



HAL
open science

CHRYSAOR: an Agent-Based Intelligent Tutoring System in Virtual Environment

Frédéric Le Corre, Caroline Fauvel, Charlotte Hoareau, Ronan Querrec,
Cédric Buche

► **To cite this version:**

Frédéric Le Corre, Caroline Fauvel, Charlotte Hoareau, Ronan Querrec, Cédric Buche. CHRYSAOR: an Agent-Based Intelligent Tutoring System in Virtual Environment. International Conference on Virtual Learning, Nov 2012, Brasov, Romania. pp.39-45. hal-00784302

HAL Id: hal-00784302

<https://hal.science/hal-00784302v1>

Submitted on 4 Feb 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

CHRYSAOR: an Agent-Based Intelligent Tutoring System in Virtual Environment

Frédéric Le Corre, Caroline Fauvel, Charlotte Hoareau, Ronan Querrec and Cédric Buche

UEB/ENIB/CERV, 25 rue Claude Chappe F-29490 Plouzané France
{lecorre, fauvel, hoareau, querrec, buche}@enib.fr

Abstract. The various existing Intelligent Tutoring Systems (ITS) models do not capitalize on all the possibilities permitted by the use of virtual reality. In this paper, we first establish the important characteristics of ITS (genericity, modularity, individualization, scenario edition, adaptativity). Subsequently we present our studies using an agent metamodel (BEHAVE) based on an environment metamodel (VEHA), in order to make a generic ITS. We focus on describing our agent model and its knowledge of the pedagogical situation and incorporate a pedagogical scenario model in our ITS. The use of this ITS is illustrated by an application of a virtual biomedical analyzer which enables to learn the technical procedures of the device.

Keywords: intelligent tutoring system, agent, metamodel, virtual environment, genericity, pedagogical scenario

1 Introduction

In the biomedical domain, there is a need to train people for the use and maintenance of diagnostic analytical devices. In the traditional training method, several students in a classroom with a live instructor manipulate the device alternately. Unfortunately in the biomedical industries, many new employees do not go through the classical training program. Moreover, these industries can not provide their devices for the employees training. We aim to use virtual reality and virtual environments in order to provide more freedom to users during training programs. For instance, using virtual reality, learners can train when they want, and where they want without any constraint.

Virtual reality and virtual environment technologies are now widely used in the fields of training and education. Virtual environments can be combined with **Intelligent Tutoring System** (ITS) in order to adapt the learning situation depending on the learner activities [5]. In the literature there are plenty of ITS models [17, 11, 15], however they seem incomplete. One of the main lacks is the genericity, which means that we need to modify the ITS as soon as we change the environment [5]. For example, modifications of the ITS will be needed everytime we change the device or the exercise. Another lack is the possibility for the teacher to build the training by adding the concepts of objectives and prerequisites. These concepts can be provided by a pedagogical scenario. From our point of view, genericity is a major characteristic of the ITS, but it is not the only one. Indeed, ITS must have several important

The objective of this paper is to propose the most complete ITS called **Chrysaor**. This paper is organized in five sections. In section 2 we determine the necessary characteristics of an ITS and listed some ITS having these characteristics. In section 3 we expose the

first studies made on our generic ITS. These studies focus on using multi-agent system with the objective of endowing the ITS with modularity, which is a part of genericity, and incorporate a scenario model [10]. In order to test our generic ITS we have made an application based on the biomedical devices: **VirtualAnalyzer**, presented in section 4. Its purpose is to train people to use a biomedical analyzer. At least, we conclude, in section 5, by listing the characteristics of **Chrysaor** that needed a further improvement and detail the terms of an incoming experimentation. It is implemented using AReVi and runs in real-time.

2 Related work

The goal of this section is to find out the best ITS. In section 2.1 we present the most important characteristics that define an ITS. In section 2.2 we list the existing ITS based on these characteristics and try to find the most complete one in order to based our own ITS on it.

2.1 Major characteristics

As mentioned above, **genericity** is a major characteristic needed for our ITS. However, the actual methodology of ITS engineering is not ideal. Indeed, each application is developed independently, tutoring expertise is hard-coded into individual applications, and there is a little reuse of tutoring components [13]. Moreover, in computer environments for human training, some components are hardly reused in different simulations. Such is a case with HAL [?], in which pedagogical model assistances must be reformulate for every exercises. It therefore seems essential to set our ITS as much generic as possible.

Some elements, such as the significance of the pedagogical strategy, which can substantially varies according to the way the tutor want to intervene, when and how intervene [19], seem quite interesting. Based on this, we may want to add a disruptive [2] or a companion-component [11]. In the same way, some ITS behaviors could be replaced by a human contribution. In order to obtain this freedom to arrange the pedagogical strategies by adding or replacing some behaviors, we need some kind of **modularity** of our ITS behaviors. Modularity permits to increase the genericity of the ITS. The multi-agent systems enable components, structures and occurred interactions-diversity. The use of agents in virtual environments for human training, such as applications like **SECUREVI** [?], **STEVE** [?] or **Jacob** [?], has already shown its benefits.

In order to have a better efficiency, we want suitable assistances for each learner. So we have to work on the **individualization** of the ITS. The classical ITS are composed of a learner model which represents the learner's knowledge at the problem level [18]. This model has to contain a representation of the learner's profile, initialized and updated by an external device (e.g. a questionnaire) or by the learner/environment interactions directly. Learner modeling is a well known difficult problem, Self [?] proposed a clear explanation of its role and usage. Therefore our ITS need to have a learner model in order to individualize the assistances.

However the proposed assistances may not be ideal from the instructor point of view. Therefore we wish that our ITS can modify his behavior depending on what happens in the simulation. The ITS reasoning processes must self-adapt in order to take past experience into account. Thus, one of our objectives is that our ITS, along with its past experience, could automatically suggest the appropriate interventions by taking into account both

the learner and the context of the simulation: the system therefore becomes **adaptive** [5].

Another of our goal is to enable the teacher to customize each exercise. He might have the possibility to write a **pedagogical scenario**. This scenario would be a knowledge base connected to a single exercise. In virtual environments for education, we go from an outline where the tutor lecture (traditional education) to an outline where the knowledge presentation is done by the device while the tutor task is to organize and capitalize on interactions between the learner and the learning device. In the domain of informatic environment for humans training, there is a lot of researches about the pedagogical scenarios problematic. One of the most significant proposal of the community is the IMS Learning Design (or IMS-LD) standard [8]. In ISM-LD a scenario is considered as a sequence of pedagogical activities. Lately, the pedagogical scenario model called **Poseidon** was proposed. It enables a description of all of the components in pedagogical scenarios including activities in virtual environment. **Poseidon** enhances existing models and particularly IMS-LD within the framework of virtual environments for human training.

For these reasons, several characteristics seems important in the conception of an ITS: individualization, modularity, genericity, adaptativity, scenarios. Now, we must find some ITS related with these characteristics.

2.2 Existing ITS

In the ITS domain, many research works were made during the past few years and most of them focus on two or three characteristics described in section 2.1 (Table 1.).

| | Genericity | Modularity | Individualization | Scenario | Adaptativity |
|----------------|------------|------------|-------------------|----------|--------------|
| [1] | ✗ | ✗ | ✓ | ✗ | ✗ |
| [6] (METISSE) | ✗ | ✗ | ✓ | ✗ | ✗ |
| [9, 4] | ✗ | ✓ | ✓ | ✗ | ✗ |
| [14] (MAEVIF) | ✓ | ✓ | ✓ | ✗ | ✗ |
| [16] | ✓ | ✗ | ✓ | ✗ | ✗ |
| [17] (SimCore) | ✓ | ✗ | ✗ | ✓ | ✗ |
| [15] | ✗ | ✗ | ✓ | ✗ | ✓ |
| [5] (PEGASE) | ✓ | ✓ / ✗ | ✓ / ✗ | ✗ | ✓ |

Table 1. Comparison of current ITS

Over the past five years, many projects aim at developing the genericity part of ITS [14, 16, 17] or the modularity part by using multi-agent system technology [9, 11]. Some projects aim at individualizing the simulation for each learner. This could be done for example through emotional agents [1] or by using the Hollnagel classification [6]. At lower scale, some have highlighted the adaptativity characteristic of ITS [15] while others offer the possibility to create and change some scenarios like Marion [10] and Sorensen [17]. However, none of this ITS contain all of the five characteristics defined previously. There is another ITS called **Pegase**, which is based on the four classical models (expert, learner, pedagogical, interface) [?]. **Pegase** also contains an error model and a definition of the pedagogical model (usable independently of the exercise to do) in order to generalize it. One last important part of **Pegase** is its adaptivity allow by the auto-modification of the pedagogical model. **Pegase** is the most complete ITS with all these characteristics, so we have decided to based our own ITS on it. Nevertheless it lacks the pedagogical scenario

characteristic and some other can be improve, like the modularity. In the next section, we expose the improvement made in **Chrysaor** model based on the modularity and the pedagogical scenario.

3 Model

In order to provide a pedagogical help for the learner and a pedagogical assistance for the tutor, we aim at integrating an ITS into virtual environments described by the Mascaret metamodel and using the agent metamodel detailed previously. Indeed, we want to improve **Pegase** by endowing it with the five characteristics detailed in section 2 : genericity, modularity, individualization, scenario edition, adaptativity. For the moment, we primarily based our studies on the modularity and the scenario parts of the ITS. This enhancement in modularity improve the genericity too.

The ITS **Pegase** is the most complete ITS according to the five characteristics detailed in the previous section, we have decided to based our own ITS on it. Nevertheless **Pegase** lacks the pedagogical scenario characteristic and some other features can be improve like modularity. To do so, we choose to rely our ITS on MASCARET [12] like **Pegase**, in order to use the multi-agent system technology for the modularity. Moreover MASCARET enables to easily reify the expert or the pedagogical scenario knowledges so that the tutor could reason on it. MASCARET is a metamodel to describe virtual environments and the agents evolving in this environment. This metamodel (founded on the Unified Modeling Language : UML) provide an unified modeling language to describe the structure of the environment (entities, positions...), entitie's and agent's behavior.

3.1 Modularity

Then, we focused on the agents knowledge bases so that it could effect the modularity of **Chrysaor**. Indeed, with our work we can easily add a new knowledge to an agent, which could modify the result of its reasoning.

In order to compensate the lack of genericity, we based our work on the MASCARET metamodel and the agent metamodel [12].

MASCARET is based on many agent concepts like agent communication, behaviors and knowledge bases. In order to improve the modularity of our ITS, we have used the multi-agent system technology so that we could work on these concepts.

3.1.1 Multi-Agent System

We started to transform **Pegase** by providing it the multi-agent system technology. In the previous model, **Pegase** was represented by one unique agent. The pedagogical behavior of this agent was structured around several step (detect an error, propose an assistance, etc.) and set by several datas (Learner model, Pedagogical model, etc.). Using MASCARET and the agent architecture, **Chrysaor** keep this pedagogical behavior but divided in several agents (Error Agent, Pedagogical Agent) and several behaviors (classify the error, propose an assistance, etc.). With this technology we are able to modify one behavior without modifying the global behavior. Moreover, if a specific behavior is not efficient enough, this behavior can be played by a human teacher. Indeed, the main parts of **Pegase**: learner, teacher, expert, compare, error, pedagogical and interface were not really a part of a multi-agent system: all the agents were fixed. With this technology we would be able to easily add or modify an agent or its behaviors, like remove the teacher agent and let the human play this role.

3.1.2 Communication

Afterwards, we based our work on the agent communication, by adding the communication protocol FIPA-ACL (Agent Communication Language) to our system. Due to this standard communication protocol, we can now easily add or remove an agent or communicate with a new application on the network.

3.1.3 Organization

Similarly, we have tried to endow modularity to the agents behaviors. Thus, we have chosen to describe the multi-agent system behaviors with an activity diagram because we based our model on MASCARET which enables to describe behaviors with it. We defined the proceedings and the roles within the organization. Thanks to the multi-agent system technology, we can easily change the organisation of the ITS agents: for example, we may want to add a disruptive or a companion-component. In the same way, some ITS behaviors could be replaced by a human contribution. In order to obtain this freedom to arrange the pedagogical strategies by adding or replacing some behaviors, we have increased the modularity of the ITS.

3.1.4 Knowledge base

Our studies also focus on the agents knowledge base of our ITS. Indeed, the FIPA standard proposes to provide a knowledge base to the agent, but without giving a formalism for the knowledge base. One interest of the knowledge bases is to enable the agents to reason on the datas contained in it. Agent Knowledge base is a crucial point of our ITS model and thankfully many researchs are based on it. In MASCARET, the agents have a knowledge base. In our actual model, the knowledge could be expressed in two models : the expert model (entity, behavior, procedure) and the pedagogical model (agent, behavior, procedure). The OWL ontology is frequently used indepentently of the knowledge bases structures and is often associated with a the communication protocol FIPA. MASCARET provide agents able to perform a model expressed with a syntax based on owl. These two knowledge bases are described on the same language provided by MASCARET which is founded on UML. Consequently, some models expressed in UML could be used as a knowledge base of *Chrysaor* agents. A pedagogical scenario can be described through a procedure, therefore we could use the pedagogical scenario as a knowledge base (Fig. 1).

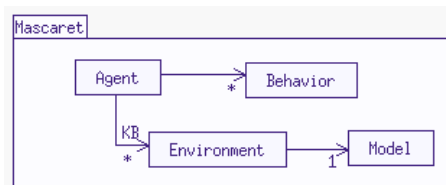


Fig. 1. UML model of the knowledge base of MASCARET

This improvement enables us to easily change the knowledge of each agent. In this context we wanted to have the pedagogical scenario to be a knowledge for the agents.

3.2 Pedagogical Scenario

As seen in section 2, *Chrysaor* lacks of a pedagogical scenario knowledge. Previously, we said that one of the most significant proposal of the community is the IMS-LD standard which focuses on the organization of learning activities. We wanted to use the IMS-LD standard so we have chosen to couple *Poseidon* (a pedagogical scenario model) with *Chrysaor*. *Poseidon* is an implementation of IMS-LD for virtual reality.

First, we aimed to rebuild the structure of *Poseidon* in order to perform it with MASCARET, so it will be considered as a knowledge base for the agents. The main class of *Poseidon* is *LearningSession* (a pedagogical scenario) and includes (Fig. 2):

- a prerequisite list : *Prerequisite*
- an objective list : *LearningObjective*
- an environnement : *EducationalEnvironment*
- the activities of the scenario : *EducationalScenario*
- the pedagogical ressources : *PedagogicalResource* linked to entities : *LearningObject*

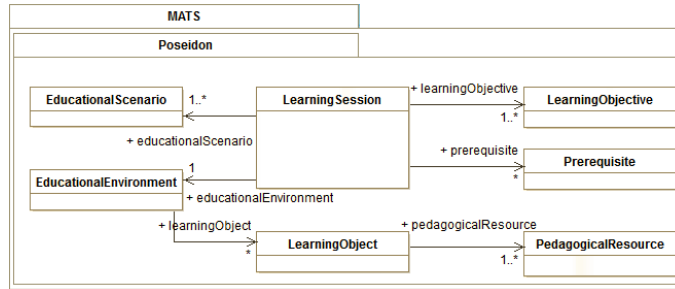


Fig. 2. UML class diagram of the package Poseidon

The *EducationalScenario* inherits from a MASCARET Activity (Fig. 3) and the *EducationalEnvironment* is an instance of a MASCARET Environment (Fig. 4).

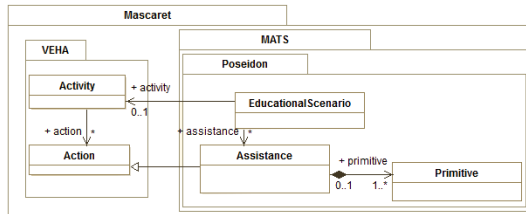


Fig. 3. UML model of Educational Scenario

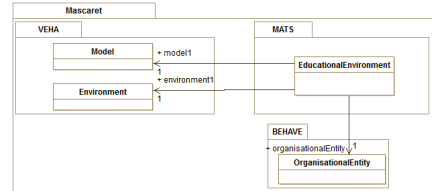


Fig. 4. UML model of Educational Environment

In section 3.1.4, we have seen that MASCARET has been modified and now contain an agent knowledge base which could be an environment. The *EducationalEnvironment* is an instance of a MASCARET *Environment*, thus we can now use the pedagogical datas as an agent knowledge, so that the agents could reason on it. For example, our pedagogical scenario could be composed of multiple exercices. In one of them, there are some observables

like a TimeEvent (from *UML:Activity*) at a critical point of the procedure. **Chrysaor** can reason on this scenario and choose to apply or not the observables.

These contributions enable us to improve **Chrysaor** which can be combine with **Poseidon** and which is more modular than before (communication, behaviors, knowledge base). We can see the implementation of our new ITS in the next section.

4 Application

The use of **Chrysaor** is illustrated by an application of a virtual biomedical analyzer which enables to learn the technical procedures of the device. Many biomedical analyzers can be found in hospitals and analytical laboratories. There is a need to train people for the use and maintenance of these analysers due to the employee turnover. We decided to base our virtual reality application on a real analyzer (Fig. 5).

4.1 Virtual analyzer

We based our application called **VirtualAnalyzer** on the real analyzer and we have implemented a routine procedure called **basicUseProcedure** which is composed of 120 basic actions and an execution time of approximately forty minutes for a beginner. **VirtualAnalyzer** (Fig. 6) is a virtual reality application for the use of the biomedical analyzer, in which the learner have to do some reagents reconstitutions and use them in the analyzer in order to start a biomedical test. In **VirtualAnalyzer** we have define a routine procedure. There are plenty of possibles actions and only one role for this procedure. All this data are used by our ITS. In reality, users can directly interact with the analyzer (open a door, insert a reagent,etc.) or interact with the associate computer (launch test, open the drawer, etc.) We choose to simulate this computer interface with Android and execute it under a touchpad which communicate with the virtual analyzer. In the event of the learner making an error, an assistance will be proposed by the ITS, and the human teacher will choose one of them, like increase transparency of all the environment elements apart from the correct object (Fig. 7).



Fig. 5. Picture of the real biomedical analyzer

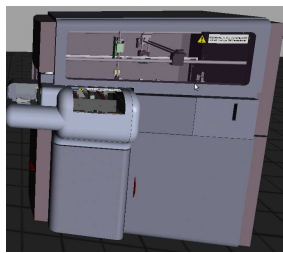


Fig. 6. Picture of the biomedical device in the virtual environment

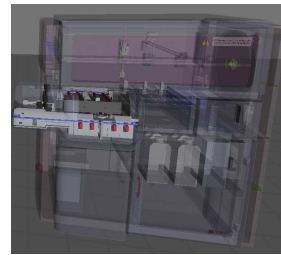


Fig. 7. Picture of the biomedical device with an assistance

With the use of our ITS in a virtual environment, learners can train at the execution of a procedure, and repeat it as much as they want. Like for the ITS model, this application uses the generic models, *i.e.* the structure of the environment, objects, organizations and procedures present in the application are described by a MASCARET model (close to UML model).

4.1.1 Environment

Previously, we exposed that an environment could be an agent knowledge. Thus, all the datas of the environment could be a knowledge for the agents : the entities (e.g. a reagent and its volume) and the organization (e.g. the `basicUseProcedure`).

Moreover, we have described some procedures using an activity diagram (Fig. 8). All these informations can be used as an agent knowledge.

4.1.2 Pedagogical Scenario

Using the new `Poseidon`, we have described a pedagogical scenario which can be split in multiple pedagogical exercises. We can see an example of exercise in Figure 9: an activity diagram with all the actions for the learner, and some observable (e.g. a `TimeEvent`) for the teacher coupled with a pedagogical action.

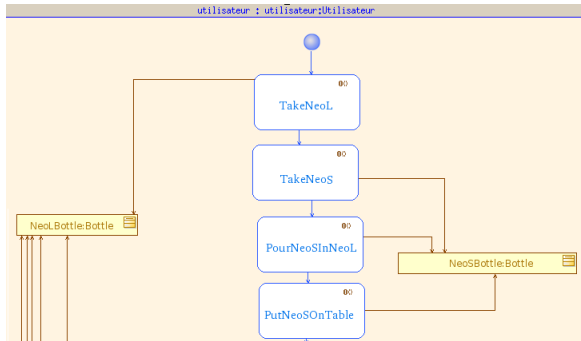


Fig. 8. UML activity diagram describing an exercise of a pedagogical scenario

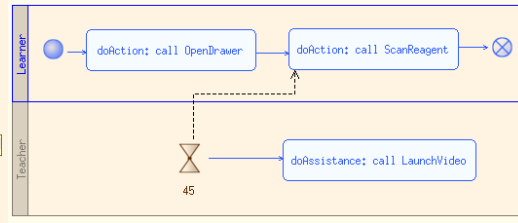


Fig. 9. UML activity diagram describing an exercise of a pedagogical scenario

4.2 ITS

Our pedagogical scenario is composed of two exercises. In the first one, the tutor presents all the important parts of the analyzer, and the learner has nothing to do. In the second part, the learner has to do a routine procedure. There are some observables like a `TimeEvent` or an imposed assistance at some critical points of the procedure. In our example, we choose to have the tutor play the teacher role of the pedagogical scenario. As though we have explain in section 3.2, the pedagogical scenario could be a knowledge of the tutor. For the first exercise, the tutor reason and choose the best way to present the parts of the analyzer (for example, grow up a part). For the second exercise, the learner do an action and the tutor has all the knowledge about the correct action (expert model) or the observables (pedagogical scenario). For example, the learner do a wrong action, so the tutor reason and propose an assistance like increase transparency of all the environment elements apart from the correct object. Later, if the learner take too long to do the correct action the tutor could choose to apply or not the observable define in the pedagogical scenario.

Due to the use of the multi-agent system technology, `Chrysaor` is more modular than before: we can easily change an agent by a real human. For example, the teacher role can be played by an autonomous agent and have a reasoning based on the behaviors described

in the model. One of its role is to choose a pedagogical assistance in a list generated by the pedagogical agent. The teacher role could also be played by a real human: he would have to decide to apply an assistance or not, and to choose one in the list.

Moreover, we can easily change the knowledge bases of each agent. For example, we could set the pedagogical scenario in the teacher knowledge base, so as to the teacher would have to choose between the assistance recommended by the pedagogical scenario or one proposed by the pedagogical agent.

One interesting point is that we can choose to have an ITS which could reason on the pedagogical scenario, or only have the pedagogical scenario and choose to execute the exercices.

4.3 Experimentation

Figure 10 represents the global architecture of the model used in the `VirtualAnalyzer` application.

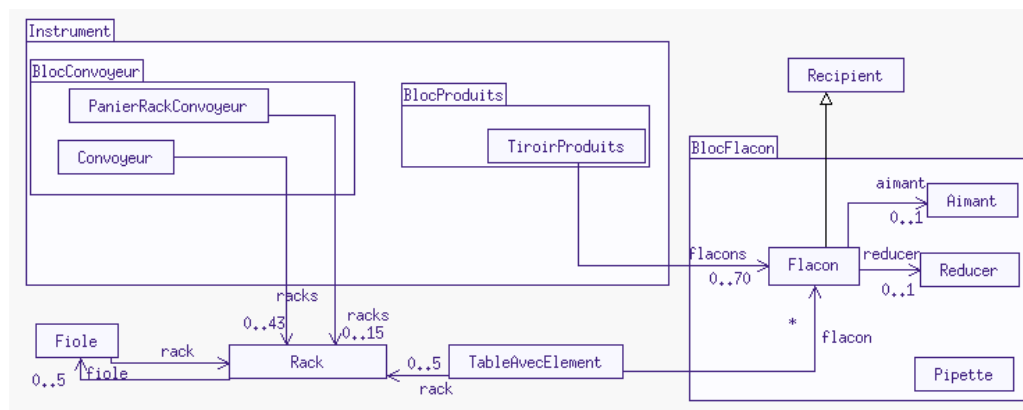


Fig. 10. UML model describing the biomedical analyzer application

In order to prove the interest of `Chrysaor` in the example of training on the use of a biomedical analyzer, we organize an experimentation with real students. Sixty high school students will use the `VirtualAnalyzer` application coupled with `Chrysaor`. Some cognitive psychologists have written an experimental protocol : There will be one group of people with a traditional training (human instructor), one group with the virtual application and one group with the virtual application and the ITS. The purpose of this experiment is to show that we can learn with virtual reality and an ITS at least as well as in the traditional training.

With the help of some cognitive psychologists, we would like to evaluate the transfer of the learning skills acquired in a virtual environment to the real life. Moreover, we would like to evaluate if `Chrysaor` improves this skill transfer. We would lead this experimentation in comparing three training methods based on the biomedical analyzer : a traditional one (theoretical training by a teacher and handling on a real analyzer), a virtual one (theoretical training, then training on the `VirtualAnalyzer`) and the assisted virtual one (theoretical training, then training on the `VirtualAnalyzer` with `Chrysaor`).

It would be interesting to compare some performance measures and also some subjective measures of the satisfaction and the mental workload.

To confirm the utility and the efficiency of **VirtualAnalyzer** and **Chrysaor**, we set up the experimentation based on a cognitive psychology literature scientific analysis [3]. This experimental working is designed and holded by a cognitive psychologist. This research have a double goal : to check how the learning of a procedure takes places within a virtual environment for training (**VirtualAnalyzer**), and to estimate the tutor (**Chrysaor**) contribution for the learning. Behavioural measurements (time to perform a task, number of instructions consulting, number of incorrect actions) will be collected as thirty people will carry out a procedure. The thirty people will be divided into two independents groups : one group will perform the procedure on virtual environment and one group will perform the same procedure on virtual environment increased by the tutor. Ten consecutive trials of the procedure will be performed first. After a period of one week, they will carry out three more trials in order to check that the learning procedure follows the theoretical framework and to attest the memorization of the procedure. Another group of fifteen people will sit in a traditional training.

5 Conclusion and futurs works

In this paper, we presented all the important characteristics needed for an ITS. On this basis, our first studies focused on the modular part of our ITS. In order to bring off this studies, we based on MASCARET metamodels. First we use the multi-agent system technology on **Pegase**. Now we can easily add a new agent, or remove one in order to play its role by a human. In the same way, we can easily change the behaviors of each role. Then we have modified MASCARET and use the environments as some knowledge of the agents. The genericity and the modularity of **Pegase** have been increased. At least, we have incorporate **Poseidon** in our model in order to use it as some knowledge for our ITS. The application of the virtual environment figuring a biomedical analyzer and containing a procedure enables us to test **Chrysaor** and the use of the new **Poseidon**.

The incoming experimentation is very important because we can improve our model according to the experimental results.

Our futur studies will focus on the learner model, in order to individualize more and more the simulation for the learner. For example we started to work on including some datas from external questionnaires (like Hollnagel [7]) in our learner model. With these works, we will be able to have a more specific pedagogy for each learner.

References

1. Ailiya, Shen, Z., Miao, C.: An emotional agent in virtual learning environment. In: Transactions on Edutainment IV, LNCS 6250. pp. 22–33 (2010)
2. Aimeur, E., Frasson, C., Dufort, H.: Cooperative learning strategies for intelligent tutoring systems. In: Applied Artificial Intelligence. vol. 14, pp. 465–489 (2000)
3. Anderson, J.: The architecture of cognition. Cambridge, Massachussetts (1983)
4. de Antonio, A., Ramirez, J., Imbert, R., Mendez, G.: Intelligent virtual environments for training: An agent-based approach. In: CEEMAS, LNAI 3690. pp. 82–91 (2005)
5. Buche, C., Querrec, R.: An expert system manipulating knowledge to help human learners into virtual environment. Expert Systems with Applications 38, 8446–8457 (2011)
6. El-Kechai, N., Desprès, C.: Proposing the underlying causes that lead to the trainee’s erroneous actions to the trainer. In: EC-TEL : European Conference on Technology Enhanced Learning. vol. 4753, pp. 41–55 (2007)

7. Hollnagel, E.: Cognitive reliability and error analysis method. In: Oxford: Elsevier Science Ltd (1998)
8. Koper, B., R.O., T, A.: Ims learning design information model. In: IMS Global Learning Consortium (2003)
9. Li, X., Ma, F., Zhong, S., Tang, L., Han, Z.: Research on virtual experiment intelligent tutoring system based on multi-agent. In: Edutainment 2010, LNCS 6249. pp. 100–110 (2010)
10. Marion, N., Querrec, R., Chevaillier, P.: Integrating knowledge from virtual reality environments to learning scenario models. a meta-modeling approach. In: International conference of computer supported education. pp. 254–259 (2009)
11. Pesty, S., Webber, C.: The baghera multiagent learning environment: An educational community of artificial and human agents. In: UPGRADE. vol. 5 (2004)
12. Querrec, R., Buche, C., , Le Corre, F., Harrouet, F.: Agent metamodel for virtual reality applications. In: International Symposium on Methodologies for Intelligent Systems (2011)
13. Rodrigues, M., Novais, P., Santos, F.: Future challenges in intelligent tutoring systems - a framework. In: Formatex (2005)
14. Sanchez, L., Imbert, R.: An agent-based adaptable and configurable tutoring module for intelligent virtual environments for training. In: Edutainment 2007, LNCS 4469. pp. 499–510 (2007)
15. dos Santos, C.T., Osorio, F.S.: Integrating intelligent agents, user models, and automatic content categorization in a virtual environment. In: ITS 2004, LNCS 3220. pp. 128–139 (2004)
16. Shi, R., Lu, P.: A multi-criteria programming model for intelligent tutoring planning. In: KES 2006, Part I, LNAI 4251. pp. 780–787 (2006)
17. Sorensen, B., Ramachandran, S.: Simulation-based automated intelligent tutoring. In: Human Interface, Part II, HCI 2007, LNCS 4558. pp. 466–474 (2007)
18. Webber, C., Pesty, S., Balacheff, N.: A multi-agent and emergent approach to learner modelling. In: European Conference on Artificial Intelligence (2002)
19. Wenger, E.: Artificial Intelligence and Tutoring Systems. Morgan Kaufmann, Los Altos, California (1987)