https://dx.doi.org/10.22161/ijaers/3.12.20 ISSN: 2349-6495(P) | 2456-1908(O)

Pedagogical Scenarization for Virtual Environments for Training: Towards Genericity, Coherence and Adaptivity

Mohamed Fahim, Abdeslam Jakimi, Lahcen El Bermi

[1]Software Engineering & Information Systems Engineering Team, My Ismail University, FSTE, Errachidia, Morocco

***Abstract*— With the appearance of new computer technologies, having access to information, learning a language, a job, or a skill, etc, is become possible and easier at any time, and at any place. Virtual Environments for Training (VET) is one of those technologies. VETs are based on virtual reality technique to put learners in training situations that emulate real situations. VETs have proven to be advantageous to put learners into varied training situations to acquire knowledge and competencies, especially when these situations are very dangerous, unrealizable, or expensive in reality. To exploit an EVT's potentials and control the behavior of it users, the trainer create a pedagogical scenario. A pedagogical scenario must let the possibility to the learner to build his own path, at the same time maintaining the coherence of learning. We present in this paper a new approach, which aims at adapting the pedagogical scenario to learner's needs, and focuses on maintaining the learning coherence.*

***Keywords*— Virtual Environments for Training, pedagogical scenario, coherence.**

I. INTRODUCTION

One of the basic requirements for education in our days, and in the future, is to prepare learners for participation in a networked, information society in which knowledge will be the most critical resource for social and economic development. Learning through Virtual Environments for Training (VET) is very useful to improve teaching and learning with the help of modern information and communication technology and virtual reality technique [1]. As learning situations grow more complex, more dangerous, more expensive or just sometimes unrealizable, VETs gain more and more interest, as they allow for a successful training.

In this paper, we propose a model making it possible to describe educational activities (EA) taking place in a VET. We identify three main needs for such a model:

Genericity: the model should be generic regardless of the learning field, of type of tasks, and of the pedagogical strategy. This level of genericity is obtained using the MASCARET (Multi-Agent System for Collaborative Adaptive and Realistic Environment for Training) meta-model [2].

The freedom of the construction of the path: the model must not impose a specific path to the learner during the learning session.

Coherence: the model should ensure coherence, whether it is temporal (the duration of the learning activity does not allow to achieve the prescribed actions), or semantic (what will happen to the following of scenario if an action changes the state of an entity?).

In the following, we first explain the concept of an informed virtual environment, and then we present the related works to the design and implementation of pedagogical scenarios. Then, we present our methodology; finally, we conclude and talk about our future works.

II. INFORMED VIRTUAL ENVIRONMENT

A VET is generally composed of a set of objects, with a graphical representation and eventually a behavior. Make a VET informed is adding to those objects semantic information. That information may be about the properties of an object, the interaction means it offers, or relations between objects, etc. Adding information to objects in the VET has two benefits: 1) avoids some expensive treatments related to the virtual agents perception i.e. an informed VET is necessary to guide the virtual agents and avatars on what can be done, how it can be done, why it should be done and how it has be done. 2) Provides input data to a decision making algorithm.

To more explain the interest of making a virtual environment informed in the context of pedagogical scenarization, we consider the following simple part of a pedagogical scenario:

"An user acting as a car driver, realized the action "open_garage_door", ..."

To write such a scenario and control its execution in a simulation requires access to the following properties of the virtual environment: there is a room named garage, which has a door that can be opened, there is also an user

https://dx.doi.org/10.22161/ijaers/3.12.20 ISSN: 2349-6495(P) | 2456-1908(O)

who can perform as a driver. So, the virtual environment is no longer seen only as a set of objects represented in three dimensions, but also as a knowledge base (Informed VET).

Informed VETs were introduced firstly by Aylett [3] which recommended the use of artificial intelligence techniques to represent explicitly knowledge in VET. After that, several models have been proposed, such as [4], which aims to include in the objects of the environment, knowledge necessary for their handling. These objects are called Smart Objects. Or STORM [5] that allows defining interaction not only between an agent and an object but also between multiple objects.

The disadvantage of these models or languages is that they cover only a part of the representation of the system and require assembling several of them in order to cover the whole pedagogical scenario. We have fixed a list of points that must ensure a language (or a model) to allow the description and operationalization of educational activities within the VET:

- It should allow the description of concepts and their properties, also the relationship between different concepts.
- It should allow the description of virtual entities: that means the fact that these entities are located in a 3D environment, they have graphical representation that can evolve, and finally a behavior that can evolve their internal state.
- It should allow the description of human activity in the environment: description of the actions that can be performed in the VET, also, the description of the sequence of actions (procedures for example).

The MASCARET meta-model, overcomes the main disadvantage mentioned above and ensure all the points already cited. It allows the creation of informed VET by adopting an object modeling approach. It allows respectively describing the concepts of an environment, its dynamic and the activities that is possible to achieve, through three profiles:

- Environmental model VEHA: it is an informed and structured virtual environment meta-model defined as an extension of UML 2.1. VEHA covers modeling of semantic properties, structural, geometric and topological entities of the virtual environment and their reactive behavior.
- Activities model HAVE : it aims to formalizing the concepts of actions and activities achievable by an actor in the simulation, whether it is an autonomous agent or an avatar controlled by a human.
- Organizational model, BEHAVE: The organizational model allows defining the role of agents within a collective activity.

III. RELATED WORKS

Current methods for the modeling of pedagogical scenarios can be divided into three groups:

1. Non-contextualized models of pedagogical scenarios

This kind of models is useful to describe educational activities called ”black boxes”: only the input parameters and the output values may be known. The most important proposal of this first category is the IMS-Learning Design (IMS-LD) standard [6]. IMS-LD focused on the concept of learning unit as a basic element of the description of the learning process. In IMS-LD a scenario is considered as a series of educational activities. Each of these activities is described by a text or a set of documents explaining the purpose of the activity, the task to achieve, the instructions to follow, etc. however, IMS-LD has it own drawback as mentioned in[7]: IMS-LD don't allow the description of the interaction between users in collaborative tasks.

We find also among this category, the language LDL (Learning Design Language) [8], considered as an enrichment for the IMS-LD . It is based on a meta-model which is come to design cooperatives and heterogeneous educational situations. The CPM (Cooperative Problem-based learning Meta-model) language [9] is a proposal for a language based on UML(Unified Modeling Language) [10] to model the learning situations known as “Cooperative Problem-based situations “.

2. Oriented-simulation models of pedagogical scenarios

These models describe very precisely the context of implementation of educational activities, but are not reusable (not generic) due to their specificity. Included in this category are Simquest [11], and RIDES [12]. The first work is an authoring system for creating computer simulation embedded in an instructional environment. The former is also an authoring system for creating interactive simulations for learning. It is used to create simulations on different subjects (medical devices working, orbital mechanics, etc. We also find The FORMID (FORMation Interactive Distance) [13] project which focus on pedagogical scenario activities in which learners interact with an interactive pedagogical object. These methods are quite limited in term of interaction and immersion and they are not generic i.e. not possible to reuse the task description in another exercise.

3. Oriented-virtual reality models of pedagogical scenarios

In addition to their similarity with the models of the second category, this third category adds the immersion and the interaction sides using virtual reality technique. Among those models, we can cite: the product GVT(Generic Virtual Training) [14], The main goal of

https://dx.doi.org/10.22161/ijaers/3.12.20 *ISSN: 2349-6495(P) | 2456-1908(O)*

GVT is to propose a set of software to cover the life cycle of a VET from conception to exploitation by learners and trainees. Another proposal is called FIACRE [15], this project led to the creation of a VET of train operators of high speed rail service. We find also the VTS(Virtual Training System) project [16], it is a training device for assembly operations in virtual environments. [17] Proposed HUMANS (HUman Models based Artificial eNvironments Software) platform, it is a generic framework, designed to build tailor-made virtual environments, while it generic, this approach don't allow to describe collaborative tasks, also it is not easy to handle by a non computer scientist trainers.

Moreover, [18] proposed a model of a pedagogical scenario called POSEIDON (PedagOgical ScEnario for vIrtual and informeD envirONment), which aims to be directly reusable in different environments. POSEIDON is the most inspiring approach for us, especially with its generic side due to the use of MASCARET meta-model. However, this approach still has two drawbacks: 1) it cannot ensure the coherence of pedagogical scenario, 2) it impose a specific path to the learner.

IV. PROPOSITION

Our main objective is to immerse learners in their professional environment simulated using virtual reality technique. This enables them to manipulate the environment so that they can "learn while doing".

We are interested in the activity of the trainer who wants to design and implement training sessions which has activities in virtual environments. In this context, the role of the trainer is to prepare and exploit the interaction between the learner and the learning device. To achieve that goal, he must create a pedagogical scenario. A pedagogical scenario defines generally six type of information [19]:

- Learning objectives: describe skills and knowledge to be achieved by learners that accomplish the pedagogical scenario.
- Prerequisites: they are skills and knowledge necessary for a learner so he can start an educational activity correctly.
- Activities: they consist of the description of educational activities performed in the environment by different actors (learners, trainers, etc).
- Roles: they describe roles of everyone involved in a educational activity.
- Resources: resources necessary to accomplish a learning session perfectly (Audio file, images, etc).
- environment : it describes the context of execution of learning activity, it contains also necessary resources to the execution of activities.

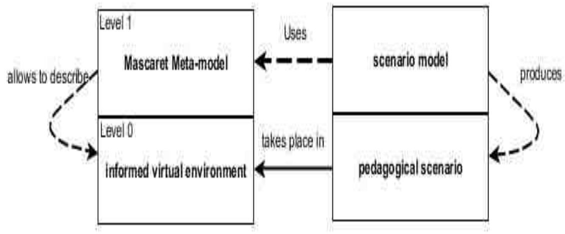

Fig.1: relationship between different modelization levels

The figure 1 shows representation of links between pedagogical scenarios, pedagogical scenario model and the different virtual environments modeling level. A pedagogical scenario takes place in a virtual environment (level 0). The scenario is produced using a pedagogical scenario model that uses the MASCARET meta-model (level 1).

We propose POSVET (PedagOgical Scenarization for Virtual Environments for Training), a generic pedagogical model scenario for the VETs, i.e. applicable in multiple environments, regardless of the field or type of task to be performed, this level of genericity is obtained through the MASCARET meta-model describing the virtual environment explicitly.

The figure 2 above shows the main class of POSVET:

- Prerequisite : A prerequisite list.
- Learning Objective : An objective list.
- Educational Environment : An environnement.
- Educational Activity : The activities of the scenario
- Educational Organisation : role of agents.

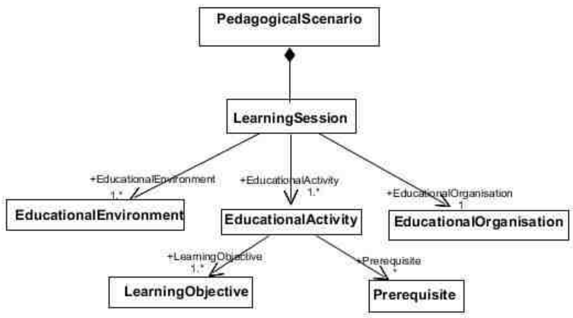

Fig.2: synthetic UML class diagram of the main elements of the model.

In POSVET , an educational activity takes place in a virtual environment. The Educational Environment is an instance of the Mascaret Environment. A POSVET educational environment (Educational Environment) contains the description of an informed virtual environment as defined by MASCARET. Moreover, POSVET allows the integration of learning Objects in an

https://dx.doi.org/10.22161/ijaers/3.12.20 ISSN: 2349-6495(P) | 2456-1908(O)

educational environment. It can be for example a multimedia file (text, audio, video, etc.), that is possible due to the link between a Learning Object and the element of the informed environment related to. For instance, a video can be used to show how to use a set of entities (relationship between a learningobject and an element of the environment). Different links between POSVET educational environment and MASCARET environment are shown in the figure 3.

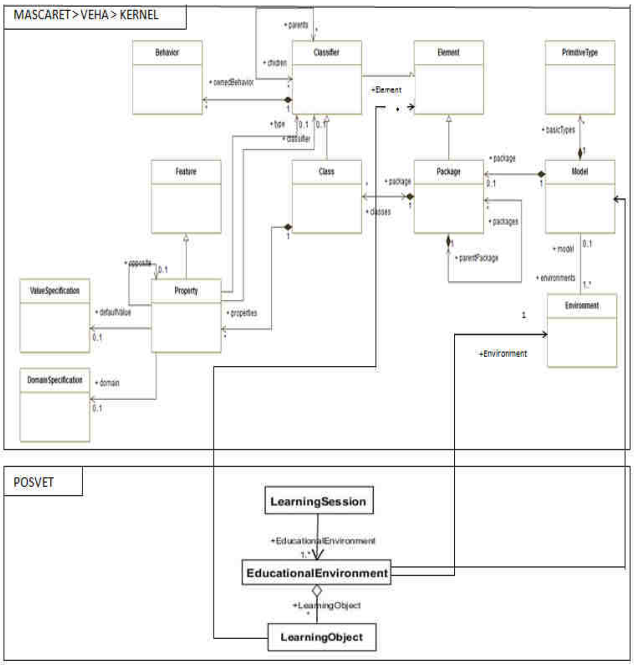

Fig.3: The entire POSVET educational environment model

The trainer can express how to organize actions to form an educational activity. The role of an educational activity is to describe the actions that the trainee must do to complete a task. The sequence of a set of educational activities makes the whole pedagogical scenario. The model of the educational activity we propose is based upon the activity model (BEHAVIOR) defined in VEHA model. The link between the POSVET educational activity and the activity as defined by MASCARET is shown in the figure 4.

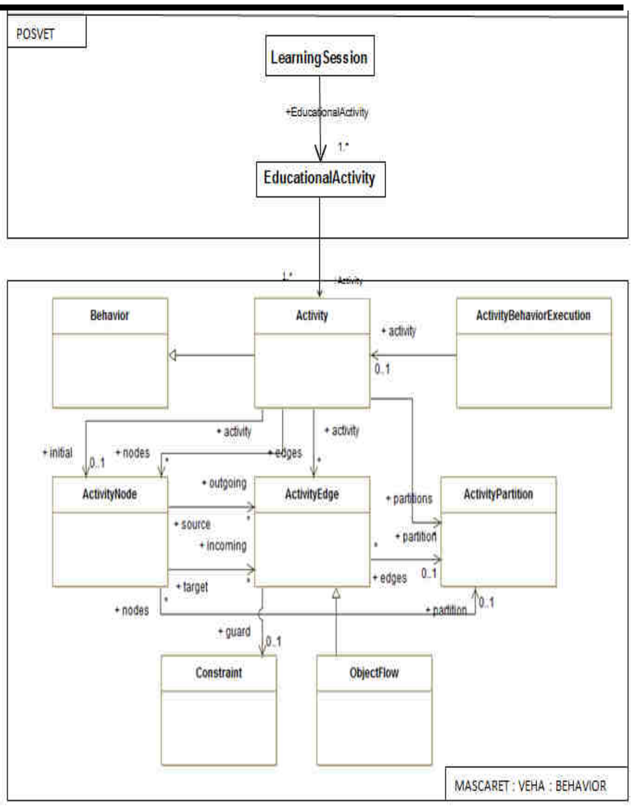

Fig.4:link between the POSVET educational activity and the MASCARET activity

In a learning session, different actors (human or artificial agents) can interact. Those actors can play different roles: trainee, trainer, etc. POSVET provides an organizational model for expressing roles based upon MASCARET organization model (BEHAVE). As part of the pedagogical scenario the trainer defines an educational organization that describes educational roles. POSEIDON defines two types of educational role: learner (trainee) and trainer. Each educational activity of the learning session must be performed by actors playing roles that are associated to. Each of these roles may be performed by a human or an autonomous agent. We have chosen the UML language for our approach. Firstly, because MASCARET is an UML extension, also the UML description enables to describe complex educational activities through loops, sequences, parallelism, events, etc. POSVET is therefore defined as an UML profile where:

- The pedagogical organization is an UML collaboration made up of roles.
- Educational activity: it is an UML activity diagram where each node is an action to be performed by the trainee. figure . 5 shows an extract of an educational activity described by an UML activity diagram. This activity allows to measure the speed of sound in the air.

UML is powerful but still have one big drawback: it is hard to handle by non-computer scientists. To facilitate the creation of pedagogical scenarios using POSVET, we are currently developing a scenario editor. This editor will allow easily the creation and modification of various components of a pedagogical scenario: educational environments, educational organization and educational activities. The creation of the pedagogical scenario using the editor will be through the use of simplified diagrams, which not require knowledge of UML.

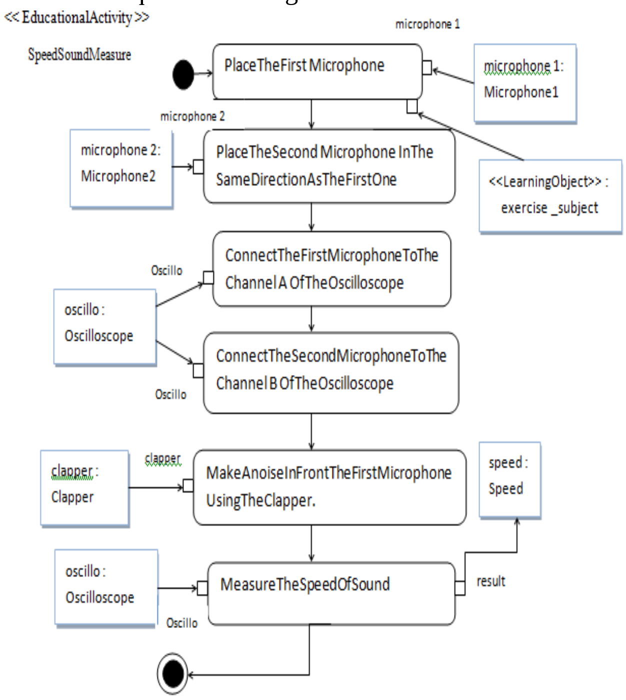

Fig.5: extract of the "SpeedSoundMeasure" educational activity represented as an UML activity diagram

In figure 6, we show the main parts of our design approach:

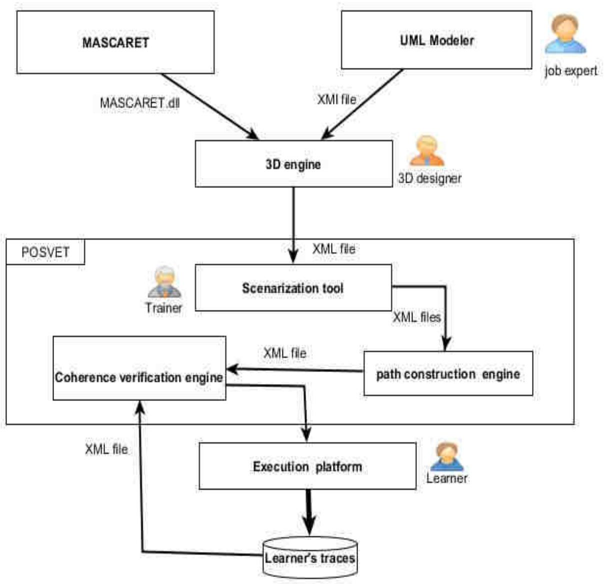

Fig.6: Our pedagogical scenario design approach

- UML Modeler: the job expert defines the virtual environment's model in the form of Uml–Mascaret diagrams exported into XMI. He has to describe the structural models and behavioral models (state machines and activities) and the human activities using UML collaboration and activities diagrams. This step is completed using a UML modeler which supports metamodels defined as UML profiles. In this work we are using Modelio UML modeler which allows importation/exportation of files with XMI format.
- 3D engine: the 3D designer creates the objects and their behavior that form the VET based on the XMI file already generated. In our work we are using the Unity 3D engine because it seems to be the most suitable with MASCARET, both are compatible on C# programming language.
- Scenarization tool: in this step, the trainer who knows very well the activity which has to be undertaken by the trainee, formalizes the sequence of actions, and interactions with the objects of the environment. He also describes educational activities, those activities are written independently from one another. This description is also independent from the execution platform. The trainer adds to each educational activity the identifier of the activity (or activities) that must be done before launching the next activity. Each educational activity is related to resources (learning objects) necessary to complete an educational activity successfully. those resources could have different formats (Pdf file, audio file, etc).
- Path construction engine: this engine deals with the generation of educational activity graph based on the learner's profile. The engine takes in input the XML file generated by the scenarization tool. The file contains educational activities that form the whole learning session and gives in the output the educational activity graph for each learner(pedagogical scenario). The engine transforms an initial list of educational activities to a hierarchical graph of several levels. Once the graph is generated the learner should validate the EA(s) to have the access to the next level. The generation of the EAs graph is based upon each learner's profile and depending on the training and competence chosen by the learner. The learner profile consists of the following information: the difficulty level, age, language chosen, duration of the learning session. The algorithm of figure 8 explains how it works The following table illustrates the different EAs of a training session and their prerequisites and the figure 7 shows the suitable educational activity graph of the table.

https://dx.doi.org/10.22161/ijaers/3.12.20 *ISSN: 2349-6495(P) | 2456-1908(O)*

Table.1: Example of different EAs of a training session and their prerequisites

Educational activities	title	Prerequisites
EA1	H	-
EA2	A	EA1
EA3	E	EA1
EA4	D	EA2
EA5	C	EA2,EA3,EA4
EA6	B	EA2,EA3,EA4

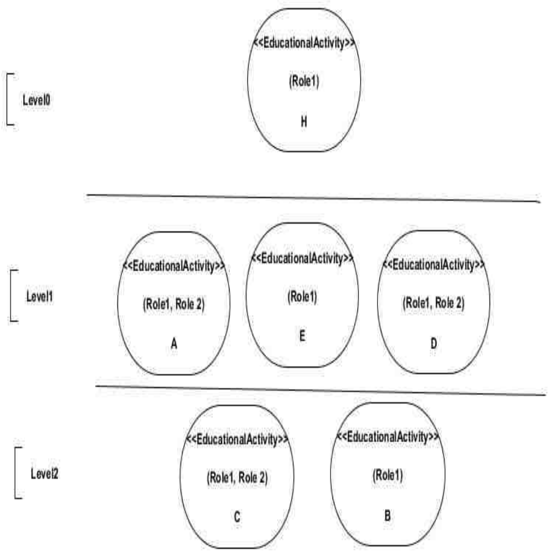

Fig.7: the educational activity graph generated according to the table above

Input : Learner Profile(ID_Learner),
Training (ID_Training)
Output: Graph of Learning Activities (graph)

```
prerequisitesList: {Collection of Educational Activities }
initialList: {Collection of Educational Activities }
level: Integer
initialList ← getEducationalActivities(ID Learner,
ID Training)
foreach Educational Activity EA: initialList do
    prerequisitesList ← hasPrerequisites(EA)
    if prerequisitesList == Φ then
        graph.put(0, {EA})
        initialList ← initialList \ {EA}
level ← 1
while initialList != Φ do
    foreach Educational Activity EA : initialList do
        prerequisitesList ← hasPrerequisites(EA)
        if prerequisitesList ∧ initialList == Φ
        then
            graph.put(level, {EA})
            prerequisitesList ← Φ
            initialList ← initialList \ {EA}
    level ← level + 1
```

Fig.8: Graph construction algorithm

In the figure 7, the trainee begins the level 0 of the learning session by execution of the educational activity " H". after that, in the level 1 the trainee has the possibility to choose between educational activities "A", "E" or "D". once the three educational activities of level 1 are validated, the trainee complete the learning session by performing the educational activitie "C" and "B" of the level 2.

- Coherence verification engine: based upon learners traces and the file generated by the path construction engine, the coherence verification engine fix the two kind of incoherence already cited: temporal and semantic incoherence. for the first case the engine indicates to the user that the duration of an activity is not enough to achieve all the actions mentioned in the pedagogical scenario. For the second kind of incoherence, it is the case when an action changes the state of an entity, in this case the engine restores the last correct situation.

The user traces file is based on a MASCARET tracking function of agent activity, provided by the pedagogical agent "observer".

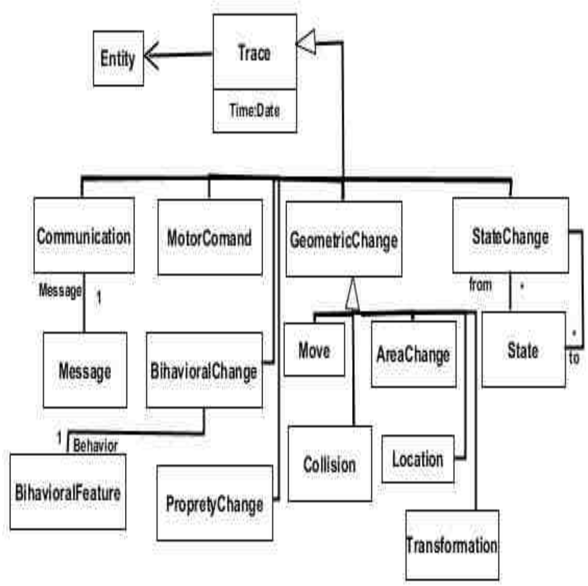

Fig.9: Mascaret observable types that form traces (UML class diagram)

Traces are recorded in an XML file. The example below illustrates the contents of a trace recorded during an experiment. It includes the following information:

- The event identifier.
- The trace date.
- Type of the performed action.
- The name of the agent and its location before the current action.

- The entity to which the agent interacts with its name, its position before and after the action.


```
<eventid=843>
<name>Takeentity(startTurn)</name>
<date>d11m04y2016_13-00-54-0646</date>
<type>take</type>
<content>User sujet13 grabbed the entity sphereRouge
(startTurnaction)
</content>
<actor>
<name>sujet13</name>
<locationBefore><tx>1.97</tx><ty>-1.07</ty><tz>1.7
0</tz><rx>0.00</rx
><ry>0.06</ry><rz>1.57</rz></locationBefore>
</actor>
<target>
<name>sphereRouge</name>
<locationBefore><tx>2.40</tx><ty>1.44</ty><tz>1.00
</tz><rx>-0.00</rx
><ry>-0.00</ry><rz>-0.00</rz></locationBefore>
<locationAfter><tx>2.40</tx><ty>1.44</ty><tz>1.00</
tz><rx>-0.00</rx>
<ry>-0.00</ry><rz>-0.00</rz></locationAfter>
</target>
</event>
```


Fig.10: example of traces file

V. CONCLUSION

We proposed in this paper, POSVET, a model that externalizes the pedagogical scenario from the application. POSVET which is based upon UML, produces an interpretable pedagogical scenario by the virtual environment. The generic side of our methodology, due to the use of MASCARET meta-model, allows the pedagogical scenarios to be reused in different platforms. The main advantage of this model is permitting adaptation of educational activities and leads the learner to control learning. A prototype of our approach is being developed. In the future, we plan to test POSVET on different type of applications.

REFERENCES

[1] P. Fuchs, G .Moreau, Le traité de la réalité virtuelle (Les Presses de l'Ecole des Mines de Paris, 2006).

[2] P. Chevaillier, Les systèmes multi-agents pour les environnements virtuels de formation. Habilitation à Diriger des Recherches, Laboratoire d'Informatique des Systèmes Complexes, Université de Bretagne Occidentale , Brest, France, 2006.

[3] M. Luck, R. Aylett, Applying Artificial Intelligence to Virtual Reality: Intelligent Virtual Environments, (2000)
Applied Artificial Intelligence, 1(14), pp. 3-34.

[4] M. Kallmann, Object interaction in real-time virtual environments. Ph.D. Thesis, École Polytechnique Fédérale de Lausanne, Suisse, 2001.

[5] N. Mollet, B. Arnaldi, Storytelling in Virtual Reality for Training, Proceedings of the First international conference on Technologies for E-Learning and Digital Entertainment (Page: 334-347 Year of Publication: 2006 ISBN: 3-540-33423-8 978-3-540-33423-1).

[6] R. Koper, B. Olivier, T. Anderson, IMS Learning Design Information Model (IMS Global Learning Consortium, 2003)

[7] C. Ferraris, A. Lejeune, L. Vignollet, J.P David, Modélisation de scénarios d'apprentissage collaboratif pour la classe : vers une opérationnalisation au sein d'un ENT, Proceedings of the 2nd EIAH'05(Page : 285-296 Year of Publication : 2005 ISBN : 978-2-7342-0999-7).

[8] M. Christian,V. Laurence, F. Christine, D. Guillaume, LDL: a Language to Model Collaborative Learning Activities,. Proceedings of 9th EdMedia: World Conference on Educational Media and Technology (Page: 838-844 Year of Publication: 2006 ISBN: 978-1-880094-60-0).

[9] P. Laforcade, Méta-modélisation UML pour la conception et la mise en œuvre de situations-problèmes coopératives. Ph.D. Thesis, Université de Pau et des Pays de l'Adour, Pau, France, 2004.

[10] Object Management Group (OMG), Unified Modeling Language (UML) specifications version 2.1 , 2007. http://www.omg.org/

[11] W. Van Joolingen, T. Jong, SimQuest, authoring educational simulations, In T. Murray, S. Blessing & S. Ainsworth (Eds), Authoring tools for advanced technology educational software : Toward cost-effective productionof adaptive, interactive, and intelligent educational software, (Dordrecht: Kluwer Academic Publishers, 2003, 1–31).

[12] A. Munro, M. Johnson, Q. Pizzini, D. Surmon, D. Towne, J. Wogulis, Authoring simulation-centered tutors with RIDES, (1997) International Journal of Artificial Intelligence in Education, 8(3-4), pp. 284–316.

[13] V. Guéraud, J-M. Cagnat, "Automatic semantic activity monitoring of distance learners guided by pedagogical scenarios", Proceedings of the First European conference on Technology Enhanced Learning: innovative Approaches for Learning and Knowledge Sharing (Page: 476–481, Year of Publication : 2006 ISBN: 3-540-45777-1 978-3-540-45777-0).

[14] S. Gerbaud, N. Mollet, F. Ganier, B. Arnaldi, J. Tisseau, Gvt : a platform to create virtual environments for procedural training, , Proceedings of the 9th IEEE Virtual Reality Conference(Page: 225–232. Page: 254-259 Year of Publication: 2008 ISBN: 978-1-4244-1971-5).

https://dx.doi.org/10.22161/ijaers/3.12.20 ISSN: 2349-6495(P) | 2456-1908(O)

[15] D. Lourdeaux, Réalité Virtuelle et Formation : Conception d'Environnements Virtuels Pédagogiques, Ph.D. Thesis, Ecole des Mines de Paris, Paris, France, 2001.

[16] J. Brough, M. Schwartz, S. Gupta, D. Anand, R. Kavetsky, R. Pettersen, Towards the development of a virtual environment-based training system for mechanical assembly operations, (2007) Virtual Reality, 11(4) , pp.189–206.

[17] V. Lanquepin, V, K. Carpentier, D. Lourdeaux, M. Lhommet, C. Barot,K. Amokrane, HUMANS: a HUman Models based Artificial eNvironments Software platform, Proceedings of the 15th International Conference of Virtual Technoloogies (VRIC'13)(Page: 59-68 Year of Publication: 2013 ISBN: 978-1-4503-1875-4).

[18] N. Marion, R. Querrec, P. Chevaillier, Integrating knowledge from virtual reality environments to learning scenario models. a meta-modeling approach, Proceedings of the 1st International conference of computer supported education(Page: 254-259 Year of Publication: 2009 ISBN: 978-989-8111-82-1).

[19] R. Koper, Modeling units of study from a pedagogical perspective: the pedagogical meta-model behind EML (Open University of the Netherlands : Educational Technology Expertise Centre, 2001).