IT for Innovative Educational Environments:
Exploring, Authoring and Programming

P. Boytchev[1], S. Kamenova[2], E. Sendova[3], E. Stefanova[1], E. Kovatcheva[1], N. Nikolova[1]

[1]Faculty of mathematics and Informatics, University of Sofia
[2]High School of Math and Science "Akad. I. Tsenov";
[3]Institute of Mathematics and Informatics, Bulgarian Academy of Science

Key words: *educational environments, exploratory and interactive learning, authoring tools*

Abstract:

The paper presents a novel approach based on using a single educational environment to achieve various methods for supporting educational activities – exploring prebuilt learning modules, using authoring tools to create educational modules and programming activities. This approach builds a solid foundation for subject-neutral and multidisciplinary applications and utilized modern IT techniques like virtual reality, interactivity and explorativity.

1   Introduction

Being a relatively new educational option Information Technologies (IT) could play a significant role in the learning process if they are applied in a fostering and constructive way. The computer is a landmark of IT and it should be used in the classroom in a way that increases students' motivation and interest, hence amplifying their anxiety to learn and explore. This paper describes a novel approach of using micro-applications in a multidisciplinary environment which engage and inspire both students and their teachers.

There are many reasons for having IT applied in a broad range of secondary school subjects: the possibility of effective activities of students and teachers, increased volume of information they must deal with, improvement of the quantity and quality of thinking and gaining skills for utilization of information sources from inside and outside the school [1].

A proper use of computers in education might lead to an increase of students' motivation for learning and experimenting. Unfortunately, there are cases when computers in education just provide an additional layer isolating students from subjects. When applying IT to other subjects the main goal is to provide new possibilities of activities which have not been available earlier. Educators are still looking for the best approach, although it might happen that the best one is just a collection of several approaches.

The use of computers in the classroom is dominated by the software which is used. Thus the focus of having IT embedded in non-IT classes is on the software applications which are available for the students and their teachers; as well as on the possible educational approaches available for these applications.

This paper presents an approach based on a single educational environment. This environment is called Elica (Educational Logo Interface for Creative Activities [2]) and can support three methods discussed in the paper: (a) exploring prebuilt learning modules; (b) using authoring tools to create educational modules and (c) programming activities. Elica and Elica-based software build a solid foundation for subject-neutral and multidisciplinary applications and utilized modern IT techniques like 3D scenes and virtual reality, interactivity and explorativity.

2   Interactive Exploring

Traditionally educational software is disseminated in the form is a monolith software application. The main advantage of this approach is that both developers and users can focus on just a single application developed for a specific educational topic. This may lead to inconvenience when a teacher needs to broaden the usage of the software: either she should start to use additional (possibly incompatible) applications, or the application should incorporate more features and become heavier.

Our approach is to deliver a set of interactive micro-applications which support growing functionality within a consistent user interface. To evaluate the benefits our team took part in the DALEST project [3]. DALEST stands for Developing Active Learning Environment for Stereometry and was co-funded by the European Union under the Socrates program, Minerva 2005 selection. Partners in this project were educational and software development institutions across Europe, namely University of Cyprus, University of Southampton, University of Lisbon, University of Sofia, University of Athens; Cyprus Mathematics Teachers Association and N.K.M Netmasters.

The same approach is currently used in the on-going European project InnoMathEd [4] – Innovations in Mathematics Education on European Level, which is supported by the Lifelong Learning Programme of the European Union. Partners in this project are University of Augsburg, Bulgarian Academy of Sciences, University of South Bohemia, University of Bayreuth, Projekt Bildung Institut, German School Board Bolzano, University of Cyprus, Tyrolean Educational Service, University of Cambridge and University of Oslo.

2.1   DALEST Applications

The DALEST project supported the creation of two distinct applications. Chronologically the first one was a .Net monolith application for exploring stereometric problems. The other was a set of 10 Elica micro-applications covering various topics from Stereometry [8]. In this section we will present some of these micro applications. Their design was focused on providing tools which: (a) foster interactive using; (b) set up a platform for exploring various kinds of problems; (c) engage students with its 3D virtual worlds; and (d) encourage teachers to uncover their educational creativity.

The set of micro-applications include:

- *Cubix*, *Cubix Editor*, *Cubix Shadow* – applications for experiments in a cubic world: building structures, calculating volumes and surfaces, exploring symmetry and similarity, solving some class of volumetric and linguistic problems.

- *Potter's Wheel*, *Bottle Design*, *Math Wheel* – applications for educational and mathematical "playing" with rotational solids, calculating volumes and surfaces, decomposition of complex solids, approximation of curves and surfaces, 3D puzzles with solid constructions and Boolean constructive geometry.

- *Origami Nets* – application for constructing solids and their nets, making animations of folding, solving problems of plane tiling, building objects from polygons, measuring properties of polyhedrons.

- *Slider* – application to study the intersection of a solid and a plane, exploring conical sections and non-trivial sections of solids, demonstrations of properties of unusual surfaces like the Moebius strip.

- *Scissors*, *Stuffed Toys* – applications to solve the reverse problems of net folding, developing special thinking, activities with symmetry and rotation in 3D space, initial steps in topology.

All micro-applications share a consistent and intuitive interface. Students are able to work within the virtual world straight away. In fact they are encouraged to do so by manipulating virtual 3D objects. Students can rotate them and view them from different points of view (actually some situations do require such type of activities), then they start to modify the object by adding, removing or moving 3D elements.

Many of the problems encompassed in the Elica applications explore ideas which are rarely mentioned in conventional textbooks. Sometimes, they explore known ideas, but from a new perspective. Let's consider one trivial example of intersecting a cube with a plane. Most likely students would expect that this intersection is a square. However, the intersections could also be triangular, pentagonal and even hexagonal. **Figure 1** shows two snapshots from the Slider application. The left one is a composition of three non-square intersections of a cube and a plane; the right one visualizes the orientation of the cube in respect to the plane while producing pentagonal intersection.
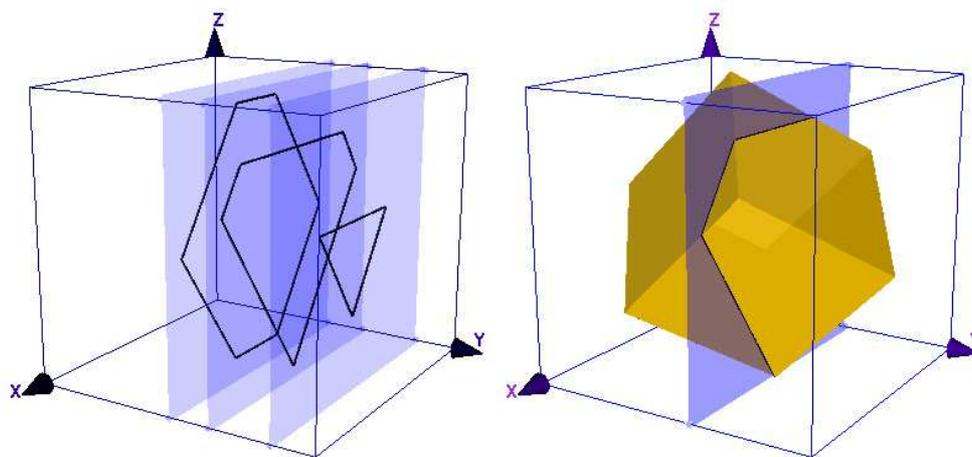
Figure 1 Exploring intersections of a cube and a plane with *Slider* application

In all applications students develop sense of space – this includes but is not limited to the ability of imagining special relations, relative and absolute positions, orientations, etc. All DALEST applications have been designed with some specific target set of educational activities.

Surprisingly, when the applications were used by students and there teachers, we realized, that they expand the range of possible activities into areas which we have never planned in advance. For example, using the *Cubix Editor* applications, students can build 3D constructions. The initial intention was to practice activities in the areas of special symmetry, expressing special properties with words, exploring the basics of solids' volumes and surfaces. However, students used the application to make their own discoveries.

3(11)
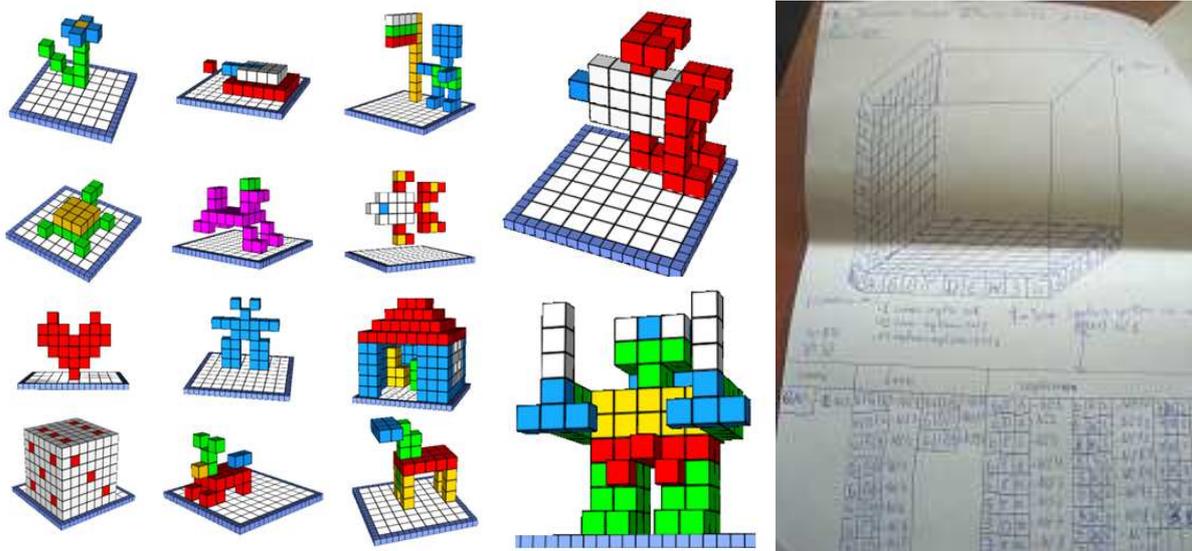
Figure 2 Rediscovering the 3D coordinate system

Figure 2 shows some figures made by students. The task was to build a 3D figure from differently coloured cubes following some constraints for the number of cubes. After finishing the construction the students have to describe it in such a way, that it can be reconstructed by somebody else. The figures on the left represent several typical constructions of students. On the right there is a photograph of student's paper showing the discovery of the three dimensional coordinate system. This discovery was neither planned, nor expected by us. The student discovered it while trying to describe his 3D figure in an unambiguous way.

2.2   InnoMathEd Applications

Although the DALEST project ended few years ago, our work on using IT in the classroom never stopped. Currently we use Elica to build applications in the frame of the InnoMathEd project. This is an active project and this section will describe briefly just one of the many applications. As in DALEST, the software uses virtual reality and 3D graphics to provide an educational playground for experiments and exploration. The inclusion of virtual reality provides two immediate and easily observed benefits: students and teachers become intrigued and inspired by the software; and it makes it possible to implement new and unique mathematical problems which are not possible to put in a paper textbook or in a hands-on pen-and-pencil activities.

The InnoMathEd applications are organized into sets and every set contains several modules accompanied with teaching activities.
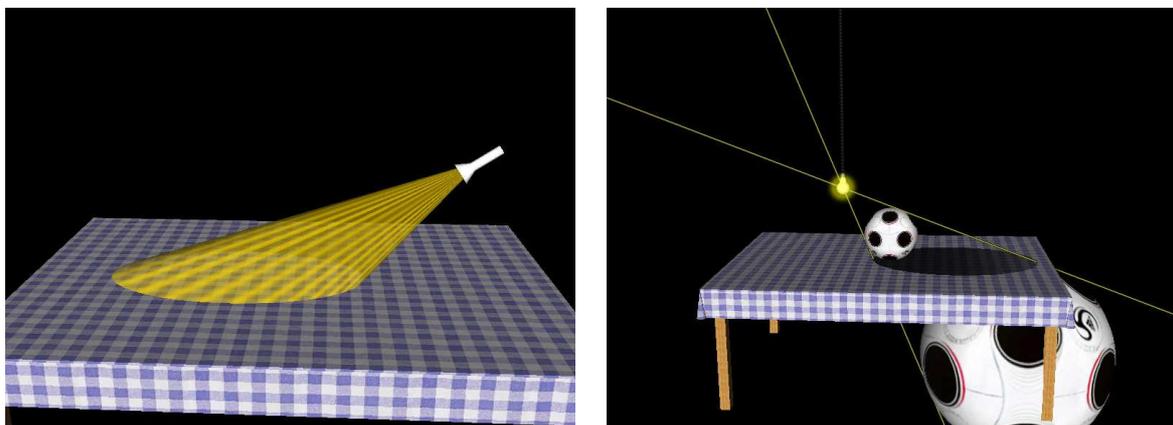
Figure 3 Virtual experiments with conical sections and light

4(11)

One of the set is dedicated to conical sections. The applications in the set represent virtual experiments that students can replay at home with real objects. Two snapshots are shown in **Figure 3**. The left one demonstrates the construction of an ellipse by using the light cone of a torch. The right one represents an ellipse made of the shadow of a ball. The construction is related to several mathematical theorems, like the one that the ball touches the table exactly at the ellipse's focus. The good thing about virtual worlds is that they can show things which are not easily available in the reality. For example, the right snapshot of **Figure 3** shows a second ball (the one below the table), which is circumflexed by the extended shadow and touches the other focus of the ellipse.

The InnoMathEd sets are still under constructions and we do not have any observations from their practical application. However, they have been demonstrated to teachers and educators who shared quite positive feedback.

3   Authoring tools and games

Most educational software has a specific target within the scope of one or two school subjects. IT solutions are easier to implement in scientific classes, mainly Mathematics and Computer Science. The driving power in the educational process is the creativity of the teacher. By using predefined applications the teacher becomes merely a narrator. To address this issue we have initiated the developing of new authoring software which allows the teachers to generate easily educational software by themselves.

We base our efforts on micro-applications, virtual reality and gaming. However, we also add subject-neutralism, so that teachers from virtually all subjects could use the software. There are three pilot prototypes representing skeletons of gaming scenarios, where teachers add their own content. The applications generated by the prototypes were tested with students and teachers. The evaluation showed that this novel approach was highly anticipated and accepted.

Many students are fans of computer games. This natural interest could be used to increase the educational effectiveness. Young students which start to learn a new subject are specifically pliable to gaming approach. Availability of computers, continued curiosity of students and affinity to everything new are the best factors to support education.

Being not just games, but also authoring tools, these applications are used by teachers. We did a survey about teachers' preferences in order to get first-hand feedback and ideas. What teachers expect from authoring gaming software is:

- To have rich resources
- To have intuitive interface
- To require no programming skills
- To be able to construct games quickly
- To be easy to learn how to use the software
- To provide a distinct aim for the student to strive for

Having these requirements in mind we have created several prototypes of educational authoring gaming applications. Although they are based on simple games, they provided a lot of challenges while designing and implementing them.

3.1   *Tic-tac-toe*

The first game that we implemented was *Tic-tac-toe*. It is a famous game played by two persons. They put their marks on a 3x3 grid. The person who manages to align three marks

horizontally, vertically or diagonally wins the game. Nowadays Tic-tac-toe is a widely computerized game. The first implementations were made several decades ago [5]. It is a perfect game for teaching winning strategies, although the game is not challenging for experienced players, they can always win or draw even [6]. It is not meaningful to implement the classical Tic-tac-toe, because this would be just one of the many existing implementation. Also, a classical Tic-tac-toe may not provide background for educational activities.

That is why we designed a new single-player version of the game. The board is still 3x3, but each cell covers a question. When a student clicks on a cell its question appears. A correct answer marks the cell with O, while a wrong one puts an X.

Being a single-player game, the student plays the role of both players. The first player represents the knowledgeable half of the student, and the second player is the ignorance half. **Thus while playing the game the student tries to defeat her lack of knowledge**.

The teacher has a full control over the game – from setting the background and cell images, to providing a pool of questions (and their correct answers), sound effects, complementary messages and so on. During the game the software will pick questions from this pool, so the same game with the same questions is not likely to be played twice.

Figure 4 Two instances of tic-tac-toe games

One of the first implemented demo-games is on an astronomical topic. The questions behind the cells are about the planets in the Solar System. Because of some historical reasons Pluto is included as a planet. The next table lists the questions included in the first implemented game:

Statement	Answer
This planet is closest to the Earth	Venus
This planet is the biggest	Jupiter
This planet has one more satellites than the Earth	Mars
The biggest part of the planet's surface is covered with water	Earth
There is no hotter planet than this in the Solar system	Venus
This planet is smaller that one of the Neptune's moons	Pluto
This planet has the biggest number of rings	Saturn
This planet makes the quickest tour around the Sun	Mercury
Only one planet is smaller then this one	Mercury
The planet is usually the outermost planet	Pluto
Only Jupiter is bigger than this planet	Saturn
This planet is the best place for life in the Solar system	Earth
This is the smallest planet	Pluto
Only Pluto is further away from the Sun than this planet.	Neptune
This planet has the largest inclination of rotational axis	Uranus
This planet has some of the deepest valleys	Mars

An interesting side effect of the game is that by playing the students tend to modify their winning strategy. A traditional Tic-tac-toe game is quite predictable, but this is not the case with our implementation. Consider a possible scenario of a game beginning as shown in

Figure 5. The student starts with two successful answers (O stands for correct answers, X – for incorrect). The third move is an attempt to make three O's in a line and win the game with just three answers. Unfortunately the third answer is wrong. The forth answer is also wrong. This situation shows a case where the traditional Tic-tac-toe reasoning becomes void. If the student selects to open the free cell on the top row, the game will continue independent on the correctness of the answer. However, if the student chooses the free cell in the right column, then a wrong answer makes her loose the game.
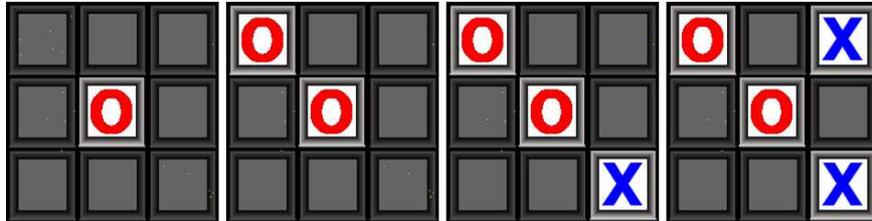
Figure 5 Example of possible first four moves of a Tic-tac-toe single-player game

We expect that the actual long-term strategy may depend on the self confidence of the student. An exact study of how students play the game may be carried out in the future.

3.2   Matching game

One of the main properties of the Tic-tac-toe game is that it can be applied to any subject. It is up to the teacher to choose a topic and provide a list of questions about it. This subject neutrality is a common feature of all our authoring games. For example, another game in the set is the Matching Game. The main goal in this game is to put a given set of items on their correct positions. Possible applications of this type of games are:

- Geology: Position of ores to their extraction sites
- Geography: Place names of cities over a map of a country
- Geometry: Match images of platonic solids with their names
- Foreign Languages: Move a translated word next to its original word
- Music: Align note signs on the staff to compose a specific melody
- Biology: Order creatures according to their hierarchy in the food chain
- Physics: Set force vector to the correct joints of a truss structure
- Algebra: Pair folded expressions with their equal unfolded ones
- Literature: Determine the author of each novel
- History: Match historical events with dates or people

Every problem which requires mapping of one set onto another can be virtualized as a matching game. As with the previous game, the teacher cat set background maps, images of the draggable objects, textual explanations and hints, sounds effects.

3.3   Labyrinth

The third game module is based on labyrinths. It is not implemented completely, so some of the things described in this section are still under design.

There is a maze of rooms and the student should find the exit. Rooms have questions in them – i.e. when the student enters a room its question is shown. Each question has several answers. Only one of them is the correct one. For each answer there is a door leading to another room. There is also a backdoor which can be used if a student wants to go back to the previous room (thus undoing her last answer).
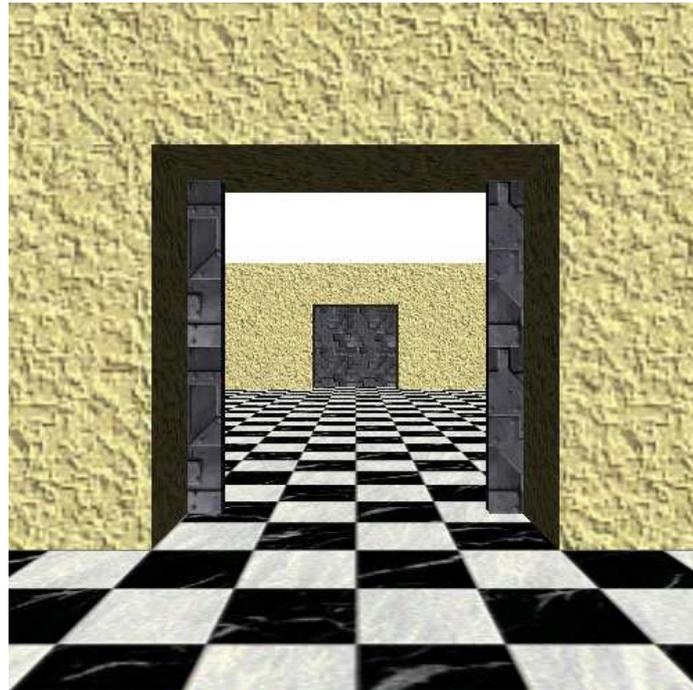
Figure 6 Interactive prototype with 3D animation, sound effects and mouse-based navigation

When the student answers correctly to all questions, then she has actually passed through the correct doors and manages to exit the labyrinth.

Initially wrong answers have been designed to lead to rooms without doors or with locked doors, but this would trigger the students to use a trial-and-error strategy. Thus it is decided that choosing the wrong door should not provide any immediate feedback to the player. Instead, sometime in the future (let's say after 2, 3, or even more rooms) the student will reach a dead-end. It is expected that this should provoke some pondering over the passed questions. When the student reaches a dead-end she does not know which door was wrong. She has to go back one step, recheck her answer. If she is absolutely sure that this answer was correct, then she can go backward one more room.

As with the previous games the teacher can make custom labyrinths by providing a set of questions and possible answers. To make the game more attractive (see **Figure 6**) the labyrinth is drawn and animated in 3D like a first-role game [7]. The doors make distinct sound while opening and closing and the steps of the player are heard as she walks through the rooms.

4   Learning by Programming

From all possible methods for creating educational software may be the most functionally rich is to use a programming language. At the same time this approach is considered too heavy for many of the students and their teachers, however, there are programming languages designed specifically for non-professional programmers. Elica is an example of educational software which provides the foundation of all three approaches described in this paper – exploring with prebuilt interactive environments, using authoring educational gaming software and learning through expressing ideas, experiments and knowledge in a high-level user-friendly programming environment.

Nowadays Elica is used as a core language for several university-level courses with students in Mathematics and Computer Science. It is suitable for teaching the basics of Computer

Science (Data structures and Algorithms), Object Oriented Programming [10], and Computer Graphics [11].

Elica by itself is also useful for building other programming-based systems which can be used both for teaching and as well for building a next level of programming educational environments.
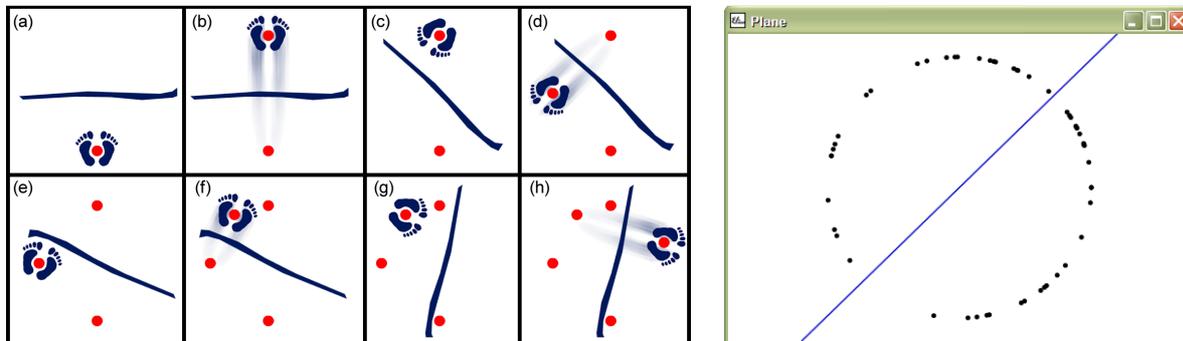
The player stands in front of a stick (a). Then jumps over it landing in exactly the symmetrical point – on the other side of the stick (b). Then the stick is rotated on an arbitrary angle but around its center (c). After this the players jumps again (d), the stick is again turned (e), the player jumps (f), the stick is turned (g), the player jumps (h) and so on.

```
run "reflect
ob "Angle 0
ob "Player point 100 0 [historical]
ob "Stick line point 0 0 :Angle
ob "Player reflect :Player :Stick
repeat 50 [ ob "Angle random 360 ]
ob "Positions :history.Player
```

Figure 7 Jump-over-the-stick game - description (*left*) and implementation (*right*)

A notable example for this is Geomland [12]. Initially it has been developed as a standalone Logo implementation with a focus on plane geometry. It has been used to teach programming to mathematicians; and to teach mathematics to programmers [9]. The new version of Geomland is re-implemented as a library in Elica. Geomland is used as a core development language for the InnoMathEd module on reflections. It describes a programmable and constructive approach for learning and practicing geometrical reflections. The module starts from building programs for the simplest cases of reflections and ends with a virtual simulation of a game based on reflections – see **Figure 7**.

Another implementation of a programming language build on top of Elica is the geographical language described in [13]. The initial idea of this language was to make it possible to reconstruct the first airplane flight over the North Pole in 1926. The language contains commands to control take off, landing and flight speed. Except for the original flight it is possible to construct and study mathematically a broad set of trajectories, some of them are shown in **Figure 8**.
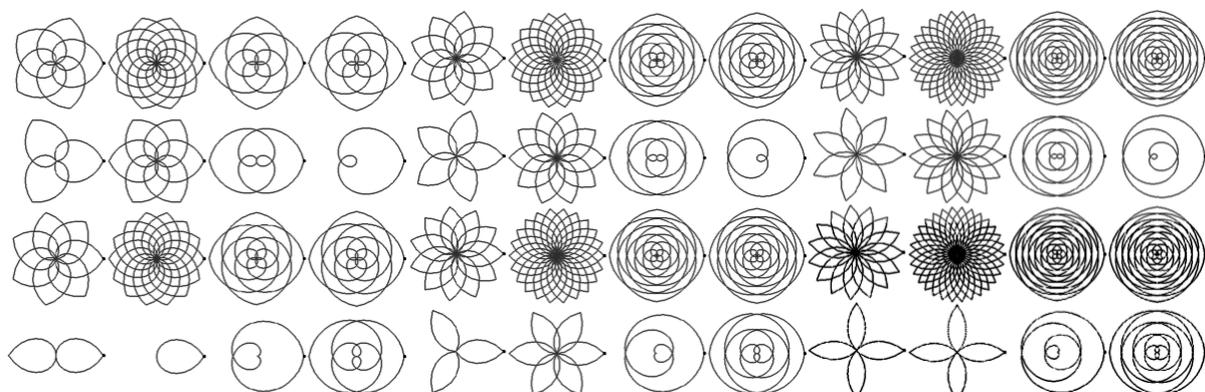
Figure 8 Flight trajectories generated within the North Pole Adventures

5   Conclusion

The paper presents a new approach of using IT for education. It described briefly the educational environment Elica and its multi-purpose usage. Elica can be used as a programming environment suitable for students which start to learn programming. It can also be used to build other domain-specific programming environments like The North Pole Adventures, Geomland and Reflection – **Figure 9**.

Being a general-purpose system, Elica can be used as a development platform for virtual worlds implemented through intuitive and interactive virtual reality. More than a dozen of such applications are already built along two European projects – DALEST and InnoMathEd. Finally, Elica can also be used to build authoring tools for teachers with no programming background.
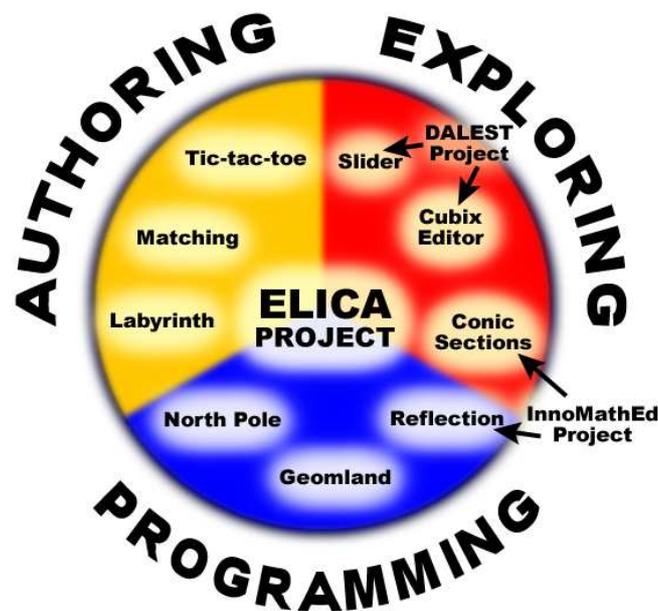
Figure 9 Elica as a core development platform

All Elica applications are distributed as Open Source, so the software can be investigated and modified freely. Several of the Elica-based applications have been already improved by users. For example, the original version of the Equation Balance application (not included in **Figure 9**) worked only with positive numbers and has been modified by users to also accept negative numbers.

The main outcome from using a single development system is that all applications, independent on whether they are authoring tools, virtual models or programming environments, share the same core and features. This makes it possible to gradually move from using applications through authoring games and up to building own programs. This transition would be much harder to perform if different programs are used for each type of activity.

Several courses at Sofia University are based on Elica and they are for students which will become teachers in Mathematics and Computer Science. Being in touch with the system that is used to implement a classroom software is an important factor, because several of the applications are already a part of the IT textbooks for 6th and 7th grades.

Nowadays Elica is used in several national and international projects. Within the next year a dozen of new applications is supposed to appear. Additionally it will be possible to collect a more significant feedback from teachers and students.

Acknowledgements:

References:

[1]   Department for Children, Schools and Families: Learning, Teaching and Managing Using ICT, http://www.teachernet.gov.uk/wholeschool/ictis/
[2]   Boytchev, P.: Elica Site, http://elica.net
[3]   DALEST Project Web Page, http://www.ucy.ac.cy/dalest
[4]   InnoMathEd Project Web Page, http://www.math.uni-augsburg.de/prof/dida/innomath/
[5]   The History of Noughts and Crosses, http://www.adit.co.uk/html/noughts_and_crosses.html
[6]   Tic Tac Toe Strategy Guide, http://www.chessandpoker.com/tic_tac_toe_strategy.html
[7]   Landerer Cl.: Programming for the Nintendo-Generation from Student's Minds to Virtual Reality. In: Dagienė V., Mittermeir (Eds): ISSEP Proceedings (2006) 361-372
[8]   Boytchev P., Chehlarova T., Sendova E.: Enchancing Spatial Imagination of Young Students by Activiies in 3D Elica Applications. In: Boyvalenkov P., Rusev P. (Eds): Mathematics and Education in Mathematics (2007) 109-119
[9]   Sidericoudes O.: Integrating Logo into Mathematics Classroom Activities. In: Nikolov R., Sendova E., Nokolova I., Derzhanski I. (Eds): Proceedings of 7th Eurologo Conference (1999) 317-326
[10] Boytchev, P.: Natural Object-Oriented Programming: The Evolution Meta-phor behind Elica, in Proceedings of the 2nd International Conference "Informatics in Secondary Schools: Evolution and Perspectives, ed. V. Dagiene and R. Mittermeir, Vilnius, Lithuania (2006) 351-360
[11] Boytchev, P.: Design and Implementation of a Logo-based Computer Graphics Course, Informatics in Education, Volume 6, No. 2, Vilnius: Institute of Mathematics and Informatics (2007) 269-282
[12] Sendov B., Boytcheva S.: GEOMLAND - Computer Laboratory for Mathematical Explorations, Technologies in Information Society, Volume 1, TEMPUS-JEP-07272-94, Published by IK Znak and IK Intela (1996)
[13] Boytchev, P.: North Pole Adventures, International Journal of Computers for Mathematical Learning, Volume 7, No. 2, Ed. S. Papert and R. Noss, Netherlands: Kluwer Academic Publishers (2002) 217-242

Author(s):

Pavel Boytchev, Associated Professor, Ph.D,
Eliza Stefanova, Assistant Prof,
Evgenia Kovatcheva, Assistant Prof,
Nikolina Nikolova, Assistant Prof,
FMI-KIT, Faculty of Mathematics and Informatics, University of Sofia "St. Kl. Ohridski"
5, James Bourchier blvd, 1164 Sofia, Bulgaria
boytchev@fmi.uni-sofia.bg
eliza@fmi.uni-sofia.bg
epk@fmi.uni-sofia.bg
nnikolova@fmi.uni-sofia.bg

Evgenia Sendova, Associated Professor, Ph.D
Institute of Mathematics and Informatics, Bulgarian Academy of Sciences
Acad. G. Bonchev, bl. 8, Sofia 1113, Bulgaria,
jenny@math.bas.bg

Stanislava Kamenova
High School of Mathematics and Science "Akademik Ivan Tsenov"
18, Demokratsia blvd, 3000 Vratsa, Bulgaria
kamenova@bitex.bg