Chapter IV
Observation as a Requisite for Game-Based Learning Environments

Jean-Charles Marty
University of Savoie, France

Thibault Carron
University of Savoie, France

Jean-Mathias Heraud
Graduate Business School of Chambery, France

ABSTRACT

In this chapter, the authors propose a Game-Based LMS called the pedagogical dungeon equipped with cooperation abilities for particular activities. The main purpose of this chapter is to explain how to keep awareness of the on-going activities while remaining involved in the game itself. The difficulty is to provide the teacher with this awareness in an immersive way, making the teacher more involved in the game when s/he obtains feedback on the activity. The chapter is split into three sections. The authors propose a first section that deals with the description of our view of learning games illustrated through the pedagogical dungeon. They briefly describe the generation of a dungeon from activity preparation and the links between pedagogical concepts and their representation in the dungeon. The second section concentrates on the observation features needed in these environments in order to obtain interesting facts on what is going on. The authors need to collect traces of the collaborative activity during the enactment phase. They describe the trace life cycle and explain how facts constituting awareness can be calculated from the traces. The third part deals with the restitution of this awareness to the teacher. The problem here is to find an appropriate way to represent awareness both of students’ knowledge and behavior. This awareness must be perceived through appropriate graphical representations to preserve
Observation as a Requisite for Game-Based Learning Environments

INTRODUCTION

Nowadays, Learning Management Systems (LMS) offer functionalities that are recognized as being valuable from different points of view. For instance, students can learn at their own speed. These environments also allow the teacher to evaluate specific activities in a uniform way. However, although these environments enable powerful features, they also incur two major kinds of criticism. The first one deals with the non-attractiveness of such environments for the students, as very often students tend to consider them as unexciting. The second one relates to the lack of awareness (see (Greenberg, Gutwin, & Cockburn, 1996) for a definition of awareness) from the teacher's point of view as shown by (Kian-Sam & Chee-Kiat, 2002): s/he no longer has the usual and helpful student's feedback (eye contact, general attitude). As reported in (Hijon & Carlos, 2006), where the authors compare the built-in student tracking functionality of various CMS tools, this functionality is far from satisfactory. The regulation of the activity is thus much more difficult.

Concerning the first point, agreeing with Vygotski’s school of thought and activity theory, we consider that the social dimension is crucial for the cognitive processes involved in the learning activity. Consequently, the question was how to enhance the social dimension in such environments.

Observing the emergence and success of online multiplayer games with our students – the so-called “digital natives”-[Summit on educational Games, October 2006 (http://www.fas.org/gamesummit/)], more generally in the world (Rosenbloom, 2004) and even in education (Purdy, 2007), (Scott, 2007), it was decided to use one as a support for our course. This led us to apply the metaphor of exploring a virtual world, a dungeon, where each student collects knowledge related to a learning activity. It is our view that the way to acquire knowledge during a learning session is similar to the exploration of a dungeon. This approach reveals advantages such as a recreation-type process, a large usability of the tool or its adaptation to the student’s speed. Such game-based learning environments can thus be proposed as a way of implementing learning sessions, in which teachers can prepare and follow a pedagogical scenario (see (Kinshuk, & Patel, 1996) for a definition of a pedagogical scenario).

Concerning the second point; for usability purposes, it is essential that Computer-Based Education offer the possibility of monitoring the activity performed by the students and of obtaining information or feedback about it. For example, being aware of the learning progression of each student is an important goal for the teacher. Here, we explain how we can avoid the loss of perception for the teacher in these environments.

In this chapter, we propose a Game-Based LMS called the pedagogical dungeon equipped with cooperation abilities for particular activities (see (Dillenbourg, Baker, Blaye, & O’Malley, 1996) for a list of cooperation abilities). The main purpose of this chapter is to explain how to keep awareness of the on-going activities while remaining involved in the game itself. The difficulty is to provide the teacher with this awareness in an
immersive way, making the teacher more involved in the game when s/he obtains feedback on the activity. The chapter is split into three parts. We propose a first part that deals with the description of our view of learning games illustrated through the pedagogical dungeon. We briefly describe the generation of a dungeon from activity preparation and the links between pedagogical concepts and their representation in the dungeon. The second part concentrates on the observation features needed in these environments in order to obtain interesting facts on what is going on. We need to collect traces of the collaborative activity during the enactment phase. We describe the trace life cycle and explain how facts constituting awareness can be calculated from the traces. The third part deals with the restitution of this awareness to the teacher. The problem here is to find an appropriate way to represent awareness both of students’ knowledge and behaviour. As stated previously, this awareness must be perceived through appropriate graphical representations to preserve the “immersion” property, implying that these representations must be directly present in the game.

**DESCRIPTION OF THE EDUCATIONAL PLATFORM: THE PEDAGOGICAL DUNGEON**

In this part, we describe our game-based platform. We first address the links between an educational activity and this game. Then, we demonstrate how the different participants interact with this game. Although the concepts presented in this part are not key research issues, we prefer to explain them to the readers in order to make easier the comprehension of our approach presented later and linked to the observation and awareness in the collaborative activity.

**Links Between a Learning Session and the Objects of the Dungeon**

We have chosen to derive a set of principles from a formal theory of Human Work Activities called Activity Theory (see (Dunne, 1996) for a definition of Activity Theory) issued from (Vygotski, 1934) proposals. In this theory, the social dimension is crucial for the cognitive processes involved in the learning activity. A learning activity consists of one or more (sub) activities linked and ordered to achieve a given pedagogical goal. Actors (students or teachers) can perform these (sub) activities when their associated conditions (or prerequisites) are satisfied. They carry out these activities in collaborative spaces called arenas, through social interactions or through personal actions. An activity is mediated by tools (such as communication tools or evaluation tools) and uses artefacts (defined in (Dunne, 1996)).

To enhance this social dimension, we have chosen to put the students together in a common virtual environment during the entire learning process. In order to link the game world to the learning one and according to (Hainley & Henderson, 2006), we propose in this section to link the objects used in our game-based framework with the concepts that we usually find in a learning system. Table 1 summarizes these links.

**Breakdown of a Learning Session: Rooms and Topology**

The learning session (or learning activity) is very often split into different activities. It is the case when the teacher proposes to her/his students a set of exercises linked together in order to reach a pedagogical goal. Each activity has its own local goal, generally a concept to acquire. For a student, performing all the activities ensures that s/he has reached the general goal of the session, i.e. s/he has gained the knowledge associated with the session.
Table 1. Correspondence between AT concepts and game-based LMS representation

<table>
<thead>
<tr>
<th>Classical concept in the activity theory</th>
<th>Corresponding representation in our Game-Based LMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arena / Collaborative space</td>
<td>Dungeon for the learning activity</td>
</tr>
<tr>
<td></td>
<td>Room for a (sub) activity</td>
</tr>
<tr>
<td>Link between activities</td>
<td>Corridor</td>
</tr>
<tr>
<td>(sub) Activity</td>
<td>Crystals (Exercises)</td>
</tr>
<tr>
<td>Condition / Requisite</td>
<td>Room Door</td>
</tr>
<tr>
<td>Resources</td>
<td>Knowledge Spheres</td>
</tr>
<tr>
<td>Assessment, Validation</td>
<td>Door Key</td>
</tr>
<tr>
<td>Communication tool</td>
<td>Chat window</td>
</tr>
<tr>
<td>Actors</td>
<td>Avatars (teachers, students)</td>
</tr>
</tbody>
</table>

The dungeon represents the place where the learning session takes place. A particular dungeon is dedicated to a particular learning activity, for a particular subject. Each room of the dungeon represents the place where a given (sub) activity can be performed. The dungeon topology represents the overall scenario of the learning session, i.e. the sequencing between activities. There are as many rooms as actual activities, and rooms are linked together through corridors, showing the attainability of an activity from other ones. An example of a scenario seen as a dungeon topology is presented in Figure 1.

Application to the Dungeon: Use Case 1 - Creation of a New Pedagogical Session by the Teacher

The creation of a pedagogical session is not an easy task for the teacher. This activity can be seen as the creation of a scenario, usually written with IMS-LD described in (Koper, Oliver, & Ander-
Observation as a Requisite for Game-Based Learning Environments

son, 2003) or more flexible languages like LDL proposed by (Ferraris, Martel, & Vignollet, 2007).

If the teacher wants to construct a pedagogical session in the dungeon, s/he interacts directly with a session builder. This tool allows the four creation steps of a scenario:

- The definition and type of activities; for instance, an activity can be collaborative.
- The description of the available resources for each activity. Resources could be either local files (content included within the scenario’s definition) or links to on-line material. These files usually explain the topic of the activity. The teacher chooses the most appropriate form for these resources: a simple text, videos or even simulation applications, as is the case in (Michelet, Adam, & Luengo, 2007).
- The definition of the validation procedure for each activity. Obtaining a key related to an activity depends on the evaluation of the activity. For each activity, the teacher can choose how to evaluate it. The simplest way to obtain a key is just to read a text. But, most of the time, the student must answer a question or a set of questions. Each of these questions can be a Multiple Choice Question or an open question. In this last case, the teacher will be in charge of validating the answers to that question. We would also point out that questions could be collaborative, in which case the whole team gives the answers. We have developed here a simple way to obtain the keys.
- The definition of the constraints on the activities (organisational and logical temporal links).

According to these constraints, the map of a dungeon is automatically generated and saved. (see (Carron, Marty, & Heraud, 2008) for details). Figure 2 is an example of the result of such a generation.

![Figure 2. A dungeon map](image)

**Enactment of a Learning Session with Students**

Actors (students or teachers) can move through the dungeon, performing a sequence of sub activities in order to acquire knowledge. Activities can be carried out in a personal or collaborative way: students can access knowledge through resources (documents found inside the game), via help from teachers, or from work with other students. The dungeon can be flexible. For instance, "teleportation portals" can lead to new rooms created dynamically.

Each room is dedicated to an activity. One can find explanatory resources such as texts, links, and videos. These provide the student with useful information. The student reaches the local goal of the activity if s/he answers a quiz successfully. This quiz is thus also located in the room.

As users move through the dungeon, they can meet other students or teachers involved in the same session. When a student is in the same room as another one, it only means that these students are performing the same activity. They can of course access the resources at the same time.

Each room can be accessed through doors. These doors are the guardians of the activity. They ensure that the student has the necessary
prerequisites to perform the activity correctly. When users answer a quiz correctly, the associated key is obtained.

Activities need not necessarily be ordered in the dungeon. However, most of the time, they are well ordered: it is quite rare for a teacher to provide the students with a set of exercises without any order. By ordering the activities, teachers may want either to define an order representing a progressive approach to the general goal of the session (logical order), or simply to force the group to carry out the activities in the same order with the purpose of following the students more easily (temporal order). When users play out a session in the dungeon, this ordering is ensured by the fact that they have to have obtained the key from previous activities before entering a new room.

Application to the Dungeon: Use Case 2 - Embarking on the Quest: Moving, Answering Questions, Obtaining Keys

In our virtual environment, we represent a student by an avatar (see avatar choice in Figure 3) whose characteristics can evolve dynamically over time, as we will see in the third part of the chapter.

Most of the time, a student is present in a virtual room representing an activity. S/he can access several resources related to the activity.

Figure 3. Avatar choice window for the student

In Figure 4, two students and the teacher (with the helmet) are present in a room. Touching a sphere/globe item (a resource) opens a text window with explanations or provides a web link, a file, etc. Touching a crystal item proposes an exercise, a test or a quiz. A correct answer to a crystal question generally gives the student a key to open the door and lets her/him continue the quest (see Figure 4).

The visualisation is updated in real time and the student may move inside the room and see other avatars move and progress in the dungeon. When a question is related to the concept presented in the room, clues may be found in the resources displayed in the room. The answer can be automatically identified as correct (closed question or key words present). In this case, the result is instantly notified to the student by the system and a door is possibly opened (see Figure 5). But very often, the involvement of the teacher is necessary. S/he corrects the exercise dynamically, can add remarks, and validates the answer or not. Whatever the case, the student is notified via a window (see Figure 5 on the right) containing all the stated information.

An overall view from above (mini-map) is always supplied for the teachers during the game (Figure 2). This view is dynamic, since one can see all the users involved in this pedagogical session moving through the different rooms. This input provides the teachers with awareness about the on-going activity. The student’s view is restricted: s/he can see only the rooms that are accessible to her/him (i.e., the rooms whose keys s/he possesses).

Collaboration in the Dungeon

The teacher may want several activities to be collaborative. In that case, the rooms associated to these activities are collaborative places. These places require the students to answer in groups as indicated on the access door to the room. The advantage is to facilitate the exchange of data
Observation as a Requisite for Game-Based Learning Environments

between students, making them asking questions, co-resolving a problem. As in traditional teaching, these collaborative places offer the opportunity to create small groups in order to take advantage of the abilities of the various members of the group, explaining their resolution strategy to each other.

Currently, a chat facility is provided in the dungeon rooms (see the translucent window in Figure 4), but we can of course imagine other collaborative tools available in the collaborative places (shared space, forums, etc.). If the teacher uses collaborative work in a session, s/he must set up teams of students: students belonging to the

Figure 4. Student view of the dungeon and answering a crystal question

Figure 5. Successful answer (left) and teacher validation (right)
same team are supposed to carry out collaborative activities together. The teacher may thus create groups and dispatch students into them.

In collaborative rooms, the quiz is also collaborative. The crystal hiding a group activity has a specific colour and notifies the first student to arrive (see Figure 6). Students in the same team must all be present in the room. They may exchange via the chat tool before answering the question. In the event of a correct answer being given for a collaborative quiz, a collaborative key is provided to all the members of the team.

As in “traditional classrooms”, a student may also collaborate with a teacher, for instance if s/he needs help from her/him.

Experimenting the Dungeon

Our research work is always situated between two observations/experiments: one, which serves as a starting point for reasoning, and the other, which serves as a conclusion. From each experiment, new phenomena result which must be observed and so on. Therefore, the pedagogical dungeon has been experimented during several practical works with real classrooms at the University of Savoie and the Graduate Business School of Chambery, France. Some of the screenshots shown previously are taken from these experiments carried out with the pedagogical dungeon.

The very first experiment with the pedagogical dungeon aimed to validate the overall approach and to test the system technically. The pedagogical session modelled concerned a computer science lesson about operating systems. Several independent concepts were set out to control an operating system through console commands. The objective of the work was to practice such commands and to verify that the links with theoretical concepts had been acquired. The students were 18 years old and familiar with computer use. The students were working on Linux, and had to find or test some text commands in the console.

During the experiment, about fifteen students and their teacher were present in the classroom. Each student worked on a station equipped with the student client software and the teacher used the teacher client software. The students were allowed to consult part of their lesson or web sites (files or URLs found during the exploration of the desktop).

Figure 6. Student’s view: “Waiting for a group companion to begin the test”
dungeon). The students were informed that all their actions would be observed through traces. The students were free to refuse this observation but everyone agreed to follow the proposed protocol. They were explicitly allowed to communicate through the chat tool provided with the system. An avatar of the teacher available for students’ questions was always present in the dungeon.

As a result, students used the proposed environment without any explanation. Moreover, forms filled in by the students just after the experimentation gave us interesting feedback about the software itself and suggested possible improvements to the interface. However, the main goal of the experiment remains to establish whether such an environment improves the learning process. From the students’ point of view, the feedback was very positive: a similar “classical” practical exercise (i.e. without the support of such an application) had been proposed before to the same students. They were much more enthusiastic about the system version: the multiplayer aspect was a great factor of motivation and commitment for the students. An informal competition appeared between users exploring the dungeon. The chat tool was used intensively to communicate about the exercises, boosting the competition and reinforcing the feeling of immersion.

This description of the game gives an idea of how a pedagogical session can take place in the dungeon. The basic rules of the game are well established but, as stated previously, the teacher is used to perceiving informal information in a real classroom. She often wants to observe specific information or behaviour and providing her/him with the expected feedback on the overall activity is still a research problem. Our approach consists in taking advantage of the traces left by the actors participating in the mediated learning activity to calculate awareness indicators for the pedagogical dungeon. We describe this idea more precisely in the following section.

**OBSERVATION**

The dictionary defines observation as the act of making and recording a measurement. In traditional teaching, we are always observing and monitoring students; measuring progress, confusion, or students’ motivation. In a computer-based learning environment, this rich type of observation is no longer possible. However, another kind of observation, based on the trace left by the student on their computer, is possible. From now on, we will refer to this specific type of observation. The tracing activity is an appropriate means of reflecting in-depth details of the activity and of revealing very accurate hints for the teacher. Unfortunately, traces are objects which are very difficult to manage and understand. We propose first to briefly list the problems connected with traces and expose how to deal with them.

**Facts from Experiments**

**Fact 1: Log Files are Rich but Correspond to Information which is Difficult to Exploit**

A first aspect to consider, central to the observation area, is the form of the traces. Many e-learning Platforms or Learning Management Systems are based on Web Servers (Zaïane & Luo, 2001), (Burton & Walther, 2001). These servers easily supply logs (information concerning the connections on this server) stored in specialised files.

Here is, for instance, a line in the SQUID format, extracted from the traces provided by our local e-learning platform used by thousands of users.

```
```
It is obvious that these traces are not directly interpretable. They should be transformed and rewritten, in order to make their understanding possible. For instance,

“Connection to the e-learning platform from the university”.

**Fact 2: Traces Need to be Transformed in an Organised Way**

In order to manage this enormous amount of fine-grained information better, we specified a transformation chain allowing the manipulation of traces (Figure 7). The main purpose is to reach a good level of granularity, allowing a better comprehension of the user behaviour (Loghin, Carron, & Marty, 2005).

This chain proposes several functionalities to manipulate the traces: filtering in order to reduce the huge quantity of logs, aggregation in order to change the level of granularity (abstraction) of the traces, transformation into a uniform format in order to take into account several log formats (SQUID, APACHE, I2S), or storage in a database and use of a Database Management System through SQL requests.

**Fact 3: Traces Contain Hidden Information; Searching them can be an Interesting Avenue of Research**

The approach presented above is valid, since the analyst knows exactly what s/he is searching for and if s/he is able to express it through the proposed interface. From usage of the tool, we can say that there is a need for other approaches, especially when the analyst or the teacher does not know precisely what s/he would like to observe. This is the case, for instance, when the analyst tries to discover new types of usage. In that situation, we are faced with a new issue, one in which the information included in the traces contains hidden behaviour patterns which are to be revealed. Tools implementing “sequence mining” algorithms, and providing significant patterns, must be powerful enough to explore very large data sets (e.g., 1 GB per week for approximately 15,000 people using our e-learning platform).

**Figure 7. Transformation chain for manipulation of traces**
Fact 4: In Order to Understand the activity better, as well as the Links with Predefined Learning Scenarios, Multiple Sources for Traces should be Considered

As mentioned in the introduction, a certain amount of research linked to pedagogical platforms concerns the formalisation of educational scenarios (Koper et al., 2003). The teacher frequently foresees a sequence of activities to be performed during the learning session. This sequence, also called a scenario, guides the session, and it becomes crucial to compare the learners’ activities and the predefined scenario (France, Heraud, Marty, & Carron, 2005). This comparison allows one to provide the teacher with awareness of the ongoing activities, and to improve the scenario itself (Marty, Heraud, Carron, & France, 2004).

This is not an easy task, since the users can use simultaneously tools that are not integrated in the educational platform (forums, web sites, chat rooms). We do not want to restrict our understanding to the tasks included in the predicted scenario. We want to widen the sphere of observation, so that other activities performed by a student are effectively traced. Even if these activities are outside the scope of the predicted scenario, they may have helped him/her to complete the exercise or lesson. We thus need to collect traces from different sources. It is therefore interesting, from a general point of view, to be able to take into account more than one source of data. Such an approach allows the deduction, from the multi-source traces, of unforeseen behaviour, as demonstrated in (Marty, Heraud, Carron, & France, 2007).

Fact 5: Visualisation Improves the Teacher’s Awareness

Detecting potential problems as soon as possible is a crucial issue. In order to alert the teacher to the fact that the collaborative activity is not progressing as expected, we need to compare the traces representing the actual activities with the ones mentioned in the predefined scenario and to try to establish links between them. It is essential for the teacher to have a view of what is going on, in order to be able to react to given situations. New observation goals can also appear during the session. For instance, it can be useful to observe the status of the students during the first part of the session and to synchronise them before starting the second part of the session, making sure that everyone has acquired the required concepts.

This adaptive observation, needing high flexibility from the system, can be implemented through agents. A set of “pedagogical observation agents”, set up on the students’ computers, inspects certain user actions (the ones that are on focus for the observer) and notifies an awareness agent before invoking a visualisation agent to provide the teacher with the appropriate information. This distributed system is thus able to collect the significant logs directly on the machines through specialised agents (Loghin, Carron, Marty, & Vaida, 2008). The visualization agent interprets the traces sent by the observer agents in order to display them on a dashboard for the teacher, through indicators computed from activity traces.

Architecture of the Observation Software

From the facts pointed out in the previous section, we propose a process, namely the observation lifecycle, suited to taking these points into account. This process has three phases that can be described as follows:

A collecting phase, wherein relevant traces are identified and collected;

A transformation phase (structuring, abstraction) of collected data in order to make the rough traces more explicit and more understandable for the observer (researcher, teacher, or student);

And, a visualisation phase, whereby visualisation techniques will be used in order to reveal the semantics from the traces, make them easier
to understand and enable an analysis from a particular viewpoint. This phase aims at facilitating the interpretation of the on-going activity by a non-specialist.

We base our work on a model elaborated in collaboration with the SILEX Team of the LIRIS laboratory. This model, called Trace-Based System (TBS), defines the different modules associated with the different phases mentioned previously.

Figure 8 illustrates the process which allows the observer to interact with a traced e-learning platform in order to visualise and regulate the activity using the traces. The observer plays the role of a “trace composer”. S/he furnishes both the pedagogical scenario possibly expressed with IMS-LD (Koper et al., 2003), and the description of the experiment by pointing out the analysis needs (Carron, Marty, Heraud, & France, 2006). S/he thus sets up the e-learning platform by adjusting collection and transformation tools. Then, the experiment can be enacted, providing the analysts with usage feedback.

Collection Phase

As demonstrated in Figure 8, the collection phase is prepared before using the TBS and consists in gathering the traces generated in the e-learning platform. Trace collection is a complex computer science problem, due to the large volume of rough traces that it is possible to collect. This collection can be carried out through instrumented software according to the trace composer’s intentions (Talbot & Courtin, 2008) or through files generated by the operating system, or through dedicated spy software, such as key loggers. Another problem related to trace collection is the heterogeneity of rough traces, which requires studying a way to model them (Iksal & Choquet, 2005).

Transformation Phase

The transformation phase is performed inside the TBS. The trace being an object in itself, the notion of Trace-Based System has emerged over...
the last few years, in order to allow and facilitate
the exploitation and the interpretation of traces
(Laflaquiere, Settouti, Prie, & Mille, 2006). The
functionalities of such systems therefore concern
trace manipulation. From the rough traces, a TBS
offers a set of operations among these objects:
filtering, joining or abstracting them. When the
results of these operations are still traces, they
remain inside the TBS and they can possibly be
used for other manipulations. A TBS also offers
services allowing trace organisation, such as stor-
age or historical mechanisms. Research questions
related to this phase meet trace cleaning (Cooley,
Mobasher, & Strivastava, 1999), trace aggrega-
tion according to temporal (Marquardt, Becker,
& Ruiz, 2004), semantic or syntactic constraints
(Tanasa & Trousse, 2004), trace rewriting or
modelling (Champin, Prié, & Mille, 2004).

Trace Visualisation

The visualisation phase consists in making re-
quests among traces and in visualising traces.
These visualisation tools are part of the interface
between the TBS and the trace composer. We
have decided to situate the visualisation and the
request system outside the TBS, since these tools
do not fit the definition of trace manipulation
as given in (Laflaquiere et al., 2006). Indeed,
visualisation techniques produce results that are
not traces. Visualisation consists in elaborating a
graphical representation, adapted to the analyst’s
objective, from traces contained in the TBS. This
representation can take many forms, such as a
temporal 2D visualisation of a trace (France,
Heraud, Marty, & Carron, 2006), or of several
traces (Mazza & Dimitrova, 2005), or a spatial
3D visualisation (Cugini & Scholtz, 1999). The
visualisation system relies greatly on the analyst’s
objective. For instance, the visualisation system
must be able to provide the analyst with a real
time visualisation of the enactment of the users’
activities, and particularly to detect and show the
users in difficulty. The system must also provide
him/her with information about activities causing
problems for these users. Finally, a visualisation
of individualised paths showing the path of
activities for each user must allow the analyst to
make an intermediate assessment of the users’
progression.

In this part, we have described the trace life
cycle and explained how facts can be calculated
from the traces left by the actors. These facts can
be presented to the user in order to make him/her
aware of what is going on.

Experimenting the Observation

A second experiment with the pedagogical dun-
geon aimed to validate the observation tools.
The pedagogical session modelled concerned a
lesson about operating systems. Concerning the
teacher, we noted that the use of such a tool may
be somewhat disturbing: the way of following the
learning progression has entirely changed and
s/he may sometimes be overloaded by the number
of questions from students waiting. The teacher
also complained about the cognitive overload
generated by the system: too much information
was displayed concerning the different events
occurring in the dungeon. Although it was pos-
sible to filter the traces in order to display only
those related to particular events, this required
some additional time from the teacher, who was
already overloaded. This remark raises the ques-
tion of an adaptable control board containing
suitable indicators, enabled or not according to
the context.

Thus, the most interesting point is that the
pedagogical session has to be well prepared and to
be specifically adapted to such environments. As
stated previously, for the teacher, the experiments
revealed a great difficulty in reading and validat-
ing at the same time: a lot of student propositions
are generated from open questions. The whole
session has thus to be well designed in order to
protect the teacher from such an overload: s/he
must also have time to observe the pedagogical
activities and to react dynamically.
This awareness must be perceived through appropriate graphical representations to preserve the “immersion” property, implying that these representations must be directly present in the game. The following part deals with the restitution of this awareness to the teacher and students.

IMMERSIVE CONCEPTS FOR OBSERVING A LEARNING GAME

It is well known that a great part of the success of a game is due to the immersion of the player into its environment. One of our first experimentations (Carron, Marty, Heraud, & France, 2007) showed that it is crucial for a learning game to pay attention to the immersion factor too.

In this part, we will present concepts or artefacts added to enhance learning and to reinforce the feeling of immersion. We first propose to gain awareness concerning the students’ skills and behaviour by introducing new visual representations into the game. We then want to go further by allowing a meta-cognition process to take place via the game. In order to be efficient, the reflexive analysis of the student of his/her own trace must also be made in the same context. We propose a way to carry out this post-experiment activity in an immersive way, too.

Representing Skills through Equipment

We propose to display the skills via visible equipment (pieces of armour, clothes, weapons: a sword, magic wands, a shield) shown directly on the avatar representation inside the game. Each student begins a dungeon without any piece of armour. Each important exercise assessment lets him/her win an item: for example a sword, a hat (diadem for the girls or helmet for the boys). At the beginning, students start almost naked (see Fig 9).

This way of representing a skill presents a dual interest:

First, the student is proud to see his/her avatar evolving and this motivates him/her to obtain new items by exploring new parts of the dungeon. Moreover, another motivating factor for the student is to show his/her skills to other students. The latter are immediately impressed by this new appearance and try to get the same items as soon

Figure 9. Representing skills through equipment
Observation as a Requisite for Game-Based Learning Environments

as possible. As a side effect, it generally pushes the students to communicate and cooperate in order to progress.

Second, this representation is also very useful for the teacher seeking information about the progression of each student in the dungeon (learning session). He/she is thus instantly aware of the progression of each student without needing to activate or launch a specific visualization tool. Crucial pedagogical information is directly displayed within the game. Currently, the teacher may choose to allocate seven different rewards to specific activities of the pedagogical scenario: these are considered as main or fundamental activities and will let the students obtain, in the event of success, a piece of equipment revealing a particular skill. The attribution of items to specific activities is done when designing the pedagogical scenario. The task editor (Figure 10) allows one to address the allocation of awards to activities (identification through a “risk icon”). We can observe in Figure 10 that both solo and cooperative activities may be marked as key activities (unique player or multiplayer item in Figure 10).

In fact, depending on the topology of the dungeon and the path followed by the students, some students may appear amusingly in the dungeon with just a helmet or almost naked with a sword.

More seriously, we have considered here that a skill is something persistent, similar to a diploma, and has also to be reified into a concrete visible object worn by the avatar. The teacher and students are able to see the skills of everybody in the current dungeon. Naturally, the teacher is always fully dressed (see the top avatar in Figure 11).

Seven items are currently available. The last one is a pair of wings that is obtained when all the exercises in the dungeon have been successfully completed by a student. Such students are naturally identified by the others as the “best players” (most competent in the domain) and then possibly allowed to help the others: they may receive a new “tutor” role in order to help the teacher. We have planned to offer them new skills associated with this role (e.g. a teleportation skill) in order to enhance their cooperation abilities.

Figure 10. Allocating rewards to particular activities of a pedagogical scenario
The students’ skills are represented by equipment. Some students exhibit specific types of behaviour during a session such as brilliant, cooperative, laborious, talkative, rigorous, lax, which are important from a pedagogical point of view. In the following paragraph, we propose a way of representing such behaviour.

**Representing Behaviour through Auras**

Behaviour is the explicit way a student has of being of an actor, revealing properties particular to the way this actor behaves in the dungeon, from a general point of view or from a specific one linked to the pedagogical scenario. Representing specific behaviour may have an impact on other persons present in a room or even in the dungeon. As a matter of fact, we can predict some of these types of behaviour from the work on traces explained in the second part of this chapter. We can also assimilate this to a distinctive atmosphere or quality that seems to surround or be generated by a person. We have chosen to represent such behaviour with an aura around the avatar. In Figure 11, two auras are represented. Some snow is falling on the two girls revealing that they are in difficulty and a spiral is turning around the boy meaning that he is talkative. Naturally, the auras are absolutely not linked to the skills.

Different auras have been defined. They can be identified through their different forms (circle, snow, pulsation and spiral) or through their colour (any colour is possible).

The advantages of such a representation are numerous:

First, an aura may represent temporary behaviour and thus may be activated or deactivated when specific conditions are achieved. For example, a student in difficulty generally gives several wrong answers to the same exercise.

Second, a person may exhibit several behaviour patterns at the same time. In this case, the different auras alternate.

Third, the teacher may decide to define, attribute, activate or deactivate some auras dynamically according to his/her pedagogical needs or wishes. The level of triggering may also be adjusted at any time (see Figure 12 for the configuration tool).

We can note that in this case, the teacher has to use an external tool to define new behaviour patterns, decreasing his/her immersion factor (for a short time) but the results of these settings are directly visible through the avatars in the game.

*Figure 11. Representing particular behaviour via auras*
An Immersive Means for Reflexive Analysis: Ghosts

In order to allow a meta-cognition process (Paris & Winograd, 1990) via the game, and the reflexive analysis of the student concerning his/her own trace, we propose to carry out this post-experiment activity in the same context. This process can only be performed seriously if the access to many items is possible: look at correct and wrong answers, perceive past hesitations, time spent on specific activities, knowledge of which resources have been consulted. Meta-cognition can thus be facilitated through replaying the whole session. For this purpose, we have chosen to save all the actions of an avatar in order to be able to replay them later: this is replayed as a ghost player (see Figure 13).

The teacher is able to see a posteriori all the ghosts saved during the session. Although it is possible, the teacher does not replay the session with all the ghosts at the same time but s/he generally selects only a few specific ghosts. As seen in Figure 13, special items are displayed on the ghost when they are obtained and the ghosts blink green for correct answers and red for wrong ones.

Figure 13. Replaying a session with ghosts
Currently, only the teacher may decide to confront the students with their ghosts. Future work will address the possibility of confronting several ghosts, allowing the associated students to exchange information with each other on their ways of carrying out a particular task.

**Experimenting Immersion**

The screenshots shown in this chapter are taken from a real experiment carried out with the pedagogical dungeon based on a project management course assessment. In this third experiment, the whole dungeon was composed of 60 rooms (see the topology with students dispatched in Figure 14). During the experiment, two groups of fifteen students with their teacher were present successively in the classroom.

These students used the pedagogical dungeon for approximately one hour and were very enthusiastic about this kind of assessment. Some possible improvements to the interface were suggested and we observed a great motivation for obtaining items. To detect talkative students and those in difficulty, the teacher activated observation tools in the way stated previously. The students were not informed of the meaning of the auras but they were very proud to show their avatar with auras to other students. Apparently, the meaning of an aura was not the most important aspect for the students. We observed particular collective behaviour patterns. For instance, some students decided to pursue their quest in pairs, using the chat tool systematically in order to share their experiences.

According to the teacher, the pedagogical aim of the assessment (evaluating the knowledge level about project management) was achieved. Moreover, he was able to use the ghosts in order to examine more precisely what particular students had done. Other observation parts and new immersive factors were there simply for testing purposes. For the moment, this first experiment does not enable us to draw firm conclusions about the value and the interpretation of auras from a pedagogical point of view. New observation indicators applied to other experiments will give us more clues to demonstrating the pedagogical justification of such tools with effective learning enhancement. Nevertheless, the current results encourage us to go on with this research work.

*Figure 14. Topology of the dungeon for assessing the Project Management Course*
CONCLUSION

As for classical games, we think that the immersion factor is a key point for all the actors (students and teachers). In this chapter, we have proposed some solutions in order to keep the teacher involved in the game by providing him/her with immersive indicators and tools concerning pedagogical objectives.

Through a game-based learning system equipped with observation facilities, the teacher may define observation indicators before and during the game according to his/her pedagogical needs or wishes. The observation system is generic, and we show how to exploit traces left in a game-based LMS in order to obtain relevant information about the pedagogical session.

For appropriation and acceptability reasons, the feedback of information must be adapted and integrated dynamically within the game. The players and the teacher thus remain immersed in the game. These concepts (items, auras) are implemented in the pedagogical dungeon through indicators. Future work will focus on observation indicator classification, and more precisely on the definition of collaborative indicators. The immersion problem is also of great importance when users examine their own traces. Meta-cognition thus implies a ghost re-player. As stated previously, improvements to this player are required in order to provide a more effective tool.

In this kind of environment, personalization is of primary importance. Actors must have a picture of themselves with respect to their current knowledge, and the behaviour they tend to exhibit. This information makes the adaptation of the environment possible for the users. This topic is of course directly connected with research on user modelling. As a matter of fact, these learning games are widely spreading among the educational community. The user model represents this picture, valid at a given time, which will be refined and changed, according to the results obtained from the different uses of the pedagogical tools.

Certain features of the pedagogical dungeon are particularly interesting for feeding this research. Firstly, in the pedagogical dungeon, we can represent the knowledge acquired but we can also keep traces of the way this knowledge has been acquired. To some extent, the ghosts can be considered as part of the user model. Secondly, the dungeon is general enough to be used in diverse areas (e.g. Geography, English). A link between several learning dungeons is therefore possible. This general approach can benefit students and teachers since it can break down the divisions among the topics taught. For instance, the students’ skills in communication will be available during a computer science lesson, allowing the teacher to propose adapted activities which are more or less collaborative.

REFERENCES


Observation as a Requisite for Game-Based Learning Environments


