Teaching with game-based learning management systems: Exploring a pedagogical dungeon

Thibault Carron
Jean-Charles Marty
Jean-Mathias Heraud
University of Savoie, France

The work reported here takes place in the educational domain. The authors propose a learning environment based on a graphical representation of a course. The emergence of online multiplayer games led the authors to apply the following metaphor to the digital work environments: The method of acquiring knowledge during a learning session is similar to the exploration of a dungeon, where each student collects knowledge related to a learning activity. In the first part of this article, the authors focus on a description of how concepts of learning activities can be represented in the dungeon view. The second part deals with the support of the observation task for the teacher during a learning session and more generally with providing users with awareness. The authors thus propose a multiagent system using data collected from traces resulting from the collaborative learning activity. Finally, this environment allowed the authors to set up experiments with students at their university.

KEYWORDS: collaborative activity; digital work environments; game-based learning environments; learning environment; learning management systems; multiagent system; online multiplayer game; pedagogical dungeon; trace observation

Nowadays, learning management systems (LMSs) 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 teachers to evaluate specific activities in a uniform way. However, although these environments enable powerful features, they also incur two major kinds of criticism (Lowe, 2002). The first deals with the unattractiveness of such environments for students: Very often, students tend to consider them unexciting. The second involves the lack of awareness (see Greenberg, Gutwin, & Cockburn, 1996, for a definition of awareness) from teachers’ point of view, as shown by Kian-Sam and Chee-Kiat (2002): They no longer have the usual and helpful student feedback (e.g., eye contact, general attitude). As reported by Hijon and Carlos (2006), who compared the built-in student-tracking functionality of
various course management system tools, this functionality is far from satisfactory. The regulation of activities is thus much more difficult.

Concerning the first point, agreeing with Vygotsky’s school of thought and activity theory, we believe that the social dimension is crucial for the cognitive processes implied in learning activities. 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 (Federation of American Scientists, 2006)—more generally in the world (Rosenbloom, 2004) and even in education (Purdy, 2007; Scott, 2007), it was decided to use them as 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, the large usability of the tool, and its adaptability to students’ 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, & Oppermann, 2006, 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 activities performed by students and of obtaining information or feedback about them. For example, being aware of the learning progression of each student is an important goal for teachers. Here, we explain how we can avoid the loss of perception for teachers in these environments.

In this article, we propose a game-based LMS called a pedagogical dungeon, equipped with cooperation abilities for particular activities (see Dillenbourg, Baker, Blaye, & O’Malley, 1996, for a list of cooperation abilities). The second section demonstrates the links between pedagogical concepts and their representation in the dungeon. The following section concerns observation of what is occurring in the dungeon; some awareness indicators are provided, thanks to specific observation probes. Moreover, we present some means of providing feedback and retrieving perception, thanks to activity trails or traces left by users (see Marty, Heraud, Carron, & France, 2007, for details on these traces). In the fourth section, the system itself is described: We focus on activity preparation and the generation of the dungeon, explaining our metaphor more precisely and describing each artifact. We then explain the enactment of the game. The last section deals with a description of lessons from the experiment.

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), derived from Vygotsky’s (1978) proposals. In this theory, the social dimension is
crucial for the cognitive processes involved in a 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 artifacts (defined by Dunne, 1996).

To enhance this social dimension, we have chosen to put students together in a common virtual environment during the entire learning process. To link the game world to the learning world, and according to Hainley and 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.

### Decomposition of a learning session: Rooms and topology

The learning session (or learning activity) is very often split into different activities. This is the case when a teacher proposes to his or her students a set of exercises linked together 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 he or she has reached the general goal of the session (i.e., gained the knowledge associated with the session).

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.
Actors (students or teachers) can move through the dungeon, performing a sequence of subactivities to acquire knowledge. Activities can be carried out in a personal or collaborative way: One can access knowledge through documents, with help from teachers, or from work with other students. As we will see later, the dungeon can be flexible. For instance, “teleportation portals” can lead to new rooms created dynamically for instance.

**Achievement of activities**

Each room is dedicated to an activity. Explanatory resources such as texts, links, and videos can be found. These provide students with useful information. Students reach the local goal of an activity if they answer a quiz successfully. This quiz is thus also located in the room. In Figure 2, we can see an example of a room in the dungeon.

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 student, it only means that these students are performing the same activity. They can of course access the resources at the same time.

The teacher may want several activities to be collaborative. In that case, the rooms associated with these activities are collaborative places. Currently, a chat facility is provided in the dungeon rooms, but we can of course imagine other collaborative tools available in these rooms (shared space, forums, etc.). If the teacher uses collaborative work in a session, he or she must set up teams of students: Students belonging to the same team are supposed to carry out collaborative activities.
together. In collaborative rooms, the quizzes are also collaborative. Students on the same team must all be present in the room. They may exchange via chat before answering the question.

As in “traditional classrooms,” students may also collaborate with teachers, for instance if they require help from teachers.

**Sequencing of activities**

Each room can be accessed through doors. These doors are the guards 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. In the event of a correct answer given for a collaborative quiz, a collaborative key is provided to all the members of the team.

Activities must 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 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 must have obtained the key of previous activities before entering a new room.

However, these conditions are not sufficient to ensure the correct progress of the pedagogical activity. As stated previously, teachers are used to perceiving some informal information in real classrooms. They often want to observe specific information or behavior, and providing them with expected feedback on the overall activity is still a research problem. Our approach consists of 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 next section.

Observation and traces of collaborative activities

To obtain acute awareness for the actors, we first situate the kind of expected needs according to a categorization of awareness. We then describe how such a system can be observed by agreeing on a certain number of basic observation features. Finally, we set up a way for combining easily observed basic events to notify more complex events to the users.

Awareness needs in such an environment

Teachers using such educational environments need a means of following the progression of the learning activity accurately and of avoiding the terrible lack of feedback already mentioned in the introduction of this article. Generally, teachers in virtual classrooms perceive less information than in real classrooms, because they do not see their students. We must first point out that it is possible to perceive partially, thanks to relatively classic (in the video game domain) and basic awareness indicators
such as the ones proposed by Nova, Wehrle, Goslin, Bourquin, and Dillenbourg (2007). Because we want to deal in more depth with how the problem of awareness is tackled in the pedagogical dungeon, we propose to examine the different facets that make up the categorization of awareness proposed by Greenberg et al. (1996) (see Figure 3).

Informal awareness concerns the general consciousness of others’ presence, capability, or availability, all things that are implicitly felt when a teacher is in a real classroom. Part of this awareness is brought about by the dynamic representation of users through avatars, but little information can be inferred from the behavior of an avatar in the game, for instance, the fact that an avatar is frozen in the same activity for a long time (total inactivity).

Social awareness deals with the feeling that is given by another person in a dialogue context. Is he or she listening? Is he or she interested in the discussion? What is his or her emotional state? These pieces of information are generally obtained thanks to nonverbal indicators, such as facial expressions, looks, or body behavior in a real classroom. Social awareness is often supported by webcams. No feedback about this is integrated in the graphical user interface of the pedagogical dungeon.

Group structural awareness concerns the organizational consciousness role, responsibility, or status of each user. The task realization process may also be taken into account here. In the pedagogical dungeon, the user who has the teacher’s role is able to create groups and to place students in them. Some indicators are given through the interface about group structural awareness: the notification message of group joining, the specific representation of the teacher (teachers’ avatars are magicians).
Workspace awareness is the most intuitive aspect of awareness. In fact, it represents the group activity consciousness (of each user) relative to the workspace. Major points concern users’ personal identity information, their positions, their activities, and sometimes immediate changes that they are making in the workspace. In the dungeon, a user is represented with an avatar and his or her name; the discussion in the chat window is displayed, with all participants identified by their names; the user position is displayed both in the room and on a spatial representation of the dungeon called the minimap (see Figure 4); the color of each room on the minimap reminds the user of his or her own progression (succeeded, failed, or not visited); the behavior of the crystal in the main game window also provides this kind of awareness; and help calls as well as answer propositions for an activity validation are instantly displayed with a notification window on the teacher’s screen.

As shown in the literature, most of the work on awareness focuses on workspace awareness; in particular, Greenberg et al. (1996), who defined the concept, mainly worked on solutions in this domain. For the moment, it is also the case in our software: We dealt mainly with workspace awareness through our graphic user interface, as explained later.

However, we can provide teachers with improved awareness, because computers provide the possibility of obtaining much more interesting information: the visualization of traces left by users during their activities within the system. We have already proposed some trace visualizations as a solution to specific observation problems in Carron, Marty, Heraud, and France (2006); France, Heraud, Marty, and Carron (2005, 2006); Heraud, France, and Mille (2004); and Marty et al. (2007). The purpose here is to be more general and offer observation features on any pedagogical tool. The following paragraph explains how this problem is addressed in our approach.

An observation application programming interface (API)

We have chosen an approach that is as generic as possible and thus possibly independent from our application. The idea is to equip any application (here, the whole of the pedagogical dungeon) with a tracing possibility. This implies the definition of an API of required basic observations. For instance, in the dungeon, actions such as entering a room or correctly answering a quiz may be traced and thus collected by specific elementary probes.

Basically, in our context, we defined 17 elementary probes that may be flagged at any moment by any client of our application (Table 2, Figure 5). As seen in Table 2, each probe contains some parameters and has a particular aim in order to complete a specific category of awareness that is not often addressed.

These elementary probes raise two problems:

**Quantity**: There is a large amount of (sometimes useless) information coming from all the clients, and some selection or filtering methods are thus needed. As a matter of fact, in such an environment, many users play simultaneously. All these indicators create a great number of signals from which we have to select the interesting ones. The parameters of a probe may help this configuration to describe the indicator. For
<table>
<thead>
<tr>
<th>Probe Name</th>
<th>Parameters</th>
<th>Awareness Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>WorkshopArriving</td>
<td>&lt;UserName&gt;, &lt;WorkshopName&gt;</td>
<td>Group structural awareness</td>
</tr>
<tr>
<td>WorkshopLeaving</td>
<td>&lt;UserName&gt;, &lt;WorkshopName&gt;</td>
<td>Group structural awareness</td>
</tr>
<tr>
<td>WorkshopAnswering</td>
<td>&lt;UserName&gt;, &lt;WorkshopName&gt;</td>
<td>Group structural awareness</td>
</tr>
<tr>
<td>WorkshopAnsweringWithContent</td>
<td>&lt;UserName&gt;, &lt;WorkshopName&gt;, &lt;AnswerContent&gt;</td>
<td>Social awareness</td>
</tr>
<tr>
<td>WorkshopTeacherValidating</td>
<td>&lt;UserName&gt;, &lt;WorkshopName&gt;, &lt;TeacherName&gt;</td>
<td>Group structural awareness</td>
</tr>
<tr>
<td>WorkshopTeacherValidatingWithContent</td>
<td>&lt;UserName&gt;, &lt;WorkshopName&gt;, &lt;TeacherName&gt;, &lt;Comment&gt;</td>
<td>Social awareness</td>
</tr>
<tr>
<td>WorkshopCorrectlyAnswering</td>
<td>&lt;UserName&gt;, &lt;WorkshopName&gt;, &lt;Boolean&gt;</td>
<td>Group structural awareness</td>
</tr>
<tr>
<td>StudentConnecting</td>
<td>&lt;UserName&gt;</td>
<td>Informal awareness</td>
</tr>
<tr>
<td>HelpConsulting</td>
<td>&lt;UserName&gt;, &lt;HelpName&gt;</td>
<td>Group structural awareness</td>
</tr>
<tr>
<td>Chatting</td>
<td>&lt;UserName&gt;, &lt;ChannelName&gt;</td>
<td>Group structural awareness</td>
</tr>
<tr>
<td>ChatContentListening</td>
<td>&lt;UserName&gt;, &lt;ChannelName&gt;, &lt;SentMessage&gt;</td>
<td>Social awareness</td>
</tr>
<tr>
<td>GroupCreating</td>
<td>&lt;GroupName&gt;, &lt;GroupType&gt;, &lt;UserName1&gt;, &lt;UserName2&gt;, etc.</td>
<td>Group structural awareness</td>
</tr>
<tr>
<td>GroupSplitting</td>
<td>&lt;GroupName&gt;, &lt;GroupSplitter&gt;, &lt;UserName1&gt;, &lt;UserName2&gt;, etc.</td>
<td>Group structural awareness</td>
</tr>
<tr>
<td>StudentDeconnecting</td>
<td>&lt;UserName&gt;</td>
<td>Informal awareness</td>
</tr>
<tr>
<td>TeacherConnecting</td>
<td>&lt;TeacherName&gt;</td>
<td>Informal awareness</td>
</tr>
<tr>
<td>TeacherDeconnecting</td>
<td>&lt;TeacherName&gt;</td>
<td>Informal awareness</td>
</tr>
<tr>
<td>TeacherHelpCalling</td>
<td>&lt;UserName&gt;, &lt;TeacherName&gt;</td>
<td>Social awareness</td>
</tr>
</tbody>
</table>
example, the teacher may want to observe a specific activity or a particular student, and the appropriate probe will thus be configured to collect the right information.

**Quality:** These probes can handle information that is too basic and not very easy to understand or not very meaningful. That is why the user or the teacher must provide some specific treatments to transform the information into something understandable.

These statements led us to offer the possibility of developing more complex probes on the basis of the combination or transformation of elementary ones.

**Complex probes: Personalization of the expected perception**

For a teacher, expectations concerning perception through the system are somewhat difficult to express. The level of what needs to be perceived may vary, as is also the case in traditional teaching: A teacher may want to observe basic facts (e.g., who starts a new activity) or more abstract facts (e.g., who regularly cooperates before answering a quiz properly). The API presented provides users with elementary probes. They are thus useful for observing basic facts. However, they may not be helpful enough when the level of abstraction needed is higher. For instance, being aware only of a student consulting a help file can be not very meaningful. However, if the same observation occurs just after the student has given a wrong answer and then followed that with success in the same activity, the teacher may be reassured as to the usefulness of the help file related to the activity. The combination of these three indicators (simple probes) allows the creation of a complex probe and thus the provision of a higher level explanation about the ongoing activity.

**FIGURE 5:** Probe visualization (colored text mode)
As stated previously, 17 basic indicators (simple probes) have been extracted from the pedagogical dungeon (see Table 2), and three operators have been proposed (AND, OR, and THEN) to combine one probe with another. All these indicators may be combined with one another to define new complex probes. Figure 6 shows an example of such a definition. The new probe is available in the educational platform, with the same properties as the basic probes. In particular, the new ones can be reused to create a more complex probe. Thanks to this mechanism, the set of probes naturally increases with the help of the users, guided by the needs of observation in the platform.

The possibility of having basic or complex probes in the platform helps solve the problem of a lack of feedback for the teacher. The definition of the API allows one to define properly what is observable on the platform, whereas the possibility of defining complex probes enables a personalization of the expected teachers’ perception.

Observation architecture

Technically, we propose a multiagent architecture to achieve such an observation task. In fact, this is justified by the fact that we need to collect relevant information from each student’s workstation and to forward it to the teacher’s station. Furthermore, the system offers the possibility of reacting by adapting the session (e.g., sending messages to particular students, modifying some collect parameters, providing help files, even dynamically adding a part of the dungeon to provide new activities).
This decentralized system, exhibiting a strong need for collaboration and interaction, led us to base our observation architecture on a multiagent system, as described by Carron et al. (2006). In this approach, specific software agents are distributed on every machine: a collecting agent on students’ workstations (“student agents”) or servers (“server agents”) and a structuring and visualization agent on...
the teacher’s station (“teacher agent”). The whole system (pedagogical dungeon and multiagent observing system) is described in Figure 7. The teacher agent enables one to configure the collect agent of any student agent by specifying the traces of relevant activities that have to be sent to the student. All these traces must of course be interpreted and presented to the teacher (as seen previously in Figure 5).

As a matter of fact, when the system is installed, the observation may be finely tuned. Adapted complex probes may be defined by combining different simple probes through an interface (as seen in Figure 6). Each probe may be generic or filtered on one or several specific parameters (see Figure 8). For example, it is possible to be notified only when a student gives a wrong answer to Exercise X of Activity Y after having consulted Help File F.

All pieces of information related to the system may also be observed: the topology of the system, the probes deployed on each agent, or the status of the different agents (see Figure 9).

We have described the correspondence between a learning session and the objects of the dungeon and the way of setting appropriate indicators in the system. The next section is related to a more descriptive view of our system: the creation of the dungeon (associated with a given learning session) by the teacher and the enactment of a pedagogical activity in this game based environment.

Description of the pedagogical dungeon

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 Learning Design (Koper, Oliver, & Anderson, 2003) or more flexible languages such as LDL (Ferraris, Martel, & Vignollet, in press). If the teacher wants to construct a pedagogical
session in the dungeon, he or she interacts directly with a session builder. This tool allows the four creation steps of a scenario: the definition of activities, the description of the available resources for each activity, the definition of the validation procedure for each activity, and the definition of the constraints on the activities (organizational and logical temporal links). According to these constraints, the map of a dungeon will be automatically generated and saved.

**Definition of a new activity.** The user starts with a pedagogical session containing an initial activity called “prologue.” The students start their quest in the room associated with this activity. Usually, there is only one document available in this room, explaining the purpose of the quest and the main rules of the game.

The teacher defines his or her session by entering as many activities as he or she wants. Activities may be added in a session. The teacher then needs to define the information contained in the four tabs corresponding to the scenario creation steps (see Figure 10).

**Providing general information.** The teacher first supplies the name of the activity. Of course, this is useful to link this activity with the others, but also for the system to provide adequate feedback on a specific activity, as seen above. The teacher also reports here whether the activity is collaborative or not.

**Selecting the resources.** The teacher chooses the different resources that will be available for a specific activity. This could be either local files (present on the teacher’s computer) or links to online material. These files usually explain the topic of the activity. The teacher chooses the most appropriate form for these resources: simple texts, videos, or even simulation applications, as in Michelet, Adam, and Luengo (2007).

**Setting up links between activities.** Ordering activities can be a difficult task. Let us consider independently the two kinds of ordering mentioned above. A teacher can construct his or her session as a sequence of activities: “The students will start with
Exercise 1; they will continue with Exercise 2 and finish with Exercise 3.” In that case, the teacher only wants to declare an organizational order among the three activities of the session (see Figure 11).

Another way of defining links between activities is to consider the prerequisites for the different activities:

The students will start with Activity 1 (presenting and testing Concept C1); if they succeed, they can begin Activity 2 (which introduces Concept C2), but if they fail, they need to do Activity 3, which provides additional and more detailed information about Concept C1, because this item has not been acquired.

In that case, we can consider that the teacher is defining the order by considering prerequisites, that is, he or she uses a logical order. In Figure 12, a prerequisite for Activity 3 is a failure in Activity 1. There is also the possibility of stating parallel activities.

The teacher can thus easily use both of these ordering possibilities to describe the most suitable succession of activities for the session. Although the interface is quite simple, the tool allows expression of rather complex situations. For instance, we can say that Activity 7 must occur after Activity 1, that the prerequisites for it are success in Activity 3 and success for all the group in Activity 4, and that Activity 7 can occur at the same time as Activity 6. From a collaborative point of view, we can see that there is an easy way to synchronize a group: All the keys of the group should be needed. In that case, the student must wait for all the members of his or her team. In fact, most of the time, the student goes back into the previous rooms to help them.
Evaluating the 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. However, 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 can be collaborative, in which case the answers are given by the whole team. We have developed here a simple way to obtain the keys.

Once the teacher has defined all the activities of the session, the dungeon can be generated. Figure 13 is an example of the result of such a generation.

Graphical awareness of the activity status. A similar overall view from above (minimap) is always supplied for the teachers during the game. This view is dynamic, because one can see all the users involved in this pedagogical session moving through the different rooms. This input provides the teachers with some awareness about the ongoing activity. The student’s view is restricted: He or she can see only the rooms that are accessible to him or her (i.e., the rooms whose keys the student possesses).

Enactment of the pedagogical activity: Exploring the dungeon

Two roles may be played in the dungeon: teacher or student. Both sides are described in this section.

Each user is identified with a login. The progression of the student is saved when he or she quits the application and is restored when the student reconnects to
continue the learning session or the exploration of the dungeon. The system may also be accessible from home, but in this case, other students or the teacher may not be present in the dungeon at the same time.

**Student.** In our virtual environment, a student is represented by an avatar (see Figure 14), whose characteristics evolve dynamically over time, according to the activities completed during the session (keys acquired). For instance, the avatar can wear different clothes that the student wins during a session.

Most of the time, a student is present in a virtual room representing an activity. He or she can access several resources related to the activity.

In Figure 2, two persons are present in a room; the avatar on the bottom (with the helmet) is the teacher. The other one (a student) has his nickname written above his avatar (Antony), and the name of the activity (prologue) is written on the floor of the room. Touching a sphere/globe item (a resource) opens a text window with explanations or provides a Web link, a file, and so on. 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 him or her continue the quest (see Figure 15).

The translucent white area is a chat window for collaborative features. Each person present in this room can see what is said (see Figure 15). Clicking on a specific avatar may open some private chat windows.
FIGURE 14: Avatar choice window for the student

FIGURE 15: Answering a crystal question
The visualization 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, some 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 16). However, very often, the involvement of the teacher is necessary. The teacher corrects the exercise dynamically, can add remarks, and validates the answer or not. Whatever the case, the student is notified via a window (see the right side of Figure 16) containing all the stated information.

**Teacher.** The teacher view is specific, and he or she may have an overall view of the dungeon (see Figure 13). One of the teacher’s main aims is to follow the progression of the students. The overall map view shows the position of each student and provides the possibility of teleporting himself or herself instantly to a specific room to see, for instance, the teamwork in a collaborative activity. Nevertheless, a minimap is always available in the game view (see Figure 17).

Another part of the role of the teacher is to answer the questions or to validate the answers proposed by the student. When a question is awaiting a teacher validation, a specific window appears in the teacher interface (see Figure 17).

As we will see in the next paragraph, collaborative activities are also possible. The teacher may thus create groups and dispatch students into them, as shown in Figure 18. Thanks to these notification windows or even visualization windows, the teacher is aware of the current activity of each student. The teacher may choose to help slow students or to propose other activities or resources to particularly fast students. Some rooms also enable collaborative work and thus the synchronization of students with each other, as presented in the next section.
Collaboration in the dungeon. In our system, some activities must be realized collaboratively. The associated rooms require the students to answer in groups, as indicated on the access door to the room. The crystal hiding a group activity has a specific color and notifies the first student arriving (see Figure 4). The advantage is to make a rapid student help another one before continuing the progression. The groups are thus homogeneous. Collaborative keys are given to all the avatars that constitute the group.

This serious game visualization is dynamic, because all the elements are updated in real time and because the teacher can interact with this view. Nevertheless, we will see in the next section concerning real experiments that interactions with students may be very time-consuming.

Experiment

The pedagogical dungeon has been tested during several practical works with a real classroom at the University of Savoie.
Description of the learning activity and of the system

*The pedagogical objective.* The pedagogical session modeled concerned a lesson about operating systems. Several independent concepts were exhibited 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 were acquired. The students were 18 years old and familiar with computer use. All the types of artifacts or exercises described above were used in this experiment. The students were working on Linux and had to find or test some text commands in the console.

*The methodology of the experiment.* During the experiments, about 15 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 links found during the exploration of the dungeon). The students were warned that all their actions would be observed through trails. 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.
All the observation probes were configured at the beginning of the session and were active during the whole session. Moreover, forms filled in by the students just after the experimentation gave us interesting feedback about the software itself. Some necessary improvements have already been highlighted and resolved; for example, many students tried every possibility to validate a multiple-choice questionnaire.

**Conditions and objectives of the experiment**

The very first experiment aimed to validate the overall approach and to test the system technically: As expected, students used the proposed environment without any explanation. 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: Similar “classical” (without the support of such an application) practical work 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 developed between users exploring the dungeon. The chat tool was intensively used to communicate about the exercises, boosting the competition and reinforcing the immersion feeling.

Subsequent experiments focused on improvements realized on the system. Three evolutions have already been implemented and validated. We can notice that “cosmetic improvements” have already been realized on the version presented here because it seemed that the first world proposed was not very attractive to girls. Special adaptation must be envisaged to adapt to both male and female students.

**Pedagogical results**

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 the teacher may sometimes be overloaded by the number of questions from students waiting.

From another point of view, some advantages were exhibited about the level of understanding of the lesson. Concerning the activity regulation, direct explanations were given to students proposing wrong answers, and additional files were included during the game. Collaborative activities have not been evaluated sufficiently, since they were located at the end of the scenario, and some students did not reach that point.

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 possible to reduce the volume of the traces displayed, this requires some additional time from the teacher, who was already overloaded. This remark raises the question of an adaptable control board containing suitable indicators, enabled or not according to the context.

The teacher thus prefers to use the system with multiple-choice questions and with a minimal awareness tool. The teacher prefers to enrich the probes as needed during the activity’s progression. As a matter of fact, other experiments have been
set up concerning a lesson about object-oriented design, but we failed to set up an experiment in which students had to write pieces of software (too difficult to evaluate quickly enough).

Some students pointed out that the style used in the exercises did not match the style of the game. This means that teachers should think about the form of their pedagogical activities to make them fit with the dungeon’s look and feel (a medieval style). A good immersion of the session subject in the dungeon ensures a better motivation on the part of the students, as shown in Rosenbloom (2004) or Cheng and Cairns (2005).

Thus, the most interesting point is that the pedagogical session has to be well prepared and specifically adapted to such environments. As stated previously, for the teacher, the experiments revealed a great difficulty to read and validate at the same time: A lot of student propositions are generated from open questions. The whole session must thus be well designed to protect the teacher from such an overload: He or she must also allow time to observe the pedagogical activities and to react dynamically. Moreover, in the first experiment, the teacher was disappointed because he could not change the pedagogical scenario dynamically: Because of misunderstanding of a concept, most of the students had difficulties in a particular room. It would have been useful to add a special room for additional explanations.

Improvements on the basis of this last remark were therefore implemented between the first and second experiments: A new part of the dungeon may thus be created and added to an existing one. The latter is linked with the extension via a teleportation portal (see spiral item in Figure 2).

Discussion and perspectives

The prospects at the end of this experiment are numerous.

Several instances of the dungeon have currently been tested to date, but an interesting perspective is to propose the various scenarios to the students. Many teachers from other domains (electronics, mathematics) are very interested in testing it in their lessons. From our point of view, this perspective will show that the tool is not restricted to a specific domain and could be integrated as a tool into the “old” LMS we developed in Martel et al. (2004), which is currently used by 25,000 persons in our university.

Today, there are too few results to evaluate the impact of such a system on understanding, but the students have asked whether other lessons could be proposed in this way and are motivated by the possibility of continuing at home. To them, the idea of finishing the exploration or of replaying a dungeon game seems to be a natural manner in which to continue and complete the learning process at home.

Many environments, such as that of Mazza and Dimitrova (2005), propose statistical tools and sometimes also specific functionalities for tracking students (e.g., Moodle, a popular course management system proposing tracking; see http://moodle.org) or for tutoring student groups tutoring, but in most LMSs, this analysis can only be realized a posteriori and is static. The different ideas proposed in this article give
good reason for being particularly interested in real-time visualization: The knowledge of the current user activities is considered more helpful than only past activities. We need it to be as aware and as skilled as possible to take the best decision and to regulate the collaborative activities.

Another possible use of the dungeon deals with the “a posteriori” analysis of a session. Saving, storing, or retrieving past activities may give sense to a learning session or a specific serious game. Each signal or trace may be saved in a database to be consulted again at a later date.

Several advantages of this a posteriori analysis may be shown. For example, a teacher does not always have the time during the session to analyze what is going on, and thus, it could be interesting to replay some parts or even the complete session. In the educational domain, it is well known that the reflexive aspects of the trace may be interesting: A student may want to consult his or her own trace for metacognitive purposes or even to compare it with another student’s.

The teacher needs to be aware of the ongoing activity, and he or she needs to follow the progression of all the students. Indeed, the overall view of the dungeon presented earlier is a good tool to perceive the ongoing activity, and an a posteriori analysis of a session gives information about the progression of all the students. For instance, the traces can be used to draw the path of each student in the dungeon, or the room can be highlighted with colors to indicate the state of a student with respect to an activity. It may be noted that this last type of information, regarding what has happened in the previous few minutes, can be revealed to the teacher in real time.

The results and students’ returns show that it is not only or just a transformation of a pedagogical scenario into a dungeon structure. Thanks to these digital environments, we are able to observe a learning session in a much more accurate manner. Therefore, we are able to adapt or regulate the pedagogical scenario in real time. This new dimension offers the possibility of providing additional help or exercises, or various paths to acquire knowledge, adapted to each sort of student. Today, it is difficult to give different exercises directly, according to the skill of each student, or to just present an alternative between several exercises.

**Conclusion**

In this article, we have demonstrated through examples how educational games are relevant to providing students with a dynamic and pleasant learning platform. Although the user interface was quite important for usability purposes, we wanted to present a new way of designing learning activities. One of our next goals is to check the links between the creation of a pedagogical scenario with standard languages, such as IMS Learning Design, and the generation of the dungeon.

Collaboration between different roles (within a group of students or between students and teachers) has been considered. We think that this aspect can be improved: In this version of the game, only two roles are implemented, and they are static. This notion of role is interesting, and we would like to extend it to many dynamic roles (e.g., a student can become a leader for a strictly defined period of
time in a collaborative task). We can also improve the collaborative space itself, by offering more than a chat in a room. Asynchronous tools must also be considered when we want to tackle distance learning (students are not always present at the same time in the dungeon).

Another part of this article deals with the awareness indicators providing feedback to the teacher on the ongoing activity. A multiagent system is proposed to create, configure, and deploy specific probes to observe the activity in the dungeon. We have shown that such an approach allows one to address specific categories of awareness that are generally not tackled.

Finally, visualization of the probes must be improved as well as the operators to elaborate new complex probes. For example, we imagine proposing specific visualization with calculus functions. We have also planned to work on collaborative indicators to suggest adapted actions to enhance the overall group activities. These indicators must be defined through personalized tools for the teacher’s requirements.

References


Thibault Carron is an associate professor of computer science at the University of Savoie, France. He is a member of the Syscom Laboratory. He obtained his PhD in computer science at École Nationale Supérieure des Mines de Saint-Etienne in 2001. His current research interests deal with the study of collaborative activity observation.

Jean-Charles Marty is an associate professor of computer science at the University of Savoie, France. His research interests are mainly in the observation and regulation of collaborative activities. His research results are mainly applied to the educational field and more recently to serious games.

Jean-Mathias Heraud is an associate professor in the Graduate Business School of Chambery, France. He obtained his PhD in computer science at the University of Lyon in 2002. His research interests deal with the study of collaborative activity in information systems.

**ADDRESSES: TC & J-CM:** Syscom, University of Savoie, UFR SFA, Campus Scientifique, 73376 Le Bourget du Lac, France; telephone: +33-4-79-75-88-54, +33-4-79-75-86-16; fax: +33-4-79-75-86-90; e-mail: thibault.carron@univ-savoie.fr, jean-charles.marty@univ-savoie.fr

J-MH: ESC Chambery, Savoie Technolac, 73381 Le Bourget du Lac Cedex, France; telephone: +33-4-79-25-33-48; fax: +33-4-79-25-33-54; e-mail: jm.heraud@esc-chambery.fr.