# Developing a Virtual Environment for Learning Geometry

Venustiano Soancatl, María-Luisa Cruz, Luis Huerta,
Antonio León, Andrea Herrera, Lucina Torres*, William Zurita, and
Inti Reyes

Universidad del Istmo, Departamento de Informática,
Carretera Chihuitan. S/N, 70110. Ciudad Ixtepec, Oaxaca, México
{venus,mlcruz,luisdh2,leonborges,
aismenee,ezurita,inti}@bianni.unistmo.edu.mx
luz83_83@hotmail.com*
http://www.unistmo.edu.mx

**Abstract.** In Mexico and in many other countries, students have many problems in the mathematical learning process, particularly in problem solving. The purpose of this project is to design and develop an educational computer-game system using a virtual environment to teach junior high school students how to solve geometric problems. It is based on two main approaches: game-based learning and question-led learning. The system includes a database that stores geometric problems and their step by step solutions, thereby leading students through the process of problem solving while keeping track of the students' performance. The design of this system permits the separation of the domain area and the game programming.

**Keywords:** virtual environment, game-based learning, question-led learning, geometry, problem solving.

## 1 Introduction

The fast growth in computer processing power has opened the door to new communication and entertainment forms, which includes virtual environments, complex computer games, social networks and immersive environments. Concerning computer games, they have evolved from simple 2D graphic interfaces in the 70's to 3D models in the 90's with virtual environments arriving at the beginning of the XXI century [1]. Now, the challenge is to use these new technologies to improve the quality of life. In education, the use of computer games to strengthen the learning process is becoming more and more popular [3]. This approach is called game-based learning (GBL) which is defined by Conolly and Stansfield as "the use of a computer games-based approach to deliver, support, and enhance teaching, learning, assessment, and evaluation" [4]. Many educational games have been developed that have their own characteristics and cover domains such as adventure, arcade, simulation, strategy, logic, puzzle and immersion [2]. Some research groups and companies are developing different kinds of educational games with great success. The aim of GBL is to use the engaging property


Advances in Computer Science and Electronic Systems
Research in Computing Science 52, 2011, pp. 26-36

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

of the games in the learning process, such as: Virtual U to teach administration students [5], Timez Attack for learning the multiplication tables [6], Flooding Control Trainer *FCT* a game for training US Navy [7], UFractions (Ubiquitous Fractions) an educational mathematical mobile game, developed in SouthAfrica [8], and 80days a cultural game, developed in Europe [9]. These computer games have proven that it is possible to mix fun and knowledge to the benefit of many segments of society. The functionality of the GBL approach has not been proven, however, there are some reports about the impact of games in the learning process. The Teachers Evaluating Educational Multimedia (TEEM) company in the United Kingdom has done some research about the educational use of games [10]. They analyzed about 20 different games including different surveys applied to parents, teachers and students. The results are divided into four key stages and each stage represents a different set of knowledge or skills that the game helps to develop. The results show there are some topics that the parents think are very important for their children to learn. Parents were asked which subject they preferred in student games. Math, spelling and reading were most frequently mentioned. The prioritized skills by the parents were decision making (40%), design (25%), problem solving, strategy and cooperation (20%), and mouse control and taking turns (10%). In these surveys about (30%) of the students played computer games every day while the rest played once, 2-3 times or 4-6 times per week.

In Mexico, math competence is one of the greatest problems that students face, since many students score poorly in accordance with some national and international assessments such as the Assessment for Academic Achievements in School Centers (ENLACE), Educational Quality and Achievement Tests (EXCALE) and the Programme for International Student Assessment. They report that more than 50% of Mexican students are in a low proficiency level in the mathematics scale [11]. We propose using the GBL approach to help mexican students improve their mathematical skills. In this work we present the development of a virtual environment to teach geometry problem solving. Our goal is to join the fun of a videogame and the learning of geometry. Also we focus on the question-led learning approach to guide students through the process of problem solving [17]. To achieve this goal, we analyzed some important features of videogame content creation and in the following sections we explain the content design of our system including the virtual environment.

## 2 Content Creation

In videogames, the term content usually takes two different forms: assets and gameplay [13]. The term assets denotes every object that is presented to the user such as 3D models, objects, characters, animations, music and sound effects. Gameplay defines what the user does, the actions the user can do, and what the game has to do in response [13]. The creation of the assets involve graphic artists using diverse kinds of tools such as Maya, 3D Studio Max, Blender and POV-Ray to create 3D models and textures which also involve script writers and music composers [14]. Behaviors are the dynamic part of the game and define what players can do with the objects in the environment and how they react to the players' actions. Usually, behaviors are programmed by artificial intelligence (AI) programmers. Instructional content creation is similar to gameplay content creation because it defines what the user can do and what the system has to do in response. In educational applications, teaching material is divided into a set of learning objects or instructional units. They are designed to be studied with no interruption. Learning objects contain some kind of text or image with an explanation. Also, there are different form of tests to assess the student.

Educational applications are sometimes called computer assisted instructions (CAI) systems, which require a domain expert cutting the domain knowledge into individual pieces to be presented to the students. A lot of tools can be used for building learning objects such as Power Point, Open Office, Macromedia AuthorWare or web based platforms. The simplicity and generality of these programs allows nearly all kind of domains to be represented. Unfortunately, the knowledge representation in these programs is inadequate and interaction capabilities is also very simple. Students learn reading and thinking instead of putting concepts into practice. This kinds of learning objects are not appropriate for procedural or problem solving skills, since the exponential demand of more content exceeds the capabilities of a single human author [18]. It occurs especially when the system should provide specific explanations for each common error. Some improvements to this model have been proposed to overcome these difficulties, such as "generative CAI". Another proposal is to use AI techniques for representing the instructional content, which is called Intelligent Tutoring Systems (ITS).

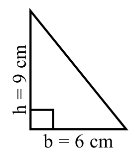
Using more sophisticated content representations allows us to create automatic explanations using natural language techniques, so user interactions can be richer, more open responses can be managed and analyzed, and the system will deduce user misconceptions or skills. More complex domains afford systems that follow the learning-by-doing approach [15], where students spend most of the time solving exercises in an environment with richer interaction. These kinds of systems have to decide the next exercise and infer how the student solves it, giving some feedback to help when mistakes are detected. Exercise creation and correction constitute the domain knowledge, so, systems must know the valid response and the more likely human mistakes and their causes. These systems also need the ability to propose exercises themselves. All the content material is highly dependent on the domain being taught and the representational techniques. This causes the creation of universal tools to be very difficult for ITS [13].

### 2.1 Content Design

In this paper we present a content design for an educational game system for teaching geometry. Problem solving is considered the most important part of math. However, teaching problem solving is not an easy task because there are lots of difficulties that students face when they try to solve problems [17]. In addition, when solving a problem there are a great variety of paths to obtain a correct solution causing a combinatorial explosion impossible to afford. Our system stores math problems in a database, which includes the step by step solution of every problem. Therefore, the system is able to lead the student through the process of problem solving.

Each problem statement in the database consists of text, images or 3D models, a time limit, difficulty level, and tips or recommendations, along with steps that lead the students to the solution of the problem. Each problem is associated with one of the following geometry topics: triangles, quadrilaterals, circles, similarity, the Pythagorean theorem, trigonometry, perimeters, areas and volumes. Each step includes a multiple choice question and a recommendation if required. The possible answers and recommendations can be represented by text, figures, images or 3D models. The problems are designed following the question-led learning approach. Questions in this approach are focused to help students to reinforce their process of thinking, show them how to formulate precise statements and questions, bolster their habit of questioning, improve the quantity and quality of questions asked by them, and teach them different techniques and strategies for problem solving [11]. The next example shows a simple problem that could be stored in the database.

Problem: *Find the area of the following triangle.*




1. The formula you need to find the area is

   (a) $A = \frac{bxh}{2}$ (b) $A = bxh$ (c) $A = bxb$ (d) $A = \frac{bxh}{3}$

   tip: *remember that the triangle area is half of the area of a rectangle*
2. where *b* is

   (a) height (b) width (c) base (d) perimeter

3. and *h* is

   (a) base (b) width (c) height (d) perimeter

4. then compute the area using

   (a) $A = \frac{6x9}{3}\,cm^2$ (b) $A = \frac{6x9}{2}\,cm^2$ (c) $A = \frac{6x8}{2}\,cm^2$ (d) $A = \frac{9x6}{1}\,cm^2$

Each problem is linked to a specific object in a virtual environment and the object can be an image, a painting or a 3D model. An object can be associated to zero or more problems. A filename, description, location, scale and rotation angles are stored for each object. This information allows the system to display the objects in the virtual environment.

The database stores information about users, such as name, nickname, password, birthday and gender. Also, the number of solved problems, number of mistakes, scores, and number of sessions in the system is recorded. Figure (1) shows a diagram which includes three main entities: PROBLEMS, STEP and MUSEUMOBJECT, and their attributes.

Some advantages of this design are that the system will be able to update problems without modifying any programming code, store statistics about students performance, minimize dependencies between the domain expert and game programmer, use the same

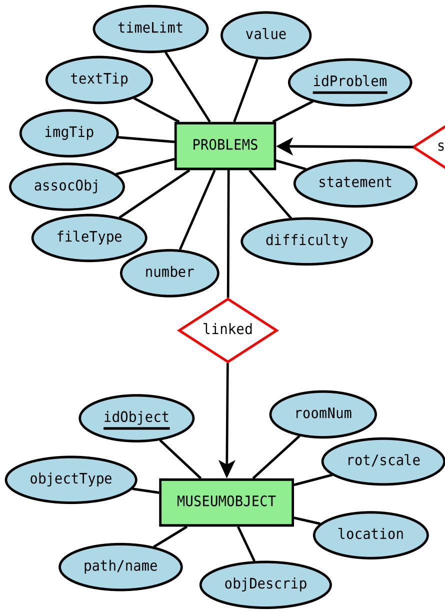



**Fig. 1.** ER Diagram of the database.

map to execute different exercises, use the same presentation to teach different contents, and to reduce the development cost by presenting different exercises on the same scenario.

The main strength of this approach is that domain experts can design problems without any programming skill. Also problems can have a high level of complexity and include as many steps as the domain expert wants. Another advantage to this approach is that almost any kind of domain can be represented.

## 3 Development of the Virtual Environment

The software is composed by two main modules, the graphic environment and the database control. Both modules are in constant communication while the system is running. Figure (2) shows the relationship between these two modules. The DB control module performs reading and writing transactions on the database. The graphic environment module creates the virtual world and displays the data for the geometric problems taken from the database. The graphic environment also reads and writes user information from and to the database by mean of the DB control.

### 3.1 Development of the Graphic Environment

The programming language we are using to develop the system is C++, since it is one of the most common programming languages to develop videogames. The 3D scenery

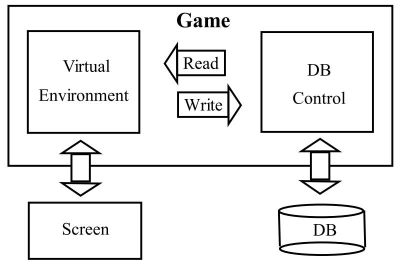



**Fig. 2.** System components and their interaction.

and graphics are displayed using the open source Irrlicht 3D Engine, which is written in C++ and compatible with NET languages. The irrKlang sound library is used to play sounds and music. Fonts and text are displayed using the FreeType library, which is a free, high-quality, and portable font engine. The database has been embedded using SQL Server Compact 3.5, a free easy-to-use database engine.

### 3.2 The Gameplay

Currently the museum has five functional rooms where the user can navigate and explore, with each room dedicated to a specific geometry domain. The domains of each room are the following: triangles, quadrilaterals, circles, areas, and solids. Each room has paintings on the walls of famous mathematicians and places. Each painting is associated with one or more problems. While the user navigates in the museum, a red light pointer is used to guide the direction of the movements and when the user points to a painting, a brief description is displayed at the top of the screen.

When users try the system for the first time, they must create an account which includes a nickname and password. All the user's statistical data are initially set to zero which includes the points, number of solved problems, and number of mistakes. These statistical data allows the system to keep track of student activities such as problems solved, mistakes committed and how long the system is used.

The aim of the game is to solve all the problems in the museum, which are designed using the question led learning approach [15]. When the user selects a painting by pointing and clicking on it, a problem is displayed on the screen and a chronometer starts running backward showing the time remaining to solve the problem. After several seconds, the system starts displaying the steps (multiple choice questions) one by one on the screen. When the user selects the correct answer the next step is shown, until the solution is found or the time expires. Figure (5) shows a screenshot of the system displaying a problem, question and four possible answers. In this version these answers are shown inside 3D objects which are floating like moving balloons. However, if the user makes a mistake, the system shows educational feedback and sends the user to answer several questions back, this is done as a kind of punishment. This make the user think twice before selecting a choice by chance. This process is shown in Figure (3) as a time dependent flow chart. When users solve a problem, the system gives them a certain number of points and additionally, depending on how fast the students solve the

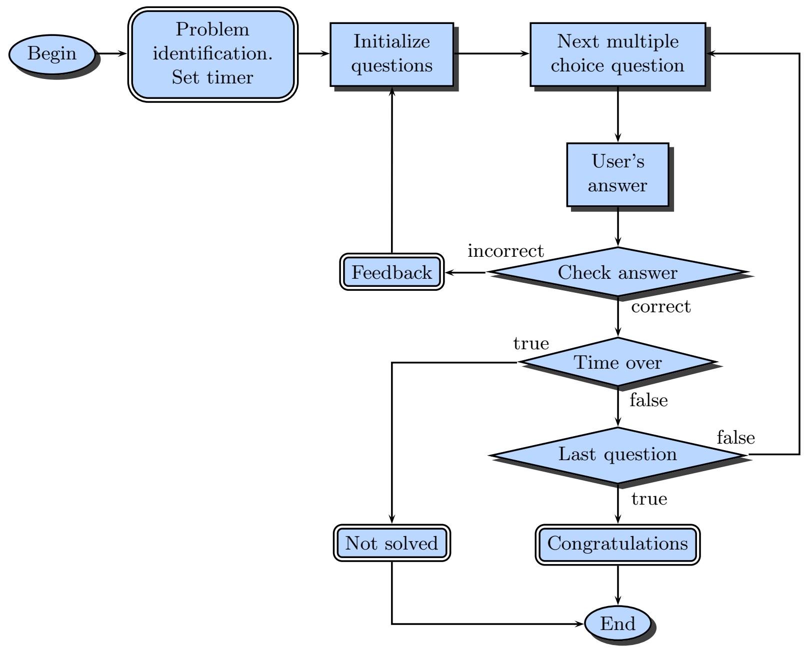



**Fig. 3.** Problem solving flowchart.

problem, they can earn extra points. The system stores the time required to solve the problems, number of mistakes and add points to their record. There is only one way for the students to fail when they are playing the game and it occurs when the time expires. In this case, the system stores the number of incorrect choices that the students made and their points do not decrease. This gives us information about of how the students tried to solve the problem. Additionally, we can infer if the students responded by guessing or if the time period was insufficient.

Once a problem is solved, it is deactivated and is not shown to the user anymore. That is why there must be many similar problems to assure the user understands the topic.

The users usually get feedback from the system, whether the user selects an object, responds a multiple choice question correctly, makes a mistake or solves a complete problem. The feedback could be recommendations, congratulations, sounds and images. When the user makes a mistake, the message ”Think about it and try again” appears. We propose to send only positive messages to the user.

### 3.3 Virtual Environment Components

We define the three principal components of the virtual environment as Scenery, Problem Manager, and Player. The Scenery component displays all the elements in the virtual environment, such as images, 3D models, textures, text and lights. The Problem Manager component reads the problems from the database, presents them to the user,

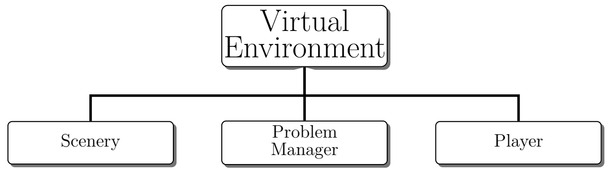



**Fig. 4.** Components of the virtual environment.

display the steps associated with the problems, evaluates user responses, and manages a chronometer to keep track the time limit for the problems. The Player component manages all the information about the user, such as scores and points, game sessions and solved problems. Figure (4) shows the main components of the virtual environment.

The screen is divided into four horizontal sections: user data, problem statement, virtual world and step statement. Figure (5) shows a screenshot of these sections. At the top of the screen, the user data is displayed in red, to the left is located the nickname (1), in the middle of the section users can see their score, and to the right, the number of solved problems in the current session. This information helps users track their progress in the game. The problem statement section is the area reserved to display the text statements of the problems in yellow and is located right under the user data section. The third section is the biggest area in the screen and is designated to display the virtual world where the user can navigate. This virtual world contains the interactive objects that display the problems, such as paintings, 3D models and possible answers to the multiple choice questions. The fourth section, at the bottom of the screen shows the questions or statements of problem steps in yellow. Figure (5) shows a view of the virtual world where the user's nickname is "Juanito", his score is 5 points and he has already solved two problems in the current session. Juanito is navigating in the triangle room and he has just started to solve a triangle geometric problem. The problem is displayed at the top of the screen and he is in the second step (question) of fourteen (7). His choices are in 3D transparent balloons (8,9,10), and at the top right corner is the chronometer showing the time remaining (6). In the scenario, there are some paintings on the walls, such as a famous mathematician (2) and a giant Olmec Colossal Head (3).

The system is in the initial phase to analyze its effectiveness as a learning tool, but teacher and student responses has been very positive. Teachers find the game useful for learning because it is based on videogames which have become part of the lives of this age group of students. Additionally, teachers like the fact that they can design their own material, set the difficulty level, and get statistical information about student performance. Students like the aspect of learning by combining 2D and 3D objects in a virtual environment. Students also asked if the level of difficulty could be increased by making the answers move faster.

### 3.4 Main Pedagogical Aspects

The main reasons we believe the system would be effective are the exciting and entertaining environment which keeps student attention on the screen. Curiosity stimulates


**Fig. 5.** Screenshot showing elements of the system.

learning and motivates a student, it can drive to seek out new and interesting information. The system can evoke a learner's curiosity by providing the environments that have an optimal level of informational complexity [12], since it can be designed by teachers and domain experts in a way that is neither too easy nor too complicated. Problems can gradually increase in difficulty and can be adapted or personalized by teachers. Problems can be designed using a pedagogical method such as Polya's four phased method [16] and include steps for critical reflexion. The problems in the system can be seen by the students as challenges, which can be overcame with feedback designed by teachers. This system also is focused on the learning by example approach since the system leads students step by step thorough the process of problem solving. An important advantage of the system is that teachers can easily identify statistical data to find students that need extra help to improve their scores.

## 4 Conclusions and Future Work

In this paper we have shown the development of a computer game system to teach mathematical problem-solving to students. First, we presented the design of the content of the system. The proposed system includes a virtual environment and a database. The graphics interface component controls the gameplay. The database stores information about the domain expert and users. This design permits updating the data domain expert without modifying the programming code. The Game-Based Learning approach was used to design the gameplay of the system that includes images, sounds and 3D models. For the domain expert we focused on the Question-led Learning approach to design the problems stored in the database. The problems are presented in a sequence of steps that lead the students to find their solution. Each step includes a multiple choice question which is displayed using the virtual environment component. The clear separation between exercises and the virtual environment significantly reduces the cost of the educational game, since the virtual environment can be reused. The system is in the last development stage. The next stage is to apply usability tests and to distribute the system to schools and libraries.

## Acknowledgments

We want to give many thanks to FOMIX-VERACRUZ and UNISTMO for supporting this work which is part of the project 95656. We also thank Kevin Mitchell for his contribution in the English translation.

## References

1. Entertaiment Software Asociation, Game player Data, `http://www.theesa.com/facts/pdfs/ESA_Essential_Facts_2010.PDF`
2. Pérez Marqués Graells, Tecnología educativa, `http://peremarques.pangea.org/videojue.htm`
3. Paul G.: What Video Games Have to Teach Us About Leraning and Literacy, Palgrave Macmillan, First Edition (2003)
4. Thomas, C., Mark, S.: 'Using Game-Based eLearning Technologies in Overcoming Difficulties in Teaching Information Systems, Journal of Information Technology Education, Vol 5, pp. 459–476 (2006)
5. Eriic, A., C.: Software Review Virtual U: A Simulation of University System Management, Information Technology,Learning,and Performance Journal,Vol. 19, no. 1, Spring (2001)
6. BigBrainz : The ultimate multiplication tables tool, `http://www.bigbrainz.com/`
7. Hussain, T.S., Roberts, B., Bowers, C. Cannon-Bowers,J.A., Menaker, E.S., Coleman,S.L., Murphy,C., Pounds,K., Koenig,A., Wainess,R.,Lee,J.: Designing and Developing Effective Training Games for the US Navy, Interservice, Industry Training, Simulation, and Education Conference *(IITSEC)*,Paper No. 9477,(2009)
8. Turtiainen, E., Blignaut, S., Els, C., Laine, T. and Sutinen, E.: Story-based UFractions Mobile Game in South Africa: Contextualization Process and Multidimensional Playing Experiences. Proceedings of the Second Workshop of Story Telling and Educational Games, (STEG 2009)
9. Kickmeier-Rust, M. D., Albert, D., and Mattheiss, E.:An educational guide to planet earth: Adaptation and personalization in immersive educational games. In Y. Cao, A. Hannemann and B. F. Manjn (Eds.), proceedings of the 2nd International Workshop on Story-Telling and Educational Games, (STEG 2009)
10. Mc Farlane, A., Sparrowhawk A., Heald, Y. :Report on the educational use of games. In :Teem publications, `www.teem.org.uk/publications/`
11. Venustiano, S., Lucina A., María-Luisa C., Antonio L., Carmen M.: Un enfoque para la Enseñanza-Aprendizaje de solución de problemas de Matemáticas basado en preguntas, 11th Congreso Internacional y 14th Nacional de Material Didáctico Innovador, UAM, México, pp. 97–103,(2010)

12. Thomas, W., M.: What Makes Things Fun to Learn? A study of Intrinsically Motivating Computer Games. Cognitive and Instructional Sciences Series CIS-7 (SSL-80-11), (1980)
13. Marco, A. G., Pedro, P.G., Pedro A. G.: Games-Based Learning Advancements for Multi-Sensory Human Computer Interfaces. Information Science Reference (an imprint of IGI Global), Hershey New York (2009)
14. John, P. F., Omar, S.: Software Engineering for Game Developers. Thomson Course Technology PTR., Boston, MA (2005)
15. Marc, P.: Digital Game Based Learning. MacGraw Hill, USA (2004)
16. Polya, G.: How to Solve It: A New Aspect of Mathematical Method, Princeton University Press, (1989)
17. Venustiano, S., Antonio, L., Carmen, M., Lucina, T.: Leading Students to Solve Math Problems Using Question-led Learning. In: 4th European Conference on Games Based Learning, pp 368–376. Copenhagen, Denmark (2010)
18. Joshua, T., Karen, T., Magy-Seif E., Marek, H.: Authoring Tangible Interactive Narratives Using Cognitive Hyperlinks. In: INT3 '10 Proceedings of the Intelligent Narrative Technologies III Workshop ACM, New York, NY, USA (2010)