge,
- a *learner model,* permitting to get the state of knowledge at a given moment,
- a *pedagogic model,* permitting to carry out teaching choices according to the learner behaviour,
- an *interface model,* allowing the information exchange between system and user.

Domain Model

The domain model represents the expertise on the domain (Nkambou, Gauthier & Frasson, 1997). It is also called expert model since it defines the expert knowledge in a field of knowledge. The domain model does not contain only a description of competences to acquire, but also a representation of knowledge to transmit. It has to be able to propose several paths possibilities to achieve an objective. It consists of two components: the *declarative knowledge* and the *procedural knowledge.* On one hand the declarative brings a base of knowledge representing elements the professor would like to transmit. On the other hand the procedural proposes a reasoning system able to interpret the knowledge base.

The purpose of expertise on the domain is to allow a comparison with solutions suggested by the student. For that, the domain model has to be able to generate solutions on the problems in the same context as the student one, so respective answers can be compared. Thus the system is able to determine differences and correspondences between student actions and those awaited. It can also evaluate performances and locate student difficulties. Lastly, the knowledge on the domain allows the generation of explanations related to the expert solution.

The knowledge representation requires the use of formalism. The *logical* formalism was one of the first formalisms suggested to represent knowledge. It uses a language, axioms and rules (logic proposal, fuzzy, modal, etc.) allowing the representation of veracity, uncertainty, temporality, and so forth. Moreover, cognitive sciences, which are interested in intelligence mechanisms, generally use a formalism based on *graphs.* It gathers, in the shape of graph concepts, notions representing knowledge and their inter-connections. Semantic networks (Quillian, 1968), networks with markers propagation (Fahlman, 1979) or conceptual graphs (Sowa, 1984) allow the representation of knowledge. An inference mechanism brings know-how. Lastly, a logical description formalism of knowledge representation called *frame* (Minsky, 1975) is also used. A frame is a structure to represent a concept or a situation as "a room" or "to be in a room". This formalism associates for each knowledge a know-how.

Student Model

Whereas the domain model contains expert knowledge on the problem resolution, the learner model (Leman, Marcenac, & Giroux 1996; Py, 1998) brings a measurement of student knowledge on this problem. It is as called a diagnostic model since it allows measuring the student progression. Ideally, this model must contain an advanced representation of the student profile. It must provide all the specific aspects of each student's behaviour and knowledge in the form of a model. The student profile is established and updated through the interactions he or she operates with his/her environment.

Using information of the learner model, the system adapts itself to the student. The profile must integrate learner knowledge on the domain model, called epistemic sub-model (Delestre, 2000). The objective of the epistemic sub-model is to determine the state of student knowledge for concepts present in the domain. Moreover, it must integrate his or her not-epistemic characteristics representing his/her pedagogical preferences or objectives, called the behavioural sub-model (Delestre, 2000). It defines pedagogical objectives of the current exercise. The system must take into account these objectives, while being more or less flexible according to the training type. In the same way, the student must have the possibility to choose the point of view of such or such teacher. Lastly, the system must deal with the specific student capacities according to fields of teaching.

One of the learner model objectives is to allow the evaluation of each knowledge element to acquire. Incorrect behaviours can occur; also incorrect knowledge must be represented with the aim to identify errors. Several methods exist to evaluate knowledge to be acquired:

1. *Model tracing* compares stages achieved by the student and existing stages in procedural rules defined in the domain model. This approach was used by tutor LISP (Anderson & Reiser 1985);
2. *Issue tracing* is a modification of *model tracing*. The purpose of this model is not to model the problem resolution process but rather to determine competences and solutions to use. An update of the competences acquired by the student is used. Tutor WEST (Burton & Brown, 1982) uses this method;
3. *Expert systems* analyze the student answers. The conclusions of the domain model are used to update the learner model. HANDLEBAR (Clancey, 1983) uses this method.

The representation of the epistemic and behavioural sub-models can use various methods:

1. The method called the *overlay* proposes to cut out the expertise in basic units. The student model is composed of a subset of these entities. The student knowledge is regarded as under part of the expert knowledge. An empty model corresponds to a student who would not have any domain knowledge, while a model identical to the expert corresponds to a student who would have reached the same level of control as a domain expert. Each knowledge item can be labelled with a discrete value (known/unknown) or a continue value (from 0 to 1). This principle was used by tutors WUSOR (Stansfield, Carr & Goldstein, 1976) and HANDLEBAR. The errors made by the student are explained in terms of knowledge absence: it is the ignorance of such rule or such concept which leads the student not to do the best possibility. This model considers the student will learn nothing except what the expert had provided. Thus there is no

mechanism to know not acquired knowledge and those which were not still presented;
2. The method called *differential,* which is an extension of the *overlay,* where knowledge is separated considering if the student was exposed to such or such knowledge or was not. This method was used for tutor WEST;
3. The method called the *buggy model*, which is also an extension of the *overlay.* The approach consists to present in the learner model rules whose application produces an incorrect result. This method was used by Brown and Burton in systems BUGGY (Brown & Burton, 1978) and DEBUGGY (Burton, 1982). Knowledge is represented by an elementary procedures network. Any correct procedure can be replaced by an incorrect procedure having the same domain applicability. From a set of answers given by a student, DEBUGGY builds the correct and incorrect procedures network whose behaviour approaches most the student.

Pedagogical Model

The pedagogical model allows defining the pedagogical activities aiming at helping the teacher in the training process. It allows simulating the teacher decisional behaviour relative to a pedagogical intervention, based on differences between the domain model and the learner model. Then, the main objective of the pedagogical model is to answer three questions (Lourdeaux et al., 2001; Wenger, 1987):

- **When intervene?** The pedagogical model determines when an intervention is desirable, if the student must be stopped or not. The trainers can intervene at various times: following errors made by formed, before errors in order to show up the risks of errors, following student questions, and so forth. To determine when to intervene is a subtle decision. To guide a student, it is sometimes more effective to let the student seek during one moment than to stop him or her each time. On another side, been left to him or herself, the student probably will be lost.
- **Why intervene?** Moreover, the pedagogical model determines why to intervene. The objective can be to check student knowledge acquired or guide him or her in its training. Thus, Lourdeaux et al. (2001) proposes two types of interventions:
  1. Pedagogical strategies related to the scenario modification allowing trainers to check student knowledge;
  2. Pedagogical strategies related to the student guidance are distributed according to two categories: active methods (training by the action) and explanatory methods (training by the explanation).
- **How intervene?** Much more, the pedagogical model determines the nature of the assistance. It can be a modification of the environment, the exercise or simply an addition of information. For that, it must take into account the student profile and the environment characteristics. Lourdeaux proposes a categorization in the various ways of representing the "pedagogical assistances" according to various levels of realism: enrichment, degradation, simplification, and so forth (Lourdeaux et al., 2001).

The pedagogical model must choose possible interventions allowing helping the student. For that, it can specify and control its interventions based on one or more "methods":

- ***Socratic* method:** the system asks questions to the student in order to encourage him/her to analyze its own errors (used by SCHOLAR and WHY);

- ***Coaching* method:** the system lets the student act and waits until he/she asks for assistance (used by SOPHIE, WUMPUS and WEST);
- ***Learning by doing* method:** the system is active and encourages the student to select information and deduce orientations on the domain model;
- ***Learning while doing* method:** the system remains in background task and only provides punctually tips.

Interface Model

The interface model allows communication and finalizes the form by which the system wants to transmit information. This model is in co-operation with the diagnostic and didactic of the system. It transforms the internal representation of the system into comprehensible information for the learner. This model can transmit the same knowledge more or less clearly. Indeed, even if the pedagogical model decides course and contents of the didactic actions, the interface model deals with its final form. More generally, this model takes care of communications between the student and the system remainder (Miller, 1988). Thus, it is in charge of the bidirectional communication between the internal representation of the system and a comprehensible interface for the student (Wenger, 1987).

We can define two directions for the communication:

1. **System towards learner:** it is the ergonomic of the interface, the way of defining which type of media to use to translate the system information. One of the difficulties in this context is the difficulty of adaptation to users. Indeed, various users will have different behaviours with various types of interface. For example, certain people will be more receptive to a purely visual interface, others with sound, and so forth. The interface designer will be able to discover which type of stimuli the user will be most receptive to, based on the learner model.
2. **Learner towards system:** it is the detection of action/intention, the way of recovering information coming from the user in order to be able to analyze them. Burkhardt (2003) underlines the fact that the intention detection of the user is a central problem regarding interactions. It is necessary and important to separate action detection from of intention detection. Indeed, the intention does not imply the action and conversely.

We noticed three systems setting up one or more facets of the interface model:

- **METADYNE** (Delestre, 2000) is an adaptive hypermedia able to get student actions following hypertexts links used. Ergonomics is directed by the hypermedia, and consequently presents little interest in an immersive virtual environment;
- **HAL** (Lourdeaux et al., 2001) proposes to recognize actions using message communication between environment agents and compares these data with a preset scenario to extract important information;
- **STEVE** (Rickel et al., 1999) also proposes to recognize actions using communication of messages between environment agents and build a scenario considering a final objective. It sets up an animated virtual tutor able to recognize and generate speech.

The use of the interface does not have to impose a burden for the training that would block the real training (Wenger, 1987). The interface must use the advantage of existing communication conventions, like the natural language, while introducing some news, such as mouse (Vigano, Mottura, Calabi & Sacco, 2003).

MASCARET MODEL

Our goal is to train teams to collaborative and procedural work in a physical environment. In this case, we have to simulate in a *realistic* way this physical environment and the *collaborative* and *adaptive* team member's behaviour in the social environment. Evolution of those environments results from simulation of autonomous agent's local behaviour and their interactions. We propose a model, called ***MASCARET,*** where we use **m**ulti-**a**gent **s**ystems to simulate **c**ollaborative, **a**daptive and **r**ealistic **e**nvironments for **t**raining. This model aims at organizing the interactions between agents in the virtual environment and provides them with abilities to evolve in this context. In addition, it allows the establishment of models necessary to the creation of an intelligent tutoring system. In this context, the stress is put on the organisation facet of the MAS. The organisational model specifies if agents may interact or not, the way they can do it and what they are expected to do according to their capabilities (their potential behaviour and the information they are supposed to hold). Because agents may have to adapt their behaviour according to this information, they must maintain knowledge about organisations. It is a major issue in a VET where learners and teachers have to build representations about their environment. They have to know how this environment is structured and it is necessary to control what kind of interaction may arise.

The generic model of organisation is given in the next section. Its derivation to the modelling of specific interaction contexts is detailed in the followings.

Organisational Model

The generic organisational model is presented as a UML class diagram (Figure 1). It is based on the concepts of agent, organisation, role and behavioural feature. Hannoun, Boissier, Sichiman and Sayettat (1999) have already proposed an organisational model for multi-agent systems, but this model, dedicated to the collaborative realisation of procedures, is not generic enough to solve our problem. Ferber and Gutknecht (1998) have also proposed such a model called agent/group/role, but this model seems to be more a pattern for MAS design than a model that formalises the concepts of organisation and roles. In our model, the aim of the organisation is to structure interactions between agents; it enables each agent to know its partners and the role they are playing in the collaboration. The concept of role represents agent responsibilities in the organisation (corresponding to their behavioural features). Agents have then an organisational behaviour that permits them to play or abandon a role in an organisation. This behaviour also enables agents to take into account the existence of other members.

This model is a generic model in the way that all the resulting classes are abstract. The organisational model is then derived to implement specific organisations.

Figure 2 gives examples of different instances of specific organisations that can be modelled using *MASCARET.* This figure is not an exhaustive view of what the organisation may be and many other organisational entities may be imagined. Interaction patterns may differ by the number of agent involved and the roles they play. Notice that one agent can play roles in different organisations.

Physical organisation specifies the interactions that occur between entities compounding the physical environment of the trainees. Because the participation of an agent to an organisation is explicit, the designer of the training session can decide to activate or ignore different kinds of physical interactions. Thereby the difficulty level of the exercise can be controlled. Virtual human and users' avatar may be involved (or not) in physical organisation because they have to manipulate it and to undergo their environment

Figure 1. Generic organisational model (UML class diagram)

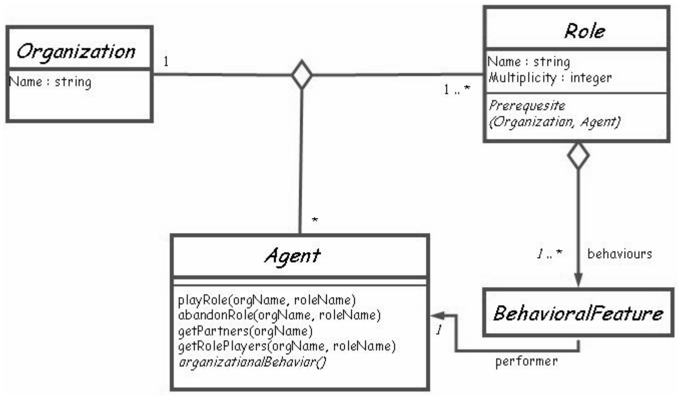
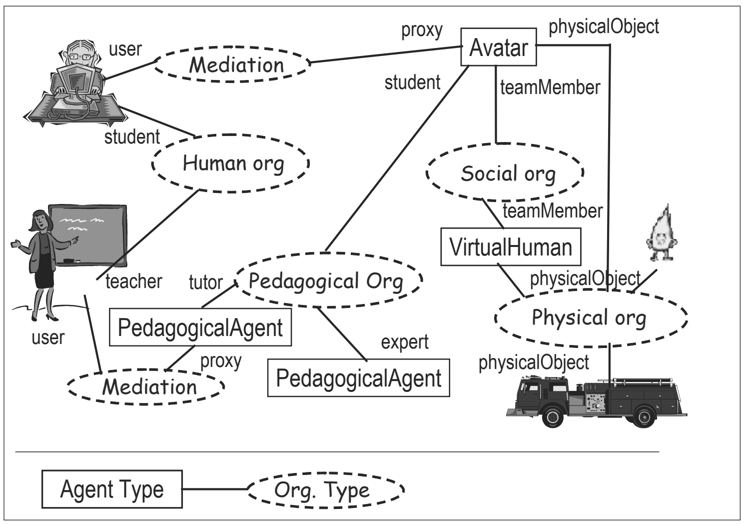


Figure 2. Different types of organisation in MASCARET



(e.g., depending on its learning experiences, a fire-fighter trainee may or not have to undergo the toxicity of a gas due to an accidental lick).

Social organisation represents social interactions that are a major point in collaborative training. This kind of organisation involved agents representing humans: virtual humans (their expected behaviour is simulated), avatars that represent users' actions (trainees, teachers or co-workers).

Pedagogical organisation allows specifying what trainees are supposed to do and what kind of actions and knowledge may be performed by pedagogical agents. These agents can be artificial agents (as an intelligent tutor in an ITS) or humans. In this last case, their actions are mediated by their proxy (see mediation organisation below). Pedagogical agent can play a wide variety of roles, for example, tutor, domain experts, learning companion, troublemaker …

Mediation organisation makes explicit the way the user can interact in the virtual environment and how far he or she can delegate actions to his avatar: for example the actions corresponding to movement from points to points can be delegated to the avatar so that the learner can focus his or her attention on more specific learning objects. Actions the learner is supposed to do and actions delegated to his/her avatar are specified by their respective roles.

Human organisation is not completely under the control of the VET: corresponding interactions are not mediated by the system. Anyway, assumptions may be made about these kinds of interactions and it can be decided to prevent them (when users are in different rooms) or to encourage them (by an invitation to ask a question to the teacher for example).

Physical Interactions

In a virtual environment for training, the user's (learner and teacher) physical environment must be realistic, interactive and act in "real-time". Then, to reach the constraints of virtual reality, models we use to simulate physical phenomena are obviously simplified. Therefore, the teacher may want, for pedagogical reasons, to inhibit some phenomena. For that, we must propose models that are compatible to a disconnection between the phenomena. Moreover, although all interactions have potential effects on the two agents involved, we consider that the interactions between agents have a privileged direction.

Then, the reactive agents' behaviour evolving in physical environment is to perceive situations and to act consequently. A practical limit of the individual based model is that each agent can potentially perceive all others. The complexity of the algorithm is in this case $O(n^2)$. Then, we have to design rules to organise these interactions between reactive agents. For that, we use the generic organisational model we have proposed before. In this case the organisation is a network where agents are connected together when they are in interaction. We call this organisation an interaction network (InteractionNet, Figure 3). To represent the concept of privileged direction in interactions, we define two particular roles called source and target. The goal of source agents is to give information on their internal states to other agents (targets) so that they can compute the interaction's "strength" and their internal state. The interaction can be detected by the two agents involved, but, for "real-time" computation reasons, it is better if only one agent detects it (one of two agents or another one else). We then define a recruiting role, which has the responsibility to maintain the knowledge of each agent upon the structure of the organisation. This means that an agent playing this role has to detect the interaction situations. The internal architecture of reactive agents matches the constraint of physical phenomena disconnection presented before, because an agent can have several reactive behaviours, each one participating in a different interaction network. This elementary behaviour (see ReactiveBehaviour class in Figure 3) consists in the computation of a vector of internal state variables after the evaluation of inputs (from the interactions where the agent is a target) and presents a pertinent external representation of its state (output) to other agents (potentially targets of an interaction where the former agent is a source).

Social Interactions

The environment is also populated by more "intelligent" agents representing humans. They are undergoing the environment and acting on it as reactive agents, but the way they choose their actions is carried out on a higher level of abstrac-

Figure 3. Interactions network

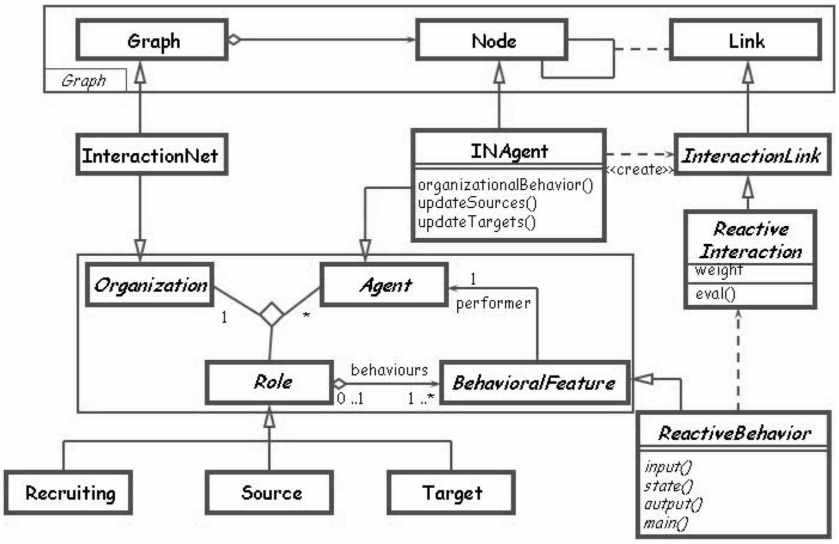
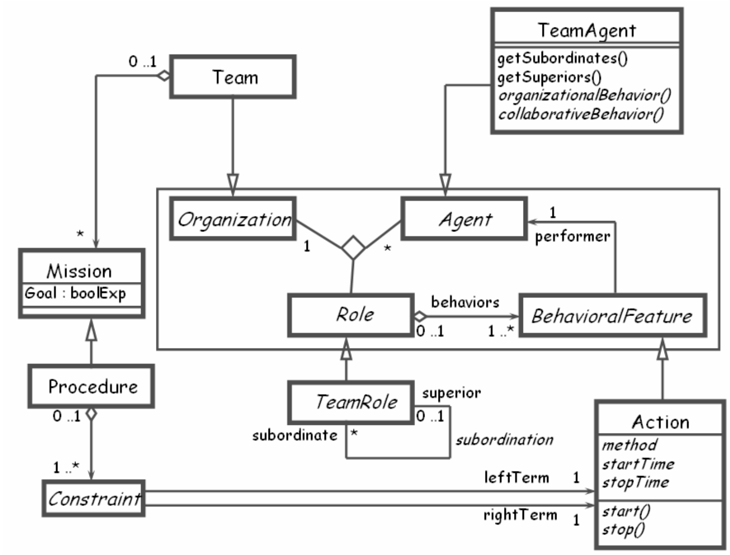


Figure 4. Team model



tion. Those agents are various humans involved in the formation (learners, teachers) who are played by autonomous agents. In our case, the social environment is structured and each member knows its roles and those of its partners. In a specific organisation the interactions between the team members are also structured by the mean of domain specific procedures known by all members. We thus derive our generic organisational model to formalise this concept of team (Figure 4). We are interested in the case where the action's coordination between team members is already stated and written in procedure. On the other hand, the environment being dynamic, agents can need to adapt the scenario to the environment. The procedure must then have a semantic representation so that agents can reason above. Then we use a high level language to describe a procedure (Allen temporal logic).

The reasoning of team members relates on organisation, procedures and actions. We propose a model of agent having local organisational knowledge. A rational agent is divided into a decisional part and a part represented by modules of perception of the physical environment, communication and actions (Figure 5). It must carry out actions of the procedure and adapt to situations not envisaged. The procedure describes interactions between agents in an optimal situation, and leaves to the agent the responsibility to build implicit plans (not clarified in the procedure) considered as "natural" within an applicative situation. Moreover, the procedure coordinates actions of a semantic level which we call "actions trades" such as "sprinkling a fire" in the case of firefighter procedures, whereas the implicit plans arrange actions of a generic semantic level for humans such as "going at a point". For that, the agent must be able to reason on actions and we propose a model of goal directed actions having pre-conditions and post-conditions. Thus, before carrying out an action, the agent must make sure that pre-conditions of this one are checked. If it is not the case, it builds itself a plan by back chaining on pre-conditions and post-conditions of actions. When an agent starts or stops actions, it broadcasts a message that enables other members to update their knowledge on the evolution of the procedure. When this behaviour failed, the agent calls its organisational behaviour, which can help it to find a solution with another team member. Thus, in a hierarchical organisation, when an agent has a problem that it cannot solve, it refers to its superior. Then, the superior has the responsibility to find a solution among its subordinates (if it does not find any, it refers to its own superior about it). We represent this mechanism by a method like a *Contract Net Protocol.*

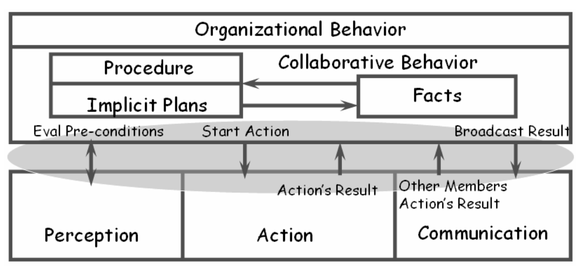
Users Interactions

The avatar in *MASCARET* is not only the representation of the user but has also its own behaviour (reactive and rational). Therefore the avatar model in *MASCARET* is the same as rational agent. In order to provide the decision making responsibilities to the user (student) the link between the collaborative behaviour and the action module (Start Action message) can be inhibited. This inhibition is dynamic, meaning a learner can take the control of an autonomous agent during the simulation. It becomes then his or her avatar. At any time, the user can release control on the avatar. Their adaptive behaviour is supported by the dynamics of roles attribution between the user and his or her avatar (see mediation organisation Figure 2).

All modules composing the avatar are still active and thus remain potentially usable. The Action's Result message provides information to the Facts database. The avatar has the knowledge on the action historic and on the current action running. In addition, the avatar is still informed about result of other member's actions and still informs them about action done by the user. The knowledge on the evolution of the collaborative procedure is still consistent.

As the collaborative behaviour of the avatar is inhibited, it does not call any more the organisational behaviour (in case of an action failure). It becomes the responsibilities of the user to find a solution.

Figure 5. Architecture of rational agents



Management of ITS Models

Our objective is to integrate a differentiated pedagogy based upon context. Intelligent tutoring system (ITS) presented earlier aim at providing students with dedicated tutoring. In *MASCARET,* the information presented by classical ITS models are accessible. The goal of this paragraph is to show how we can represent the four models of an ITS. In *MASCARET*, the models are seen as bases of knowledge constructed and manipulated by agents or humans (human or virtual).

The Domain Model

The expert model represents the knowledge that the learner has to acquire. In our case this knowledge is essentially procedural, but also *"know-how"* to act in the environment. Therefore the domain model presents knowledge on the social and physical environment. In order to represent this knowledge, this model is based upon the *Team* and *InteractionNet* models proposed by *MASCARET.* This means that comparing to models presented in the ITS overview section, our formalism is based on a logical representation (action pre-conditions and goals), but also on a graph representation to explicit links between the actions of roles and interaction between the physical entities.

The expert model is not represented as a special class but is represented by the set of organisations. But in no way must the expert model reference each instance of organisation. It only refers to the knowledge on organisations structures. An agent asking information to the agent maintaining the expert model never asks information in a specific context (*"which action the user can do now?"*), which is the responsibility of agents maintaining the user model for example, but asks question in a generic case (*"what actions can be done after this action in this type of organisation?"*). In that way the reification of the concept of organisation is very important.

Then, about the social environment, the agent maintaining knowledge on the expert model is able to answer questions about roles, actions and procedures. This information is written by the expert according to models proposed by *MASCARET.* Therefore, as STEVE for example, the expert model is able to inform on pre-conditions or goal of actions, the order of those actions and responsibilities of each intervenient. The way this information is used depends on other agent behaviour. For example an agent can ask information on the goal of actions to explain it to the learner.

For the physical environment, the expert model maintains information on casual links between the different roles (sources, targets …) intervening in physical phenomena. This knowledge is written by the expert and the application designer. An agent can then ask information, for example what can change the direction of the lick of gas and have the answer that it is an agent playing role "source" in a "gas propagation" interaction net.

Errors Model

As in some ITS presented by Py (1998), we consider errors as crucial information. Therefore we decided to introduce the errors model in our ITS. An error model is a knowledge base on classical errors done by learners. It can be compared to the *buggy model* presented earlier; its goal is to help to identify the error and determine the reason for this error. This information is written by the teacher and takes the form of the rule: "usually students do action C after action A; it is an error because something," where "something" is classically a domain-specific rule.

Using the organisation model of *MASCARET,* a pedagogical agent can detect errors on procedure ("not the right action or not done by the right team member"). The responsibility of the agent managing errors is then to exploit its knowledge about typical errors (the left part of the rule) in order to give more accurate information on the error (the right part of the rule). Therefore the tutor can perform a particular action, for example explain the "something".

The agent managing errors can record information about error occurrences. This information may be used to enrich the knowledge about typical errors. It is under the control of the human teacher; in the future, the agent might learn it.

Learner Model

As seen earlier, the learner model could be divided on psychological information, curriculum, and learner current actions and state.

In *MASCARET,* a special agent maintains information on the learner current state and action. This agent is the avatar agent; it maintains knowledge on the actions done by the user, which roles he or she plays in which organisations. In that way it can assimilate to an *overlay method* because it contains a subset of the expert model. Moreover, the avatar can plays roles in physical phenomena (InteractionNet); then it maintains also this knowledge and can inform on the influences the user is currently overcoming.

We could have decided to give its pedagogical autonomy to this avatar. In this case, the avatar could decide to explain/do the next action or decide not to overcome some physical phenomena for pedagogical reasons. But we prefer to give the avatar its pedagogical responsibilities by the means of pedagogical roles in pedagogical organisation and then let the teacher express its pedagogical rules which is the role of the pedagogical model.

Psychological criteria of the learner are, for example, student's emotion state or level. In *MASCARET,* this information is not yet accessible, but we are planning to work on it, based on Kermarrec's (2002) works.

Interface Model

This model permits the communication between the system and the user. It presents information to the student and detects his or her actions. Virtual environment for training has the specificity to tend to be immersive. Therefore the user is supposed to perceive and to act as naturally as possible. In *MASCARET,* this model has not been represented yet. But we can consider that we have the same problems as Lourdeaux et al. (2001), whose work is addressed to decision making and not technical gesture. This method is based on virtual behaviour primitives. This means the correspondence between values of sensors (FOB, mouse, gloves…) with actions (walk, take…) and the procedure knowledge.

We also want to detect the intention of the user to act differently according to the difference between the user actions and intention. We consider that the user verbalises his or her intentions and then they can be detected by the system with voice recognition.

Pedagogical Model

The pedagogical model defined pedagogical strategies issued from psychological and didactic research. We are particularly interested in collaborative strategies. Therefore the pedagogical model is based on knowledge about roles (pedagogical actions), interaction (exchange of information about the different ITS models) and organisation (action coordination to reach the shared pedagogical goal). To our point of view, the pedagogical model shares characteristics of the domain model, where the domain is "pedagogy". As in the former case, general knowledge about pedagogical organisation is needed, instead of particular configuration of a training session. The pedagogical organisation structure is explained in the next section.

Pedagogical Interactions

Pedagogical agents are defined as agents playing pedagogical roles. In MASCARET, potentially any agent can hold pedagogical skills. So these agents may have the following representations:

1. an autonomous artificial agent, which can have or not a representation in the virtual environment;
2. an avatar of a teacher playing no roles in the social environment of the trainees;
3. an avatar of a user (teacher or trainee) playing an active role in the environment (in this case the agent has three types of roles: domain specific, proxy of the user and pedagogical ones);
4. an autonomous artificial agent simulating a human or a physical object (it is part of the environment and also exhibits pedagogical skills).

The contribution of these agents to the different ITS models presented in this chapter is obviously not the same. The first category of agents corresponds typically to tutors of traditional ITSs: they maintain information about the ITS models and communicate it to other pedagogical agents. The teacher can be personified in the environment in order to facilitate the communication with learners. The contribution of the learner avatar is important because it can gather information about the learner (user-avatar interactions). As the avatar participates also to social organisation it holds information about the domain (procedures to be performed). A pedagogical agent participating to the realisation of a collaborative procedure can play pedagogical roles as a companion or a troublemaker; in this way its behaviour is part of the pedagogical model. Physical objects of the environment are knowledge items of the domain model; they constitute also elements of the interface model (to control the way information is presented to the learner and the type of behaviour they can trigger in reaction to a user action in a particular pedagogical context); finally, to a certain extend they can also contribute to the pedagogical strategy: they can perform actions like to inform the user about its structure or its potential behaviour. Generally, these agents are mostly reactive

Figure 6. Pedagogical agent/role/actions

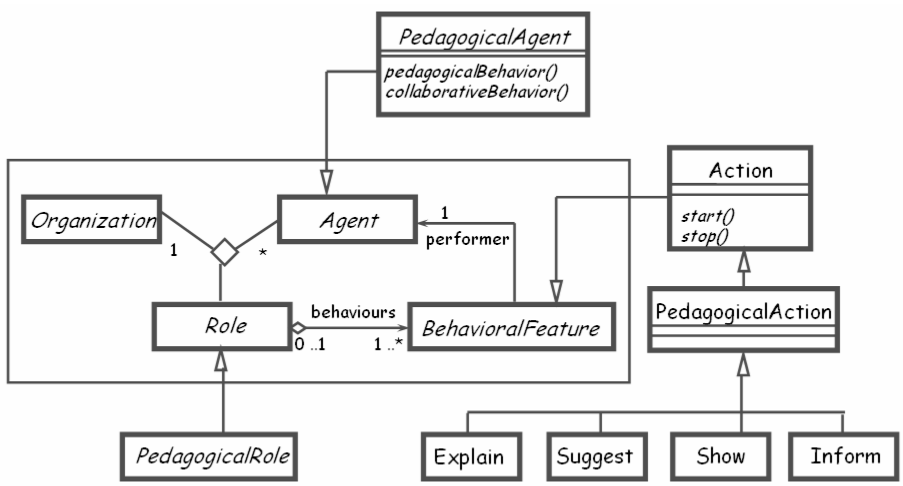
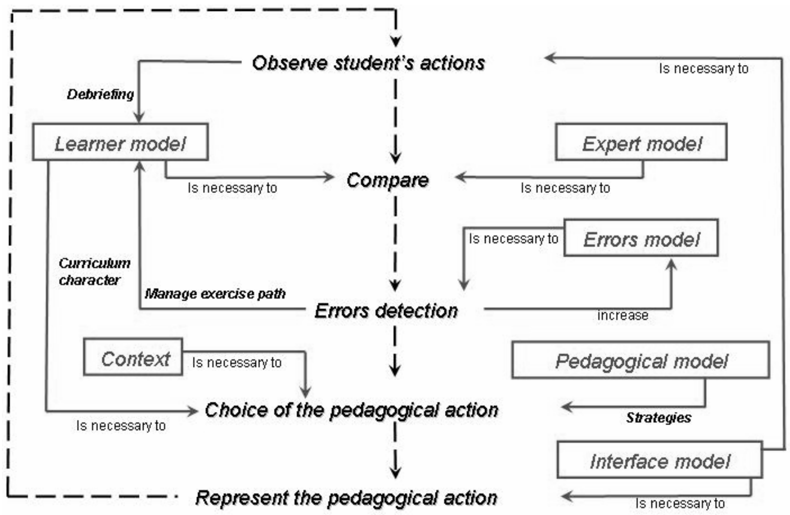


Figure 7. Pedagogical process



as proactive agents and the pedagogical strategy is under the responsibility of deliberative agents like artificial tutors.

Every pedagogical agent shares the same goal: to increase the student's skills. Different strategies may be applied to reach this goal. The pedagogical organisation defines the different roles that have to be played in a particular pedagogical context, that is, the sets of pedagogical action to be performed. Figure 6 gives examples of pedagogical actions. Notice that each action can contribute to a different objective.

The pedagogical action Suggest informs the user about the next action. It provides goal and pre-conditions on the next action. The pedagogical action Show simulates such action. The pedagogical action Explain provides current action goal.

Strategies are defined by roles, corresponding to sub-goals. For example, the goal of a disturbing strategy is to suggest solutions that can be erroneous (Chan & Baskin, 2000). It is a means to force the learner to evaluate his/her self-confidence in his or her own solutions. For example this strategy may consist to modify the orientation of the wind in order to show up gas propagation. Disturbing the learner is achieved by modifying the behaviour of the agents playing the Source role in this InteractionNet.

Using the four models, a pedagogical multi-agent system can help students with dedicated tutoring. We proposed an overall process resulting in interactions (data-flows) between the five ITS models (Figure 7). The pedagogical process, which constitutes the resulting behaviour of the MAS, is a five-step cycle. First, a pedagogical agent observes the student's action using the interface model. The avatar of the student has the knowledge on the current running action. Such observation permits updating the learner model.

Second, we compare the expert model to the learner model. That way, we are able to detect an error. For example, if the goal of an exercise is to respect a procedure. The student achieves action A and starts action C. The expert model provides information on the procedure. Therefore we know that after action A the student should start action B. The comparison detects an error corresponding to a layout.

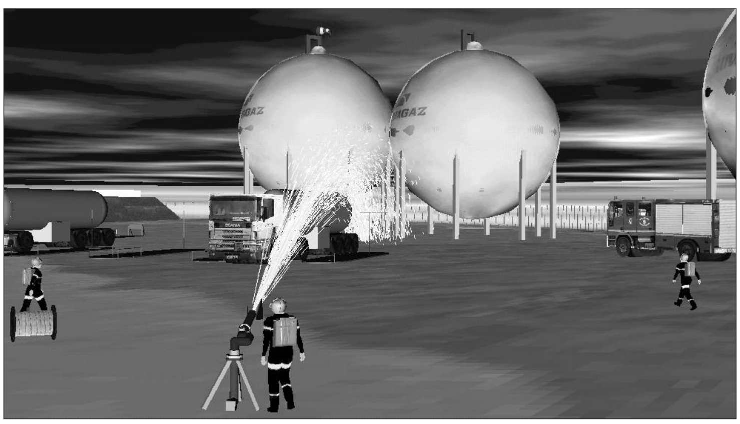
The error detection uses the error model and updates the learner model. In addition, we can increase the errors model. According to information on the error and using the pedagogical model (strategies), the learner model (level, emotion state ...) and the Context (Figure 7), the pedagogical agent selects a pedagogical action. Finally, the pedagogical action selected is represented to the student using the interface model.

SÉCURÉVI

SÉCURÉVI (security and virtual reality) is an application of *MASCARET* to civil safety (Figure 8). It is dedicated to the training of fire-fighter officers for operational management and for commandment. A complete description of *SÉCURÉVI* is presented in Querrec, Buche, Maffre and Chevaillier (2003). In a typical exercise, gas lick in an industrial site, the physical environment is constituted of the site where exercises take place as well as physical phenomena (fire, smoked, water jets...) being able to intervene. The application designer's work in *SÉCURÉVI* is essentially to implement elements of physical and social environment by inheriting *MASCARET.* Thus, the application designer has to conceive his or her own agents to simulate a specific phenomenon. This is possible by inheriting INAgent as well as its reactive behaviour (inheriting Reactive Behaviour). Then he or she defines the interactions networks (InteractionNet). The application designer of social environment composed of FPT teams (Fire engine Thunders Pumps) in charge of the incident attack or CMIC team (the Intervention Chemical Movable Cell) follows the same path by inheriting model of social environment proposed by *MASCARET.* Thus, the application designer has to describe new teams and roles as well as actions that agents have to perform. *SÉCURÉVI* is implemented using the platform *AREVI/ORIS* (Harrouet, Tisseau, Reignier & Chevaillier, 2002).

The domain model consists first of knowledge on FPT and CMIC team structure. Each type of organisation is structured by four or five roles and up to 20 procedures. The learners play roles of leaders intervening in those teams during the incident. Teachers can participate in the simulation to trigger malfunctions, help the learners or play a role in a team. Then we can create two mediation organisations, one for the learner and the other for the teacher. In each one, the user avatar intervenes. In the learner mediation organisation, the role of the learner is to choose the action he or she has to do according to the action done by the

Figure 8. Picture from SécuRéVi



other team members. The role of the avatar is to realise in the environment the action ordered by the learner and to overcome some of the physical phenomena in the environment. In that way the avatar knows student historic and current action running. Therefore the avatar plays a role in a pedagogical organisation where he or she has the responsibility to maintain the student model. As the avatar is also in charge of detecting a learner's action, it has the responsibility to manage the interface model. In fact, this agent has information about roles and actions played by the learner. It can, using communicating with the agent maintaining the domain model, have information about all actions done by the learner. Then, according to pedagogical rules, it can choose to propose an interface where the learner has to verbalise the next action or select in a list constructed from the information retrieved by the avatar. But this agent is not the one to manage this interface model; another one can propose information on the environment. For example, the agent could display elements that are not visible in the real world as wind curve or gas cloud, in order to specify to the student specific conditions.

The teacher can also intervene in the simulation, by the mean of his or her avatar for example to play a role in the same team of the user. In addition, he or she can participate to a pedagogical organisation where he/she has the responsibility to decide which pedagogical action to do, and another agent, not visible in the environment realises those pedagogical actions. For example the teacher may want to disturb the learner by modifying the environment. Then, if the mission goal is to stop gas propagation, disturbing consists of showing up gas propagation. That means an agent will ask the agent maintaining the domain model the roles that can modify this phenomenon. Then it will ask the agents playing these roles to change their behaviour to increase the wind power or show the wind direction.

CONCLUSION AND FUTURE WORK

Considering a virtual environment for training as a multi-agent system, we propose the *MASCARET* model. It provides a framework to design multi-agent systems dedicated to collaborative, adaptive and realistic environment for training. This VET aims to put trainees in operational position by simulating their physical and social environment in such a way that they can learn to perform some collective tasks. These tasks are described as sets of actions, which define roles, allocated to agents representing human actors. We stated that the system is adaptive in the way that the set of instantiated agents, roles allocation and agent decisional rules are not imposed upon pedagogical designer: all these elements can be dynamically defined in order to adapt pedagogy to learner profiles and pedagogical objectives. Not only has the behaviour of pedagogical agent to be adaptive but the physical environment ones too: it is necessary to remain in control of reactive objects manipulated by learners, or having influence on them, and more precisely of the type of physical phenomena to be activated or inhibited. Physical environment behaviour has to be controlled but must remain consistent in order to be intelligible to learners.

We stressed the point that information about potential interactions and action realisation are a major issue in the management of pedagogical knowledge, not only for teachers but also for learners. This knowledge is based on a piece of information about organisations that structure interactions. We have made no assumption about the nature of these interactions and the way agents perform their actions because we think that this issue is still open. We agree that it will be helpful to propose solutions concerning this point to VET designers. It is the major objective of our future works.

We have shown how to take into account the four classical ITS models (enriched by the error model) in an agent-based VET. A key point is that knowledge about these models is distributed among agents that perceive actively their environment and exchange information, according to the pedagogical role they play. The resulting behaviour of the multi-agent system defines the pedagogical process we propose.

Our objective is that *MASCARET* will be multi-strategic VET, as in Mengelle and Frasson (1996). It will be achieved by the specification of different pedagogical roles. First, we are interested by the tutor role. Using the notion of procedure as described in the social environment, the agents playing this role can provide deductive reasoning and explications. Second, we propose the use of the role of companion (Chan & Baskin, 2000). The companion is a virtual actor that will cooperate for the realisation of tasks, exchange ideas on the problem and share the learner's goals. We are also interested in the role of troublemaker (Aïmeur et al., 2000), whose goal is to disturb the learner by proposing solutions that can sometimes be erroneous.

Finally, we want to provide to our system the possibility to adapt pedagogical behaviour to a specific student. In this option,, the choice of pedagogical actions will be more adaptive. Due to the number of input variable and pedagogical rules, we envisage the use of machine learning techniques as classifier systems (Wilson, 1994).

REFERENCES

Aïmeur, E., Frasson, C., & Dufort, H. (2000). Cooperative learning strategies for intelligent tutoring systems. *Applied Artificial Intelligence, 14,* 465-489.

Anderson, J.R. (1988). The expert module. In M.C. Polson & J.J. Richardson (Eds.), *Foundations of intelligent tutoring systems* (pp. 21-53). Hillsdale, NJ: Erlbaum.

Anderson, J.R. (1990). Analysis of student performance with the lisp tutor. In A.L.M. S.N. Fredericksen & R. Glaser (Eds.), *Diagnostic monitoring of skill and knowledge acquisition* (pp. 27-50).

Anderson, J.R., & Reiser, B.J. (1985). The lisp tutor. *Byte, 10,* 159-175.

Brown, J.S., & Burton, R.R. (1978). A paradigmatic example of an artificially intelligent instructional system. *Int. Journal of Man-Machine Studies, 10,* 323-339.

Burdea, G., & Coiffet, P. (1993). *Virtual Reality Technology.* Wiley Interscience.

Burkhardt, J.-M. (2003). Réalité virtuelle et ergonomie: Quelques apports réciproques. *Le Travail Humain*, *66*(1), 65-91.

Burns, H.L., & Capps, C.G. (1988). Foundations of intelligent tutoring systems: An introduction. In M.C. Polson & J.J. Richardson (Eds.), *Foundations of intelligent tutoring systems* (pp.1-19). Hillsdale, NJ: Erlbaum.

Burton, R. (1982). Diagnosing bugs in a simple procedural skill. Intelligent tutoring systems. In D. Sleeman & J. Brown (Eds.), *Intelligent tutoring systems.*

Burton, R., & Brown, J. (1982). An investigation of computer coaching for informal learning activities. In D. Sleeman & B. Brown (Eds.), *Intelligent tutoring systems.* Academic Press.

Chan, T., & Baskin, A. (2000). Learning compagnion systems. In C. Frasson & G. Gauthier (Eds.), *Intelligent tutoring systems: At the crossroad of artificial intelligence and education* (pp. 159-167). Bristol.

Clancey, W. (1983). Guidon. *Journal of Computer-Based Instruction*, *10*(1), 8-14.

Clancey, W. (2002). Simulating activities: Relating motives, deliberation and attentive coordination. *Cognitive Systems Research*, *3*(3).

Delestre, N. (2000). *Metadyne, un hypermédia adaptatif dynamique pour l'enseignement.* PhD Thesis, University of Rouen.

Demazeau, Y. (1995). From interactions to collective behaviour in agent-based systems. *The European Conference on Cognitive Science* (pp. 117-132). Saint Malo.

Fahlman, S. (1979). *Netl: A System for Representing and Using Real-World Knowledge.* Cambridge, MA: MIT Press.

Ferber, J., & Gutknecht, O. (1998). A meta-model for the analysis and design of organizations in multi-agent systems. *Third International Conference on Multi-Agent Systems* (pp. 91-105).

Halff, H.M. (1988). Curriculum and instruction in automated tutors. In M.C. Polson & J.J. Richardson (Eds.), *Foundations of intelligent tutoring systems* (pp. 79-108). Hillsdale, NJ: Erlbaum.

Hannoun, M., Boissier, O., Sichman, J., & Sayettat, C. (1999). MOISE: An organizational model for multi-agent systems. *IBERAMIA-SBIA 2000* (pp. 156-165).

Harrouet, F., Tisseau, J., Reignier, P., & Chevaillier, P. (2002). oRis: Un environnement de simulation interactive multi-agents. *Revue des Sciences et Technologie de l'Information, série Technique et Science Informatiques, 21*(4), 499-524.

Kermarrec, G. (2002). *Stratégies d'apprentissage et autorégulation en EPS, une recherché descriptive en contexte scolaire.* PhD Thesis, University of Rennes II.

Leman, S., Marcenac, P., & Giroux, S. (1996). Un modèle multi-agents de l'apprenant. *Sciences et Techniques Educatives, 3*(4), 465-483.

Lourdeaux, D., Fuchs, P., & Burkhardt, J.-M. (2001). An intelligent tutorial agent for training virtual environments. *Fifth World Multiconference on Systemics, Cybernetics and Informatics,* Orlando, USA.

Mengelle, T., & Frasson, C. (1996). A multi-agent architecture for an ITS with multiple strategies. *CALISCE* (pp. 96-104).

Miller, J.R. (1988). The role of human-computer interaction in intelligent tutoring systems. In M.C. Polson & J.J. Richardson (Eds.), *Foundations of intelligent tutoring systems* (pp. 143-189). Hillsdale, NJ: Erlbaum.

Minsky, M. (1975). A framework for representing knowledge. In P.H. Winston (Ed.), *The psychology of computer vision* (pp. 211-277). NewYork: McGraw-Hill.

Nkambou, G., Gauthier, R., & Frasson, C. (1997). Un modèle de représentation des connaissances relatives au contenu dans un système tutoriel intelligent. *Sciences et Techniques Educatives, 4*(3), 299-330.

Piaget, J. (1978). *Behavior and Evolution.* New York: Pantheon Books.

Py, D. (1998). Quelques méthodes d'intelligence artificielle pour la modélisation de l'élève. *Sciences et Techniques Educatives, 5*(2).

Querrec, R., Buche, C., Maffre, E., & Chevaillier, P. (2003). SécuRéVi: Virtual environments for fire-fighting training. In S. Richir, P. Richard & B. Taravel (Eds.), *Fifth Virtual Reality International Conference* (pp. 169-175).

Quillian, M.R. (1968). Semantic memory. In M. Minsky (Ed.), *Semantic information processing* (pp. 227-270). MIT Press.

Rickel, J., & Johnson, L. (1999). Animated agents for procedural training in virtual reality: Perception, cognition, and motor control. *Applied Artificial Intelligence, 13.*

Self, J. (1988). Artificial intelligence tools in education. In P.E. R. Lewis (Ed.) (pp. 73-85). North-

Holland: Elsevier Science Publishers B.V.

Shute, V. (1990). Rose garden promises of intelligent tutoring systems: Blossom or thorn? *Space Operations and Research (SOAR) Symposium*.

Sowa, J.F. (1984). *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley.

Stansfield, J., Carr, B., & Goldstein, I. (1976). *Wumpus advisor 1: A first implementation of a program that tutors logical and probabilistic reasoning skills* (Tech. Rep.). MIT, Artificial Intelligence Laboratory.

Tisseau, J. (2001). *Virtual Reality: In virtuo autonomy* [Accreditation to Direct Research]. University of Rennes I.

VanLehn, K. (1988). Student modeling. In M.C. Polson & J.J. Richardson (Eds.), *Foundations of intelligent tutoring systems* (pp. 55-78). Hillsdale, NJ: Erlbaum.

Vigano, G., Mottura, S., Calabi, D., & Sacco, M. (2003). The virtual reality design tool: Case studies and interfacing open topics. *Virtual Concept 2003* (pp. 364-371).

Wenger, E. (1987). *Artificial intelligence and tutoring systems.* Morgan Kaufmann.

Wilson, S.W. (1994). ZCS: A zeroth level classifier system. *Evolutionary Computation, 2*(1), 1-18.

Woolf, B.P. (1992). Building knowledge based tutors. In I. Tomek (Ed.), *Fourth International Conference of Computer Assisted Learning* (pp. 46-60). Berlin, Heidelberg: Springer.

This work was previously published in the International Journal of Distance Education Technologies, Vol. 2, No. 4, pp. 41-61, copyright 2004 by Idea Group Publishing (an imprint of IGI Global).