Analysis of a selection of interactive learning environments for mathematics

Jana Trgalova, Denis Bouhineau, Jean-François Nicaud

kaleidoscope
concepts and methods for exploring the future
of learning with digital technologies

D20.5.2 (Final)

Analysis of a selection of interactive learning environments for mathematics

Main author : Jana Trgalova (UJF)

Nature of the deliverable : Report

Dissemination level : Restricted

Planned delivery date : November 2007

No part of this document may be distributed outside the consortium / EC without written permission from the project co-ordinator

Prepared for the European Commission, DG INFSO, under contract N°. IST 507838
as a deliverable from WP20
Submitted on 19-10-2007

Summary

1) Analysis of the ILEs developed by TELMA teams: Aplusix, AriLab2, and e-Slate.
2) Critical analysis of the grid yielding a refined methodological tool for ILEs analysis.
3) Study of T-algebra, an additional ILE for the learning of elementary algebra;
4) Comparison of 10 ILEs based on the exploitation of the analysis grids.

History

Filename	Status	Release	Changes	Uploaded
D20-05-02-F.pdf	Final	1		19/10/2007

Contributor(s)

Name, First name	Contractor
Bouhineau Denis	(UJF)
Nicaud Jean-François	(UJF)

Contents

TELMA Teams

(ITD-CNR)	Consiglio Nazionale Ricerche – Istituto Tecnologie Didattiche – Genova - Italy
(UNILON)	University of London - Institute of Education – London - UK
(DIDIREM)	University Paris 7 Denis Diderot - DIDIREM – Paris - France
(NKUA-ETL)	National Kapodistrian University of Athens - Educational Technology Lab – Athens - Greece
(MeTAH)	MeTAH and Leibniz - IMAG, Grenoble, France
(Siena)	University of Siena - Department of Mathematics – Siena – Italy

1. Introduction

This is a second deliverable devoted to a study of several educational ICT systems in the field of arithmetic and algebra. In accordance with the previous deliverable (Del 20.05.01, 2006), the systems studied will be called ILEs (Interactive Learning Environments) in the rest of the text since they are interactive software.

As announced in the first deliverable, "*the goal of this study is to have a good knowledge of the main existing ILEs for arithmetic and algebra, produced by research laboratories or companies, which are either products or advanced prototypes, are in use in several schools or have been experimented with a lot of students, and which have a high level of interactivity*" (ibid, p. 3).

The first phase of this study, reported in the first deliverable, had been carried out from January to October 2006 and allowed to elaborate a software analysis grid which was used to study 6 ILEs selected a number of systems according to the above mentioned criteria.

The second phase of the study, which is reported in the present deliverable, was conducted from November 2006 to August 2007 and consisted in three stages:

1) Analysis of the ILEs developed by TELMA teams, namely

- Aplusix developed by MeTAH team, study carried out by Siena team;
- AriLab2 developed by CNR-ITD team, two independent studies carried out by DIDIREM and ETL-NKUA teams;
- e-Slate developed by ETL-NKUA team, study carried out by IoE team.

2) Critical analysis of the grid yielding a refined methodological tool for ILEs analysis;
3) Study of T-algebra, an additional ILE for the learning of elementary algebra;
4) Comparison of 10 ILEs based on the exploitation of the analysis grids (3 ILEs developed by TELMA teams, and 7 ILEs developed by teams outside TELMA).

2. Analysis of the ILEs developed by TELMA teams

The ILEs developed by TELMA teams were analysed by teams who used these ILEs in the cross-experimentation.

2.1. Analysis of Aplusix

By Laura Maffei, Siena team

Design and development

Name of the ILE: Aplusix (developed by MeTAH team)

Date of the study: November 2006

Who

People/organisation involved in the design : J.F. Nicaud, D. Bouhineau, S. Mezerette (MeTAH)

Interaction between these participants

Why

Aims of the design

Aplusix is a computer-system in which students work within the domain of arithmetic and algebra (Nicaud & al. 2004). In this environment pupils may develop the calculations as they are used to do in paper and pencil.

The authors, starting from the assumption that acquiring skills in reasoning by equivalence is the main goal to be reached for solving algebraic tasks, developed a program providing a feedback on two consecutive steps by means of the equivalence verification.

The peculiarity of the microworld consists of its twofold functionality (action /revision): on the one hand it offers the user (for instance a student) a well-structured space where to perform tasks; on the other hand it allows the user (a teacher, a researcher, or a student) to revise step by step the given solution. Moreover, in the ‘action’ functionality, the microworld provides two main kinds of interaction modes, called *training mode* and *test mode*. In training mode Aplusix verifies the calculations by checking the equivalence between two consecutive steps and points out the presence of errors (Fig. 1). Blue cross lines show that the expression you are writing is not well-formed (i.e. a plus sign need an argument), black lines show that the first expression is equivalent to the second, red cross lines show that the first expression is not equivalent to the second.

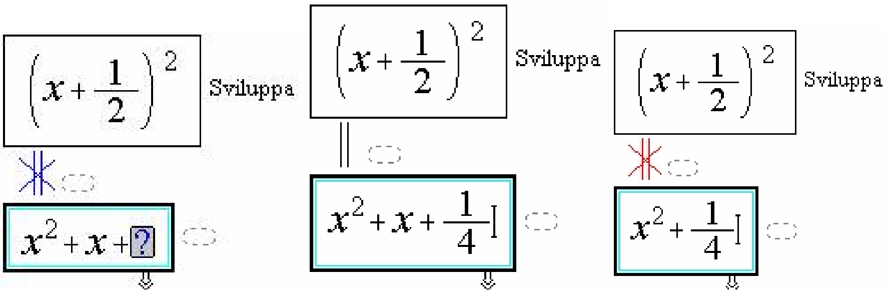

$\left(x+\frac{1}{2}\right)^2$
Sviluppa
$x^2+x+?$
$\left(x+\frac{1}{2}\right)^2$
Sviluppa
$x^2+x+\frac{1}{4}$
$\left(x+\frac{1}{2}\right)^2$
Sviluppa
$x^2+\frac{1}{4}$

Fig. 1. The three different signs provided by Aplusix in training mode.

In the test mode no feedback is provided (Fig. 2).

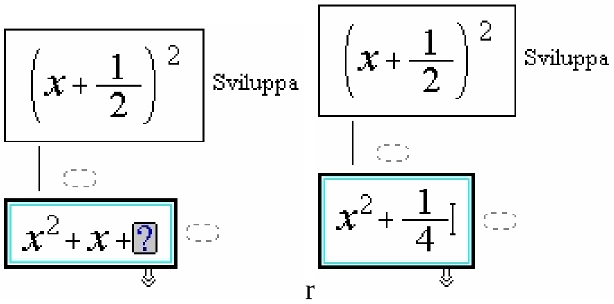

$\left(x+\frac{1}{2}\right)^2$
Sviluppa
$x^2+x+?$
$\left(x+\frac{1}{2}\right)^2$
Sviluppa
$x^2+\frac{1}{4}$
r

Fig. 2. In test mode, a single black line links permanently two consequent steps.

Thanks to a good editor, tasks can be organized creating suitable lists of exercises, so that the teacher may plan pupils’ activities according specific didactic goals. Afterwards, it is possible to revise the work done by means of the *Replay System.* This facility is very useful for the teacher who, passing through the sequence of his /her actions, can observe the difficulties encountered and the errors committed by a pupil; but it is also useful for the students who can revise their work and correct their own errors. This command presents significant advantages compared with the revision of a work done with paper and pencil, where many of the traces of the solution process are lost, so that one can't reconstruct the precise order in which the calculations have been developed.

Research questions

1. Assuming that the most important mathematical skill for students who start to study algebra is to carry out correctly literal calculations, the developers address the question:

How is it possible to make a computer system to help pupils learning such a calculation skill?

2. Then, the second goal to be reached in algebra consists of gaining abilities in solving problems. That means, first of all, formulating a mathematical model of the situation. According to this educational goal, the developers built the problem activity which aims to help the writing in equations of a problem.

How is it possible to make a computer system to help pupils learning such a symbolization skill?

How

Constraints affecting the design

When Aplusix verifies the correctness of an equation requires that the solutions are given in a way which is not correct from a mathematical point of view. In example, the solutions of the equation $x^2 - 1 = 0$ have to be given in this form: $x = 1$ or $x = -1$, that is by means of two equations. But the solutions are the real number 1, -1.

Underlying theories and principles

The design of the software is inspired to the assumption that 'learning algebra requires gaining competence in calculations'. Lots of students, for different reasons, meet great difficulties in getting this basic competence, which on the contrary assumes a great value, both for the students and the teachers. Aplusix aims at helping pupils (but also teachers who could better use their time) to gain this competence in an autonomous way.

In fact, according to the theory of didactical situations (Brousseau, 1997) Aplusix is considered a *milieu for validation* thanks to its peculiarity to show to pupils if their answers are correct or not without the intervention of the teacher.

Start of the design and number of versions

Aplusix has had two phases:

The first phase began in 1982 and ended in 1999. Aplusix was a command driven software developed in the Lisp programming language.

In 1985, J.F. Nicaud designed a language, called SIM, that represents objects and rules.

Then, the second phase began in 2000. The current Aplusix is editor-based software developed in the Delphi object Pascal programming language. As it was adequate for our goal, J.F. Nicaud implemented the SIM in Delphi.

Development status (advanced prototype, beta product, product): Product

For whom

Intended Audience

Aplusix it is intended for use by students aged 13-16 years old.

Distribution

License

Aplusix is a commercial product sold by several publishers.

There is a free version for researchers.

Price

Aplusix is currently commercialized in French language by Les Editions Archimède. The price of single software is 55 euros. The price for a school site license is 600 euros.

Aplusix is commercialized in UK and Italy since January 2007.

Artefact

Technical aspects

Operating System (or Web)

APLUSIX runs on Windows 95, 98, NT, Millenium, 2000, XP, Vista.

Programming Language

Delphi object Pascal programming language

Translations

Aplusix is available in French, English, Italian, Spanish, Portuguese and Vietnamese.

Interface

Representations

- **what rep.**

Aplusix contains a 2D editor of algebraic expressions. It is also possible to write comments for each step of the solution.

*** nature (usual, new)**

Algebraic expressions are displayed in the usual 2D representation.

*** quality (poor, medium, rich)**

rich

Manipulations

- **on what (objects, rules…)**

Aplusix allows manipulating expressions in a *structure mode*. This means that actions are executed with respect to the algebraic structure. In example, if $x+2$ is copied in the clipboard and y is selected in $3xy$, a paste provides $3x(x+2)$.

*** quality (pleasant, painful)**

It is considered pleasant to use by students once having acquired a good level of familiarization with the use's schemes of the software.

Domain

Mathematical content

Aplusix has two domains: the first one concerns the verification of the calculations (it is the set of expressions for which Aplusix can do this verification); the second one concerns the solutions and the scores.

In the current version, the domains are:

Type of exercise	Domain *Verification of the calculations.*	Domain *Solutions and Scores.*
Numerical calculation	The expressions must not include variables.	The expressions must not include variables.
Expansion	Polynomial or rational expressions.	Polynomial expressions.
Factorisation	Polynomial expressions with a degree less or equal 4.	Polynomial expressions with a degree less or equal 2.
Solving equations	Polynomial equations with a degree less or equal 4 and rational equations giving such polynomial equations.	Polynomial equations with a degree less or equal 2.
Solving inequations	Polynomial inequalities with a degree less or equal 4 and rational inequalities giving such polynomial inequalities.	Polynomial inequalities with a degree equal 1.
Solving systems of equations	Systems of equations with maximum 10 equations and 10 unknowns.	Systems of equations with 2 equations and 2 unknowns.

Sort of exercises

* formal exercises

The editor of Aplusix allows creating exercises entering an **exercise type** (Calculate, Expand, Factor or Solve) and an **expression**. Aplusix recognizes the solved form of these types of exercises. It is possible to add an **instruction** to an exercise. In that case, the instruction is shown to the student instead of the exercise type.

* word problems

The editor of Aplusix allows problems expressed in natural language. They may include figures having the form of bitmap (BMP format). A problem may contain several questions, linked or not. The questions are called sections.

Each section is made of:

- An information zone with the section title and a text describing the situation and asking a question. The text may include images.
- A calculation zone that can be left empty by the teacher or be filled with an exercise,
- An answer zone in which the teacher writes the expected answer. This zone can be empty if an exercise of the type Calculate, Expand, Factor or Solve has been put in the calculation zone.
- Equivalent expressions. In this mode, Aplusix will use the equivalence of expressions to make the comparison. If the student's answer is “x=1” and the expected answer is “2x+4=6”, the student's answer will be accepted.

There are three modes for comparing the student answer and the Aplusix expected answer:

- Identical expressions.
- Syntactically closed expressions. In this mode, Aplusix will use commutativity, associativity and simplifications with 0 and 1 for the comparison. If the student's answer is "y+x=1 and 2x+1y=2" and the expected answer is "2x+y=2 and x+y=1", the student's answer will be accepted.
- Equivalent expressions to make the comparison. If the student's answer is "x=1" and the expected answer is "2x+4=6", the student's answer will be accepted.

Types of activities

Lecture

No

Examples solved by the ILE

No.

Exercises to be solved by the student

Yes.

Feedback

Immediate/delayed

Aplusix provides both immediate feedback and delayed feedback. In fact, when the software environment is set up in *training mode*, the system displays step by step the equivalence of two consecutive steps, that it gives an immediate feedback. On the contrary, if the software is set up in *test mode* no feedback is provided. However, a pupil can revise and, if necessary, correct the solved exercises by means of the Self-correction activity. In this activity the accomplished calculation are displayed to the in training mode: the feedback has been delayed.

Nature

The feedback is based on the equivalence correctness.

Diagnosis

The verification is shown on the links between steps. These links are:

- A double black line means "The expressions are equal" or "The equations, inequalities, systems of equations are equivalent"
- A double blue crossed line means "One expression is achieved or undefined".
- A double red crossed line or arrow means "The expressions are not equal" or "The equations, inequalities, systems of equations are not equivalent".
- A single black line means "No verification is done".

Interactivity

Microworld

X

Hypermedia

Answer

Multiple choice question: No

Final open answer: Yes

Step by step answer

The student may give step by step answer and the final answer (in training mode).

No answer

In test mode Aplusix gives no answer.

Autonomy

Degree of student's autonomy

The program gives to students a high level of autonomy thanks to the provided feedback. Students may interact freely with the software, that means editing and solving exercises they want, without the teacher's intervention.

Self-evaluation

The provided feedbacks constitute the best way students have for evaluating on their own. In addition, Aplusix displays the gained scores in solving an exercise.

Scores are calculated considering the progress of the reasoning made with correct calculations. When nearly in solved form, with correct calculations, the score is close to the maximum. When a solved form has been given but "Solved" has not been indicated, the score will not be the maximum.

If there are incorrect calculations, Aplusix considers the situation before the first incorrect calculation. If the score is not high, Aplusix looks at the calculations after the incorrect one and may increase the score.

There is no score for problems.

Knowledge of the task

The ILE knows what is the student's task

Usually students choose the task.

The ILE can provide hint

Yes.

The ILE can provide the solution

Yes.

The ILE can solve the exercise step by step: No

Teacher's role

Parameterization

The teacher can set up the software as he /she wants.

Production of scenarios

Production of exercises

The teacher can prepare lists of exercises and problem using the Editor application.

Validation, usage, experimentation

Experimentation

Usage analysis

Experiments with Aplusix in France, Brazil, Italy, India and Vietnam have shown that APLUSIX:
- Is easy and pleasant to use by students (it has a good editor and relevant feedbacks)

- Allows a significant improvement of students' skills even for rather short uses (it has been measured with tests on paper)

- Facilitates the teacher's work during sessions (autonomy of the majority of the students), which allows him/her to spend more time with students having difficulties with algebra.

Qualitative results

Qualitative results are very promising: interacting with Aplusix in training mode not only reinforce algebraic skills, but also allows student to acquire a self-consciousness of their difficulties. In addition, the direct interaction with the software (without the teacher's mediation) lead pupils to take personally charge of their errors and, consequently, to find strategies to overcome them (Maffei & Mariotti, 2006).

Quantitative results

High school of Seyssinet, France, with 28 students: after 50 minutes of use, the rate of well solved isomorphic exercises is up 100%.

City of Campo Grande, Brazil, with 1120 students: after 50 minutes of use, the rate of well solved is up 30%.

Usage

Familiarization with the ILE

The ILE is user-friendly: a short time is needed for familiarizing.

Evolution

A step by step solver has been developed in a prototype and could be added to the product.

Annotations by the teacher on the student calculation sheets have been developed and will be added to the product at the end of 2007.

Webservices are being developed and will be added to the product in 2008.

Tree representations of algebraic expressions have been developed in the context of the ReMath project (http://remath.cti.gr) and will be experimented at the end of 2007.

Graphical representations of expressions are currently developed in the context of the ReMath project (http://remath.cti.gr).

2.2. Analysis of AriLab2 – Symbolic microworld

By DIDIREM team

Design and development

Name of the ILE: AriLab 2

Date of the study: 25 May 2007

Who

People/organisation involved in the design : Bottino R. M., Chiappini G. / **ITD** (Institute for Educational Technologies)- CNR (Italian National Research Council)

Interaction between these participants

AriLab has been a long-term research project carried out by CNR researchers working, first, at the IMA, then at the ITD.

WHY

Aims of the design

AriLab 2 is an open multi-environment system for developing skills in solution to arithmetical problems for children of the primary and the lower secondary schools.

Research questions

The authors of the ILE formulate three main questions based on the Cole and Engeström's model of activity (cf. TELMA cross-experiment[1]):

- *"How the educational technology used can mediate new ways for the learner of accessing, representing, and interact with the concepts, procedures, and rules that are involved in the acquisition of a given mathematics knowledge which constitutes a learning object for a teaching and learning activity.*
- *How the educational technology used can contribute to the design and the enactment of didactical practices aimed at an evolution in the use of the rules related to the knowledge to be learnt and to the construction of appropriate ways and meanings for using them.*
- *How the educational technology used can contribute to mediate the assumption of new and old roles by participants in the didactical practice."*

[1] http://telma.noe-kaleidoscope.org/docs/outcomes/TELMA-Guidelines.doc

How

Constraints affecting the design

There were two possibilities in designing the communication environment present in AriLab 2: a design based on client-server architecture or a design based on a direct link among computers. Authors chose the second possibilities, because specific constraints were individuated in the structure of the primary schools in Italy

Underlying theories and principles

The authors use a social constructive perspective and refer in particular to the Activity Theory framework (Cole and Engeström).

Start of the design and number of versions

The design of AriLab 2 started in 2002 in the ITALES project. It is the new version of AriLab, software developed in 1999.

Development status (advanced prototype, beta product, product)

Product

For whom

Intended Audience

AriLab 2 addresses two kinds of users: the teacher and the pupil of the primary and the lower secondary school.

Distribution

License

DIDA.EL S.p.A. - Via Lamarmora 3/A20122 – MILANO

Tel. 025418091 Fax: 02 55181751

E-mail: catalogo@didael.it

Price

46,48 Euro

Artefact

The whole environment is composed by ten microworlds, a teacher's and a student's environment and also a communication environment. We have especially studied the Symbolic manipulator, one of the microworlds of AriLab 2.

Technical aspects

Operating System (or Web): MS Windows

Programming Language: Toolbook

Translations: English, Italian, Spanish

Interface

Representations

*** what rep.**
Two representations: literal and numerical expressions (integer and rational forms but not decimal form) and some fixed rules of manipulation of the (sub) expressions. There is no representation for exponentiation.

*** nature (usual, new)**
The window is divided into three spaces: the problem text space, the button space, the demonstration space. The demonstration space is itself divided into two columns. One column is devoted to the different steps of the manipulations and another one devoted to the rules used at each step. This spatial organisation is not usual but can exist in other environments.
In the first column, there is no equal sign between the successive expressions obtained at each step as is usual.
The representation of the rules is new. The use of the <–> or –> replaces the equal sign and thus represents the possibility of transforming one expression into another. This is not usual in the paper-and-pencil environment at least in some countries where computational aspects are highlighted.

*** quality (poor, medium, rich)**
Medium or rich: the representation with two columns is interesting because it allows reasoning but representation in natural language is missing.

Manipulations

*** On what (objects, rules…)**
Manipulations on numerical (with integers or rational) or literal expressions by applications of rules selected in the button space: selection of a part of the expression can be accomplished with the mouse. The part of the expression selected is highlighted and, in the meanwhile, the rules (axioms or properties) that can be applied to the expression selected are made active. The application of a rule to a selected expression produces a transformation that corresponds to the chosen rule. The transformed expression is written below and the rule is written in the second column of the demonstration space.

*** quality (pleasant, painful)**
Pleasant: adapted to intended audience (beginners in algebra) because it is necessary to detail each step to transform expressions.
It nevertheless can be painful and boring for more advanced pupils.

Domain

Mathematical content

Firstly, algebraic expressions of first degree but possibility to work on rational expressions which lead to second degree.

Sort of exercises

* **Formal exercises:** main possibility.

* **Word problems:** not really well adapted to solve a word problem because equations can't be solved except for some particular proof problems: for instance, it is possible to propose a problem where the goal is only to transform expressions into others to prove numerical proprieties.

Types of activities

Lecture: No

Examples solved by the ILE: No

Exercises to be solved by the student

Open environment : no predefined list of exercises inside the ILE

Feedback

Immediate/delayed

Immediate feedback

Nature

The selection of a part of an expression activates the possible rules (in the button space) that can be applied by the pupil.

The application of the selected button provides the transformed expression and the rule used is written in the demonstration space.

The “no valid expression” message is given when the pupil enters a new expression to be transformed with a syntax error. Some expressions mathematically correct are refused by the ILE (e.g. *((2+x)3+x)* is not recognized. It is necessary to enter ((2+x)*3+x)). Some other expressions are not recognized but modified without any feedback (for instance *(a+b)^2* becomes *a+b NON, the system write “no valid expression”) 3(2+x)* becomes *3* etc. because it is necessary to write the “*” between 3 and (2+x)) and some expressions are simplified without any feedback (e.g. *((3+**x**)+5)* is transformed into *3+**x**+5*).

Diagnosis

No. Pupils can only perform valid actions but can go on without ending. The ILE doesn't evaluate the pupils’ choices (the process). It doesn't evaluate if the pupils have succeeded in solving the exercises.

Interactivity

Microworld: Yes, pupils can define their own expressions to be simplified.

Hypermedia: Yes, pupils can go from a microworld to another.

Answer

Multiple choice question: No

Final open answer

A final answer can be produced by the pupils by copying and pasting objects from one of the different microworlds to the solution sheet. This final answer can only be evaluated by the teacher.

Step by step answer: No

No answer: Yes

Autonomy

Degree of student's autonomy

The degree of autonomy is quite important as regards the whole process of resolution but it is less important at the local level because pupils can only do valid actions.

Self-evaluation: No

Knowledge of the task

The ILE knows what is the student's task: No.

The ILE can provide hint

The ILE doesn't provide any hint, but the teacher can provide some hints for allowing the pupils to begin a given exercise. The communication module also allows the teacher to communicate and to give some help to the pupils during the lesson.

The ILE can provide the solution: No.

The ILE can solve the exercise step by step: No.

Teacher's role

Parameterization

The teacher chooses the available microworlds.

Production of scenarios

The teacher provides the "beginning" of each situation: the text of the problem, the microworlds available and eventually some hints. During the lesson, she has to attribute the different exercises to the pupils (individually or by group). She can prepare some a priori hints, useful during the lesson, but not in the ILE itself. There is no contextual hint about the tasks and about the result of the use of the valid mathematical rules proposed. Nevertheless, the teacher can provide interactively some contextual written help through the communication module during the lesson, and rely on the students' production for orchestrating collective discussions.

Production of exercises

The teacher prepares some exercises and makes available the microworlds attached to each exercise.

Validation, usage, experimentation

Experimentation

The Symbolic manipulator microworld has not been experimented by the DIDIREM team so we propose below a few elements for an a priori analysis.

Usage analysis

The microworld and especially the algebraic transformation rules seem well adapted for the first steps in the world of algebraic manipulations. It allows the pupils to explicit and understand precisely the formation and transformation rules of algebraic expressions. Nevertheless, in usual practices (for instance in France), teachers and pupils don't explicit these as precisely as the ILE does. This point might generate difficulties in the integration of ILE in usual practices. The teacher would have to take in charge this “new” level of detail in algebraic manipulations.

The ILE is no longer adapted as soon as the pupils become more familiar with algebraic manipulations. For instance, too many manipulations are required for transforming $x+x$ into $2x$ once a pupil has reached a first level of familiarity.

Qualitative results and quantitative results

They concern Arithmetic solving and not only symbolic manipulator

Within the activity theory framework, Cole and Engstrom’s model has allowed research equip both to perform detailed analysis of changes brought about in learning environment as a result of technology-mediated activity and to examine how these influence the students’ learning processes. The technology can play a crucial role in

- Developing responsibility with validation processes,
- Developing a social practice that provides the students with assistance for overcoming the difficulties encountered while tackling the tasks set,
- Favoring elaboration of personal experience,
- Favoring conversion of the solution into different representation registers and providing tools for coordinating them, and
- Favoring learning by analogy,
- Offering new possibilities for exploring the mathematical knowledge involved in solving problems

For qualitive and quantitative results from bibliography:

In English: http://www.itd.ge.cnr.it/arilab/biblio_eng.html

In Italian: http://www.itd.ge.cnr.it/arilab/biblio_ita.html

Usage

Familiarization with the ILE

No specific difficulties for becoming familiar with this ILE. Nevertheless, the familiarization must be thought in connection with the teaching of the use of algebraic transformation rules in paper and pencil environment.

Evolution

A new version of the Symbolic Manipulator microworld is implemented in a new system: Alnuset (Algebra on the numerical sets) which is developed in the ReMath project.

In particular, Alnuset can be considered the evolution of two microworlds of ARI-LAB 2: the Fraction microworld and the Symbolic Manipulator microworld.

2.3. Analysis of AriLab2 – Fraction microworld

By Giorgos Psycharis, Educational Technology Lab, National & Kapodistrian University of Athens (ETL-NKUA) (second study).

Design and development

Name of the ILE: AriLab2 - Fractions microworld

Date of the study; November 2006

Who

People/organisation involved in the design: Consiglio Nazionale delle Ricerche – Istituto Tecnologie Didattiche – Genoa, Italy (CNR-ITD).

Interaction between these participants:

Why

Aims of the design:

AriLab2 is a stand-alone multi-environment open system; this means that it is constituted by a number of interconnected environments that support different and complementary pedagogical opportunities. It integrates hypermedia and network communication tools oriented to the development of arithmetic problem solving activities at primary and lower secondary school levels. It can be considered as an example of a new generation of ICT-based learning systems for primary and secondary education (*a situated multi-environment system* in the words of Bottino & Chiappini, 2002) which is characterized by a strict integration of visualisation, communication and re-elaboration tools aimed to support both the creative exploration of problems and the representation, validation, and communication of knowledge. AriLab2 addresses both teachers and students by providing two main environments: the **Teacher Environment** and the **Student Environment**.

By interacting with AriLab2, the user can build her/his own solutions, by interacting with a structured and interconnected set of tools including:

- an environment for building and describing problem solutions (**Solution Sheet environment**);
- **microworlds** that embody an abstract domain of knowledge described by a model for representing problems and offering a variety of ways to achieve a goal. The available microworlds are: Euro, Abacus, Calendar, Number Building, Fractions, Graphs, Simplified Spreadsheet, Operations, Fractions and Arithmetic Manipulator. The Fractions Microworld (FM) used in the cross-experimentation combines graphical and symbolic notation of fractions represented as points on the number line;
- a communication environment that enables exchange of messages and solutions among users (**Communications environment**);
- a database of solved problem.

The best exploitation of all the functions of AriLab2 can be achieved by using it in a lab (or in a class) with computers (PC supporting Windows) connected to a local network. Nevertheless, it is also possible to use it in a stand-alone version without all its functions available. When the students and the teachers took their seats in the lab, the context of the use of the system can be schematised like this:

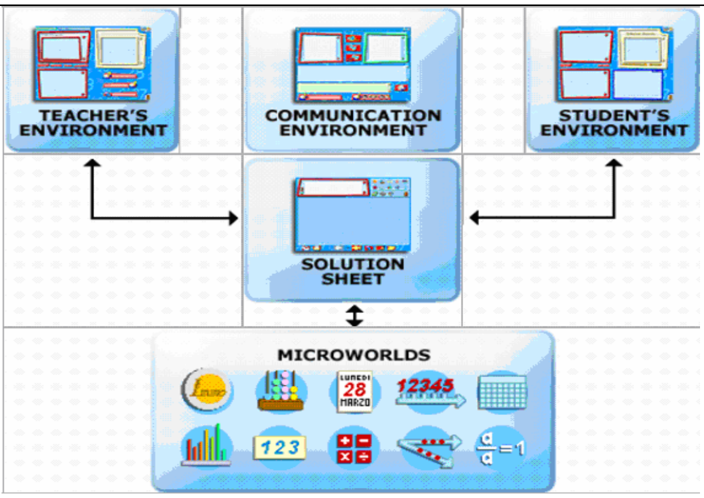


Figure 1: The environments available to the AriLab2 users.

Research questions:

The general research area is the study of learning, teaching, contextual and organizational processes during the infusion of innovative educational environments in everyday school practice. The CNR-ITD team's research is characterised by an integrated approach where tools are designed and studied as embedded in educational contexts which are themselves objects of the research. The main interest concerns the kind of mediation offered by technology to both teaching and learning processes taking into account contributions from various disciplines (mathematics education, pedagogy, cognitive science and information technology). The educational institutions (e.g. schools) are considered both as constraint systems and as entities undergoing a process of change as result of technology integration. The main aims of this research are:

- understanding how mathematical concepts, which are abstract and formal in nature, can be built by students engaged in exploratory activities including manipulation of concrete (visual, motor perceptive) representations.
- understanding the role of the teacher in the classroom and how his/her work changes if an ICT system is introduced in the teaching and learning practice

These issues have been studied in the long term in a number of arithmetic problem solving areas at different kinds of students (different school level, normal students, deaf students, students who are considered as low achievers in mathematics) using various AriLab2 microworlds. With one class, the system was used for almost the whole cycle of primary school. Some information on the experiments performed is briefly reported on the Table 1.

Type of class	*Duration of the experiment*	*Frequency of computer sessions*
Primary school class	6 months (Grade 2)	2 hours/week
Primary school class	4 years (Grade 2 to Grade 5)	2 hours/week
Primary school class (deaf children)	4 months (Grade 3)	2 hours/week
Lower secondary school class (only with low achievers in maths)	4 months (Grade 6)	2 hours/week
Lower secondary school class (only with low achievers in maths)	6 months (Grade 7)	2 hours/week

Table1: Classes involved in experimentation with AriLab2.

How

Constraints affecting the design

AriLab2 has been a long-term high – labour project including technical development as well as educational design, implementation and research.

Underlying theories and principles

The design of educational software is informed by methods and principles of the activity theory and the research conducted by the development team may be seen to be within a socio-constructivist paradigm. Rather than constrain the learning experience to be narrowly individualistic, situated multi-environments learning systems like AriLab2 support socially situated interaction and investigation by providing tools to support not only the relationship between the student and the learning object but also all the relationships that are established among the participants during a teaching/learning activity.

In particular, in the design of such systems, the following issues assumes a crucial importance:

- The computational objects and interactivity that a system makes available to the user and their relationship with the cognitive processes involved in the acquisition of the knowledge for the learning of which the system has been realized.
- The tools offered to validate student’s actions and the support they offer to the evolution of student’s knowledge.
- The tools offered to support the re-elaboration of personal experience and its sharing within the class.

The tools offered to support the setting up of a social context able to assist students’ performance and the evolution of competencies and knowledge. (Bottino & Chiappini, 2003).

In general, situated multi-environment systems offer tools for problem exploration, for representing solution strategies and processes and for communicating such processes as well as tools to support learning evaluation, the management of the teaching activity, and the evaluation of learning. Of course, tools and features cannot by themselves guarantee learning. They have to be used to support the construction of activities in which learning could be the result of a social construction of meaning and of its justification.

Start of the design and number of versions

AriLab2 has been designed and implemented within the research project ITALES (“Innovative Teaching and Learning Environments for Schools”, IST-2000-23356) partially funded by the European Commission within the 5th Framework Program. The development of AriLab2, has been undertaken major transformations from the initial prototype (Bottino et al., 1994), to the first commercially available version (AriLab, 1999), to the new version, which have been completely re-designed. An iterative approach to design has been used; that is, the design of technology has been informed by the study of the integration of initial prototypes in real class situations (Bottino & Chiappini, 2002). As the matter of fact, the process of design used was one in which the whole educational environment was designed and tested. The technology was just one component.

Development status (advanced prototype, beta product, product)

AriLab2 can be considered as a final product.

For whom

Intended Audience:

Designed to support activities in arithmetic problem solving, AriLab2 is intended to be of use to students, who are engaged in solving a problem, and teachers, who are to design and manage educational activities for their classes.

- The student has at his/her disposal tools to develop the solution process (Microworlds), a tool to describe and present the solution product (the Solution Sheet) and a tool allowing communication between users (e.g. exchange messages and problem solutions).
- The teacher can edit texts of problems and impart them to a student (or a group of students, or to the whole class) through the local network connection; construct examples of problems solutions and send them in the same way to the whole class or to a specific student of the class; to store all the actions that the students performed while solving a problem (monitoring).

The various microworlds, being exploratory learning environments, appear to be accessible to a wide range of students. The Fractions microworld used in the cross-experimentation is intended for use with students in upper primary or lower secondary school.

Distribution

License

ARI LAB 2 is a Copyright of the CNR-ITD, **distributed** by Dida*EL Srl (http://www.didael.it).

Price: 61 Euro.

Artefact

Technical aspects

Operating System (or Web)

AriLab2 is realised to be used in a class laboratory equipped with PC working under 32-bits Windows (Windows 98, Me, Windows 2000, Windows XP). It is developed to be used on a local network with TCP-IP protocol. It is also possible to be used in a standalone computer but in this case some functions of the system cannot be exploited (e.g. the communication function, for instance). In the local network, a computer is intended to be used by the teacher and the others by the students.

Programming Language

The development of AriLab2 is currently based on the following technologies:
1. Windows 98, Me, 2000, XP as OSs supported.
2. ToolBook 8.5, Visual Basic, and C++ as development environments.
3. Microsoft Agent for the voice synthesizer.

Translations: Versions are available in Italian, English and Spanish.

Interface

The details provided from now on refer specifically to the Fractions microworld (FM) used in the TELMA cross-experimentation.

Representations

*** what rep.**

When the user accesses the FM the following interface is presented to him/her:

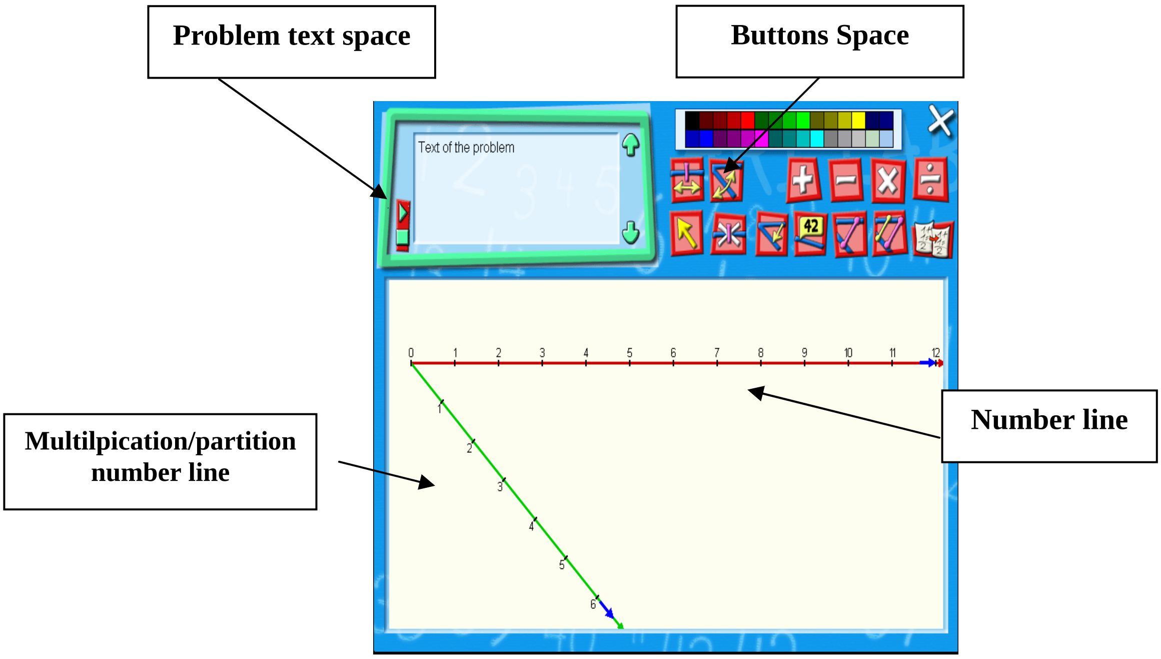
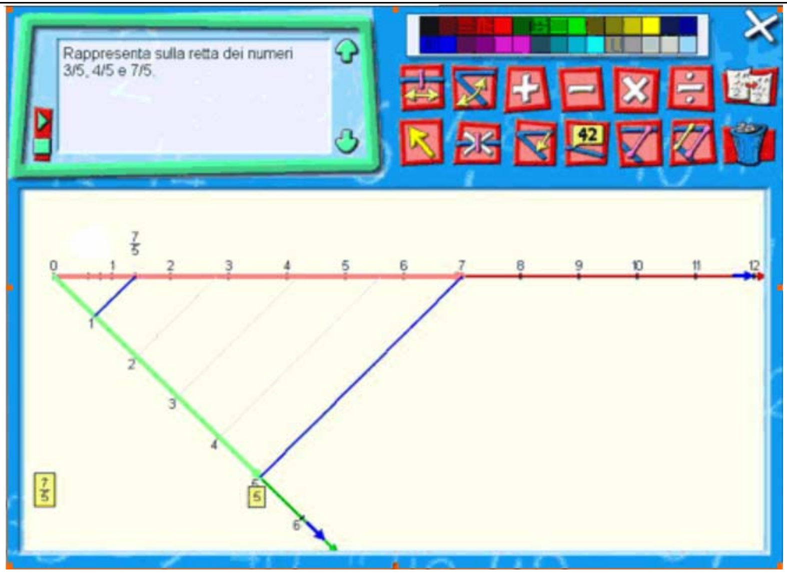


Figure 2: The interface of the FM.

The FM primarily makes use of graphical and symbolic representations of a fraction. The graphical representation consists of two half number lines: one horizontal and one slanted (called multiplication or partition number line). The construction of a fraction is realised as a quotient of a division: the divider and divisor are selected from the horizontal and the slanted line respectively (see Figure 3).



Rappresenta sulla retta dei numeri 3/5, 4/5 e 7/5.

Figure 3: 7/5 represented as a point on the number line.

The symbolic notation of on the horizontal line in a post-it form. The geometric part of the representation concerns the partition of a length selected on the number half line, into the number of parts defined on the partitive half line. After the selection of two numbers from the slanted lines it is also provided instantly a geometrical representation of a fraction based on the Thales Theorem (see Figure 3). (Since this theorem is introduced at the secondary level, the NKUA-ETL team chose to bypass this interpretation in the experimentation and to use it as a 'black-box' for the students).

In the FM the user can also:

- dynamically change the unit of measure on a half-line by clicking on a specific button and then move the unity of measure on one of the number half lines;
- change the angle between the two half lines by first clicking on a specific button and then moving the multiplicative or partitive half line in the new position;
- associate a label (a 'post-it') to a point constructed on the number half line by clicking on a specific button and then moving the cursor on the expression of the post-it he/she wants to use a label. Then he/she has to click on such expression;
- create a geometric sum/subtraction of two lengths selected on the number half line by using the button for the selection of two lengths on the numbers half line. Each selection is realized by dragging the mouse from the point 0 to the point desired. A red arrow is visualized during the drag of the first length and a yellow arrow for the second length. The system determines the length sum and visualizes it as a blue arrow. The result of the sum is marked with a point on the number half line. In the "post it" associated to this point the expression of the sum appears (e.g. "3+5", "1/2+1/3");

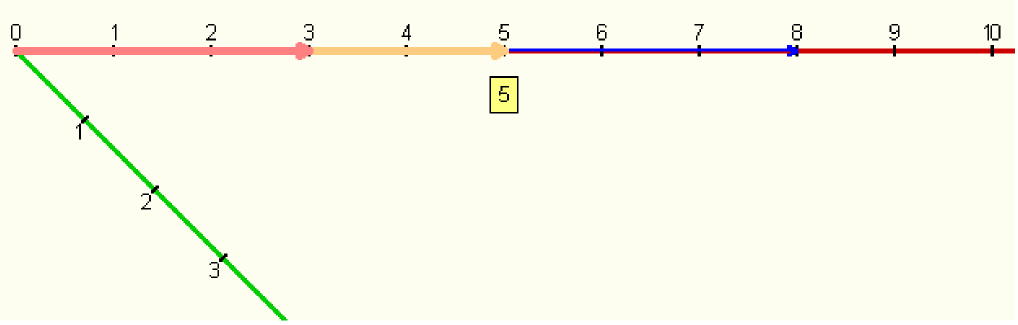


Figure 4: The selection of 3 and 5 in the FM.

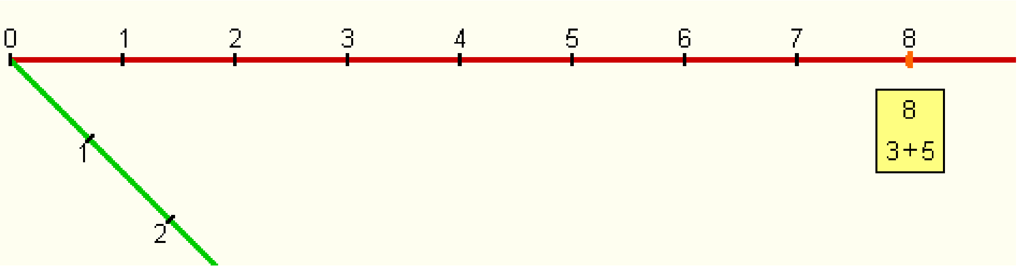


Figure 5: The sum 3+5 in the FM.

- multiply a selected length on the number line by a number of times selected on the multiplicative half-line. During this construction the symbolic notation of the process performed is automatically constructed by the system and inserted in the post-it associated to the point constructed;

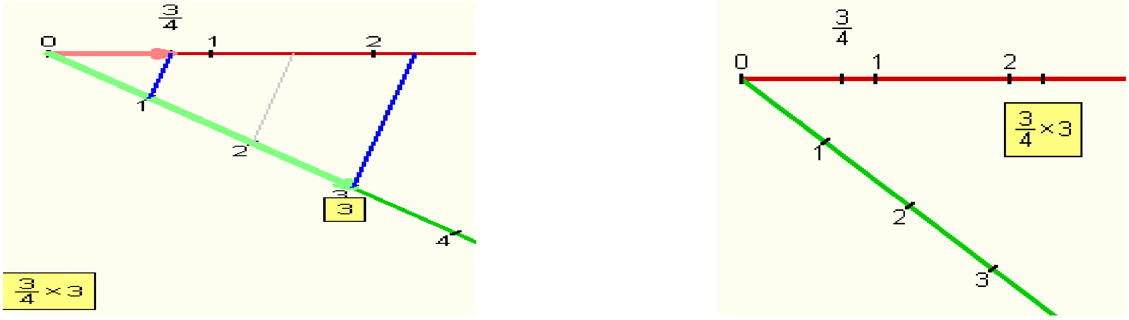


Figure 6: The multiplication ¾ ×3.

- construct a segment that connects a point on the number half line with a point on the multiplicative or partitive half line;
- transform a length selected on the number half-line on the multiplicative or partitive half-line respecting the proportion;
- construct a segment parallel to an already existing segment passing for a given point;
- construct segments that connect a point of the number line with a point of the partitive or multiplicative half line;
- change the angle between the two half lines, by clicking on a specific button and then moving the multiplicative or partitive half line in the new position;
- cancel an object constructed in the fraction space (i.e. point, segment) by clicking on a specific button and then moving the red cursor on the object she/he wants to cancel.

*** nature (usual, new)**

1. The novel character of the above representations of fraction can lead to the identification of a ‘distance’ between the mathematical objects constituting the representation of fractions in the

microworld (tool design) and those found in the traditional curriculum, based primarily on the part-whole scheme. For example, the symbolic representation of fractions with a numerator equal to 1 coincides with the part-whole representation of these fractions (e.g. the position of the fraction 1/3 indicates also the respective part-whole relationship, 1 part of the 3 in which the unit 1 is divided) which does not happen with any other type of fractions as represented in the FM.

2. The numerical representation is more or less conventional and familiar to students, though the graphical representation offers a possibly unfamiliar degree of precision in non-observable partitions of the number line. This may be considered as a characteristic that enhances the abstract nature of the representation of fractions as points among integers. It can be concluded that while the graphical representation is not novel, its use in this context, as a way of representing fractions, may be unusual for students.
3. While number lines are commonly used and thus provide a basis of familiarity for making sense of the representation, the dynamic nature of the construction offers a new way of visualising fractions. The fact that every construction and operation can be controlled by means of a perceptive motor approach enhances the dynamic nature of manipulating the provided representations.

*** quality (poor, medium, rich)**

AriLab2 provides a wide variety of representations with rich potential. Within the FM a sub-set of these is used. Nevertheless, multiple linked representations of the same mathematical object provide a rich representation of fraction. A limitation of the representation of fractional amounts may be related to the fact that the symbolic representation of a calculation in the FM includes only the numbers of the respective operation (e.g. 1/3 + 1/2) and not the resulting equivalent fraction (e.g. 5/6). Moreover, the user is not able to measure a specific part of the number line (e.g. between ½ and 3/4) if it is not starts from zero. So, students need to be familiarized with the idea that any specific segment on the number line is realized by its equivalent transformation starting from zero.

Manipulations

- **on what (objects, rules…)**

  The user can easily control the numeric values concerning the construction of fractions as well as the operations between them by means of a perceptive motor approach. For example, the user can manipulate the size of the numeric unit on a half number line by clicking on a specific button and dragging the arrow on the left of the numeric unit.
- **quality (pleasant, painful)**

1. Manipulation of the available representations is simple and dragging the arrows along the number line is congruent to moving in a kinaesthetic way along the line.

2. Manipulation of geometric parts of the number line is not accessible unless transformed to begin from zero. This can be a cause of frustration and confusion for new users.

Domain

Mathematical content

Constructing fractions as number-measures.

Comparing and ordering fractions.

Making operations with fractions (4 operations: +; *; -; /).

Sort of exercises

No exercises are provided within the AriLab2 environment. Student activity must be defined in interaction between teacher and student. It is usually the teacher who decides the task and assigns it to students.

*** formal exercises**

n/a

*** word problems**

n/a

Types of activities

Activities are not defined by the FM. For the purpose of the NKUA-ETL experimentation, it was decided to focus on exploratory tasks concerning comparison and ordering of fractions as well as operations with fractions based on integrating the available representations of fractions on the number line with persons covering certain distances in an everyday context. Two strands of activities were designed, including questions with one answer (e.g. ordering specific fractions), open questions (e.g. finding three fractions between 3/4 and 2/5) and requests for exploration (e.g. comparing distances from specific points on the number line).

Lecture

n/a

Examples solved by the ILE

n/a

Exercises to be solved by the student

n/a

Feedback

Immediate/delayed

Immediate.

Nature

FM represents fractions in specific ways and does not signal some kinds of mistakes by means of visual feedback. This characteristic of the tool gives space for pupil's interpretations of the given feedback in specially designed tasks. For example, having made a prediction that a given fraction is going to lie between two others, the student can construct the respective fraction and observe the result on the number line in order to confirm or refute their answer. Or, a student who thinks that 2/3+1/3 is equivalent to 3/6 can construct these expressions in the microworld and realise that they are not equivalent since they are not contained in the same post-it. This way the visual feedback can be considered as a metaphor of the class of equivalent of fractions.

Diagnosis

Interactivity

Microworld

There is a high degree of interactivity as all events within the Fractions microworld are direct results of student actions.

Hypermedia: n/a

Answer

Multiple choice question: n/a

Final open answer : n/a

Step by step answer: n/a

No answer

Autonomy

Degree of student's autonomy

Potentially very high, though in practice student autonomy may be circumscribed by the types of tasks posed by the teacher.

Self-evaluation

The nature of the feedback provided by manipulation within the microworld allows students to judge whether their answers are correct by evaluating the display of the number line.

Knowledge of the task

The ILE knows what is the student's task: No.

The ILE can provide hint: n/a

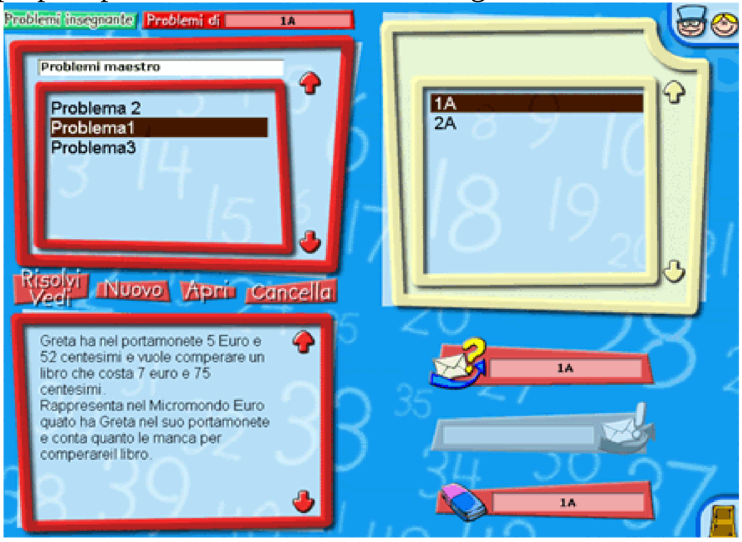
The ILE can provide the solution: n/a

The ILE can solve the exercise step by step: n/a

Teacher’s role

Parameterization

AriLab2 offers the teacher an environment, the Teacher’s Environment (Figure 7), where s/he can write texts of problems, prepare problem solutions, and manages the local network.



Greta ha nel portamonete 5 Euro e 52 centesimi e vuole comperare un libro che costa 7 euro e 75 centesimi.
Rappresenta nel Micromondo Euro quato ha Greta nel suo portamonete e conta quanto le manca per comperareil libro.

Figure 7: The Teachers’ environment.

A number of configurability and personalisation opportunities are offered. In the high left zone of the interface there is the list of the names of the problem texts prepared by the teacher. It is possible to select a problem from the list, and to access it by selecting the button “Solve/View”. This button brings to “Solution Sheet” environment. In this environment it is possible to build the solution to the current problem. In the teacher’s environment it is also possible for the teacher to write the text of a new problem (Icon “New”).

In the bottom left zone of the interface, the text of the current selected problem is visualized. In the upper right zone of the interface, the list of the teacher’s classes is shown. The teacher can visualize the names of all the students of a class just by selecting the name of the class. It is possible for the teacher to impart different problems to different students, and send messages and solutions both to groups of students, to the whole class, or to each individual student (icons bottom right side of the interface). This way a teacher can manage the complexity of the problems given to different students by setting up different kinds of activities. For example, the teacher can begin the introduction to fractions with exercises concerning fractions between 0 and 1 (the representation of which is based on the part-whole scheme) and then continue with fractions bigger than 1. The teacher can also control the work performed by a student by looking at the problems s/he has solved.

Production of scenarios

The microworld itself does not provide scenarios. These must be defined by the teacher in direct interaction with students.

Production of exercises

The microworld itself does not provide exercises. These must be defined by the teacher in direct interaction with students.

Validation, usage, experimentation

Experimentation

Usage analysis

See references in http://www.itd.cnr.it/arilab/_english/biblio_eng.html.

Qualitative results

See, Bottino and Chiappini (2003).

Quantitative results

Usage

Familiarization with the ILE

AriLab2 is a friendly and easy-to-use piece of software.

Evolution

The algebraic Line within the dynamic digital artefact ALNUSET which is being developed in the context of the **ReMath project** (http://remath.cti.gr).

2.4. Analysis of E-Slate

By UNILON team

Design and development

Name of the ILE: E-Slate - Fractions microworld

Date of the study: November 2006

Who

People/organisation involved in the design
Computer Technology Institute, Athens
National & Kapodistrian University of Athens, Educational Technology Lab

Interaction between these participants
Two large teams collaborating longitudinally through a small number of hybrid actors (see diSessa, 2004).

Why

Aims of the design

E-Slate may be thought of as a "construction kit for educational software" (Kynigos, 2003). According to its website http://e-slate.cti.gr/ it provides "*a revolutionary way for creating highly interactive educational software in a rapid prototyping manner with minimal effort and resources*". It provides generic component pieces of software and a set of pre-fabricated connectors ("plugs") for linking components. A Logo-based scripting language can be used to programme the behaviour of the components and the kind of connections between them.

The "construction kit" is intended to provide developers and authors with tools for producing educational software. However, the educational contingent of the development team has also produced microworlds within E-Slate that may be used by teachers and students as exploratory learning environments. As well as fully developed microworlds, the team has what it terms "half-baked" technological artefacts, developed specifically for use either by students or by teachers in professional development courses, with the intention that the teachers should change and customise them in order to investigate mathematics themselves or to build microworlds for students to use.

The Fractions microworld used in the cross experimentation is based on one such half-baked microworld, "Sliders", with a script to define functional relationships between the values of sliders. For the purposes of the cross-experimentation, this was specialised to multiplicative relationships with the aim of allowing students to explore the relative sizes of fractions.

Research questions

The general research area is the study of learning, teaching, contextual and organizational processes during the infusion of innovative educational environments in everyday practice. The main interest concerns the kind of cognitive, social and systemic turbulence created as a result of questioning and change in everyday practice. The NKUA/ETL team investigates ways in which this turbulence might be channelled so that **students** generate mathematical meanings, **teachers** develop strategies for creating environments where this is likely to happen, and **organizations** accept, recognize and value such change.

The main aims of this research are:

- understanding how the generation of mathematical meanings for students occurs and how this may be influenced by the classroom norms,
- understanding the role of the teacher in the classroom and how his/her work changes if an ICT system is introduced in the teaching and learning practice

These issues have been studied in a number of areas of mathematics, using various microworlds constructed using E-Slate.

How

Constraints affecting the design

E-slate has been a long term high-labour project (around 50 person-years of technical development, equivalent effort in educational design, implementation and research). It has been difficult to sustain over 8 years and has been the result of a balance between the R&D agendas of two very different organisations. It has been perceived as a vehicle for infusing innovation in an educational system, which by itself is a highly problematic venture.

Underlying theories and principles

The design and research conducted by the development team may be seen to be within a constructionist/socio-constructivist paradigm. The design of educational software is informed by methods and principles of the constructionist community:

- the software is meant to facilitate innovation and therefore some change in activity, attitude, perceptions and understandings in both teachers and pupils;
- the same piece of software may be used by different people for different activities at various levels of sophistication;
- the software is designed to be used primarily for knowledge generating activity with some personal meaning and not for following directions, gathering or observing information, answering questions or simply observing things which are going on;
- in many cases, users will construct things with the software (not just observe or test out);
- what users do with the software may well be a surprise to the original designers;
- software development needs to be integrated with use in contexts that are as realistic as possible;
- there is a tendency for more emphasis on the context within which the software is used and on the activities rather than on the actual development of the software itself;
- there is a lot of effort spent on building an understanding between actors with different expertise, perspectives and stakes in the development process;
- the roles of developing, testing authoring and using are purposefully not clearly defined.

  (Kynigos, 2002)

In studying students' generation of meaning in interaction with a technological environment. the team has made use of the notion of "situated abstraction" (Noss & Hoyles, 1996). See, for example, (Yiannoutsou & Kynigos, 2004) for an example of analysis of the evolution of meanings about proportionate relationships while students engage in map construction in a cartographic microworld.

The context of the educational environment within which teachers and students work with technological tools is considered to be both crucial to the analysis of meaning generation and an object worthy of study in its own right. The role of the teacher in interaction with students is understood referring to the notion of "classroom norms" (Cobb & Yackel, 1996) while teachers are themselves perceived as professionals with an active role to play in the design and support of educational activities (Kynigos & Argyris, 2004)

Start of the design and number of versions
The E-Slate software and microworlds have been developed since 1993. The current version is 1.3

Development status (advanced prototype, beta product, product)
It is best described as a beta product. By 'product' in this case we mean a product coming from a research team rather than a commercial product.

For whom

Intended Audience
As a construction kit for educational software, E-Slate is intended to be of use to: software developers, who can create new components while benefiting from the functionality provided by existing components; authors or publishers, who can make use of existing components, templates and microworlds to develop educational software; educators, who can personalise or even adapt software constructed using E-Slate (http://e-slate.cti.gr/); students who can engage in exploratory, constructionist or expressive activities using, constructing or de-constructing e-slate microworlds.
The various microworlds, being exploratory learning environments, appear to be accessible to a wide range of students.
The Fractions microworld, developed for use in the cross-experimentation, is intended for use with students in upper primary or lower secondary school.

Distribution

License
E-Slate and E-Slate components are © of the Computer Technology Institute 1993-2000. They may be used for any non-commercial purpose and/or redistributed under the same terms. The software is distributed on an "as is" basis without a warranty of any kind.

Price: n/a

Artefact

Technical aspects

Operating System (or Web)
Windows (release version 1.3 has been tested on Windows98/2000/ME/XP)
Suggested configuration: for best results, use a Pentium II at 366 Mhz or more with 64MB of RAM. Although the screen resolution can be of any size, to view the accompanying Microworlds properly, you will need a 1024x768 screen.

Programming Language
Component connectivity and some features of component behaviours can be programmed with a Logo scripting language based on UCB Berkeley Logo. E-Slate is currently based on the Java platform (Java 1.3) and related technologies.

Translations

Versions are available in Greek and in English. The English version has recently acquired full documentation on the ETL's site.

Interface

The details provided from now on refer specifically to the Fractions microworld used in the TELMA cross-experimentation. This is only one example of an educational environment developed using the toolkit provided by E-Slate.

Representations

*** what rep.**

The Fractions microworld primarily makes use of three types of representation: "sliders"; symbols in the form of Logo programming; and Cartesian graphs. Specifically, a fraction may be represented by:

1. a relationship between values shown by positions on two linked number lines (sliders);
2. a rational number entered into a Logo procedure in either decimal (e.g. 0.25) or ratio (e.g. 1/4) form;
3. the gradient of a linear graph.

The sliders are displayed on the main screen (Figure 1). One of these is the 'control' slider and there may be one, two or three other sliders whose displayed values are dependent on the value shown on the control. (In theory there could be a larger number of sliders but the Fractions microworld was constructed to allow up to three dependent sliders.)

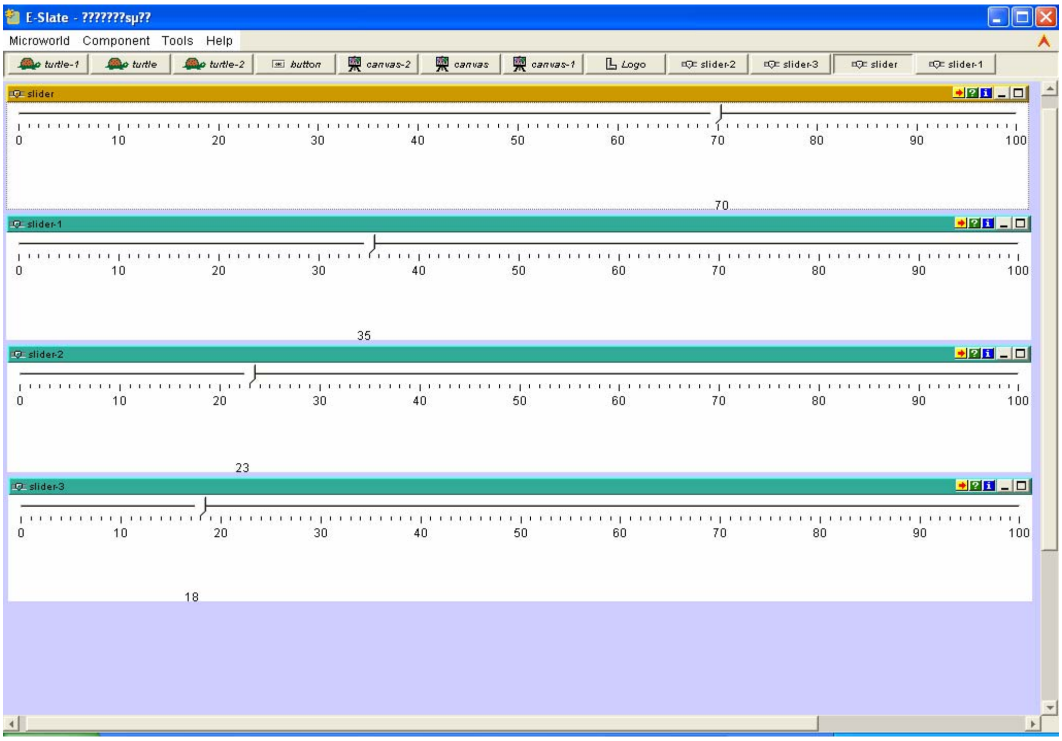
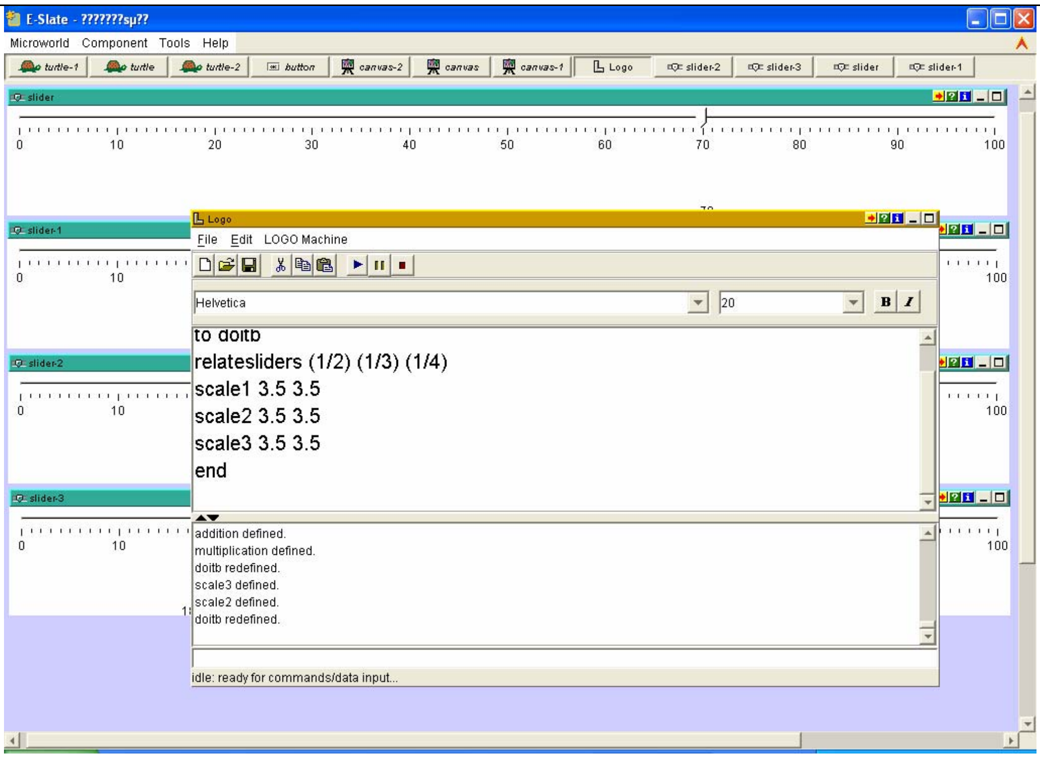


Figure 1: Fractions microworld main screen

The set of Logo procedures is displayed in a window that may be hidden when not needed (Figure 2). In the Fractions microworld the window has been formatted to show only the procedure that students are asked to edit by inserting their chosen fractions.



```
to doitb
relatesliders (1/2) (1/3) (1/4)
scale1 3.5 3.5
scale2 3.5 3.5
scale3 3.5 3.5
end
```

```
addition defined.
multiplication defined.
doitb redefined.
scale3 defined.
scale2 defined.
doitb redefined.
```

```
idle: ready for commands/data input...
```

Figure 2: Logo window and editable fraction procedure

The graphs are displayed in individual windows that may be hidden when not needed (Figure 3). The graphs show only axes and plotted points with no labelling. (In the experimentation, both teachers chose not to introduce their students to the graphs.)

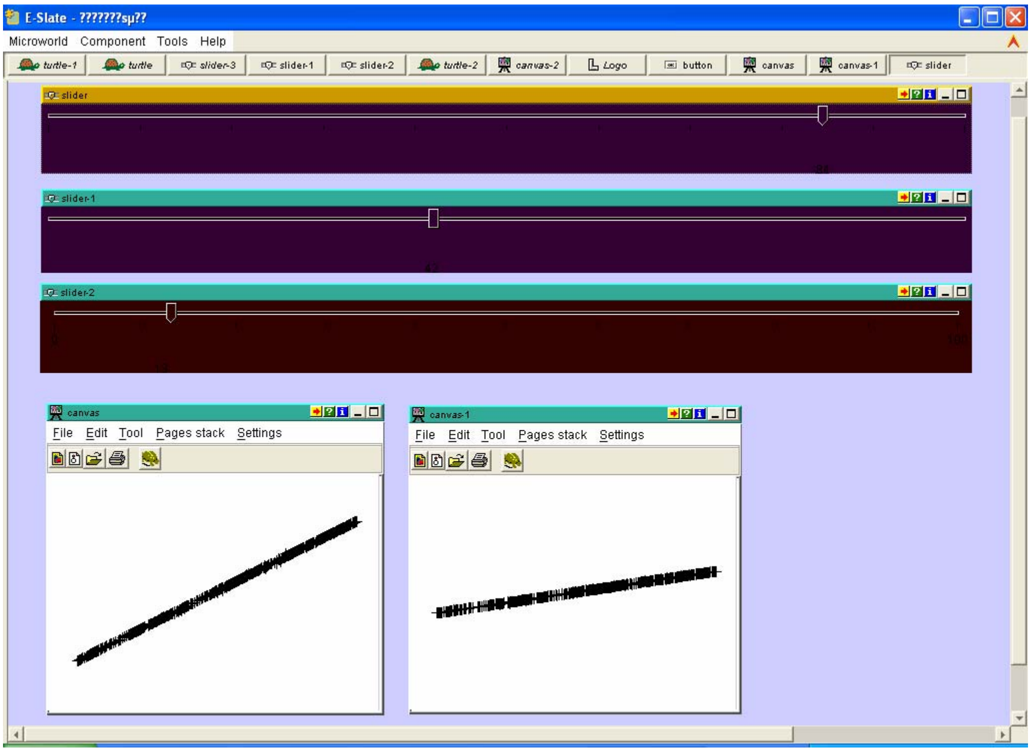


Figure 3: Sliders and graphs

These forms of representation are linked as follows:

a the numbers entered in the Logo procedure determine the relationships between the values displayed on the sliders

b changing the position of the pointer on the 'control' slider causes a dynamic generation of points on the corresponding graph

In addition to these, the control panels provide by the E-slate environment allow changes to be made to the appearance and parameters of components. Those aspects employed during the experimentation with the Fractions microworld included: numerical inputs to determine the scales of the sliders; colour to determine the colour of the sliders (choosing black has the effect of obscuring the numbers on the slider, thus changing the nature of the representation of fractions on the sliders (compare Figure 4 with Figure 1).

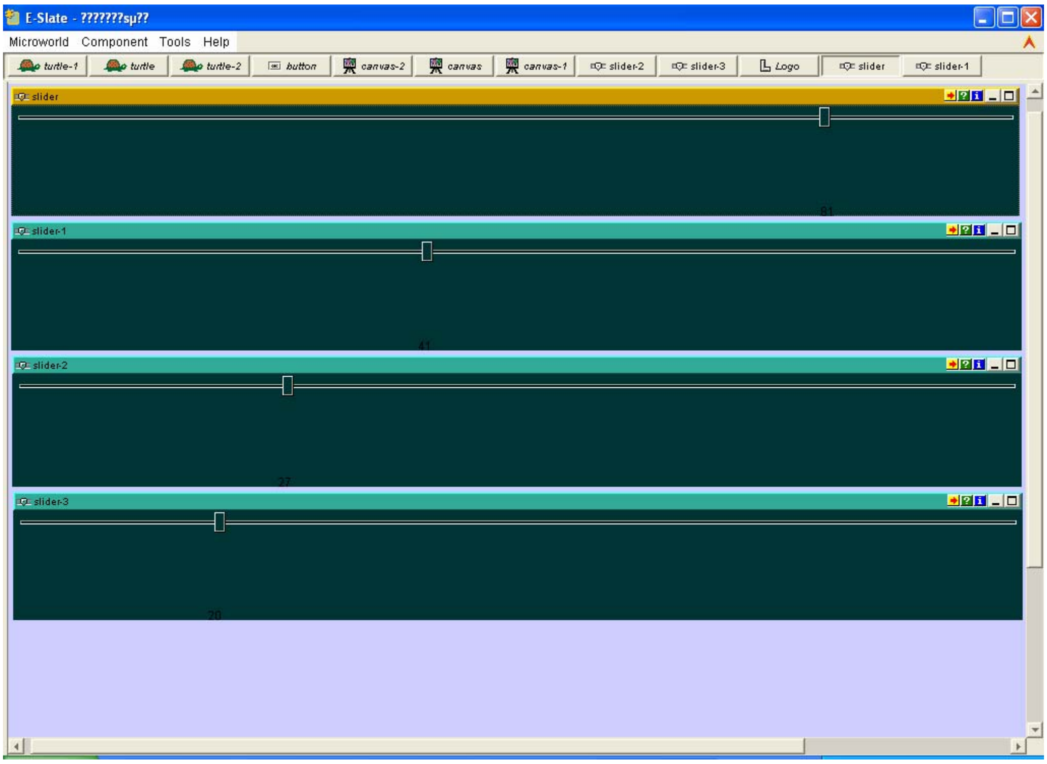


Figure 4: Sliders with numbers semi-obscured

*** nature (usual, new)**

1. The slider representation is novel. While number lines are commonly used and thus provide a basis of familiarity for making sense of the representation, the dynamic nature of the sliders offers a new way of visualising fractions.
2. The numerical representation is more or less conventional and familiar to students, though the Logo programming environment demands a possibly unfamiliar degree of precision in entering the notation. In order to facilitate this precision, one of the teachers involved in the experimentation inserted parentheses into the line of code to be edited by students around the positions where they were expected to enter numbers.
3. While the graphical representation is not novel, its use in this context, as a way of representing fractions, is unusual.

*** quality (poor, medium, rich)**

The E-Slate environment offers a wide variety of representations with rich potential. Within the Fractions microworld a very limited sub-set of these is used in a restricted way. Nevertheless, multiple linked forms of representation of the same mathematical object provide a rich representation of fraction.

A limitation of the slider representation of fractional amounts is that the pointers on the sliders will only indicate integer values. By setting the colour of the sliders to black it is possible to (almost) obscure the numbers and focus attention only on the qualitative nature of the relationship between sliders. Alternatively, students need to be familiarised with the idea that the dependent sliders display only approximate values, rounded to the nearest integer.

Manipulations

*** on what (objects, rules…)**

1. Manipulation of the numerical values in the Logo procedure. The user opens the Logo editor by clicking a button, enters a new value or values into one line of the relevant procedure and presses Insert. This has the effect of resetting the relationship(s) between the values displayed on the control slider and on the other slider(s).
2. Manipulation of the sliders. The user clicks and drags the pointer of the top-most ‘control’ slider. While changing the value displayed on this slider, this also changes the position(s) of the pointer(s) on the other, dependent, slider(s), maintaining the ratio between the values. It is also possible to move the pointer on the control slider by simply clicking on the desired new position. The positions of the pointers on all the sliders then move discretely rather than continuously.

*** quality (pleasant, painful)**

1. The number of steps involved in entering the numerical values means that this is not a simple manipulation. It is easy to make the mistake of closing the Logo editor window before pressing Insert, thus failing to affect any of the changes made to the procedure. This can be a cause of frustration and confusion for new users.
2. Manipulation of the control slider is simple and dragging the pointer along the number line is congruent to moving a finger along the line. The resulting movement of the pointer(s) on the dependent slider(s) sometimes does not keep pace with the movement of the control slider and can appear to be a set of discrete jumps rather than a smooth drag.

Domain

Mathematical content

Fractions of a quantity

Ordering rational numbers

During the experimentation, the rational numbers considered were restricted to those between 0 and 1, though it would be possible to adapt the Fractions microworld to make it possible to consider a wider range of numbers by changing the scales of the sliders.

Sort of exercises

No exercises are provided within the E-Slate environment. Student activity must be defined in direct interaction between teacher and student.

*** formal exercises:** n/a

*** word problems:** n/a

Types of activities

Activities are not defined by the Fraction microworld. For the purpose of the experimentation, it was decided to focus on the comparison and ordering of rational numbers between 0 and 1. Each of the two teachers produced paper worksheets with tasks for students, including closed questions (e.g. Which is larger: 5/16 or 7/19?), open questions (e.g. Find three fractions between 3/8 and 1/2.) and requests for explanation (e.g. Explain how you know that … is between … and …).

Lecture: n/a

Examples solved by the ILE: n/a

Exercises to be solved by the student: n/a

Feedback

Immediate/delayed: immediate

Nature

Feedback is provided to students through the manipulations they perform. For example, having made a prediction that a given fraction is going to lie between two others, the student can enter the three rational numbers into the Logo procedure and observe the behaviour of the sliders in order to confirm or refute their answer.

Diagnosis: n/a

Interactivity

Microworld

There is a high degree of interactivity as all events within the Fractions microworld are direct results of student actions.

Hypermedia: n/a

Answer

Multiple choice question: n/a

Final open answer : n/a

Step by step answer: n/a

No answer: n/a

Autonomy

Degree of student's autonomy

Potentially very high, though in practice student autonomy may be circumscribed by the types of tasks posed by the teacher.

Self-evaluation

The nature of the feedback provided by manipulation within the microworld allows students to judge whether their answers are correct by evaluating the display of the sliders.

Knowledge of the task

The ILE knows what is the student's task: No.

The ILE can provide hint: n/a

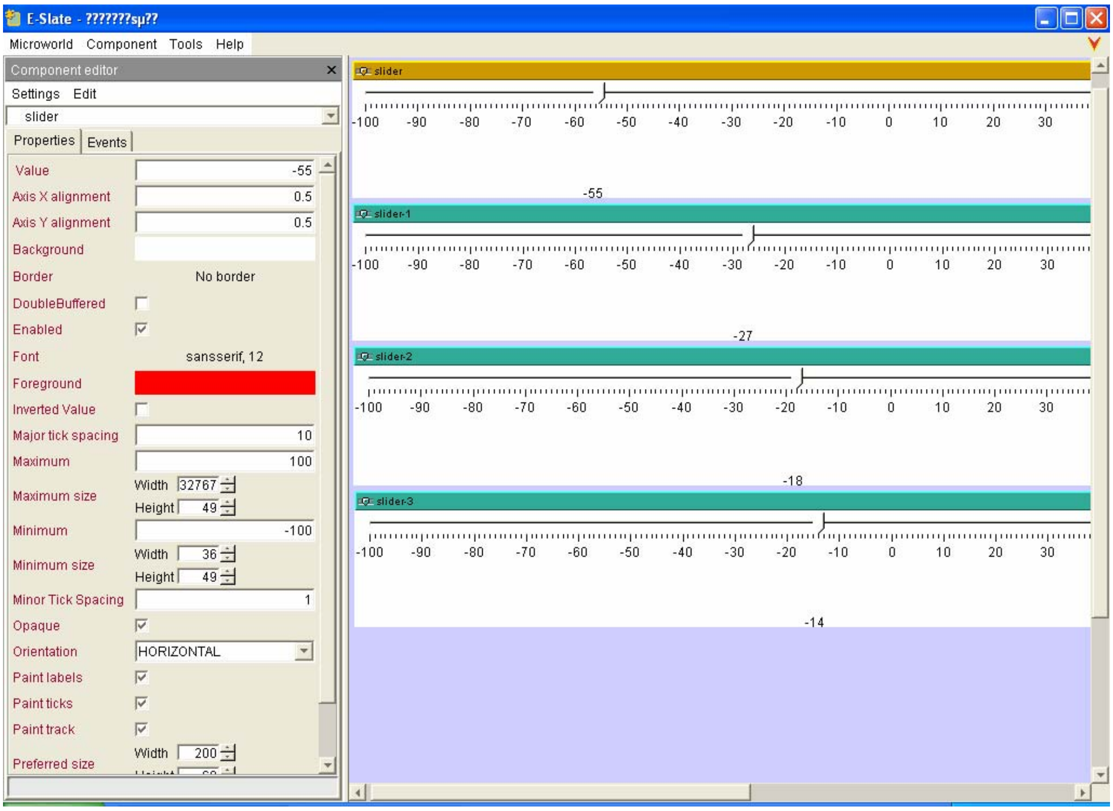
The ILE can provide the solution: n/a

The ILE can solve the exercise step by step: n/a

Teacher's role

Parameterization

By using the 'component editor' palette, the teacher can change the settings of the sliders to alter: the range of numbers considered (e.g. to include negative values); the labelling; the colours of the sliders. These changes have potential to alter the aspects of the fraction concept encountered by students.



Value	-55
Axis X alignment	0.5
Axis Y alignment	0.5
Background	
Border	No border
DoubleBuffered	□
Enabled	☑
Font	sansserif, 12
Foreground	🟥
Inverted Value	□
Major tick spacing	10
Maximum	100
Maximum size	Width 32767 Height 49
Minimum	-100
Minimum size	Width 36 Height 49
Minor Tick Spacing	1
Opaque	☑
Orientation	HORIZONTAL
Paint labels	☑
Paint ticks	☑
Paint track	☑
Preferred size	Width 200

It is also possible for teachers to interact in a more advanced way with the microworld, e.g. to add further components or to alter the Logo procedures to set up different kinds of relationships between the sliders. This could have the effect of changing the mathematical domain addressed within the microworld, i.e. by defining exponential rather than multiplicative relationships between sliders.

Production of scenarios

The microworld itself does not provide scenarios. These must be defined by the teacher in direct interaction with students.

Production of exercises

The microworld itself does not provide exercises. These must be defined by the teacher in direct interaction with students.

Validation, usage, experimentation

Experimentation

Usage analysis

The use of the Fractions microworld within the TELMA cross-experimentation is described in (Morgan, 2007). The NKUA/ETL team have developed and studied the use of a number of exploratory learning environments developed in E-Slate (e.g. Psycharis & Kynigos, 2004; Yiannoutsou & Kynigos, 2004).

Qualitative results

The study undertaken as part of the TELMA cross-experimentation focused on teachers' and students' use of various representations of fraction provided by the microworld and by the usual paper and pencil environment. Two teachers worked together to plan a sequence of lessons but implemented these plans with rather different forms of pedagogy. The pedagogic differences seemed to be associated with different kinds of use made by students in their classes of the available representations. In particular, the way each teacher introduced the software to the students and their forms of interaction appeared to influence the ways in which students used the sliders and talked about comparisons between fractions.

In one class the teacher made explicit links with previous fraction work and illustrated use of the microworld by setting up and discussing fractions of given quantities. Students in this class focused on static configurations of the sliders and made substantial use of computational methods. The other teacher adopted a more open approach to introducing the tasks and illustrated use of the microworld by demonstrating the dynamic variation of sets of sliders. When students in this class discussed comparisons between fractions they focused on visual and dynamic relationships between the sliders rather and did not use much computation.

Quantitative results

n/a

Usage

Familiarization with the ILE

As indicated above, the initial introduction to the meaning and functionality of the microworld has an important influence on students' subsequent modes of use.

Evolution

3. Critical analysis of the grid

Recall that the methodological tool for the selected ILEs analysis has been designed based on existing grids and evaluation questionnaires proposed by Tchounikine (2004), Nicaud (2001)[2] and (Nicaud & al. 2006). The first grid aims at analysing research projects in the domain of the design of educational software and proposed 4 dimensions of the analysis: (A) the definition of the research, (B) theoretical framework of the research, (C) results of the research, and (D) life cycle of the research. Nicaud proposes a questionnaire to analyse features of ILEs for algebraic reasoning, addressing the following points of view: (1) domain/capacity, (2) inference and interaction, (3) adaptability, (4) heuristics, and (5) tutoring.

The grid proposed by MeTAH team is organized around three periods of life of an ILE: (1) design and development, (2) artefact, and (3) validation.

The design and development part aims at highlighting the objectives of the authors (who and why), the underlying principles and theoretical background (how) and the intended audience (to whom).

The artefact part addresses technical aspects of the ILE, issues related to the interface (representation and manipulation of mathematical objects), the domain covered by the ILE, the interactivity (types and nature of feedback, student's autonomy, ILE's knowledge of the task), and the role of a teacher.

The validation part aims at gathering information about empirical validation of the ILE from published papers and research reports, which seems particularly important in this study, which is based exclusively on an analysis by inspection.

This methodological tool for ILE analysis allows addressing the issues of utility, usability and acceptability of an ILE (Tricot & al. 2003) commonly used in the AI-ED community for evaluating educational software. However, since the pieces of software selected for the inspection differed from each other in many points of view, it seemed necessary to have a critical insight on the grid itself and its applicability to these different systems. In what follows, we report about the work carried out with the aim to evaluate and refine the methodological tool for ILE analysis.

3.1. Questionnaire

In order to have a critical feedback as regards the methodological tool itself, MeTAH elaborated a questionnaire aiming at highlighting questions or items in the grid that were unclear, irrelevant with respect to the software under study, inappropriate to the kind of study carried out, useless regarding the aims of the study, or missing. The following questionnaire was send to all partners who were asked to provide answers based on their experience with the use of the grid to analyse one or more pieces of software:

[2] See Appendix.

Methodological tool designed for the ILEs analysis
QUESTIONNAIRE

In the year 3 of TELMA, we have studied 6 selected ILEs for arithmetic and algebra through the use of a grid specifically designed for this purpose.

This questionnaire aims at gathering feedback on the grid, its clarity, convenience and applicability.

Please, answer the following questions. Refer to the use of the analysis grid in studying the ILE(s) you were in charge of. Try to be as much explicit as possible.

Question 1: Clarity of the questions/items in the grid:

What questions/items were not clear enough? How would you reformulate them to make them clearer?

Qustion 2: Applicability of the questions/items in the grid:

Were there any questions/items that you considered not relevant with respect to the ILE under study? Explain why.

Question 3: Convenience of the questions/items in the grid:

Were there any questions/items that you considered of little interest in the ILE study?

Question 4: Missing questions/items:

Are there any aspects of the ILE you have studied that you consider important and that were not addressed by the questions/items in the grid?

Question 5: Other remarks, suggestions related to the grid?

3.2. Answers to the questionnaires

3.2.1. DIDIREM team

By Brigitte Grugeon

Question 1. Clarity of the questions/items in the grid:

Some questions /items were not clear enough.

- The indicators proposed for the item “Sort of exercises”**:** formal exercises, word problems, are in my opinion not clear enough and tend to give a limited vision of algebra, and of the diversity of possible tasks in algebra for which ILEs can be expected to support the learning process: for instance exercises supporting the introduction of new notions (letters, literal expressions, equations, ..), exercises focusing on algebraic modelling which cannot be limited to word problems, exercises focusing on the interaction between the symbolic algebraic register and other semiotic systems… It would be necessary to refine the categories introduced here.
- The indicator “Lecture” proposed for the item “Types of activities is in my opinion not clear enough: what does it cover precisely?
- The indicators proposed for the item “Experimentation”: qualitative results, quantitative results, seem very general. What is expected in the answers could be clarified (see 4).
- The indicator “Familiarization with the Ile” proposed for the item Usage is also not clear: all the more as it appears in the item evaluation and is the only one in this item.

Question 4. Missing questions / items

- The main problem I had was that I did not find how to express with the grid deals my feeling regarding the “quality” of the mathematical content and tasks proposed by the ILE. The grid does not have a place for it, it does not propose criteria for evaluating it. Thus, it is not possible to analyse and qualify the didactical potentialities of these tasks. In my opinion, it would be necessary to add another item after **Types of activities** to introduce such an analysis. We could call it **Mathematical and didactical potential of activities.**
- Some indicators could be added to the item “usage analysis” especially for making clear if usage analysis have been carried out only in laboratory conditions or have included classroom experimentations. We all know that classroom experimentations are absolutely necessary for evaluating what can be offered by an ILE to the learning of algebra, and how it can be integrated by a given educational system. Moreover the grid could make more precise the kind of information required about these experimentations (size, duration, context, organization…)

3.2.2. ETL team

Question 1. Clarity of the questions/items in the grid:

1. Some questions were a bit general. However, the description was necessary to be detailed. For example, the question '* what rep.' in the *Representations* part. My choice here was to give a general picture of the interface and then to describe with more details specific (crucial in my view) functionalities. The question should be broken in smaller (more focused) questions. The objects that can be represented in a piece of software are numerous. For example think of the same question for describing software with many different representations (including geometrical figures, tables, coordinate systems, calculators). The question can be reformulated by asking for a general description of what is represented in the main part and then to give selected (or all) of the types of possible representations included.
2. In the part 'Artifact – Type of activities' can be added one more category called 'scenario' (or something like that) which is more general and gives the opportunity to the author to describe an activity as a summary of a scenario. Some teams (and software) do not include in their tasks ('closed') exercises to be solved.

Question 2. Applicability of the questions/items in the grid:

The only problem that I faced was the following: when preparing the grid for the Fractions Microworld (AriLab2) -which is a microworld embedded in the more general 'platform' AriLab2- I had to choose when I was referring to the first and when I was referring to the second. My choice was to devote the 'Design and Development' part (plus the Technical aspects from the Artifact) as well as the last part of it (Validation, usage experimentation) to the platform AriLab2 and to state that all of the other parts were devoted to the description of the 'Fractions microworld'. A good idea is to reformulate the grid with one question at the beginning of it asking the user to declare the type of software that will be presented and indicate when the presentation will be focused on a part of it. This problem arises when one needs to describe a specific part (lets say 'microworld') of a more general software structure (lets say 'platform'). Some authors might be interested to focus only on one of those types when preparing a specific grid. As I remember the same problem was also mentioned by Candia Morgan when was preparing the grid for the microworld 'Number lines' of the platform E-slate.

Question 3. Convenience of the questions/items in the grid:

No questions/items considered of little interest in the ILE study.

Question 4. Missing questions/items:

Nothing.

3.2.3. ITD team

By Michele Cerulli

Question 1. Clarity of the questions/items in the grid:

Questions/items were all clear for me.

Question 2. Applicability of the questions/items in the grid:

Maybe the ones mentioned below in 3 and the question/item concerning research questions as the ILE didn't seem to be oriented toward research, but I am not sure about this.

Question 3. Convenience of the questions/items in the grid:

Yes, the part "Validation, usage, experimentation" wasn't relevant from my point of view; it appeared too much technical, and a bit far from the perspective of a researcher in maths education who is not too much interested in the production cycle of software. The same can could be said on "technical aspects" but in such case, it appears that the questions are relevant at least for assessing the usability of the tool in different contexts.

Question 4. Missing questions/items:

Perhaps something about the possible modalities of employment (see Didactical Functionalities) of the tool. They are only marginally considered in the "types of activities" part, but it would be interested to address the question/item of how the designers/distributors of the software think it should be employed to achieve educational goals. This would also bring this study closer to the rest of the activities developed in TELMA.

Question 5. Other remarks, suggestions related to the grid

I don't have other particular remarks

3.2.4. Siena team

By Mirko Maracci

General comments:

1. in order to "evaluate" the grid, the aims of the study should be better specified
2. It is not always clear whether we are just collecting information provided by ILE designers (structuring this information according to a specific frame) or we are developing a personal analysis of the ILE. E.g.: the item "Artefact-> Interface -> Representations -> What representations" can be understood in at least 2 different ways:
   a. What (mathematical) objects are represented according to our own perspective
   b. What (mathematical) objects are represented according to the designers' perspective.
3. The sub-sub-…-sub-items appear different in nature. Some of them seem to be different alternatives, one has to choose/check the more appropriate; e.g. the sub-items of "Artefact-> Knowledge of the task"(in these cases are they mutually exclusive? Do they cover the range of all possibilities?). Some other seem to direct the attention to specific aspects (in order to frame the analysis or the

communication), on which one has to report or elaborate; e.g. the sub-items of “Artefact-> Interface”. Can such differences made explcit? Is it suitable/desirable?

Question 1. Clarity

“Artefact-> Interface -> Representations ->…

What representation”. What kind of representations should one focus on? Representations of mathematical objects? Of didactical interactions? Of whatever?

Nature”. What does “nature” mean? Does it refer to something like the “distance” between the representations available on the ILE and “institutional/cultural” representations of the same objects? If it is the case I’d speak of “distance” of something like that.

Quality”. What does “quality of representation” mean? What are the criteria of evaluation?

“Artefact-> Interface -> Manipulations -> Quality” same comment as before.

“Artefact-> Feedback ->” the analysis of feedbacks should be explicitly related to the user’s actions. What it meant by “nature of a feedback”?

“Artefact-> Interactivity ->… ” the meanings of “hypermedia” and “microworld” should be specified: in particular, are they two mutually exclusive alternatives? Do they cover the whole range of possibilities?

“Artefact-> …” Why are “Sort of exercises”, “Answer” and “Types of activities” separate items?

Question 3. (Low) Interest

I do not consider really interesting the items *“Design and Development-> Who -> Interaction between participants”* and *“Design and Development-> How -> Constraints affecting the design”*. Maybe I am missing the point here and of course the interest in these (or even other) items also depends on the aims of the present study.

Question 5. Conclusive remarks, suggestions…

Besides specific comments on the different items, I wonder whether a global re-organization of the items in terms of Didactical Functionalities (as envisaged by the ILE designers) is suitable/desirable. Example of a possible re-structuring:

1. **Identity**: who, why, intended audience (?), distribution, developing status
2. **Educational aims**, also including the intended audience (?)
3. **Artefact**: technical aspects, interface, actions/feedbacks, interactivity
4. Envisaged (according to the designers) **modes of use**:
   a. (Envisaged or in case included) type of tasks (not necessarily math tasks):
      i. Description of the task, expected answer of the user…
   b. ILE knowledge of the tasks
   c. teachers’ role
5. **Rationale**: linking Ed.Aims, features of the Artefact and modes of use: it includes “underlying theories and principles” (distinction between design theories and educational ones)
6. Validation, experimentation…

3.2.5. UNILON team

By Jehad Alshwaikh

Question 1. Clarity of the questions/items in the grid:

In Artefact section

Interface:

Representations

* what representations

* nature (usual, new)

* quality (poor, medium, rich)

Manipulations

* on what (objects, rules…)

* quality (pleasant, painful)

Interactivity: Microworld, Hypermedia

In Teacher's role:

Parameterization

Production of scenarios

Production of exercises

In Validation, usage, experimentation

Usage: Familiarization with the ILE

Question 2. Applicability of the questions/items in the grid:

Distribution: License, Price

Question 3. Convenience of the questions/items in the grid:

In Design and development section:

Development status (advanced prototype, beta product, product)

In Artefact section:

Technical aspects (Operating System (or Web), Programming Language); Interactivity (Microworld, Hypermedia)

Question 4. Missing questions/items:

- Does ILE encourage communication among students themselves and with teachers? How?
- What modes of communication ILE offers? (such as: language, visual representations, sound, .. etc)
- Do you think that ILE causes or creates any "misconceptions" that students will construct?

Question 5. Other remarks, suggestions related to the grid

Maybe putting range to answer some questions will help to make sense of some issues. I mean using Lekert scale (1=not good….. 5=excellent) maybe will help in achieving this idea.

3.2.6. Summary of the TELMA teams critical inputs

As regards the clarity of questions / items of the analysis grid, two main remarks have been raised:

- Some items are not given the same meaning by different teams or are not well understood (e.g., *Lecture* in the item TYPES OF ACTIVITIES, *Nature* in the FEEDBACK item or Familiarization with the ILE in USAGE item);

- Some items need to be refined. This is the case of questions where several possibilities are mentioned. In some cases, proposed items do not seem to cover the whole range of possibilities (e.g., *Formal exercises* and *Word problems* for the item SORT OF EXERCISES, or *Microworld* and *Hypermedia* for the item INTERACTIVITY).

As far as the applicability / relevance / interest of the questions / items is concerned, such aspects like *Licence* and *Price* in DISTRIBUTION item, TECHNICAL ASPECTS (*Operating system*, *Programming language*, *Translations*), or the whole part VALIDATION, USAGE, EXPERIMENTATION are considered by some teams as too technical and of little interest with respect to the aims of the study.

The software analysis reveals that some aspects considered as important for the study are not covered by the questions / items in the actual grid. Some partners raise the issue of evaluating the piece of software and suggest adding a questionnaire devoted to a qualitative analysis of tasks proposed by the system from both mathematical and didactical point of view. Other issues considered interesting to study and not addressed by the grid are the following:

- modalities of employment of the system;

- does the system encourage communication among students themselves and with the teacher? If so, how?

- what modes of communication the system offers (language, visual representation, sound…)

- do you think that the system causes or creates any “misconceptions” that students will construct?

Finally, there is a suggestion for reorganizing of the whole grid so that it makes apparent didactical functionalities of the system as envisaged by the designers. Indeed, analysing software in terms of its didactical functionalities would bring this study closer to the rest of activities developed by TELMA.

3.3. Refining the methodological tool for software analysis

Recall that the aim of this study is not to judge the quality of the systems analysed, but rather to gain a global view of the state-of-the-art educational software in the field of arithmetic and algebra. An analysis grid has been designed in order to gather as much information as possible about the design and the development of the system and its usage, and to frame an inspection of various aspects of the artefact. The goal of the analysis, through exploiting grids filled out for different systems, is identifying and comparing didactical functionalities[3] of these systems, not in terms of their quality, but rather in terms of

[3] Didactical functionalities of a tool are defined as follows: “With didactical functionalities we mean those properties (or characteristics) of a given ICT, and/or its (or their) modalities of employment, which may favour or enhance teaching/learning processes according to a specific educational goal.

present (or missing) functionalities. Section 5 reports on the comparison of the educational systems studied in the framework of TELMA work.

Taking into account all teams' remarks and suggestions concerning the initial analysis grid and the goal of the study, we propose a new version of the analysis grid with the following modifications:

- some of those items that are considered not relevant with respect to didactical functionalities of a tool have been suppressed;
- items judged not explicit enough in the initial grid have been clarified;
- the whole grid has been reorganized so that it makes apparent didactical functionalities of the system under study;
- all questions /items are accompanied with a short description of the nature of the expected answer.

Name of the ILE:
Date of the study:

Design and development

This part is to be filled out based on available documents (articles, reports…).

Who:
Authors of the ILE

1. People/organisation involved in the design:
Field of people involved in the design (computer science, psychology, educators...)

2. Interaction between these participants:
What each category of designers is in charge of?

Why:
Motivation, educational goals of the designers

1. Aims of the design:

2. Research questions:

For whom:
Intended audience, school level…

How:
Main choices and decisions made in the design phase, and a rationale

1. Constraints affecting the design

2. Underlying theories and principles

Distribution:

1. Licence

2. Price

The three key elements of the definition of the didactical functionalities of an ICT tool are: (1) a set of features/characteristics of the tool; (2) a specific educational goal; (3) a set of modalities of employing the tool in a teaching/learning process referred to the chosen educational goal." (Cerulli & al. 2005, p.2)

Artefact

This part is to be filled out based on the inspection of the system.

Technical aspects:

1. Operating System (or Web):
What operating systems or Web browsers the ILE functions with?

2. Translations:
In what languages the ILE is available?

Mathematical content:

Parts of arithmetic and algebra curriculum covered by the ILE

Interface:

How mathematical objects are represented within the ILE, and how these can be manipulated?

1. Representations of mathematical objects available in the ILE:

1.1. *Description*

1.2. *Distance between institutional and/or cultural representations and the representations available in the ILE*

2. Manipulation:
What can be manipulated (objects, rules…) and how?

Types of activities:

For what purposes can the ILE be used (what modalities of employment are envisaged by the designers)?

1. Does the ILE provide lectures (definitions, theorems, proofs…)?

2. Does it provide solved examples?

3. Does it provide exercises to be solved by the student? If so:

3.1. *What kinds of exercises (formal exercises, word problems, multiple choice questions…)?*

3.2. *What kind of answer is expected (step by step answer, final answer…)*

3.3. *Is it possible to give no answer?*

Interactivity:

Features of the ILE concerning user-ILE interactions

1. Feedback

1.1. *What kind of feedback is provided by the ILE?*

1.2. *When a feedback is provided?*

2. Knowledge of the task

2.1. *Does the ILE know what the student's task is?*

2.2. *Can the ILE provide hints?*

2.3. *Can the ILE provide the solution of the given task?*

2.4. *Can the ILE solve a given task step by step?*

Autonomy:

Features of the ILE allowing an autonomous work with the ILE

1. **What is the degree of student's autonomy?**
2. **Does the ILE allow self-evaluating?**
3. **Does the ILE provide a diagnosis of the student's knowledge or skills?**

Teacher's role:

Features of the ILE concerning allowing customisation of the ILE by a teacher

1. **Are there any parameters a teacher can set up in order to adapt the ILE to her/his own objectives?**
2. **Does the ILE allow creating exercises? Learning scenarios?**

Experimentations

This part is to be filled out based on available documents (articles, reports…).

Experimentation

Empirical evaluation of the ILE (instrumental issues, evidence of an impact on learning)

1. **Usage analysis**

   1.1. *Is the interface user-friendly allowing a quick familiarization with the ILE?*

   1.2. *Is the ILE used at a regular base?*

2. **Is there evidence of an impact on students' learning?**

   2.1. *From a quantitative point of view (improvement of students' achievements…)?*

   2.2. *From a qualitative point of view (overcoming common misconceptions….)?*

4. Analysis T-algebra

The following analysis has been framed by the refined grid (cf. section 3.2.7).

Name of the ILE: T-Algebra
Date of the study: 9-10-2007

Design and development

This part is to be filled out based on available documents (articles, reports…).

Who:

Authors of the ILE

1. People/organisation involved in the design:
Field of people involved in the design (computer science, psychology, educators...)

Marina Issakova, Dmitri Lepp and Rein Prank.
Rein Prank is associate professor in computer science at the University of Tartu (Estonia). Marina Issakova and Dmitri Lepp are PhD students in computer science at the University of Tartu (Estonia).

2. Interaction between these participants:
What each category of designers is in charge of?

Rein Prank is the director of the PhD of Marina Issakova and Dmitri Lepp. They worked with mathematics teachers for the general specifications of the system.

Why:

Motivation, educational goals of the designers

1. Aims of the design:

Expression manipulation is one of the key skills needed for solving problems in practically all fields of mathematics. The learning poses difficulties to many students and is relatively labour-intensive for the teachers while the results are often unsatisfactory.

The aim is to develop a system containing an interface allowing the program to understand all decisions made by students.

2. Research questions:

Is it possible to improve the beginning of the learning of algebra with a system containing an interface allowing it to understand all decisions made by students?

For whom:

Intended audience, school level…

Student at school, during beginning of the learning of algebra (solving linear equations, inequalities and systems; expansion of polynomials).

How:

Main choices and decisions made in the design phase, and a rationale

1. Constraints affecting the design

Interaction according to the Action-Object-Input scheme which consists of asking the student to choose an action in a menu and select a sub-expression on which to apply this action, then to provide the result.

Feedback is given at many levels, including incorrect selection, action not applicable to the selection, incorrect result.

2. Underlying theories and principles

The importance of immediate feedback.

Distribution:

1. Licence: n/a

2. Price: free

Artefact

This part is to be filled out based on the inspection of the system.

Technical aspects:

1. Operating System (or Web):
What operating systems or Web browsers the ILE functions with?

Windows standalone application

2. Translations:
In what languages the ILE is available?

Estonian and English

Mathematical content:

Parts of arithmetic and algebra curriculum covered by the ILE

Solving linear equations, inequalities and systems (2 equations and 2 unknown having 1 solution). The task is either the “solve” task or a subtask like “Multiply both side of the equation by common denominator”.

Expansion of polynomials.

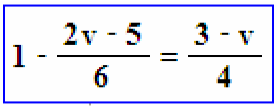
Interface:

How mathematical objects are represented within the ILE, and how these can be manipulated?

1. Representations of mathematical objects available in the ILE:

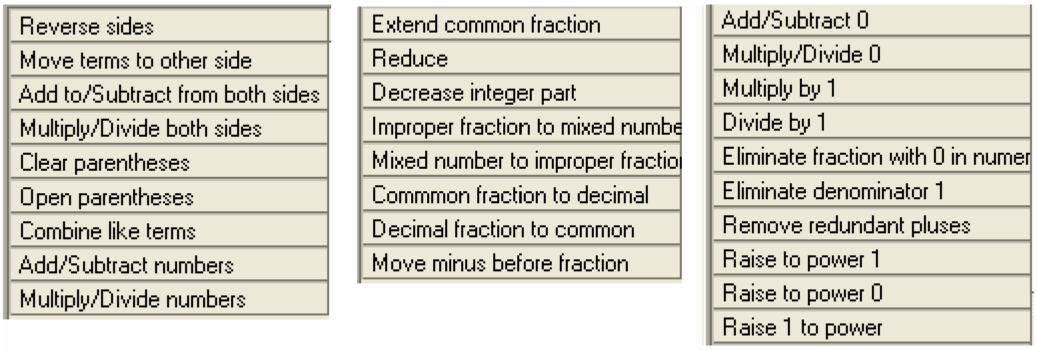
1.1. *Description*

Algebraic expressions are represented in their usual 2D form.

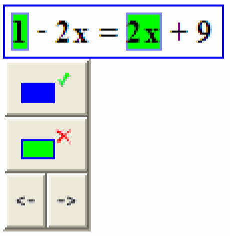


$$1 - \frac{2v - 5}{6} = \frac{3 - v}{4}$$

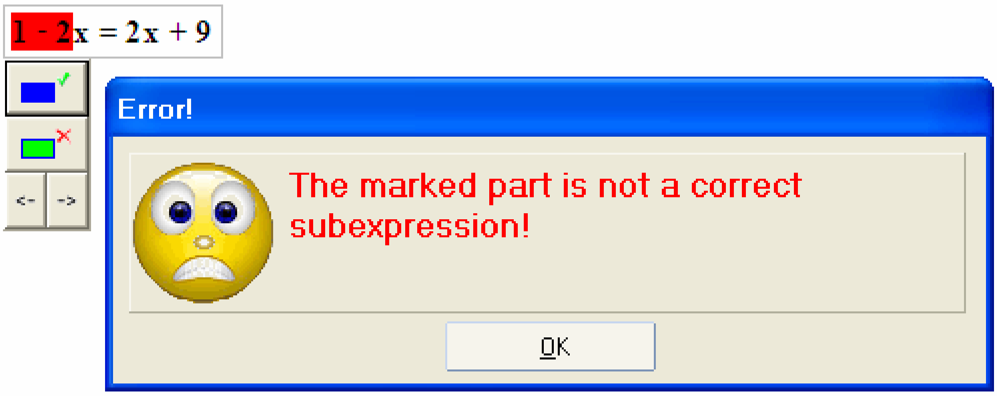
The student chooses an action in a menu. Here is the menu for the “Solve equation” type of problem.



The student also selects a sub-expression, or several sub-expressions for some actions like “Move terms to other side”. This selection is done by moving the mouse and using a 4 buttons panel.



The selection of a part which is not a sub-expression is possible. An error is indicated when the student asks to validate the selection.

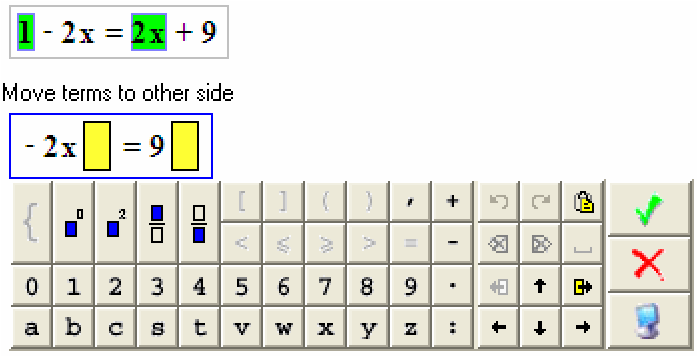


Error!

The marked part is not a correct subexpression!

OK

When the request is correct, a new expression is built by T-algebra with input areas where the student h write the result of the action.

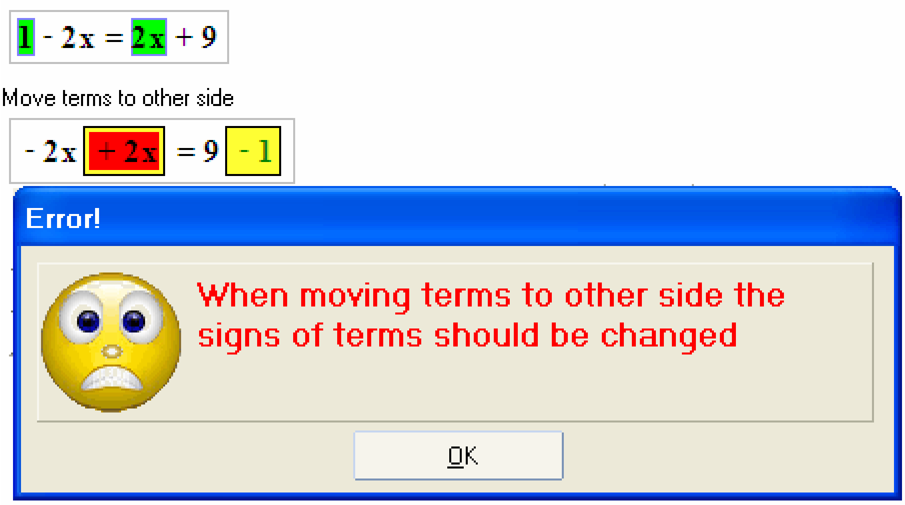


1 - 2x = 2x + 9

Move terms to other side

- 2x □ = 9 □

The result is analysed and feedback is provided when errors occur with a localisation of the part of the expression which is concerned by the error.



1 - 2x = 2x + 9

Move terms to other side

- 2x + 2x = 9 - 1

Error!

When moving terms to other side the signs of terms should be changed

1.2. *Distance between institutional and/or cultural representations and the representations available in the IL*

The representation of algebraic expressions is identical to the institutional representation.

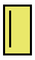
There is no institutional representation of the algebraic rules in mathematics education. The authors chosen a representation which makes sense but mixes low level rules like “Eliminate denominator 1” more complex ones like “Combine like terms”. Some transformations, like the simplification of 16/4, are r complex to perform with the system than they are on paper.

2. Manipulation:

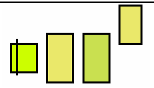
What can be manipulated (objects, rules…) and how?

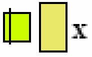
Algebraic expressions are selected. Rules are chosen in a menu. Algebraic expressions are input.

The input stage in the program can proceed in three different modes:

1. **free** input: only one box for the whole result 

2. **structured** input: several boxes for entering the result, separate boxes for sign, coefficient and/or

variable, exponent 

3. **partial** input: the program fills out some parts of the result and leaves empty boxes only for most important parts: sign, coefficient, exponent 

Types of activities:

For what purposes can the ILE be used (what modalities of employment are envisaged by the designers)?

1. Does the ILE provide lectures (definitions, theorems, proofs…)? No

2. Does it provide solved examples?

It is possible to get a solved example by using the autosolve functionality.

3. Does it provide exercises to be solved by the student? Yes

If so:

3.1. *What kinds of exercises (formal exercises, word problems, multiple choice questions…)?*

Formal exercises (check validity of a numerical inequality, add a number to both sides of an inequality, solve an equation…)

3.2. *What kind of answer is expected (step by step answer, final answer…)*

Step by step answer with indication of the action, the sub-expression and the result.

3.3. *Is it possible to give no answer?* No

Interactivity:

Features of the ILE concerning user-ILE interactions

1. Feedback

1.1. *What kind of feedback is provided by the ILE?*

Feedback on incorrect selections, incorrect action and incorrect result of action: a textual error message is sent to the user and the wrong part of the expression is highlighted with red background.

1.2. *When a feedback is provided?*

Immediate feedback when an error is detected.

2. Knowledge of the task

2.1. *Does the ILE know what the student's task is?* Yes

2.2. *Can the ILE provide hints?* Yes, on user's demand (Hint button), the ILE indicates the action to perform.

2.3. *Can the ILE provide the solution of the given task?* Yes, on user's demand (Autosolve button), the ILE solves completely the exercise. All steps are provided with the names of actions used and parts of expressions on which the actions have been applied highlighted with green background.

2.4. *Can the ILE solve a given task step by step?* Yes (see above).

Autonomy:

Features of the ILE allowing an autonomous work with the ILE

1. What is the degree of student's autonomy?

Average. The student is not obliged to follow a particular path. S/he can make many sorts of errors. However, s/he has to think his/her actions through the action names presented in the menu.

2. Does the ILE allow self-evaluating?

Yes. When the student uses an adequate list of exercises, using the number of solved exercises, the number of errors (Error counters item) and the nature of the errors (Error list item). The system keeps a history of the use of hints: number of help usage (Counters of help usage item) and nature of hint provided (List of help usage item).

3. Does the ILE provide a diagnosis of the student's knowledge or skills?

No

Teacher's role:

Features of the ILE concerning allowing customisation of the ILE by a teacher

1. Are there any parameters a teacher can set up in order to adapt the ILE to her/his own objectives?

No

2. Does the ILE allow creating exercises? Learning scenarios?

Teacher software allows creating lists of exercises.

Experimentations

This part is to be filled out based on available documents (articles, reports…).

Experimentation

Empirical evaluation of the ILE (instrumental issues, evidence of an impact on learning)

1. Usage analysis

1.1. Is the interface user-friendly allowing a quick familiarization with the ILE? Yes

1.2. *Is the ILE used at a regular base?*

It has been used in several experiments and is supposed to be used largely.

2. Is there evidence of an impact on students' learning?

2.1. *From a quantitative point of view (improvement of students' achievements…)?*

Yes, measured with pre-tests and post-tests.

2.2. *From a qualitative point of view (overcoming common misconceptions….)?*

The diminution of errors is clearly established but not really linked with overcoming misconceptions.

5. Comparison of the ILEs

In this section we report on the first attempt to compare the 10 ILEs based on the exploration of the analysis grids filled in for each of the ILEs. This part of the ILE study requires a further development and will be continued during the next three months.

In what follows, we propose comparing features of the ILEs classified into three categories:

1) features related to the knowledge domain of the ILE: educational goals of the designers, school level, mathematical content covered by the ILE, representation systems used and possible manipulation on them, and types of activities proposed by the ILE (section 5.1);

2) features related to the user-ILE interaction: feedback/reaction to errors/hint; user's autonomy, diagnosis, possibility to customise the ILE by a teacher (section 5.2);

3) technical aspects, accessibility and availability: operating system, distribution, translations (section 5.3).

5.1. Knowledge domain of the ILEs

	Goals	School level	Mathematical content	Representation	Manipulation	Types of activities
ActiveMath	Learn maths	16 years and up	Matheführerschein: equations, fractions, functions) Algebra: introduction to groups	Textual Graphical Geometric Algebraic (usual)	On graphs, texts, numbers, geometric figures, applets, expressions Hypertext available	Lectures Solved examples Proofs of theorems Exercises (open questions, multiple choice questions, computational exercises)
AnimalWatch	Raise young girls self-confidence in their maths skills	9 – 12 years	Basic arithmetic in integers Fractions Word problems	Textual Images Graphs/charts/diagrams Algebraic (usual) Numerical (answer box)	On numbers and diagrams	Word problems
Cognitive Tutor	Support students in applying algebra to real-world problems Support reasoning among multiple representation Employ modern computational tools	10 – 18 years college	Bridge to algebra (Numbers, fractions, algebraic problem solving, linear functions...) Algebra I (linear expressions, equations and inequalities; quadratic expressions and equations...) Algebra II (functions, modeling and multiple representations...)	Textual Numerical (table) Graphical Algebraic (usual)	Entering numbers, unit labels and expressions Choosing transformations from a list and entering arguments in boxes	Formal, computational exercises Word problems

Math Teacher+	Master skills and algebraic concepts	11 – 18 years	Algebra 1 (expressions, linear equations and inequalities, area and volume) Algebra 2 (quadratic equations and inequalities, equations and systems with two variables, quadratic systems and parameters, sequences)	Algebraic (usual) Graphical	On algebraic rules and expressions Experimenting with graphs of functions	Lectures (lessons) Solved examples Exercises (formal) Word problems
MathXpert	Help students learn algebra	10 – 18 years university	Algebra (linear functions and systems of linear equations; quadratic equations and functions; inequalities…)	Algebraic (usual) Graphical	On algebraic expressions, (systems of) equations and inequalities, graphs	Formal exercises (ex. eliminate rational exponents, simplify roots and fractions…)
MrsLindquist	Tutor students in writing expressions for algebra word problems	12 – 16 years	Symbolization (i.e., writing an algebraic expression for a given word problem)	Textual Numerical Algebraic (usual)	Inserting natural language sentences by means of pull down menus Inserting algebraic expressions	Word problems
T-algebra	Improve learning of early algebra with a system whose interface allows understanding all decisions made by students	12 – 16 years	Solving linear equations, inequalities and systems (with 2 equations, 2 unknowns having a unique solution); expanding polynomials	Algebraic (usual) Textual (action menu)	Selecting algebraic expressions Choosing a rule from a menu Entering algebraic expressions	Formal exercises (solve equations, inequalities or systems of equations; expand polynomials)
Aplusix	Help acquire calculation and symbolization skills	13 – 16 years	Numerical calculation Expanding Factoring Solving equations, inequalities and systems of equations	Algebraic (usual)	On algebraic expressions	Formal exercises (calculate, factor, expand and simplify, solve)

AriLab2	Help develop skills in solving arithmetic problems	6 – 12 years	Symbolic manipulator: linear expressions, rational expressions leading to second degree	Symbolic manipulator: numerical, algebraic (usual)	Symbolic manipulator: selecting a (sub)expression, choosing a rule to apply from a list of applicable rules	Not defined
			Fraction microworld: constructing fractions on a number line, comparing and ordering fractions, operations with fractions	Fraction microworld: graphical, numerical	Fraction microworld: changing units on the two number lines	
E-slate	Provide tools for producing educational software	12 – 16 years	Fractions of a quantity Ordering rational numbers	Sliders Symbols in LOGO programming Graphs	On numerical values in the LOGO procedure, on pointers on the sliders	Not defined

Table 1. Knowledge domain of the ILEs.

5.2. User-ILE interaction features of the ILEs

	Feedback	Reaction to error	Hint / correct answer	Diagnosis	User’s autonomy	Customization
ActiveMath	Immediate checking of the correctness of the answer	Explanatory note when classical error Input syntax help when syntax error Hint (part of strategy, theorem to use…) otherwise	Hint messages when error detected Hypertext links available Correct answer never provided	Estimation of the current state of knowledge (mastery colors) Providing a student’s profile	High	Possibility to create her/his own book tailored to her/his needs
AnimalWatch	Immediate checking of the correctness of the answer	Hints Eventually, the problem is solved in the help screen	Hints are provided when an error is detected Correct answer is presented step by step	Monitoring students’ progress (Reporter accessible only by a teacher)	Limited: strong guidance by the system	Teachers can create exercises

		Error has to be fixed before moving on	when a student keeps doing wrong	Evaluation based only on the final answer		
Cognitive Tutor	Immediate checking of the correctness of the answer	Erroneous answer getting red Explanatory note when the error matches a common misconception Error has to be fixed before moving on	Help message (3 levels) on demand Eventually the right answer is provided	Based on knowledge tracing (skillmeter displayed permanently)	Limited: strong guidance by the system	No
Math Teacher+	Immediate checking of the correctness of the answer and of the end of exercise Interpretation suggested by the system when possible ambiguity	Erroneous answer getting red Alert message when syntax error Up to 3 mistakes, the system provides the solution	Guidance (on demand): static message providing general idea how to start solving the problem Hint (on demand): dynamic help referring to the last answer line providing a tip on how to continue	Report on student's progress available on demand	Partial: one can choose the topic to study and decide to take a test	Teacher can chose the sequence to be studied and set up parameters to control the system behavior
MathXpert	Selection of a (sub)expression displays a context sensitive menu showing compatible actions to choose from On demand, the system tells whether the problem is solved or not (Finish)	Impossible to commit an error	Hint (on demand): advice on how to continue solving the problem AutoFinish (on demand): the problem entirely solved by the system AutoStep (on demand): the problem is solved step by step by the system	No	Partial: choice of actions, but only mathematically correct actions can be performed	Teacher can create exercises
MrsLindquist	Dialog form immediate feedback depending on the chosen tutoring strategy: asking to solve an instance of the problem and then generalize it, to define sub-goals of the problem and solve them, to explain the correct answer provided by the system, scaffolding, providing the correct answer			Based on model tracing paradigm	Low in selecting problems Low in answering in English (by means of pull down menus) High in entering expressions as answers	No

T-algebra	Immediate feedback if incorrect selection of a sub-expression, incorrect action or incorrect result of the action	Error message when incorrect selection of a sub-expression, incorrect action or incorrect result of the action (the error message cannot be closed during 3 seconds obliging the user to read it before closing) The wrong parts of an expression highlighted with red background	Hint (on demand, available only when choosing an action): indication of the action to use Autosolve (on demand, available only when choosing an action): the problem entirely solved by the system	No	Partial: choice of actions from a menu containing action names, and input of expressions	Teacher can create lists of exercises with teacher software
Aplusix	Immediate checking of the equivalence between two consecutive steps in the solving process, in the training mode No feedback available in the test mode, but delayed feedback available in Self-correction activity Checking whether the exercise is solved on demand	Syntax error message when syntax error detected Blue cross lines when the last expression is not well-formed Red cross lines between non equivalent expressions	No hint available Correct answer provided on demand	Score provided for exercises, but not for word problems	High	Teacher can set up parameters to adapt the behavior of the software Teacher can create exercises and problems with Aplusix editor

AriLab2	Symbolic manipulator: selecting an (sub)expression activates applicable rules Choosing a rule when an(sub)expression is selected provides the transformed expression and the rule is added to the prove space	Impossible to commit an error	No hint available Correct answer is not provided	No	Symbolic manipulator: partial in solving a problem (selecting a rule from a list of applicable ones)	Teacher can create exercises
	Fraction microworld: immediate visual feedback (constructing a fraction, post-its containing expressions reflecting the way a fraction has been constructed)				Fraction microworld: high	
E-slate	Immediate visual feedback observing the behavior of the sliders	Impossible to commit an error	No hint available	No	High	Teacher can create exercises

Table 2. User-ILE interactivity features.

5.3. Technical aspects, accessibility, availability of the ILEs

	Operating system	Translations	License	Price
ActiveMath	Web (Mozilla Firefox, Internet Explorer, Safari)	English, German, Spanish, Mandarin, Russian, Dutch	German Free Software License	Not yet sold product
AnimalWatch	Windows 95 and later MacOS 8.0 and later	English	Sun Microsystems, Inc.	CD available on request or freely downloadable from the website
Cognitive Tutor	Windows 2000, XP MacOS 10.3	English	Carnegie Learning	For each of the 5 curriculum products: Student Text Set: 21,25 $ Teacher Text Set: 80 $ Software Implementation Guide: 47,50 $ Complete Curriculum Kits: 99 $
Math Teacher+	Windows 95 and later, NT, XP	English, Spanish, Portuguese, French, German, Swedish, Turkish and Hebrew	MATH-KAL	For each of the components: Single CD: 99 $ Complete curriculum (5CDs): 395 $ Complete curriculum (1 CD): 300 $ Site license: 795 $ Lab-pack: 375 $
MathXpert	Windows Mac using Virtual PC Linux using WINE	English, French, German	14-days trial version Stand-alone license Lab license	Algebra Assistant: 49,95 $ Algebra lab license: 750 $
MrsLindquist	Windows - Mac	English	n/a	Free
T-algebra	Windows	Estonian, English	n/a	Free
Aplusix	Windows 95 and later, NT, Millenium, XP, Vista	French, English, Italian, Spanish, Portuguese, Vietnamese	Commercialized by several publishers in a few European countries	Single software in France: 55 € Site license in France: 600 € Free research version
AriLab2	Windows	English, Italian, Spanish	DIDA.EL S.p.A., Milano, Italy	61 €
E-slate	Windows 98 and later, XP	Greek, English	Free use for non-commercial purposes	n/a

Table 3. Technical and accessibility aspects of the ILEs.

6. Conclusion and future work

Studying these ILEs allowed us gaining a deeper insight in state-of-the art technological tools devoted to the teaching and learning algebra. The results will enable TELMA teams to position better the ILEs they develop within the landscape of computer-based educational tools for arithmetic and algebra.

The obtained results will be further explored and will be published in a special issue of the International Journal of Computers for Mathematical Learning.

References

AriLab2 (1999). Copyright CNR-IMA, Distributed by Dida*EL Srl., www.didael.it.

Bottino R. M. & Chiappini G. (2002). Advanced Technology and Learning Environments, Discussing their relationships in the Arithmetic Problem Solving Domain, English L. D. (ed.), *The Handbook of International Research in Mathematics Education*, Lawrence Erlbaum Associates Mahwah-NJ, 29, 757–786.

Bottino R. M. & Chiappini G. (2003). An innovative teaching and learning environment for school mathematics, *T.E.L.'03 Proceedings: International Conference on Technology-Enhanced Learning*, FAST, Milan, Italy, 73-80.

Bottino R. M., Chiappini. G., Ferrari P. L. (1994). A hypermedia system for interactive problem solving in arithmetic *Journal of Educational Multimedia and Hypermedia*, AACE 3, 3/4, 307-326.

Brousseau G. (1997). *The Theory of Didactical Situations in Mathematics*, N. Balacheff, M. Cooper, R. Sutherland, & V. Warfield, (Trans. & Eds.) Kluwer Academic Publishers, Dordrecht.

Cobb, P., & Yackel, E. (1996). Sociomathematical norms, argumentation and autonomy in mathematics. *Journal for Research in Mathematics Education, 27*(4), 458-477.

Cole M., Engeström Y. (1993). A cultural-historical approach to distributed cognition. In G. Salomon (Ed.), *Distributed cognitions: psychological and educational considerations*, Cambridge: Cambridge University Press, MA, 1-47.

diSessa A. A. (2004). Meta-representation: Native competence and targets for instruction. *Cognition and Instruction*, *22*(3), 293-331.

Kynigos C. (2002). Generating cultures for mathematical microworld development in a multi-organisational context. *Journal of Educational Computing Research, 1-2,* 183-209.

Kynigos C. (2003). A "black-and-white box" approach to user empowerment with component computing. *Interactive Learning Environments*.

Kynigos C., & Argyris M. (2004). Teacher beliefs and practices formed during an innovation with computer-based exploratory mathematics in the classroom. *Teachers and Teaching: Theory and Practice*, *10*(3), 247-273.

Morgan C. (2007). Variations on a theme: introducing new representations of fraction into two KS 3 classrooms. *Proceedings of the British Society for Research into Learning Mathematics, 27*(1), 66-71.

Nicaud J.-F. (2001). A questionnaire for educational environments for algebraic reasoning. Unpublished paper presented in a working group in the Melbourne ICMI study in 2001.

Nicaud J.-F., Bouhineau D., & Chaachoua H. (2006). Mixing Microworld and CAS Features in Building Computer Systems that Help Students Learn Algebra. *International Journal of Computers for Mathematical Learning* 9 (2) (2004) 169-211.

Nicaud J.-F. (Ed.) (2006): *State of the art of didactical ICT systems*. Deliverable D20.05.01.

Nicaud J.F., Bittar M., Chaachoua H., Inamdar P., & Maffei L. (2007). Experiments with Aplusix in Four Countries. *The International Journal For Technology in Mathematics Education. 13(2).*

Noss R., & Hoyles C. (1996). *Windows on Mathematical Meanings: Learning Cultures and Computers*. Dordrecht: Kluwer.

Psycharis G., & Kynigos C. (2004). Normalising geometrical constructions: A context for the generation of meanings for ration and proportion. In *Proceedings of the 28th Conference of the International Group for the Psychology of Mathematics Education*. Bergen, Norway.

Tchounikine P. (2004). *Platon-1: quelques dimensions pour l'analyse des travaux de recherché en conception d'EIAH*. Rapport de l'Action Spécifique « Fondements théoriques et méthodologiques de la conception des EIAH », département STIC du CNRS.

Tricot A., Plégat-Soutjis F., Camps J.-F., Amiel A., Lutz G., & Morcillo A. (2003). Utilité, utilisabilité, acceptabilité : interpréter les relations entre trois dimensions de l'évaluation des EIAH. In *Actes de la Conférence Environnements Informatiques pour l'Apprentissage Humain*, Strasbourg, 391-402.

Yiannoutsou N., & Kynigos C. (2004). Map construction as a context for studying the notion of variable scale. In *Proceedings of the 28th Conference of the International Group for the Psychology of Mathematics Education*. Bergen, Norway.

Appendix 1

A questionnaire for educational environments for algebraic reasoning

Melbourne ICMI Study 2001

J.F. Nicaud, IMAG-Leibniz, France

We particularly consider educational environments for ***algebraic reasoning***, i.e., computer systems that allow students to solve problems by developing a ***step-by-step reasoning***. In order to study this family of computer systems, we will use the model detailed in 2.1 and the questionnaire given in 2.2.

We also consider some educational environments for algebra that are not in this family, particularly environments that concern modelling problems, i.e., the translation in algebra of word problems.

A model

An ***algebraic problem*** is given by a problem type (e.g., factor polynomial, solve equation, calculate primitive) and an expression. There is an equivalence relationship between expressions: replacing an expression or a sub-expression in a problem by an equivalent expression provides a new problem having the same solutions. Identities allow to transform expressions with conservation of the equivalence. For that, they are oriented, taking the form of transformation rules (or rewriting rules), for example, A^2-B^2 = (A-B)(A+B) is sometimes used through A^2-B^2 $\rightarrow$ (A-B)(A+B) and sometimes through (A-B)(A+B) $\rightarrow$ A^2-B^2. There are elementary transformations rules, like A+0 → A. There are complex transformations rules like the use of the discriminant b^2-4ac to factor ax^2+bx+c.

The main inference mechanism to solve an algebraic problem is applying correct transformation rules to sub-expressions (or the whole expression which is a particular sub-expression). This mechanism can be decomposed in:

1) Matching: determine applicable transformations
2) Strategy: choose an applicable transformation
3) Application: apply the chosen rule.

As problems are not always solved a direct way, there is another strategic aspect that consists of deciding between continuing on the current way or backtracking to a previous step to try another way.

Matching consists of finding a sub-expression E1, a transformation rule R and a substitution S between the rule variables and some expressions such as replacing the rule variables by these expressions in the left hand side of the rule provides an expression equivalent to E1.

For example, the rule A^2-B^2 $\rightarrow$ (A-B)(A+B) is applicable to the sub-expression $4x^2$-1 of the expression (2x-1)(x+3)+$4x^2$-1 because its left hand side A^2-B^2 matches $4x^2$-1 with the substitution [A: 2x, B: 1].

Applying an applicable rule R with the substitution S consists of replacing the right hand side of the rule by the expressions of the substitution and replacing E1 by the result. In the previous example, one first applies the substitution to (A-B)(A+B) and gets (2x-1)(2x+1) then replaces $4x^2$-1 by the result and gets (2x-1)(x+3)+(2x-1)(2x+1).

Describing in detail the application of a transformation rule R consists of providing R, E1, S and the result. Usually, partial descriptions are given: sometimes, instead of indicating the precise rule, one indicates a family (e.g., reduction); often, E1 is not indicated; S is quite never shown.

Questionnaire: features of educational environments for algebraic reasoning

We consider computer environments allowing a student to solve algebraic problems in the sense of 2.1 and able to provide some help to him/her in this task. Furthermore, we only consider systems having some proximity with human reasoning.

Many systems do not belong to this family, for example: (1) *computer algebra systems* because they solve problems in one step with special algorithms, (2) *systems drawing graphs of functions* and *systems for modelling world problems with algebra* (in their modelling part) because the activity they manage is not algebraic reasoning in the sense of 2.1.

The domain of algebra that is the scope of a system for algebraic reasoning is an important characteristic of this system. So is the proximity with human reasoning in each sub-part.

Analysis from a domain/capacity point of view:

D1: What is the domain?
D2: How large is the domain?
D3: How powerful is the system in the domain?
D4: Use only methods that humans use?
D5: Draw expressions the 2D usual way?
D6: Edit expressions a 2D way?

A system for algebraic reasoning generally takes in charge a part of the reasoning process. We will analyse the computer systems according to this issue that we split in features coming from the previous model.

Analysis from an inference and interaction point of view:

I1: Who consider the rules?
I2: Who selects the sub-expression?
I3: Who performs the matching?
I4: Who chooses the applicable rule?
I5: Who applies the rule (the chosen applicable rule)?

We will also analyse systems on their capacity to be adapted to a knowledge level.

Analysis from an adaptability point of view:

A1: Are there pieces of knowledge that can be activated/deactivated?

A2: Are there different possible grain-sizes (small steps, large step)?

We will also analyse systems from a heuristic and tutorial point of view.

Analysis from a heuristic point of view:

H1: Does the system allow backtracking?

We consider that a system allows backtracking when an on going reasoning may have several paths and in that case appears as a tree.

Analysis from a tutoring point of view:

T1: does the system provide hint on rules and matching?
T2: does the system provide hint on strategy?
T3: does the system provide feedback on errors?