Storyline for Adaptive Learning Virtual Environment (SALVE)

Frederic Kleinermann, Jordi Ribera

Department of Computer Science, Vrije Universiteit Brussel, 1050 Brussel, Belgium
{frederic.kleinermann, ribera.jordi}@gmail.com

Abstract. Serious games can be very engaging by being visually attractive and by simulating a professional environment. But they are often too predictable, impersonal and not so user-centric. To address these issues, a number of research works have been performed on adaptations where the virtual scenario adapts according to what and how the players need to learn in a given context. But most of these works focus only on adaptation for single user. It is therefore important to look at serious games with multi-user capabilities. This is challenging as the virtual scenario should adapt according to a group of learners having different skills and having to learn a common goal. This paper describes an approach called SALVE that allow authors to create a storyline (or scenario) in which they can define the type of adaptations they want to introduce for a specific learner or a group of learners so that the storyline adapts according to what the learners do inside.

Keywords: Serious Games, Virtual Environment, E-learning, Multi-Learner Virtual Environment.

1 Introduction

Today, Virtual Environment is appearing through the internet as an online community that often takes the form of a computer-based simulated environment, through which users can interact via avatars with one another and use and create objects. An example of that is Second Life [1] where people are represented by avatars and they can meet, socialize and explore the virtual world all together. These sort of virtual environments are often referred as Collaborative Virtual Environments (CVEs) or Multi-User Virtual Environments (MUVEs).

Recently, these Virtual Environment (VE) platforms have been used to develop games with training and learning goals. These sorts of games are called Serious Games and can be seen as a Virtual Learning Environments (VLE). They provide sort of Virtual Environments based on Games with the aims that the user is learning something. To some degree, Second Life could be seen as a serious games and a number of research works have been conducted to use Second Life as a Multi-learning Virtual Environments. An example of such works is the PIVOTE project [2]. However, the richness of such a VLE can also become its weakness. The learner may be overwhelmed or get lost in the VLE [3], not knowing what to do first or next, or may be distracted too much and not be able to focus on the actual learning task. For people not familiar with VEs (novice users), the time required to get acquaint with

© C. Delgado, C. Gutiérrez, R. Velázquez, H. Sossa (Eds.)
Advances in Computer Science and Electronic Systems
Research in Computing Science 52, 2011, pp. 1-13

Received 23/01/11
Accepted 20/02/11
Final version 07/03/11

such a VLE (i.e. learnability) may be long and therefore their short-term satisfaction may be low. On the other hand, youngsters used to play video games, may spend their time in activities not very much related to the learning activities, especially if they have low motivation for learning. This then results in a low effectiveness. These concerns are confirmed in [4]. One way to solve these problems is by providing VLE with an adaptive way, e.g., adapted to the individual learner and to the progress that he makes during the learning.

A number of research work already exists (see section 2) and use adaptation for improving the learning phase. But they are mostly made for single learner. Today with the possibility of Collaborative Virtual Environments (CVE) over the internet, it is also becoming important to look at the possibility of using adaptation in the context of multi-learner environment. But this is challenging as it is important to be able to distinguish between a specific learner and a group of learners. In that kind of multi-learner virtual environment, learners share the same space and as a result any modifications are seen by all.

Based on our previous work [5] [6] aiming at bringing adaptation to VE in the context of E-learning based on single learner (i.e. VLE), we have developed a new approach called Storyline for Adaptive Virtual Learning Environment (SALVE) aiming at creating adaptive storyline (or scenario) for Multi-learner Virtual Learning Environment (MVLE). The SALVE approach provides authors a way to create a storyline (or a scenario) in which they can add a number adaptations they want to introduce for a specific learner or a group of learners so that the course adapts according to what the learners do inside. The rest of the paper is structured as follows. In section 2, we consider related work. Section 3 introduces the SALVE approach. In section 4, we give some description of the software architecture implementing the SALVE approach. Section 5 presents a case study. Finally, section 6 presents conclusions and future work.

2 Related Work

This section starts by introducing existing works dealing with storyline (or scenario). It then reviews research works that use adaptation in the context of E-learning.

2.1 Storyline (or Scenario)

The *alVRed* [7][8] is a project aiming at the creation of a set of tools for designing non-linear and interactive stories in virtual environments. This approach considers *scenario* to be a ‘story’. They have developed an authoring tool for scriptwriters that allows authors to model objects participating in the story. Although this work is interesting, it does not allow author to model adaptation somehow into the scenario.

In [9], they presented an approach on modelling technique for urban traffic which uses Environment Description Framework (EDF) and Scenario Description Language (SDL). EDF is a modeling technique concerned with modeling the static portion of the scene while SDL is a scripting language to describe scenarios in the world. The work presented is limited to a specific domain i.e. urban traffic. The scripting language is not so intuitive. Furthermore no adaptation is taken into account.

In [10] a new approach called Simulation Behavior Specification Diagrams (SBSD) has been developed and it uses a scenario description language tailored towards one particular type of VE namely military mission simulation. An interesting aspect of this work is that the approach uses a visual language to model the scenario. But this approach does not allow to model full scenarios.

In [11], they have developed a scenario description language which is reminiscent of UML use cases. Instead of modeling all possible courses of a scenario, the action frame scenario language only models a single execution of it. Although what they have presented is interesting, they have not used adaptation.

In [12], they look at adaptive and intelligent Web-based Education Systems (WBES) that take into account the individual student learning requirements, by means of a holistic architecture and Framework for developing WBES. They have developed a framework that includes an authoring tool, a semantic web-based evaluation, and a cognitive maps-based student model.

2.2 Adaptation of the content/presentation for single user

Brusilovsky et al. [13] have integrated some adaptive hypermedia methods into virtual environments by developing an approach that supports different navigation techniques in the context of 3D E-Commerce. This work is interesting because it has extended some of the adaptive hypermedia methods (such as direct guidance, hiding, sorting) to 3D environments.

In 2000, Chittaro and Ranon [14] have described how to introduce adaptation inside e-commerce. Their approach is called ADVIRT. A set of personalization rules exploits a model of the customer to adapt features of the VR store. They have also customized and personalized the navigation and different layouts of the store. In 2002 [15], they have introduced a software architecture for adaptive 3D web sites called Awe3D (Adaptive Web 3D) which can generate and deliver adaptive Virtual Reality Modeling language (VRML). In 2007, the same authors [16] has explained that adaptation can happen for navigation and interaction in order to help the users in finding and using information more efficiently. Finally, and based on their previous work [17], Chittaro and Ranon have extended the E-learning platform EVE [18]. They introduced Adaptive EVE that is tailored to the knowledge level of a student and to their preferred style of learning.

Santos and Osorio [19] have introduced another approach for adaptation in VR. Their approach is called AdapTIVE (Adaptive Three-dimensional Intelligent and Virtual Environment) and is based on agents, called Interactive and Virtual Agents that assist the users and help them to interact with the environment.

Celentano and Pittarello [20] have developed an approach for adaptive navigation and interaction where a user's behavior is monitored in order to exploit the acquired knowledge for anticipating user's needs in forthcoming interactions.

3 SALVE Approach

MVLE are made of virtual objects that have visual appearance, behaviours and for which interaction can happen either with the learner or with other virtual objects. In our case adaptation may happen on the visual appearance of virtual object, on the behaviours or on the interactions. In other words, in the SALVE approach, we state that a virtual object can be in different states. Each state can be seen as an adaptation happening at a particular time. We call these states Virtual Environment Adaptations State (VEAS).

To trigger on or off a VEAS, a number of condition will need to be fulfilled like for instance if the learner A has seen a previous virtual objects before this one, then trigger this adaptation. This can be seen as a relation type in which a virtual object is in a certain adaptation state before the condition is fulfilled and then in a different adaptation state after the condition is fulfilled. We call these relation types Virtual Environment Relation Type (VERT). VERTs will also be based on the user's profile. The third element in our approach is the adaptive storyline itself where the authors can compose the storyline using not only these VEAS and VERT, but also other concepts like *next step*, *parallel step* and *pedagogical objects*. Furthermore all these VEAS need to take into account the fact that there are several learners at the same time (i.e. multi-learner environment). In other words these adaptations can be specific to a learner or to a group of learners. To achieve this, we have developed in our approach the notion of leaner's ID using *ID*, *colour* and *sound*. Each of these components will now be described in this section.

3.1 Identification of learners

To **extend** from single learner to multi-learner, it is important to be able to distinguish between a specific learner and a group of learners. Indeed in that kind of multi-learner virtual environment, learners share the same space and as a result any modifications are seen by all. The SALVE approach allows three types of identifications that can be combined namely colour, sound and a user ID. By bringing colour, sound and user ID as identifiers, we can then make adaptations targeted to a specific learner or to a group of learners. For instance, a building A is suddenly highlighted in the red colour. This will create not only an adaptation of the Virtual World, but it will also be a way to have a specific adaptation to a learner identified by the red colour as his ID. Sound can also be used in that way. Nevertheless, there can be situations where the same adaptation happens at the same time as several learners do the same action at the same time. For instance, suppose that different learners go to see the same virtual painting at the same time, then perhaps *enableAnnotation* is triggered. Since there are several learners, the annotation will be highlighted in several colours at the same time. This gives the problem that learners have no idea if the annotation is meant for him. To overcome this sort of problem, the approach uses:

- talkInChat: Normally, chats are text in which the learner can communicate through a panel. This can be also used to communicate to a learner or a group of learners.

- HeadLearnerDisplay (HLD): It shows a panel above the head of the avatar representing the learner inside the MVLE and this can be used to prompt messages that are specific to a learner.

Another ways is to use sound. There, the sound can adapt to a specific learner or a group of learners, by assigning a different sound to each of them.

3.2 Virtual Environment Adaptation State (VEAS)

When an adaptation is applied to a virtual object, we can say that the virtual object is in a certain state. These adaptations are called Virtual Environment Adaptation State (VEAS). The SALVE approach provides a library of VEAS that an author of a storyline can choose from to introduce adaptations in his storyline (or scenario). In this paper we outline some of them as an example. They are different types and they are not all related to only adaptation of content and presentation. For instance, to visually indicate that an object has not yet been studied, we may want to give it a different colour to attract the attention of a learner and we may change the colour to the one that identifies him. We will now outline some of the adaptations which have been defined in our library of VEAS.

The first category of adaptation types for objects is concerned with the adaptation of the visualization of an object, i.e. how to display it and how to hide it. Because of the length of paper, we will only mention some of them here.

– *semiDisplay*: this adaptation type is used to display the object in a semi-manner, by having a semi-transparent bounding box around it.

– *changeSize*: this adaptation type is used to change the visual appearance of an object by changing its size.

A type of adaptation which is more specific to pedagogy is marking objects to draw the attention of the learner. Marking an object can be done in different ways and we distinguish two different adaptation types for marking :

– *spotlight*: this adaptation type allows to mark an object by putting a spotlight on the object; in this way the object becomes more visible and can be used to draw the attention of the learner to this object.

– *highlight*: this adaptation type allows to mark an object by drawing a box around the object where only the edges of the box are displayed.

Another type of adaptation which is not related to visual appearance is the ability to change behaviours and interactions. Possible adaptation types for behaviors are:

– *enableBehavior:* this adaptation type allows enabling a behavior associated with an object. Note that we have also the opposite which is *disableBehaviour*.

- *enableInteraction*: this adaptation type allows enabling an interaction type (given as parameter) for an object. Not that the opposite exists and is defined as *disableInteraction.*

So far, we have discussed possible adaptations for individual components (object, behavior, interaction, …) of a MVLE. But we have adaptations that can have an impact on several components of the VE or on a part of the VE. We call them *adaptation strategies*, as they can be used as strategies to adapt a VE.

- *navigationWithRestrictedBehavior*: this adaptation strategy allows restricting the possible behaviors of objects while navigating. The restricted behavior can apply on all objects (in the VLE) or on a specified list of objects.

- *freeWithSuggestions*: this adaptation strategy will allow the learner to navigate freely in the VLE but in addition some objects will be "suggested". Suggesting is done by using marking (i.e. spotlight or highlight).

- *displayAtMost*: this adaptation strategy allows to specify when some objects should not be displayed anymore. The condition can be given by means of some pedagogical criteria like the knowledge level the learner currently has for the object or by setting a limit on the number of times the object should be displayed.

The adaptation strategies presented here are only a subset of possible adaptation strategies that we have developed in our approach.

3.3 VERT and Learner's Profile

VEAS should be activated under certain conditions. These conditions correspond to some kind of rules that are called Virtual Environment Relation Types (VERT). The VERT allows us to also encode and retrieve information on a learner or a group of learners such as how many times a certain learner has interacted with an object, how fast does the learner go or has he found certain clues. Note this can also be applied to a group of learners. More can be found in [6].

3.4 Adaptive StoryLine

The SALVE approach helps authors to conceptualize a story like for instance a course. The conceptualization of a story is done through the use of VEAS and VERT. Our approach has been inspired by the work done in the video game community and the way they actually generate a storyline [21]. In our approach, the conceptualization of a story is done from a high-level (i.e. away from the code level) and then the code itself is automatically generated.

In video games, there are different ways to look at a storyline. Often, a storyline is split among different levels where a player goes from one level to the next. For instance, when the player has collected enough points in one level, he can then progress to the next one. This way is very much linear. However inside a level, the storyline can be very non-linear as in some video games for instance, the player has to collect different types of coins. The way he collects them can be in a non-linear way as he may collect two red coins and then suddenly three green coins and then another red coins. This helps the writer of a game to go from a linear way in his storyline (levels) to a non-linear way [21]. In our approach, we want to follow a similar way where the author of a story may want to see a global view of his storyline much more like sections in a linear way, but he also wants to give some freedom the way learners explore each sections. For instance, he wants to say that the learner should visit the modern arts gallery before going to the abstract art gallery. However, the way the learner explores the different paintings in the modern art gallery can be random.

To achieve this, our approach uses the notion of *next Step* that provides a general view of the storyline much more like a linear way. To make, the storyline be non-linear when several learners are involved, we have introduced the notion of *parallel steps*. Parallel steps are made by sub steps (like for instance next step) that can be run in parallel i.e. there are no precise orders in which they are executed. This depends only on the learners and how they explore the virtual world.

To introduce adaptation in the conceptualization of a storyline, each sub steps can use VEAS and VERT (described above). This will provide a way to say that if learner A does now this, adapt the virtual world like this. However, if the learner B does another thing, then the virtual world should also be adapted. The non-linearity will happen as these two types of adaptations can happen in parallel and as a result there is no way to know in advance if learner A does something before learner B or vice-versa.

4 Software Architecture

Our approach has been implemented through a software architecture shown in figure 1. They are made of different elements.

The first element is an authoring tool that allows an author to create the adaptive storyline. It provides the adaptation types (i.e. VEAS) that are used for this storyline, the VERTs are translated into rules and facts that can then be uploaded on the reasoning engine. The second element is the reasoning engine which reasons, using these rules and facts, on the storyline and what the learners do. The third element is the interpreter engine responsible to interpret the results given by the reasoning engine and communicate to the MVLE what adaptation state should be applied to which virtual object. It also does the opposite by receiving information from the MVLE and then knows what to relate to the reasoning engine based on the learner's profile. The fourth element is indeed the Multi-learner Virtual Learning Engine. Note that MVLE has been implemented using OpenSim [22] and the reasoning engine has been implemented using Prolog Engine [23]. The interpreter has been written as Web Service written in Java. The authoring tool has been developed using Air Adobe builder [24]. The MVLE architecture is a distributed architecture.

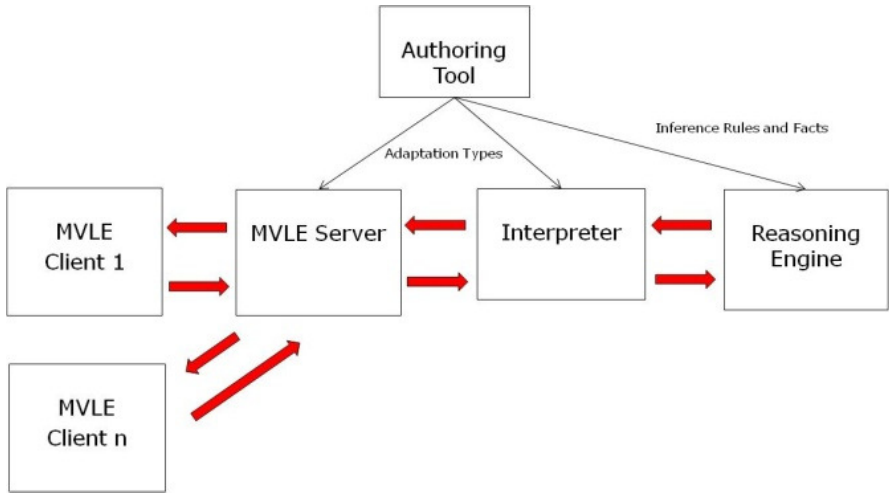

Fig. 1. Diagram of the structure of the system

5 Case Study

In this section, we will show some initial results based on our software architecture which implements the SALVE approach. The storyline is a course aiming at introducing the Etterbeck campus of the Vrije Universiteit Brussel (located in Brussel, Belgium) to new students. Every new students are represented by an avatars.

5.1 Creating the storyline

To create an adaptive storyline, the author of the course uses our authoring tool to compose his storyline (i.e. the course). We will now introduce elements of that storyline based on that case study. From figure 2, it can be seen that the storyline is deployed over a timeline axis. This gives the author an overview of the course in a linear way. This timeline is split into a number of stages identified by a number. Each stage corresponds to a step in our approach. We will now review them. The first stage is a *Start Step* which is represented graphically in the storyline by a red triangle. It allows the author to introduce the time to complete the course and the initial indications on how to reach the targets. It also contains a number of initial parameters like for instance, colours used to identify group of learners or individual learners.

The second stage in the storyline corresponds to a *Next Step* in our approach which is represented graphically by a blue round. In the case study, it contains the fact of visiting the *WISE* department.

The third stage corresponds to *parallel steps* which has a number of sub-steps that can be run in parallel bringing to non-linear way of the story. They are represented graphically in the storyline by a green square placed perpendicularly to the storyline In each sub-steps, a number of VEAS and VERTS can be added to bring adaptations.

For instance in one of these sub-steps, the author can define first the conditions that will trigger adaptations. Here if the learner represented by the blue colour, has not visited the athletic track (see figure 3), then some annotation should be displayed to the learner (see figure 4) telling him what to visit. Note that in our approach, the storyline is also displayed in full text (see bottom of figure 2, *Description of the course*).

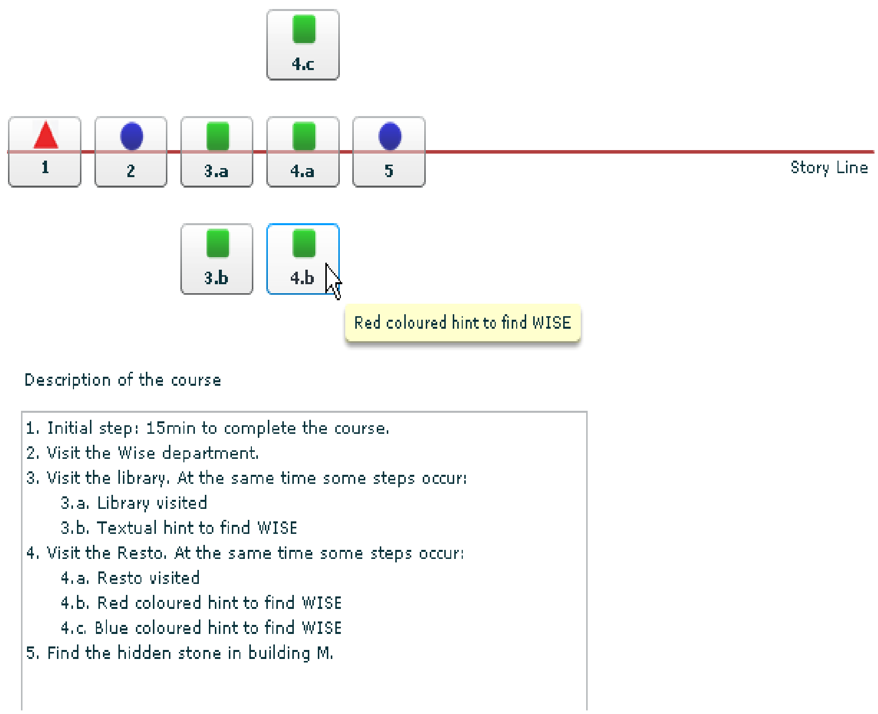

Fig. 2. Storyline and textual description of the course

In the SALVE approach, for each condition, we can also have several VEAS. We have represented that by means of a graphical notation to make a simple visual representation for the author (see figure 5). For each VEAS added, there will be an arrow from the condition to the VEAS, which is represented by a rhombus and its label (see figure 5).

5.2 Running the storyline

The course is ready to be run once the authoring tool has generated all the necessary adaptation (VEAS) for the OpenSim platform and for the interpreter. And once the different VERTs has been translated into prolog rules and uploaded to the reasoning engine. Figure 6 shows the first kind of adaptation where a specific learner receives a text message telling that he has visited the library and what he should visit next depending on his profile. Figure 7 shows an adaptation where the *WISE* department is highlighted in the colour specific to a group of learners to attract their attention by showing them where the *WISE* department is located. This adaptation is shown only when the time has passed a certain threshold so that they can hurry and stop loosing time wandering where the *WISE* department is located.

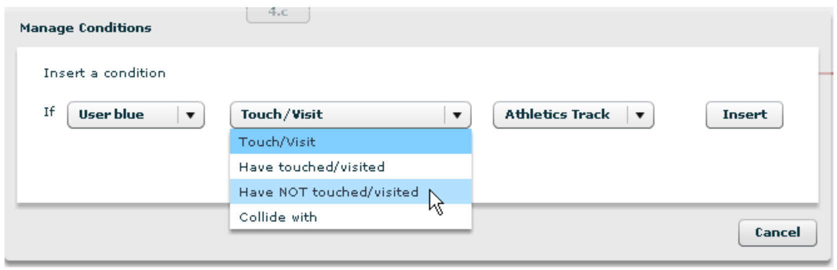

Fig. 3. Introduce a condition

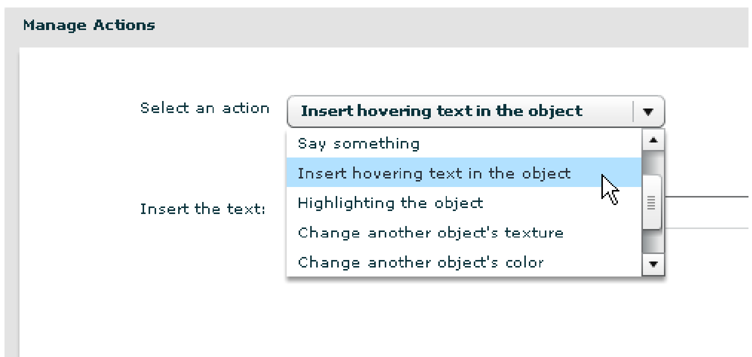

Fig. 4. Add a VEAS

6. Conclusions and Future Work

This paper presented an approach called SALVE aiming at creating adaptive storyline for Virtual Learning Environment in the context of multi-learners. It presented a number of adaptation (VEAS) and how they can be used. It also presented some initial results. The use of adaptation in multi-learner virtual environment is challenging as learners share the same space and any change will influence all the learners. It is important to find ways to make sure that adaptation can be tailored not only to a specific learner but also to a group of learners. Future work will be to address these challenges further and assess the usability of the authoring tool.

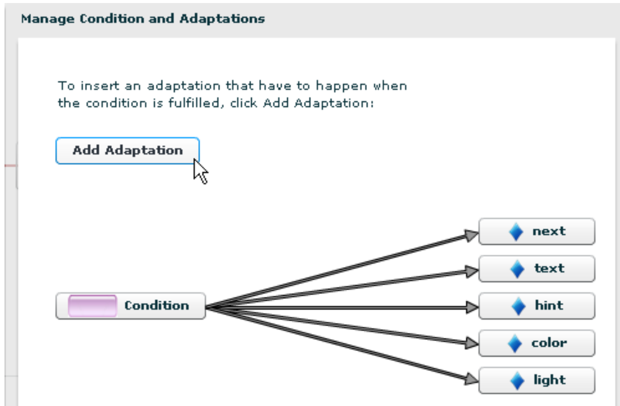

Fig.5. Add several VEAS

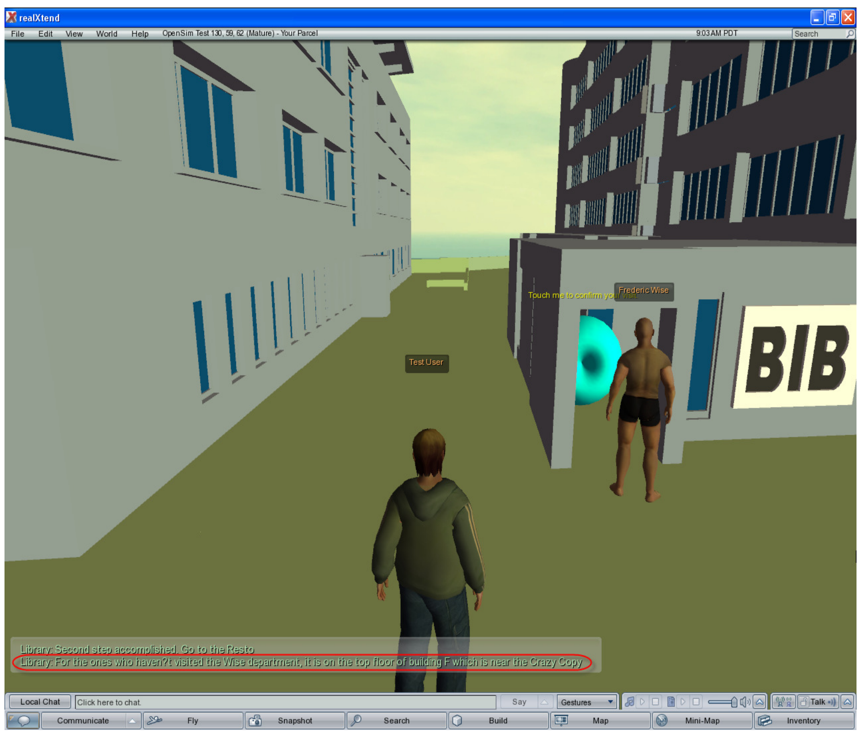

Fig. 6. Textual hint to the learners

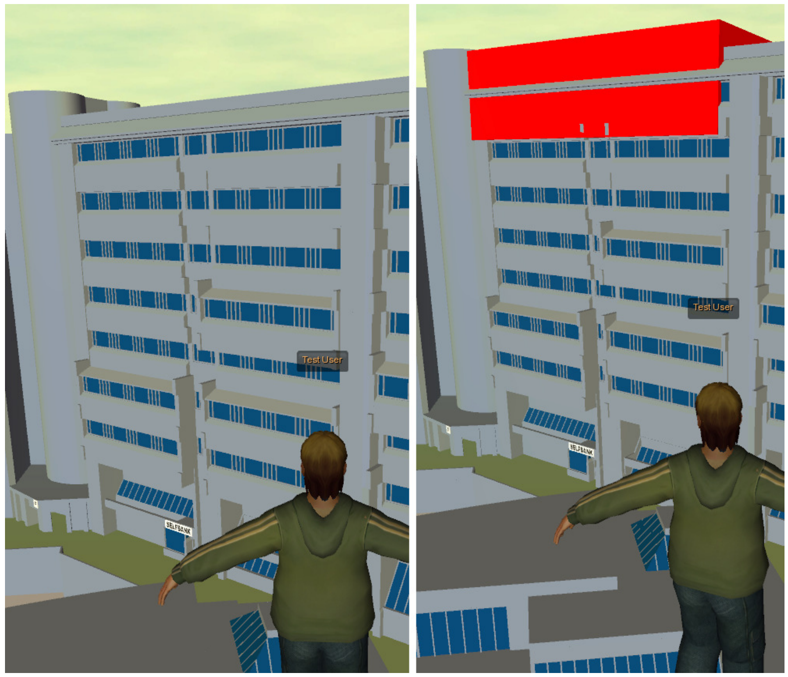

Fig. 7. WISE department highlighted in red

References

1. SecondLife, http://secondlife.com/
2. PIVOT, http://www.daden.co.uk/pivote.html
3. Bricken, M., Byrne, C. M.: Summer students in virtual reality: A Pilot Study on Educational Applications of Virtual Reality Technology. In: Wexelblat, A. (ed.), Virtual Reality: Applications and Exploration. New York: Academic Press, Inc., pp.199--218 (1993)
4. Virvou, M., Katsionis, G: On the usability and likeability of virtual reality games for education: The case of VR-ENGAGE. In: Computers & Education, Volume 50, Issue 1, January 2008, pp. 154--178 (2008)
5. De Troyer, O., Kleinermann, F., Ewais, A.: Enhancing Virtual Reality Learning Environments with Adaptivity: Lessons Learned. In: HCI in Work & Learning, Life & Leisure(USAB 2010), Austria, pp. 244--265 (2010)
6. De Troyer, O., Kleinermann, F., Pellens, B. Ewais, A.: Supporting Virtual Reality in an Adaptive Web-Based Learning Environment. In: EC-TEL 2009 : Learning in the synergy of Multiple Disciplines, LNCS, vol. 5794, Springer-Verlag, France pp. 627--632 (2009)
7. Grutzmacher, B., R. Wages, Trogemann, G.: An Authoring System for Non-Linear VR Scenarios. In: The 9th International Conference on Virtual Systems and Multimedia. Montreal, Canada, pp. 8--16 (2003)

8. Wages, R., B. Grutzmacher, Trogemann, G.: A Formalism and a Tool for Diverging Requirements in VR Scenario Modeling. In : The 13th International Conference on Artificial Reality and Telexistence. Keio University, Tokyo, Japan, pp. 8--14 (2003)
9. Willemsen, P.J.,: Behavior and Scenario Modeling for Real-Time Virtual Environments. PhD Thesis, University of Iowa (2000)
10. Bartley, C.R.: A Visual Language for Composable Simulation Scenarios. Master Thesis, Air Force Institute of Technology, Ohio (2003)
11. Ohnishi, A., Potts, C.: Grounding Scenarios in Frame-Based Action Semantics. In: Proceedings of the International Workshop on Requirements Engineering: Foundation of Software Quality, Switzerland pp. 12-18. (2001)
12. Canales, A., Peña, A., Peredo, R., Sossa, H.,, Gutiérrez, A.: Adaptive and intelligent web based education system: towards an integral architecture and framework. Expert Systems with Appplications. 33(4), 1076--1089 (2007)
13. Brusilovsky, P., Hughes, S., Lewis, M.: Adaptive Navigation Support in 3-D E-Commerce Activities. In: Proceedings of Workshop on Recommendation and Personalization in eCommerce at the 2nd International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'2002), Malaga, Spain, pp. 132-139. (2002)
14. Chittaro, L., Ranon, R.: Adding Adaptive Features to Virtual Reality Interfaces for E-Commerce. In: Proceedings of International Conference on Adaptive Hypermedia and Adaptive Web-based Systems. LNCS, vol. 1892, pp. 86-97. Springer-Verlag, Berlin (2000)
15. Chitttaro, L., Ranon, R.: Dynamic Generation of Personalized VRML Content: a General Approach and its Application to 3D E-Commerce. In: Proceedings of 7th International Conference on 3D Web Technology, Web3D 2002, pp. 145-154. ACM Press (2002)
16. Chittaro, L., Ranon, R.: Adaptive 3D Web Sites. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web - Methods and Strategies of Web Personalization, LNCS, vol. 4321 pp. 433-464. Springer-Verlag, Berlin (2007)
17. Chittaro, L., Ranon, R.: Adaptive Hypermedia Techniques for 3D Educational Virtual Environments. IEEE Intelligent Systems, vol. 22, issue 4, pp. 31-37 (2007)
18. Chittaro, L., Ranon R.: An Adaptive 3D Virtual Environment for Learning the X3D Language. In: Proceedings of the 2008 International Conference on Intelligent User Interfaces (IUI 2008), pp. 419-420. ACM Press, New York (2008)
19. Santos, D., C. T., Osorio, F. S.: AdapTIVE: An Intelligent Virtual Environment and Its Application in E-Commerce. In: Proceedings of 28th Annual International Computer Software and Applications Conference (COMPSAC'04), pp. 468-473. (2004)
20. Celentano, A., Pittarello, F.: Observing and Adapting User Behaviour in Navigational 3D interface, In: Proceedings of 7th International Conference on Advanced Visual Interfaces (AVI 2004), pp. 275-282. ACM Press (2004)
21. Ince, S.: Writing for video games. A & C Black Publishers Limited (2006)
22. OpenSim, http://opensimulator.org
23. Prolog,: http://www.swi-prolog.org
24. Adobe AIR, http://www.adobe.com/es/products/air