ISBN 978-609-95760-0-8

IFIP TC3 Working Conference
“A New Culture of Learning: Computing and next Generations”

Proceedings

Andrej Brodnik, Cathy Lewin (Eds.)

July 1st - 3rd, 2015, Vilnius University, Lithuania

Preface

Next year in November it will be 80 years since Alan Turing delivered his seminal paper *On Computable Numbers, With An Application To The Entscheidungsproblem*, which, with his description of the universal computing device, can be considered as a beginning of a new science – Computer Science. It took almost 40 years for the described device to become ubiquitously present and available as information and communication technology (ICT). The papers in this proceedings show a similar dichotomy. On one side, they deliver fresh and timely results on computer science (or informatics) education and on the other side explore the possibilities of how to use ICT in education.

What have we achieved since those euphoric days? Has our concept of education in general and computer science education in particular changed and its quality improved? Can we use the lessons of the past to prepare for the future? In an increasingly interdependent and complex world, how is technology and informatics changing society and affecting education through the subject areas of humanities, science, mathematics, arts?

The IFIP TC3 Working Conference "A New Culture of Learning: Computing and Next Generations" organized by WG 3.1, 3.3 and 3.7 has addressed these questions by offering experts from across the world the opportunity to exchange ideas and knowledge, and generate a more informed understanding of the issues of informatics and digital technologies in education. The Conference call for participation offered a number of themes providing a concise overview of the most current issues in the related fields:

- Computational thinking
- Learning analytics in programming
- Approaches to learning programming
- Curriculum challenges
- Tangible programming and physical computing
- Learner perspectives
- Teacher perspectives for ICT and computing
- Assessing computing capabilities
- Innovative approaches to teaching and learning
- Online learners experiences
- Digital literacy and competences
- Scaling up digital pedagogy and innovation in the classroom: research challenges
- Collaborative learning

More than 70 original papers were submitted; they were reviewed using a double-blind peer-review process. The Conference Programme Committee, supported by additional expert reviewers, took responsibility for selecting papers for presentation.

Notably, substantial space in the 2015 conference programme has been dedicated to a doctoral consortium chaired by Don Passey. This consortium offers PhD students a friendly forum to discuss their research, to receive constructive feedback from their peers and senior researchers, to engage in networking and to discuss questions related to research and academic life.

IFIP TC3 education aims to provide an international forum for developing research and practice in computer science (informatics) education both within schools and universities, and in e-learning across all contexts from schools to the workplace. e-We are grateful to all members participating in any way in the conference and especially to: Robert Munro for organizing the review process; Fahima Djelil who oversaw the technical platform dedicated to the review workflow; Peter Hubwieser, Don Passey and Eugenijus Kurilovas for their inspiring Keynotes; again Don Passey for taking the responsibility of chairing the doctoral consortium; Andrej Brodnik and Cathy Lewin, guest editors of the special issue of EIT and Eugenijus Kurilovas, the guest editor of the special issue of IJEER; the panel and session chairs for their facilitation of the discussions; and the members of the programme committee who provided a significant contribution to the review of papers.

Many thanks to PhD student Lina Vinikienė for a normal work of formatting the conference proceedings and book of abstracts.

Welcome to Vilnius and enjoy the IFIP TC3 Working Conference!

Conference Co-Chairs Valentina Dagienė
Anita Juškevičienė
Eric Sanchez
Mary Webb

Committee

International Programme Committee

Eric Sanchez (co-chair), Ecole Normale Supérieure de Lyon, France
Mary Webb (co-chair), King's College London, United Kingdom
Andrej Brodnik (editor), University of Ljubljana, Slovenia
Cathy Lewin (editor), Manchester Metropolitan University, United Kingdom
Christine Bescherer, University of Education Ludwigsburg, Germany
Ana Amelia Carvalho, University of Coimbra, Portugal
Valentina Dagienė, Vilnius University, Lithuania
Barbara Demo, University of Turin, Italy
Helen Drenoyanni, Aristotle University of Thessaloniki, Greece
Birgit Eickelmann, University of Paderborn, Germany
David Gibson, Curtin University, Australia
Monique Grandbastien, University of Lorraine, France
Anthony Jones, University of Melbourne, Australia
Caroline Jouneau-Sion, French Institute of Education, France
Steve Kennewell, University of Wales, United Kingdom
Johannes Magenheim, University of Paderborn, Germany
Peter Micheuz, University of Klagenfurt, Austria
Izabela Mrochen, University of Silesia, Poland
Nicholas Reynolds, University of Melbourne, Australia
Ralf Romeike, Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
Sindre Rosvik, Volda University College, Norway
Sigrid Schubert, University of Siegen, Germany
Petra Strijker-Fisser, Netherlands Institute for Curriculum Development, The Netherlands
Robert Munro, University of Strathclyde, United Kingdom
Sarah Younie, De Montfort University, United Kingdom

National Committee (Vilnius University)

Valentina Dagienė (co-chair)
Anita Juškevičienė (co-chair)
Vladimiras Dolgopolovas
Eglė Jasutė
Tatjana Jevsikova
Svetlana Kubilinskienė
Eugenijus Kurilovas
Inga Žilinskienė

Keynotes

FULL PAPERS

SHORT PAPERS

PANELS/SYMPOSIUMS

DOCTORAL CONSORTIUM

Keynotes

Personal Learning Environments in Future Learning Scenarios

Peter Hubwieser, peter.hubwieser@tum.de
Technische Universität München, TUM School of Education, Germany

Jan Böttcher, jan.boettcher@tum.de
Technische Universität München, Fakultät für Informatik, Germany

Abstract

The current educational system will change fundamentally due to arising challenges. Increasing migration rates and the demand for inclusive education will extend the already existing diversity among the learners. At the same time, modern learning theories demand new educational concepts. Learning should be motivated by the elements of the Self-Determination Theory (autonomy, awareness of competence, and social relatedness) and learners should construct their knowledge actively in a situated and social context. Facing these challenges, we propose a learning scenario based on learning tasks. The teachers should assign collaborative tasks that are individually designed and tailored to the needs and abilities of the students. The learning processes should be supported by a Personal Learning Environment (PLE), which supports the learning scenario by providing and managing learning materials and tasks and enables the collaboration among learners. The universal (system) design has to provide accessibility for various groups of users, in particular people with disabilities.

Keywords

Learning scenario, personal learning environment, social networks.

INTRODUCTION

Presumably, learning scenarios in European schools will change dramatically during the next decades, enforced by the increasing diversity of students, in compliance with the modern under-standing of human learning as situated, socially embedded, active construction of knowledge, motivated by autonomy, awareness of competence and social relatedness. It seems obvious that efficient learning cannot be restricted to closed classrooms and by fixed time-tables any more. Instead, the learning process will be promoted by comprehensive, contextualized tasks that are individually tailored to the students' individual abilities and prerequisite knowledge. The students will process these tasks in close collaboration with peers. In accordance with (OECD, 2004), we suppose that progressive learning scenarios will not be become prevalent without substantial support by specific ICT systems, called Personal Learning Environments (PLEs). One reason for this assumption is the tremendous workload of teachers that will be caused by such scenarios, another is the need for peer discussion that is inherently postulated by modern pedagogic approaches. Both problems could be alleviated by ICT systems that make use of technologies that have made the breakthrough during the last couple of years. First, nearly every student has permanently access to a very powerful computing, information and communication device in form of his/her smartphone. Second, social networks are connecting millions of people all over the world. The problem for learning is that students mostly have to seek other learners with similar interests laboriously. Third, the increasing number of MOOCs is offering more and more information and tasks, developed by specialists and often professionally presented. Forth, cloud computing allows to exchange data easily between electronic devices all over the world, provided that they have access to internet.

In this article we describe the presumed typical learning scenario of the future as well as the design outline and the development process of our PLE proposal MyLearnSpace. The requirement analysis of MyLearnSpace was explained in detail in (Böttcher, 15.11.2014). The basic ideas of the assumed future learning scenario was sketched in (Hubwieser & Böttcher, 2014) already.

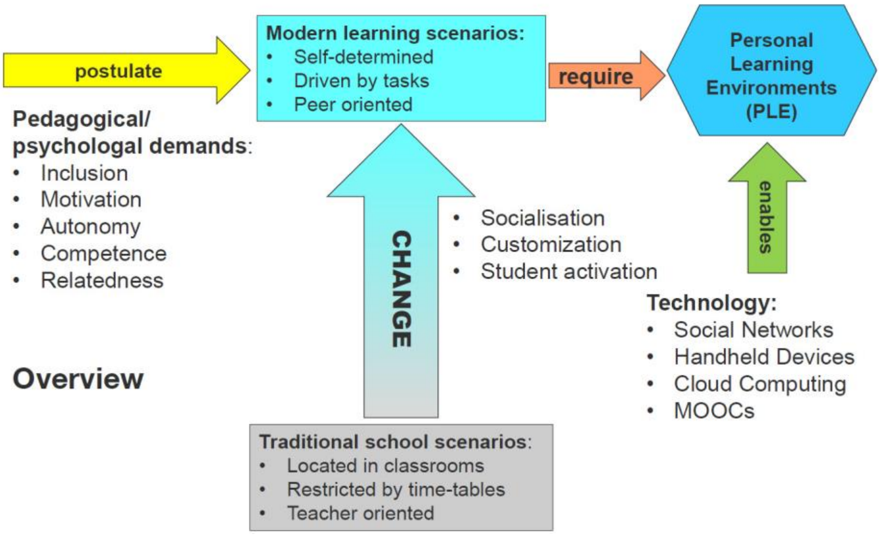


Figure 1: How PLEs support the change of learning scenarios

PEDAGOGICAL BACKGROUND

Already at the beginning of the last century, Maria Montessori (1870–1952) developed an educational approach, with the key elements independence, freedom within limits and respect (Montessori, 1993). The focus was set on individual learning, because learning would happen only intrinsically and thus cannot be prescribed externally (Montessori, 1993). Montessori postulated prepared learning environments to stimulate learning. Further, she introduced the use of adaptive learning material to stimulate autonomy (Montessori, 1993). Her approach preferred learning from working with materials, instead of direct instruction.

Célestin Freinet (1896–1966) and his wife, Élise Freinet (1898- 1983) postulated that students should learn by collaboratively working through provided learning materials that have been partly constructed by the students themselves (Freinet, 1964). Further, the learning process should be self-determined by the students rather than teacher-driven. As far as possible, the students should decide what they want to learn, with whom they want to collaborate and how much time they need for it.

The Self-Determination Theory (SDT) (Deci & Ryan Richard M., 1985), (Ryan & Deci, 2000) emphasizes that classroom practices that support students' satisfaction of autonomy, competence, and relatedness would be associated with both greater intrinsic and autonomous types of extrinsic motivation. Autonomy means that the learner has to decide what he or she wants to learn at a certain point of time. To experience one's own competence needs to work active on problems and tasks. The need of social relatedness demands to learn together with other students that are similarly motivated to do work on the same topics and problems. Further, the SDT postulates three central innate psychology needs, with all of them relating to the intrinsic and extrinsic motivation (Deci & Ryan Richard M., 1985), (Ryan & Deci, 2000), (Niemiec & Ryan, 2009): (1) competence, (2) social relatedness, and (3) autonomy.

The learning theory of Constructivism, based on the work of Jean Piaget, Lev Vygotsky, and Jerome Bruner, claims that knowledge is actively constructed by the learners rather than “transferred” from any source of instruction. It is based on the assumption that "we generate knowledge and meaning through experience" and that "knowledge is both individual and social" (Muise & Wakkary, 2010). Learning environments should support a maximum of self-control by the learner, since learning is considered as a self-driven process (Gräsel, Bruhn, Mandl, Fischer, & others, 1997). (Reinmann-Rothmeier & Mandl, 1996) remark that learners would need support in case of arising problems, which cannot be solved by the learners themselves. (Cobb, 1994) argues that knowledge is both constructed through social interaction and in the individual’s mind.

According to (Caine & Caine, 1990), the "objective of brain-based learning is to move from memorizing information to meaningful learning". Therefore, learning has to be contextual and teachers must take student interests into account: (Caine & Caine, 1991) demands that the sources of information should be quite complex, including social interactions, group discovery, individual search and reflection. Additionally, the classroom organization should support the collaborative construction of subject matter knowledge, using workstations and working on individualized projects. Many responsibilities are delegated to students, while the teachers are only monitoring.

Butler and Winne provide an analysis of cognitive processes involved in selfregulation (Butler & Winne, 1995). They review several interesting areas of research, including affect and its relation to persistence during self-regulation. Further, the role of self-generated feedback in decision making and the influence of students’ belief systems on learning are investigated. They state: “For all selfregulated activities, feedback is an inherent catalyst. As learners monitor their engagement with tasks, internal feedback is generated that describes the nature of outcomes and the qualities of the cognitive processing that led to those states.” For our purpose, their model of self-regulated learning could serve as blueprint for our learning scenario, see figure 2.

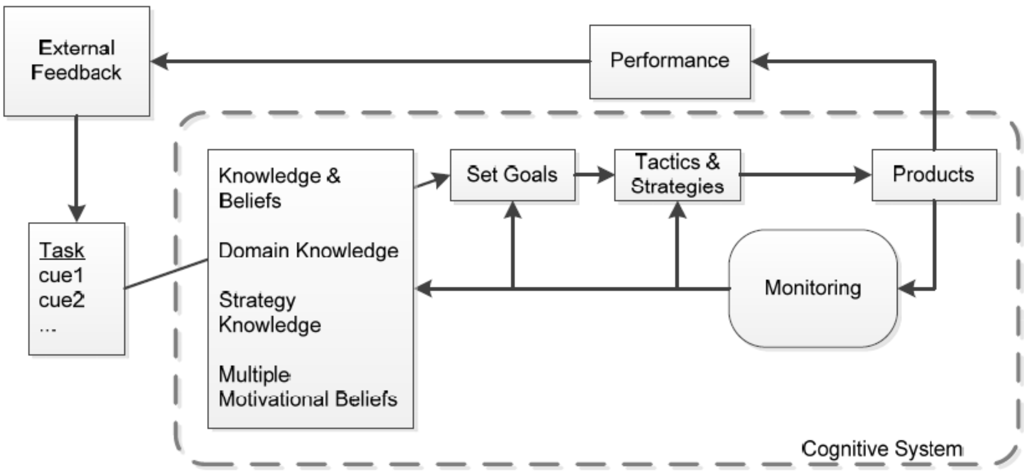


Figure 2. Model of self-regulated learning according to (Butler & Winne, 1995).

The learning strategy of peer instruction was developed by Eric Mazur at the Harvard University (Crouch & Mazur, September 2001). He is working with specific content related conceptual questions, “which probe students’ understanding of the ideas just presented. [..] Students then discuss their answers with others sitting around them; the instructor urges students to try to convince each other of the correctness of their own answer by explaining the underlying reasoning” (Crouch & Mazur, September 2001). Several surveys have demonstrated that this method works quite well. It seems able to enhance the learning success dramatically in some cases (Crouch & Mazur, September 2001), (Porter, Bailey Lee, & Simon, 2013).

In 2006, the General Assembly of the United Nations adopted the Convention on the Rights of Persons with Disabilities (United Nations, 2006). It demands that all countries have to ensure inclusive education at all levels. Separate schooling for disabled students should be cancelled and all students should be integrated into common mainstream education. Obviously, specific assistive technologies will be one of the key factors for the success of this educational inclusion (Douglas, Corcoran, & Pavey, 2007).

TECHNOLOGICAL PREREQUISITES

Social networks connect people. This can be done in two different ways, which can be combined. For one thing, users can add each other to their friends list (Werdmuller, Tosh, Files, & Free, 2005). Like in the "real world", friends on on-line platforms are more likely to collaborate than users who do not know each other (Dillenbourg, 1999). (Werdmuller et al., 2005) mention that (virtual) friendships are created by shared fields of interest. Besides bilateral friendships, communities are connecting users with shared interests. (Liccardi et al., 2007) describe communities as "informal groups of people that develop a shared way of working together to accomplish some activity". They emphasize that the membership of a community is "usually self-selected and self-organized". Typical components of communities supporting collaboration are forum, file management and wiki (Werdmuller et al., 2005).

Massive Open Online Courses (MOOCs) became an educational buzzword in 2012 and have enjoyed wide media coverage in the popular press (Marshall, 2013). The first course presented under the name MOOC took place in 2008. In contrast to traditional ways of teaching, where the size of participants is restricted, MOOCs have to be easily scalable, which causes a need for different technologies to provide or support this scalability (Wulf, Blohm, Leimeister, & Brenner, 2014): peersupport, gamification, learning-analytics, (peer-)grading, verification of identity, validation and plagiarism control. Up to now, three prominent technical platforms for MOOCs have been established: Coursera, edX and Udacity, which have their origins at American elite universities.

The term Ubiquitous Computing was created by (Weiser, 1991), where he predicted: “Specialized elements of hardware and software, connected by wires, radio waves and infrared will be so ubiquitous that no one will notice their presence” and “Ubiquitous computers will also come in different sizes, each suited to a particular task. My colleagues and I have built what we call tabs, pads and boards: inch-scale machines that approximate active Post-It notes, foot-scale ones that behave something like a sheet of paper (or a book or a magazine), and yard-scale displays that are the equivalent of a blackboard or bulletin board”. In (Krumm, 2010), the editors present an overview of the rapidly progressing field of ubiquitous computing. It covers the major fundamentals and research in the key areas that shape the field. Eleven of the most prominent ubiquitous computing researchers contributed chapters to this book. According to (Krumm, 2010), ubiquitous computing research can be categorized into three distinct areas where the research is focused: systems, experience, and sensors.

In 2011, the US National Institute of Standards and Technology (NIST) has published its commonly well-accepted definition of Cloud Computing (Mell & Grance, 2011): “Cloud computing is a model for enabling ubiquitous, convenient, ondemand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.” Furthermore, the publication lists three Service Models (Software, Platform or Infrastructure as a Service). For MyLearnSpace, the first one is relevant (Mell & Grance, 2011). Among the described deployment models described in (Mell & Grance, 2011), the Hybrid cloud. meets our demands: “The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds).”

RELATED WORK

In 2004, the International Schooling for Tomorrow Forum organized by the OECD developed a set of six scenarios for schooling in the future up to 2020, categorized according the overall role of schools in the society: On the summarizing website (OECD). The scenarios are classified in three categories: (1) Attempting to maintain the status quo (2) Re-schooling – major reform and renewal of schools and (3) Deschooling – widespread disestablishment of school systems. According to the original publication document of the scenarios (OECD, 2004), the crucial role of ICT is explained in detail. Additionally, the learning scenarios are detailed. From our point of view, our vision is a combination of the scenarios 2b and 3a.

In 2006, Microsoft has published a vision for the future of education that reflects the impact technology can have on policy and practice (Rasmus, 2008). Microsoft used its Future of Work scenarios to explore possible scenarios for learning in the future. Based on a scenario-planning process education was explored “through the lens of work, examining educators, learners, and administrators in the context of creating, synthesizing, absorbing, sharing, and managing information”.

In principle, a PLE is composed of all tools that a person uses to support learning (Attwell, 2007) Typically PLEs support learning in three main fields: (1) the learners are supported in setting and achieving their own learning goals, (2) learners can use PLEs to manage learning content and their own learning process (Cooper, Grover, & Simon, 2014), and (3) PLEs are based on the paradigm of social construction (Kreijns, Kirschner, & Jochems, 2003). Hence the communication with other learners should be facilitated. In consequence, a PLE has to be related to social network systems (Attwell, 2007).

According to (Cooper et al., 2014), PLEs have the following goals: “facilitate easy sharing, search, and retrieval of relevant knowledge and resources, help connecting people and provide easy-to-use tools for collaboration, especially around curriculum design, leverage social tools and data visualization techniques to highlight popular content as well as address burning issues and questions; provide members visibility into the relevant activities and projects, and avenues for active participation, support curation of content and facilitation of the many activities the vCoP supports, afford low barriers to sign-up and participation, and provide materials to help newcomers to get started, and incorporate features to allow busy educators to keep up with – and participate in – ongoing discussions and activity on the site”.

THE LEARNING SCENARIO

A comparison of the learning theories presented in section 2 shows that most of them recommend the following essential elements of successful learning processes: (1) activation and autonomous working of students, (2) collaboration among students, and (3) contextualization of learning tasks. Based on these elements and in respect of the increasing diversity, we suggest the following “learning scenario of the future”.

The overall organizational structure could still be quite traditional: a teacher is teaching 1-2 subjects and takes care of several classes or courses in these subjects. Every student attends classes in about 15 different subjects, depending from the grade and the type of the school.

The individual learning processes are promoted mainly by learning tasks, which are assigned to the students individually by the teachers. The students work on these tasks in small groups for 1-2 weeks, trying to solve problems on their own or by asking peers. They should ask the teacher only if this is absolutely necessary. The teachers are monitoring their progress and their motivation level. The individual learning tasks would be quite different from the simple "homework-assignments" that are currently assigned in most cases. They have to be put in a thrilling and motivating context that is related to the students' experiences. This context has to be tailored individually, for instance according gender or personal disabilities.

The basic goal for the students during the task is to acquire competencies in the sense of (Weinert, 1999), who defined competencies as "the cognitive abilities and skills possessed by or able to be learned by individuals that enable them to solve particular problems, as well as the motivational, volitional and social readiness and capacity to use the solutions successfully and responsibly in variable situations." Furthermore, he stressed that competencies may be composed of several facets: ability, knowledge, understanding, skills, action, experience, and motivation.

Apparently, the learning tasks will have to be quite complex, yet separable in different completion stages, called *milestones*. The completion of a milestone has to be reported by the students to the teachers by submitting certain artefacts, for instance a text document, a drawing or a tangible device. Again, the nature of these artefacts may depend from individual attributes of the students. The nature of the learning tasks may vary broadly. From simple summaries of texts, books or websites over the conception of systems, solution of mathematical or physical tasks to the collection of a portfolio that consists of elements of different nature. Regarding the collaborative work on projects, this could be subdivided into different learning tasks for each of the participating working groups.

The timetable of a certain class is only rudimentary. In each subject the students have to attend about one introductory lesson and one presentation meeting per 2 weeks. Assuming that these meetings take about one hour each, it results in about 15 hours per week in a traditional classroom. The rest of weekly working time (about 25 hours in all-day-schools) would be freely disposable for teamwork on individual tasks. The resulting workload of teachers according to this scenario was calculated in (Hubwieser & Böttcher, 2014).

In the terms of the model for self-regulated learning of (Butler & Winne, 1995) (see figure 2), the teacher assigns the tasks, trying to link it to the prerequisite knowledge of the students. The goals are negotiated with the students. Additionally, advice for tactics and strategies is given to the students. Their performance is assessed according to the quality of their products (representing milestones or final outcomes). The external feedback comes from the teacher and from peers.

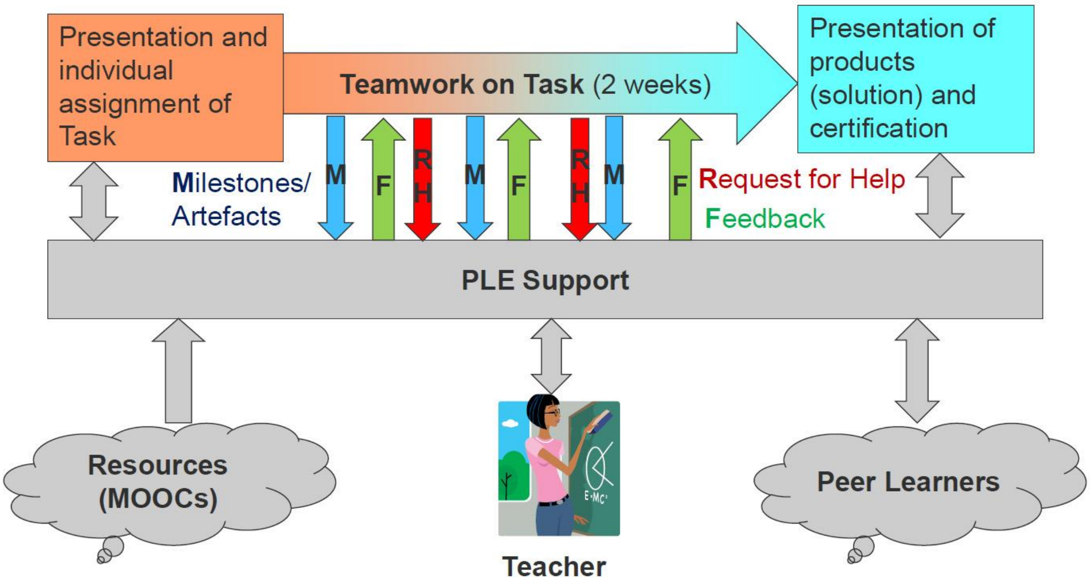


Figure 3. Structure of the task-oriented learning scenario

In our opinion, the task processing should principally (not exclusively, but in most cases) take place in the following steps (see figure 3). The teacher organizes the introduction into a new learning topic. This should provide information about the topic and its relevance for the students, suitable contextualization and activation of students' prerequisite knowledge. Additionally, most intended competencies require certain knowledge that has to be provided by some kind of information input. All provided information has to be accessible (barrier-free) by all students of the course. This requires that the teacher has to provide suitable access for instance to students with visual or auditory disabilities. This activity might take place (at least partly) in a traditional classroom, which is no more necessary for the following steps until the presentation. According to the individual abilities of the students and the respective (also individual) learning objectives, the teacher assigns different suitable, comprehensive tasks to groups of 2-3 students and agrees with the students upon the working schedule and the intended outcome of the task (solution or product). To unburden the teacher from the production, this input should come mostly from suitable prebuilt information sources, which could be analogue, originating from haptic or printed media as well as digital, e.g. from MOOCs. Nevertheless, the teachers still will have to select, restructure, recombine or reprocess original information out of different sources for this purpose. The students work for 1-2 weeks on each task. Thus, each student has to work on many tasks during the school week. As this work is performed without prescribed timetable, students are free to choose on which task they will work at a certain time. Nevertheless, there will be some restrictions, because the team members have to meet or because some specific resources are needed. The tasks are subdivided in milestones. If a milestone is accomplished, the students report this to the teacher. The teacher inspects the working progress and gives feedback. If problems arise, the students try to solve these by themselves as far as possible. For this purpose, supported by a PLE, peers are contacted that are working on the same learning topic Given that this assistance by PLE would not be restricted to the same school, age group and region, there would be plenty of people that could help. As soon as the students have completed a task, they approach the teacher and ask for acceptance of the solution. Depending on the outcomes of the work, the teacher will give feedback, demand further improvements or accept the product for presentation. In this case, the task outcome will be presented at the next presentation meeting of the whole class or course, aiming to communicate the results and seeking final feedback from peers. From time to time, the class is given a summarizing task that aims to integrate the results that have been presented by all class members.

The outcomes of the tasks in combination with the quality of the presentation are certified and graded by the teacher. Potentially, the teacher might demand an additional oral examination about the task, e.g. to assure and test individual learning progress, verify originality, defend plagiarism or to decide different performance levels of the team members. At the end, the outcome of a school career of a certain students will consist of the collection of certifications of tasks. Depending on number, completeness, requirements, quality or performance of this collection, different graduations may be rewarded. By this way, the separation of students in different school types (as Gymnasium, Middle School or Main School in Germany) could become obsolete.

DESIGN OUTLINE

Being well aware of the performance capability of current E-Learning systems, we are not intending to reinvent all their accomplishments. We are regarding Moodle (www.moodle.org; recently ranked as the top E-Learning system among the “Top 100 Tools for Learning 2014”, see c4lpt.co.uk) as the reference system for ELearning platforms and assume that our PLE should have at least the functionality and features of the current Moodle version 2.7. In consequence, the following requirement analysis will only address functionality that is not (or at least not in the desired extent) featured by Moodle.

In principle, according to the central elements of our learning scenario (see section 5), the PLE should provide the following basic functionality: (1) *task management*: support in finding, designing, assigning and managing learning tasks, their processing and their outcomes, (2) *collaboration management*: support in finding and managing peers for collaboration and (3) *media management*: support in finding and managing learning material, media and other artefacts.

MyLearnSpace should be accessible everywhere the students and teachers are able and willing to work on it. On the one hand, this requires the accessibility by desktop or laptop computers as well as by handheld devices as smartphones or tablets. On the other hand, the data have to be stored on a central server that is accessible by all these different devices. In general, the PLE should be accessed via a normal web-browser. Yet, for mobile devices, which are often restricted in the use of internet, an App would be more appropriate.

The need of inclusive education as prescribed by the UN Convention on the Rights of Persons with Disabilities (United Nations, 2006), raises the requirement of being barrier-free as far as any possible. In consequence, all material must be displayed (at least as far as possible) by all perception channels that are technically available (haptic, visual or auditory channels). For this purpose, modern assistive technology has to be integrated or properly interfaced, for instance screen readers or Braille displays. Auditory materials as sound recordings have to be transcribed as far as possible.

The role *management* has to distinguish the different types of users and their access permissions. Up to now, we have identified the following groups, each represented by a specific role, ordered by decreasing permission extent: *Programmers, System Administrators, Curriculum Admins, Teachers, Parents, Students, External Contributors and Advertisers/Sponsors.* The Teacher role has to be explicitly awarded by the supervised students respectively their parents in case of minority.

As any other ICT system, the design of a PLE has to consider issues of information security, privacy and data protection. Depending from the types of data, the demands in this regard vary from “very low” to “high” on a 5-point Likert scale. The lowest value might be valid for the learning content, as long as it is strictly separated from user content, which has the highest demands on data protection. The latter could be eased by using pseudonyms (in agreement with the teachers) instead of the real names of the students. As it concerns the “engine” of the system the administrative content should be treated on the same security level as the most sensible other content.

Basically, the data (content) of the PLE could be subdivided into learning related content (tasks, learning materials and their metadata, communication elements like wiki entries, blog entries or forum threads), user related content (role, personal data, school context, grades etc.) and administrative data (list of tags, structure of metadata, information about the specific hard- and software environment, version control, permission lists of the user groups, etc.).

To support the basic functionality, the *learning content* of *MyLearnSpace* is organized around the “main topics” of the relevant curricula, corresponding more or less with the main chapters of typical curricula or schoolbooks. In the curricula of our home state, we estimated about 10 main topics per subject and year, for instance “irrational numbers and square roots” or “prism and cylinder” in an exemplary curriculum of mathematics. If we assume that the set of main topics is similar for each school subject over all states and countries, we would have to set up about 15*10 = 150 topic rooms over all grades per school subject. The internal knowledge structure of these topics will be represented by a collaborative concept map, see (Hubwieser & Mühling, 2011). The integration of new curricula and topics or of changes in the already integrated ones will be taken over by certain distinguished teachers in the role of *Curriculum Admins*.

Apparently, this structuring principle is helpful regarding the management of learning material. For the task management this seem natural also, because all the tasks are related to learning objectives which are closely related to those “main topics” of the curricula, representing the knowledge part of the learning objectives (see Anderson & Krathwohl, 2001). Finally, the collaboration management will be much easier if the peers would be gathered around the curriculum topics they are seeking help for. Due to the diversity of existing curricula in different countries or school types, the same topic may appear in different grades. Therefore, each topic will be represented technically by a topic tag, which can be assigned to all learning content elements by the users. In consequence, searching content by giving a certain topic tag, the user will find content elements regardless of the country or the school type of its origin. By this way a student of grade 8 in Gymnasium could collaborate with a peer that attends grade 9 of Middle School in a different state.

To organize the learning and task content, we will apply our Educational Ontology (Hubwieser & Bitzl, 2010) in a slightly adopted form. Its core part is displayed in figure 4.

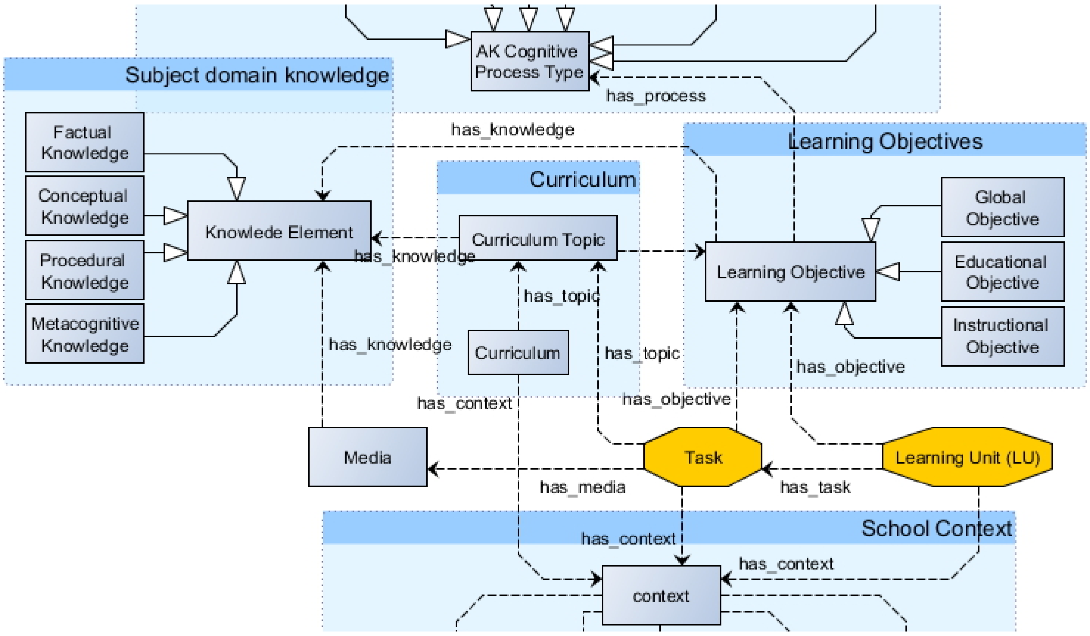


Fig. 4. The Educational Ontology

As the students are working on topics of (about 15) different subjects simultaneously, we propose to organize the student's view on the learning and collaboration content in *Topic Rooms*, one per "main topic" of each curriculum. At the end, all relevant (according to rating systems) learning content elements that are available for the student will be displayed at the respective Topic Room, arranged by their access functions, see figure 5.

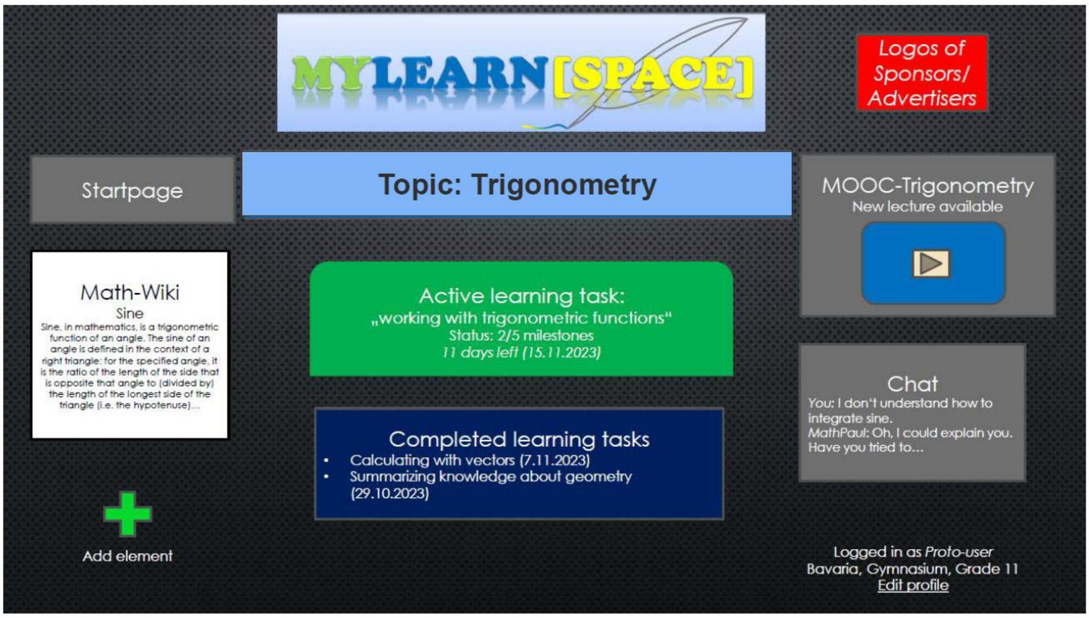


Fig. 5. Exemplary Topic Room for Square Roots (see Hubwieser & Mühling, 2012)

The preferable view on the task content will be the Task Room, both for teachers and students. Basically, the task view will present the individual progress in the task processing, reporting and displaying the status of milestones and referring to submitted artefacts and feedback documents. While the student's view is (naturally) restricted to his/her own tasks, the teacher has access to an overview of the task processes of all his students (see figure 6).

Subject	Grade	Group	Topic
Math	8	ALL	ALL

Name	Prename	Topic	Task	Difficulty	Credits	MS1	MS2	MS3	Requests for Help	Certification
Miller	Anna	SQRT	1.12	HA	12	OK	OD		3	NO
Smith	James	SQRT	1.14	EA	5	OK	OK	OD	2	A+
Hunt	Sophie	PYR	2.1	MI	10	OD	OD	OD	0	NO

Figure 6. Typical teachers' view on task processing status of their students.

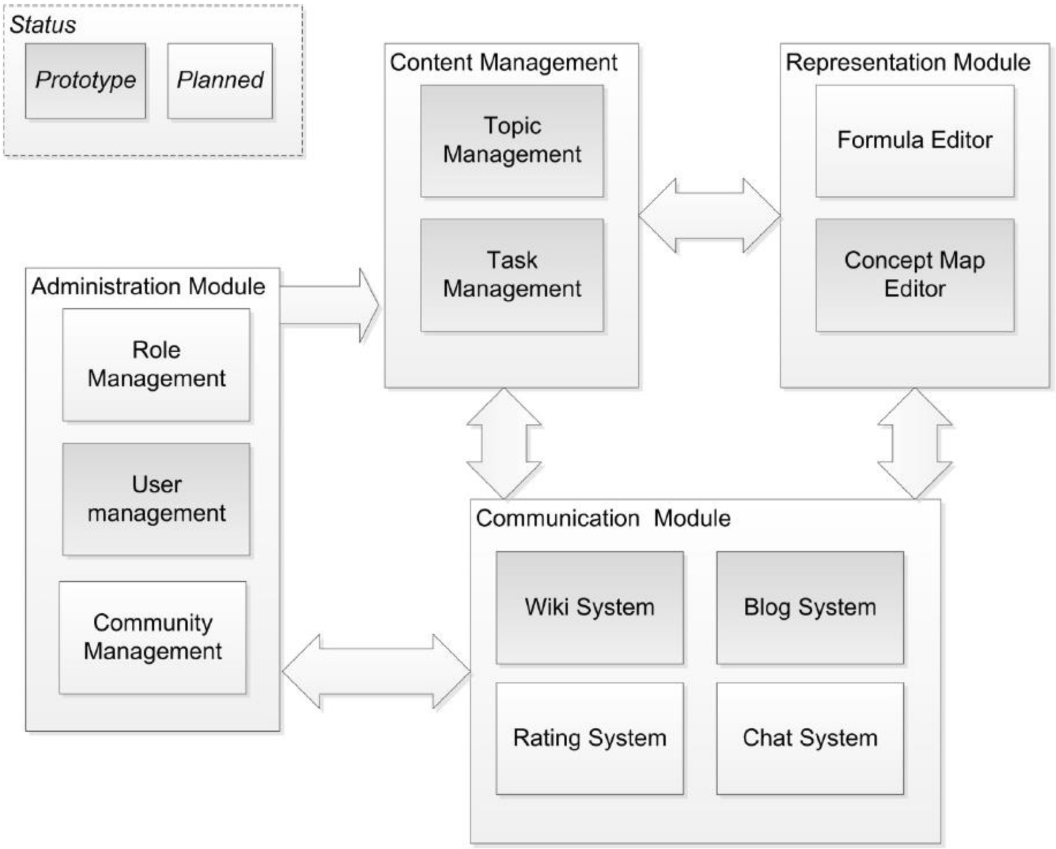


Figure 7. Modules and development status of MyLearnSpace.

According to the accessibility requirements, 3-tier architecture seems appropriate. The presentation layer will be represented by ordinary Web-Browsers, eventually supported with some local logic, implemented e.g. by JavaScript. Alternatively, we will design specific Apps for the most popular platforms of handheld devices (e.g. iOS and Android). The functionality of the PLE will be located on central servers, similar to the data management.

Basically, *MyLearnSpace* comprises the following modules, which are partly implemented in the prototypes already: (1) *administration*, managing users, roles, access permissions, communities, (2) *content*, managing curricula, topics, tasks, documents and artefacts, (3) *representation*, comprising editors for specific content as formula or concept maps, and (4) *communication*, comprising subsystems for wikis, rating, blogs and forums.

CONCLUSION AND FUTURE WORK

Although the anticipated learning scenario may already exist in some very progressive schools, it will definitely take substantial time, resources and efforts to put it into practice in mainstream schools. Yet, we are convinced that all these changes will take place, finally forced by the prescribed inclusion of all students in regular classes. Therefore we are convinced that it is high time to start the development of an ICT system that supports future learning scenarios out of several reasons. The most important one is that these scenarios will not make the breakthrough without supporting software, due to the increased workload of teachers or because of missing support for collaboration with peers. Another reason is that it offers great research perspectives in empirical pedagogics as well as in computer science, even if our learning scenario is only realized in some avantgarde-classes.

At the time this article is written, another group of students is just finishing its practical work on integrating the already existing prototype modules of MyLearnSpace. Subsequently another Bachelors' Thesis will implement this integrated prototype at an exemplary grammar school in Bavaria. Based on the evaluation of this experiment, we will proceed in the development of our Personal Learning Environment for self-determined, active and social learning.

REFERENCES

Anderson, L. W., & Krathwohl, D. R. (2001). A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives (Abridged ed.,[4. print]). New York: Longman. Retrieved from http://www.gbv.de/dms/bowker/toc/9780321084057.pdf

Attwell, G. (2007). Personal Learning Environments-the future of eLearning? eLearning papers, 2(1), 1–8.

Böttcher, J. (2014). Development of a Personalized Platform for Content-Driven Social Learning (Bachelor's Thesis). Technische Universität München, München. Retrieved from http://www.ddi.edu.tum.de/publikationen/studienarbeiten/abgeschlossenestudienarbeiten/2014/boettcher-j-bsc/

Butler, D. L., & Winne, P. H. (1995). Feedback and Self-Regulated Learning: A Theoretical Synthesis. Review of Educational Research, 65(3), 245–281. doi:10.3102/00346543065003245

Caine, R. N., & Caine, G. (1990). Understanding a brain-based approach to learning and teaching. Educational Leadership, 48(2), 66–70.

Caine, R. N., & Caine, G. (1991). Making connections: Teaching and the human brain. Cobb, P. (1994). Where is the mind? Constructivist and sociocultural perspectives on mathematical development. Educational Researcher, 23(7), 13–20.

Cooper, S., Grover, S., & Simon, B. (2014). Building a Virtual Community of Practice for K-12 CS Teachers. Commun. ACM, 57(5), 39–41. doi:10.1145/2594456

Crouch, C. H., & Mazur, E. (September 2001). Peer Instruction: Ten Years of Experience and Results. Am. J. Phys., pp. 970–977.

Deci, E. L., & Ryan Richard M. (1985). Intrinsic motivation and self-determination in human behavior. New York: Plenum Press.

Dillenbourg, P. (1999). What do you mean by collaborative learning? Collaborativelearning: Cognitive and Computational Approaches, 1–19.
Douglas, G., Corcoran, C., & Pavey, S. (2007). The role of the who icf as a framework to interpret barriers and to inclusion: visually impaired people's views and experiences of personal computers. British Journal of Visual Impairment, 25(1), 32–50.
Freinet, C. (1964). Les invariants pedagogiques: Code pratique d'Ecole Moderne. Bibliothèque de l'école moderne, (25).
Gräsel, C., Bruhn, J., Mandl, H., Fischer, F., & others. (1997). Lernen mit Computernetzen aus konstruktivistischer Perspektive. Unterrichtswissenschaft, 25(1), 4–18.
Hubwieser, P., & Bitzl, M. (2010). Modeling Educational Knowledge: Supporting the Collaboration of Computer Science Teachers. In J. L. G. Dietz (Ed.), Proceedings of the International Conference on Knowledge Engineering and Ontology Development (KEOD) 2010. 25.-28. October 2010. Valencia, Spain.
Hubwieser, P., & Böttcher, J. (2014). Personal Learning Environments for SelfDetermined, Active and Social Learning. In L. Gómez Chova, A. López Martínez, & I. Candel Torres (Eds.), ICERI2014 Proceedings. 7th International Conference of Education, Research and Innovation. November 17th-19th, 2014 - Seville, Spain (pp. 1024–1034). IATED Digital Library.
Hubwieser, P., & Mühling, A. (2011). Knowpats: Patterns of Declarative Knowledge: Searching Frequent Knowledge Patterns about Object-orientation. In Proceedings of the International Conference on Knowledge Discovery and Information Retrieval. Paris, France, October 26 - 29 (pp. 358–364). ScitePress.
Hubwieser, P., & Mühling, A. (2012). A Social Network for Learning: Supporting Collaborative Learning based on the Ontology for Educational Knowledge. In Proceedings of the 4th International Conference on Knowledge Management and Information Sharing. Barcelona, ,Spain, October 4-7 (pp. 298–301). ScitePress Digital Library. Retrieved from http://www.scitepress.org/DigitalLibrary/
Kreijns, K., Kirschner, P. A., & Jochems, W. (2003). Identifying the pitfalls for social interaction in computer-supported collaborative learning environments: a review of the research. Computers in human behavior, 19(3), 335–353.
Krumm, J. (2010). Ubiquitous computing fundamentals. Boca Ragon: Chapman & Hall/CRC Press. Liccardi, I., Ounnas, A., Pau, R., Massey, E., Kinnunen, P., Lewthwaite, S., . . . Sarkar, C. (2007). The Role of Social Networks in Students' Learning Experiences. In: ITiCSE-WGR '07, Working Group Reports on ITiCSE on Innovation and Technology in Computer Science Education (pp. 224–237). New York and NY and USA: ACM. Retrieved from http://doi.acm.org/10.1145/1345443.1345442
Marshall, L. (2013). Leveraging Online Courses to Increase Student Success in a Computer Science Degree. In : CSERC '13, Proceedings of the 3rd Computer Science Education Research Conference on Computer Science Education Research (pp. 1:9-1:16). Open Univ. and Heerlen and The Netherlands and The Netherlands: Open Universiteit, Heerlen. Retrieved from http://dl.acm.org/citation.cfm?id=2541917.2541918
Mell, P. & Grance, T. (2011). The NIST Definition of Cloud Computing: Recommendations of the National Institute of Standards and Technology (Special Publication No. 800-145). Gaithersburg, MD.
Montessori, M. (1993). Formazione dell'uomo. I libri di Maria Montessori. [Milano]: Garzanti.
Muise, K., & Wakkary, R. (2010). Bridging Designers' Intentions to Outcomes with Constructivism. In : DIS '10, Proceedings of the 8th ACM Conference on Designing Interactive Systems (pp. 320–329). New York and NY and USA: ACM. Retrieved from http://doi.acm.org/10.1145/1858171.1858229

Niemiec, C. P., & Ryan, R. M. (2009). Autonomy, competence, and relatedness in the classroom: Applying self-determination theory to educational practice. Theory and Research in Education, 7(2), 133–144.
OECD. Centre for Educational Research and Innovation (CERI) - The OECD Schooling Scenarios in Brief. Retrieved from http://www.oecd.org/edu/school/centreforeducationalresearchandinnovationceritheoecdschoolingscenariosinbrief.htm
OECD. (2004). Background OECD Papers: The Schooling Scenarios: International Schooling for Tomorrow Forum. June 6-8th. Toronto, CA. Retrieved from http://www.oecd.org/site/schoolingfortomorrowknowledgebase/futuresthinking/scenarios/scenariosforthefutureofschooling.htm
Porter, L., Bailey Lee, C., & Simon, B. (2013). Halving Fail Rates Using Peer Instruction: A Study of Four Computer Science Courses. In : SIGCSE '13, Proceeding of the 44th ACM Technical Symposium on Computer Science Education (pp. 177–182). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2445196.2445250
Rasmus, D. W. (2008). Scenario planning and the future of education. Innovate, 4(5). Retrieved from http://www.innovateonline.info/index.php?view=article&id=555
Reinmann-Rothmeier, G., & Mandl, H. (1996). Lernen auf der Basis des Konstruktivismus. Computer und Unterricht 23, S. 41–44.
Ryan, R. M., & Deci, E. L. (2000). Intrinsic and Extrinsic Motivations: Classic Definitions and New Directions. Contemporary Educational Psychology, 25, 54–67.
United Nations. (2006). Convention on the Rights of Persons with Disabilities.
Weinert, F. E. (1999). Concepts of competence: Definition and selection of competencies ([Elektronische Ressource]). [S.l.]: OFS [u.a.]. Weiser, M. (1991). The Computer for the 21st Century. Scientific American, (September), 94–104.
Werdmuller, B., Tosh, D., Files, F., & Free, P. (2005). Elgg-A personal learning landscape. TESL-EJ, 9(2). Retrieved from http://tesl-ej.org/ej34/m1.html
Wulf, J., Blohm, I., Leimeister, J., & Brenner, W. (2014). Massive Open Online Courses. WIRTSCHAFTSINFORMATIK, 56(2), 127–130. doi:10.1007/s11576-014-0405-7.

Peter Hubwieser was teaching Mathematics, Physics and Informatics at Bavarian Gymnasiums from 1985 to 2002. In 1995 he received his doctorate in Theoretical Physics at the LudwigMaximilians-Universität München. From 1994 to 2002 he was working on the implementation of Informatics as a mandatory subject at secondary schools in Bavaria. Since June 2002 he is associate professor at the Faculty of Informatics of the Technical University of Munich

Application of Intelligent Technologies in Computer Engineering Education

Eugenijus Kurilovas

Vilnius University Institute of Mathematics and Informatics, Vilnius, Lithuania

Abstract

We believe that future school means personalisation plus intelligence. Learning personalisation means creating and implementing personalised learning scenarios based on recommender system suitable for particular learners according to their personal needs. Educational intelligence means application of intelligent technologies and methods enabling personalised learning to improve learning quality and efficiency. Lithuanian Intelligent Future School (IFS) project is aimed at implementing both learning personalisation and educational intelligence. In personalised learning, first of all, integrated learner profile (model) should be implemented, consisting of (1) Selecting suitable taxonomies (models) of learning styles, e.g., Felder & Silverman, Honey & Mumford, and VARK style (Fleming); (2) Creating integrated learning style model which integrates characteristics from several models. Dedicated psychological questionnaires should be applied here; (3) Creating open learning style model; (4) Using implicit (dynamic) leaning style modelling method; and (5) Integrating the rest features in the student profile (knowledge, interests, goals etc.). After that, ontologies-based personalised recommender system should be created to suggest learning components (learning objects, activities, methods, tools, apps etc.) suitable to particular learners according to their profiles. Thus, personalised learning scenarios could be created for particular learners for each topic according to curriculum / study programme. A number of intelligent technologies should be applied to implement IFS approach, e.g. ontologies, recommender system, intelligent agents, decision support systems to evaluate quality and suitability of the learning components etc. Experience of IFS implementation by applying mobile learning scenarios in Lithuania has shown that this approach is effective in computer engineering education.

Keywords

Intelligent technologies, education, personalisation, learning styles, learner profile, ontologies recommender system, intelligent agents, decision support systems

1. Introduction

What learning content, methods and technologies are the most suitable to achieve better learning quality and efficiency? In Lithuania, we believe that there is no correct answer to this question if we don't apply personalised learning approach. We strongly believe that "one size fits all" approach doesn't longer work in education.

It means that, first of all, before starting any learning activities, we should identify students' personal needs: their preferred learning styles, knowledge, interests, goals etc. After that, teachers should help students to find their suitable (optimal) learning paths: learning methods, activities, content, tools, mobile applications etc. according to their needs. But, in real schools practice, we can't assign personal teacher for each student. This should be done by intelligent technologies. Therefore, we believe that future school means personalisation plus intelligence.

In this paper, Lithuanian Intelligent Future School (IFS) project is presented aimed at implementing both learning personalisation and educational intelligence.

The rest of the paper is organised as follows: Section 2 describes the related works, Section 3 presents Intelligent Future School concept, some examples of real-life applications of intelligent technologies in schools are provided in Section 4, Intelligent Future School implementation vision is presented in Section 5. And Section 6 concludes the paper.

2. Related work

IFS project idea is based on previous EU-funded “Future Classroom” projects co-ordinated by European Schoolnet (EUN), a network of 31 European Ministries of Education. These projects are iTEC, LSL and CCL.

2.1. iTEC (Innovative Technologies for Engaging Classrooms), 2010-2014

How did the iTEC approach impact on learners and learning (key findings):

1. Teachers perceived that the iTEC approach developed students’ 21st century skills, notably independent learning; critical thinking, real world problem solving and reflection; communication and collaboration; creativity; and digital literacy. Their students had similar views.
2. Student roles in the classroom changed; they became peer assessors and tutors, teacher trainers, co-designers of their learning and designers/producers.
3. Participation in classroom activities underpinned by the iTEC approach impacted positively on students’ motivation.
4. The iTEC approach improved students’ levels of attainment, as perceived by both teachers (on the basis of their assessment data) and students.

2.2. LSL (Living Schools Lab), 2012-2014

With the participation of 15 partners, including 12 education ministries, LSL project promoted a whole-school approach to ICT use, scaling up best practices in the use of ICT between schools with various levels of technological proficiency.

The participating schools were supported through peer-exchanges in regional hubs, pan-European teams working collaboratively on a number themes and a variety of opportunities for teachers' ongoing professional development.

Observation of advanced schools in 12 countries produced a report and recommendations on the mainstreaming of best practice, and the development of whole-school approaches to ICT.

2.3. CCL (Creative Classrooms Lab, CCL), 2013-2015

CCL brought together teachers and policy-makers in 8 countries to design, implement and evaluate 1:1 tablet scenarios in 45 schools. CCL produced learning scenarios and activities, guidelines and recommendations to help policy-makers and schools to take informed decisions on optimal strategies for implementing 1:1 initiatives in schools and for the effective integration of tablets into teaching and learning.

The 1:1 computing paradigm is rapidly changing, particularly given the speed with which tablets from different vendors are entering the consumer market and beginning to impact on the classroom. Over the next 2-3 years policy makers will face some difficult choices: How to invest most efficiently in national 1:1 computing programmes? What advice to give to schools that are integrating tablets?

To address these challenges, CCL carried out a series of policy experimentations to collect evidence on the implementation, impact and up-scaling of 1:1 pedagogical approaches using tablets. Lessons drawn from the policy experimentations also:

- Provide guidelines, examples of good practice and a training course for schools wishing to include tablets as part of their ICT strategy.

- Support capacity building within Ministries of Education and regional educational authorities and encourage them to introduce changes in their education systems.
- Enable policy makers to foster large-scale uptake of the innovative practice that is observed during the project.

3. Intelligent Future School Concept

Ifs concept is based on iTEC school innovation maturity model.

3.1. iTEC schools innovation maturity model

The model shows a number of progressive stages of innovation maturity of an institution, e.g. school. As educational institutions move from one stage to the next in the direction of the arrow, the innovation maturity of the institution progresses, e.g. the implementation of a scenario that moves an institution from the ‘Exchange’ stage of the model to the ‘Enrich’ stage would be defined as innovative in that institution’s context.

The model consists of 5 consecutive levels as follows:

Table 1: iTEC school innovation maturity model

Level	Description
5 Empower: Redefinition & innovative use	• Technology supports new learning services that go beyond institutional boundaries. • Mobile and locative technologies support ‘agile’ teaching and learning. • The learner as a ‘co-designer’ of the learning journey, supported by **intelligent** content and analytics.
4 Extend: Network redesign & embedding	• Ubiquitous, integrated, seamlessly connected technologies support learner choice and **personalisation** beyond the classroom. • Teaching and learning are distributed, connected and organised around the learner. • Learners take control of learning using technology to manage their own learning.
3 Enhance: Process redesign	• Teaching and learning redesigned to incorporate technology, building on research in learning and cognition. • Institutionally embedded technology supports the flow of content and data, providing an integrated approach to teaching, learning and assessment. • The learner as a ‘producer’ using networked technologies to model and make.
2 Enrich: Internal Coordination	• Technology used interactively to make differentiated provision within the classroom. • Technology supports a variety of routes to learning. • The learner as a ‘user’ of technology tools and resources
1 Exchange: Localised use	• Technology is used within current teaching approaches. • Learning is teacher-directed and classroom-located. • The learner as a ‘consumer’ of learning content and resources.

Using self-assessment activity based on this model, teachers and schools principals can identify the organisation’s current position on the innovation maturity model.

The highest levels of the model require learning personalisation and application of intelligent technologies. Thus, future school means personalisation plus intelligence.

3.2. IFS implementation stages

IFS implementation stages are based on iTEC school innovation maturity model. They are as follows:

1. Creating learners' models (profiles) based on their learning styles and other particular needs
2. Interconnecting learners' models with relevant learning components (learning content, methods, activities, tools, apps etc.)
3. Creating corresponding ontologies
4. Creating intelligent agents and recommender systems
5. Creating and implementing personalised learning scenarios (e.g. in STEM – Science, Technology, Engineering and Mathematics – subjects)
6. Creating educational multiple criteria decision making models and methods

3.3. Personalisation: Creating students' profiles

User model is representation of information about an individual user.

User modelling is the process of creating and maintaining an up-to-date user model, by collecting data from various sources, which may include: (1) implicitly observing user interaction and (2) explicitly requesting direct input from the user (Brusilovsky & Millan, 2007).

Regarding the information contained in the user model, there are identified six features: (1) knowledge, (2) interests, (3) goals, (4) background, (5) individual traits, and (6) context of work.

The learning style of the student also started to be taken into account, as being one of the individual traits that play an important role in learning. Learning style designates everything that is characteristic to an individual when she/he is learning, i.e. a specific manner of approaching a learning task, the learning strategies activated in order to fulfil the task.

Learning styles represent a combination of characteristic cognitive, affective and psychological factors that serve as relatively stable indicators of how a learner perceives, interacts with, and responds to the learning environment (Keefe, 1979).

Learning styles model systems differ in several aspects: underlying learning style model, diagnosing method (implicit or explicit), modelling techniques (rule-based approach, data mining, machine learning techniques), number of modelled student characteristics besides learning preferences (knowledge level, goals) and the type, size and conclusions of the reported experiments.

In what follows we will focus on the methods used for learner modelling and we classify the systems in two categories: (1) those that use questionnaires for identifying the learning style and (2) those that use students' observable behaviour.

The first adaptive educational systems that dealt with learning styles as adaptation criterion relied on the measuring instruments associated to the learning style models for diagnosing purposes. The main advantage of this method is its simplicity: the teacher / researcher only has to apply a dedicated psychological questionnaire, proposed by the learning style model creators. Based on the students' answers to the questions, a preference towards one or more of the learning style dimensions can be inferred.

The main disadvantages of this questionnaire-based approach are:

- some of the measuring instruments used could not demonstrate internal consistency, test–retest reliability or construct and predictive validity, so they may not be totally reflective of the way a particular student learns (Coffield *et al.,* 2004);

- it implies a supplementary amount of work from the part of the student, who has to fill in questionnaires at the beginning of the course (which sometimes may include over 100 questions, as in case of the Herrmann's Whole Brain Model (Herrmann, 1996));
- it can be easily "cheated" by the students, who may choose to skip questions or give wrong answers on purpose;
- there can be non-intentional influences in the way the questions are formulated, which may lead the students to give answers perceived as "more appropriate";
- it is difficult to motivate the students to fill out the questionnaires; especially if they are too long and the students are not aware of the importance or the future uses of the questionnaires, they may tend to choose answers arbitrarily instead of thinking carefully about them;
- it is static, so the student model is created at the beginning of the course and stored once and for all, without the possibility to be updated.

A method of improving this approach is to give the student the possibility to modify her/his own profile, if she/he considers that the one inferred from the questionnaire results is not appropriate (does not correspond to the reality). This is called an "open model" (scrutable and modifiable) approach and it is used either in conjunction with the questionnaires or instead of them. This direct access of students to their own learner model has several advantages: it provides an increased learner control, it helps the learners develop their metacognitive skills and it also offers an evaluation of the quality of the model created by the system (Kay, 2001).

The main disadvantages of this approach are that it increases the cognitive load of the student and that it must rely on the self-evaluation of a student who might not be aware of her/his learning style.

There is also a second category of systems, which use an implicit and/or dynamic modelling method.

Three different approaches have been identified in this respect:

1. analyse the performance of the students at evaluation tests – a good performance is interpreted as an indication of a style that corresponds to the one currently estimated and employed by the system, while a bad performance is interpreted as a mismatched learning style and triggers a change in the current learner model;
2. ask the students to provide feedback on the learning process experienced so far and adjust the learner model accordingly;
3. analyse the interaction of the students with the system (browsing pattern, time spent on various resources, frequency of accessing a particular type of resource etc.) and consequently infer a corresponding learning style.

Sometimes, these systems use a mixed modelling approach: they first use the explicit modelling method for the initialisation of the learner model and then the implicit modelling method for updating and improving the learner model.

Most of the existing systems treat learning styles in isolation of the rest of the features in the student profile (knowledge, interests, goals). The ideal is to integrate all these features in a more comprehensive and representative learner profile.

Concluding, IFS approach on creating learner model consists of the following steps:

- Selecting good taxonomies (models) of learning styles, e.g., (Felder & Silverman, 1988), (Honey & Mumford, 2000), the VARK style (Fleming, 1995)
- Creating integrated learning style model (ILSM) which integrates characteristics from several models. Dedicated psychological questionnaire(s)

- Creating open learning style model
- Using implicit (dynamic) learning style modelling method
- Integrating the rest features in the student profile (knowledge, interests, goals)

Example: the system presented in (Sangineto et al., 2008) – is based on Felder-Silverman learning style model, and uses fuzzy values to estimate the preference of the student towards one of the four categories (Sensing-Intuitive, Visual-Verbal, Active-Reflective, Sequential-Global). Initially, the system offers to the learner the possibility to use the Soloman and Felder's psychological test or to directly set the values of the category types, choosing an estimated value for each category (using a slider-based interface).

3.4. Creating recommender system

To create recommender system, it's necessary, first of all, interconnect learners' models with relevant learning components (learning content, methods, activities, tools, apps etc.), and then create corresponding ontologies.

Some examples of these steps are provided below.

3.4.1. Interconnections

Table 2: Interconnection between learning styles, learning activities, teaching methods, and learning object types (Kurilovas *et. al.,* 2014)

Learning styles (Honey and Mumford, 1992)	**Preferred learning activities**	**Suitable teaching / learning methods (iCOPER D3.1 (2009))**	**Suitable learning object types (LRE APv4.7 (2011))**
Activists are those people who learn by doing. Have an open-minded approach to learning, involving themselves fully and without bias in new experiences	Brainstorming, problem solving, group discussion, puzzles, competitions, and role-play.	3 Times 3 Things Learnt, Active Learning, Blogging, Brainstorming and Reflection, Competitive Simulation, Constellation, Creative Workshops, Creation of Personalized Learning Environments, Cultura, Cultural Awareness, E-Portfolio, Exercise Unit, Games Genre, Presenting Homework, Image Sharing, In-class Online Discussion, Mini Conference, Modeling, Online Reaction Sheets, Online Training, Peer Assessment, Process-based Assessment, Process Documentation, Project-based Learning, Resource-based Analysis, Role Play, Student Wiki Collaboration, World Café, WebQuest	Application, Assessment, Broadcast, Case study, Drill and practice, Educational game, Enquiry-oriented activity, Experiment, Exploration, Glossary, Open activity, Presentation, Project, Reference, Role play, Simulation, Tool, Website
Reflectors learn by observing and thinking about what happened. They prefer to stand back and view experiences from a number of different perspectives, collecting data and taking the time to work towards an appropriate conclusion	Paired discussions, self-analysis questionnaires, personality questionnaires, time out, observing activities, feedback from others, coaching, and interviews.	3 Times 3 Things Learnt, Blogging, Brainstorming and Reflection, Constellation, Creative Workshops, Creation of Personalized Learning Environments, Cultura, Cultural Awareness, E-Portfolio, Exercise Unit, Presenting Homework, Image Sharing, In-class Online Discussion, Listen–Do–Reflect, Ten-Plus-Two Variation, Mini Conference, Modeling, Online Reaction Sheets, Online Training, Peer-to-Peer Teaching, Peer Assessment, Process-based Assessment, Process Documentation, Resource-based	Application, Assessment, Broadcast, Case study, Demonstration, Glossary, Guide, Presentation, Reference, Textbook, Website

		Analysis, Student Wiki Collaboration, World Café, WebQuest	
Pragmatists need to be able to see how to put the learning into practice in the real world. Abstract concepts and games are of limited use unless they can see a way to put the ideas into action in their lives. They are experimenters, trying out new ideas, theories and techniques to see if they work	Time to think about how to apply learning in reality, case studies, problem solving, and discussion.	3 Times 3 Things Learnt, Active Learning, Blogging, Brainstorming and Reflection, Competitive Simulation, Constellation, Creative Workshops, Creation of Personalized Learning Environments, Creation of Personalized Learning Environments, Cultura, Cultural Awareness, E-Portfolio, Exercise Unit, Games Genre, Presenting Homework, Image Sharing, In-class Online Discussion, Listen–Do–Reflect, Ten-Plus-Two Variation, Mini Conference, Modeling, Online Reaction Sheets, Online Training, Peer-to-Peer Teaching, Peer Assessment, Process-based Assessment, Process Documentation, Project-based Learning, Resource-based Analysis, Role Play, Student Wiki Collaboration, World Café, WebQuest	Application, Assessment, Broadcast, Case study, Course, Drill and practice, Educational game, Enquiry-oriented activity, Experiment, Exploration, Glossary, Guide, Open activity, Presentation, Project, Reference, Role play, Simulation, Tool, Website
Theorist learners like to understand the theory behind the actions. They need models, concepts and facts in order to engage in the learning process. Prefer to analyse and synthesise, drawing new information into a systematic and logical 'theory'.	Models, statistics, stories, quotes, background information, and applying theories.	3 Times 3 Things Learnt, Blogging, Constellation, Creative Workshops, Creation of Personalized Learning Environments, Cultura, Cultural Awareness, E-Portfolio, Exercise Unit, Presenting Homework, Image Sharing, In-class Online Discussion, Listen–Do–Reflect, Ten-Plus-Two Variation, Mini Conference, Modeling, Online Reaction Sheets, Online Training, Peer-to-Peer Teaching, Peer Assessment, Process-based Assessment, Process Documentation, Resource-based Analysis, Student Wiki Collaboration, World Café, WebQuest	Application, Assessment, Broadcast, Course, Demonstration, Glossary, Guide, Presentation, Reference, Textbook, Website

Figure 1 shows interconnection between teaching / learning methods (M) and learning object types (T) for the case of Problem Solving activity's A1 sub-activity SA2 (Brainstorming):

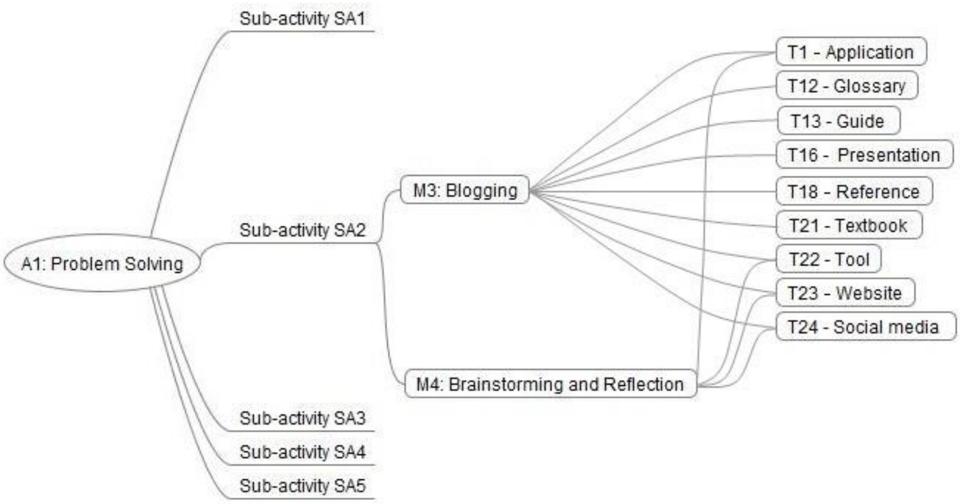


Fig. 1: Interconnection between teaching / learning methods (M) and learning object types (T) for SA2 sub-activity (Kurilovas et. al., 2014)

Table 2: Interconnection of Activists Brainstorming learning activity with suitable apps and learning object types (Kurilovas, 2014)

iOS	Android	iOS / Android	Types of LOs
Idea Sketch – lets you easily draw a diagram – mind map, concept map, or flow chart - and convert it to a text outline, and vice versa. You can use Idea Sketch for anything, such as brainstorming new ideas, illustrating concepts, making lists and outlines, planning presentations, creating organizational charts, and more!	**Mindjet for Android** – rated as one of the best mind mapping apps for Android. Create nodes and notes, add images of your own or icons provided, and add attachments and hyperlinks. Sync to your Dropbox	**Mind Mapping** – lets you create, view and edit mind maps online or offline and lets the app synch with your online account whenever connected. You can share mind maps directly from the device, inviting users via email. You can add icons, colours and styles, view notes, links and tasks and apply map themes, drag and drop and zoom	Application, Broadcast, Enquiry-oriented activity, Glossary, Open activity, Presentation, Reference, Role play, Simulation, Tool, Website

3.4.2. Ontologies

Ontology example 1: Query for finding suitable learning activities (i.e. “Problem Solving” activity could be found using “Blogging” teaching / learning method):

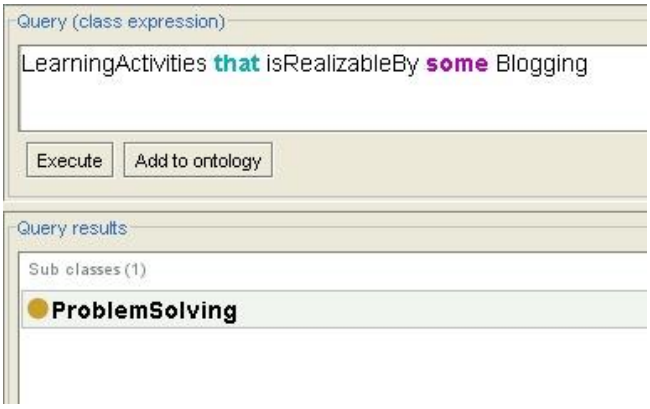


Fig. 2: Query for finding suitable learning activities by methods (Kurilovas et. al., 2014)

Ontology example 2: Query for finding suitable teaching / learning methods (i.e. “Blogging” and “Brainstorming and Reflection” teaching / learning methods could be found using suitable Application learning object type):

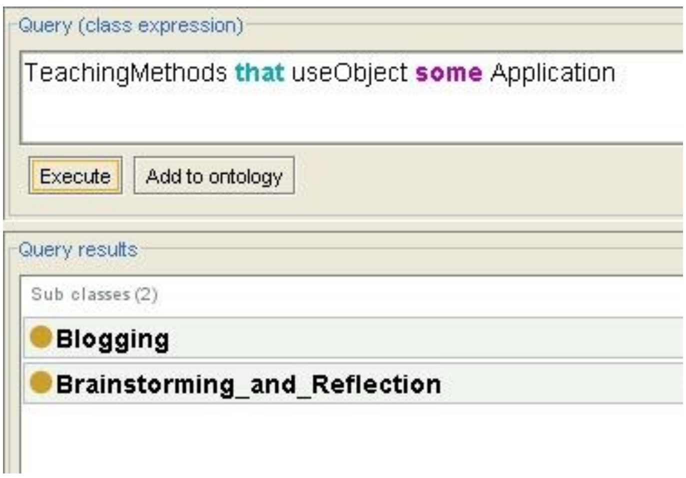


Fig. 3: Query for finding suitable teaching / learning methods by learning object types (Kurilovas *et. al.,* 2014)

Ontology example 3: Query for finding suitable iOS applications for “Brainstorming” sub-activity, i.e. “Idea Sketch” and “Mind Maping”

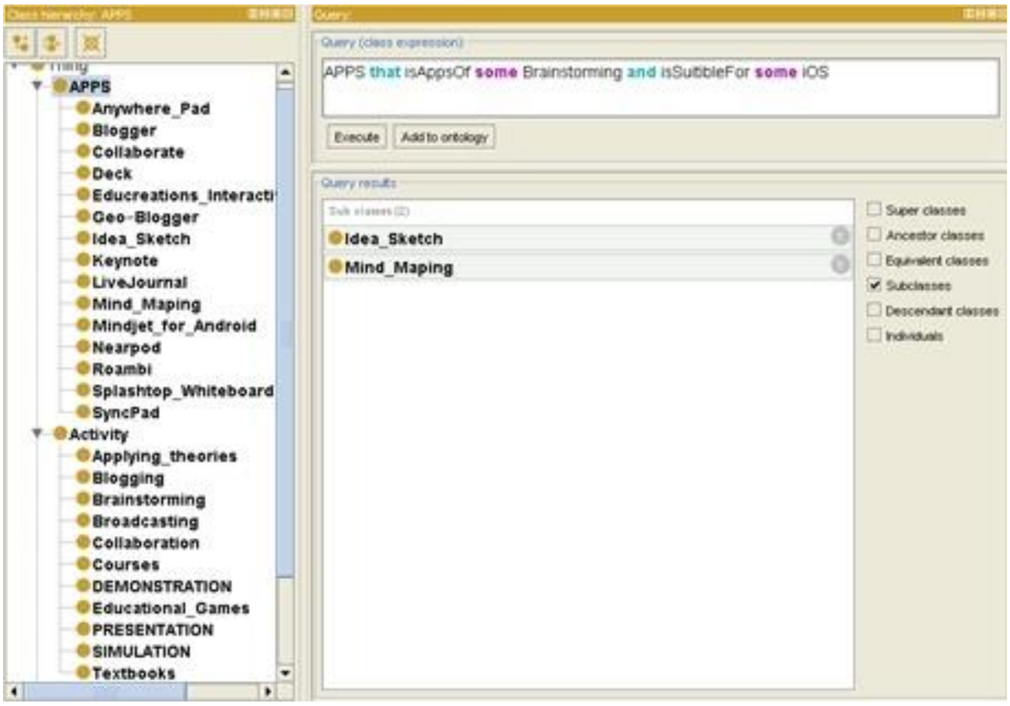


Fig. 4: Query for finding suitable iOS applications for “Brainstorming” sub-activity

Ontology example 4: Query for finding suitable Android applications for “Brainstorming” sub-activity, i.e. “Mind Mapping” and “Mindjet for Android”

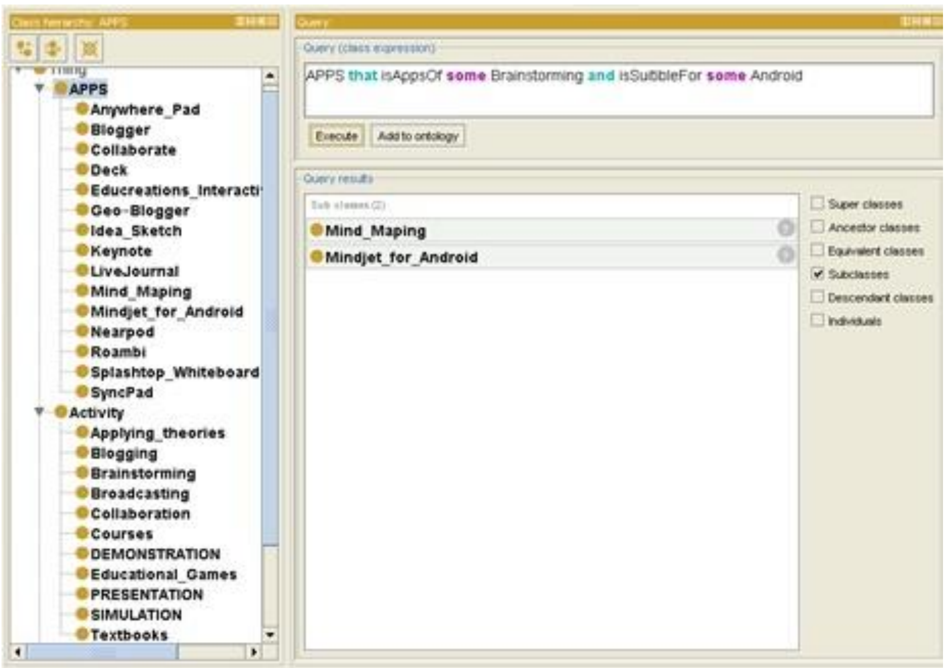


Fig 5: Query for finding suitable Android applications for “Brainstorming” sub-activity

3.4.3. Recommender system

After interconnecting learners' models with relevant learning components and creating corresponding ontologies, it's possible to start creating the prototypes of recommender systems.

One of the examples of recommender system prototypes created is presented in Fig. 6. This prototype fully implements created method to integrate Web 2.0 tools into learning activities to personalise learning according to VARK learning styles model.

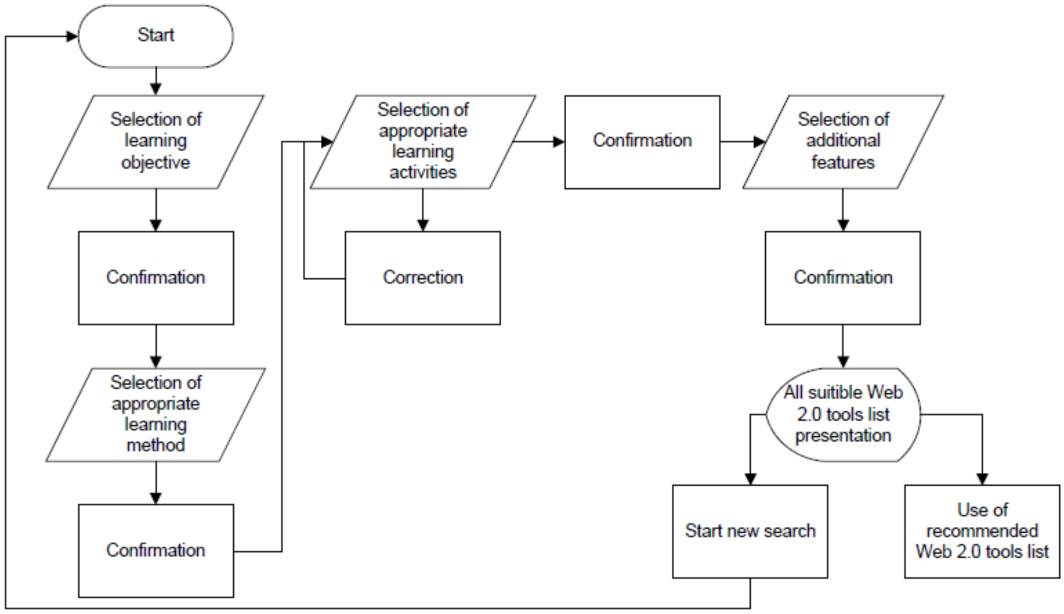


Fig. 6: Recommender system prototype integrating Web 2.0 tools into learning activities to personalise learning according to VARK learning styles model (Juskeviciene & Kurilovas, 2014)

4. Examples of Performed Research

Several research studies in STEM (Science, Technology, Engineering and Mathematics) and, more concretely, in computer engineering education have been conducted to implement IFS approach in Lithuania. These studies are as follows:

Kurilovas *et. al.* (2015) presented CCL research results on implementing mobile learning activities using tablets in Lithuania.

Creation of Web 2.0 tools ontology to improve learning is presented by Kurilovas & Juskeviciene (2015).

Kurilovas *et. al.* (2014) analysed interconnection of (Honey & Mumford learning styles with appropriate learning activities, teaching/learning methods and learning objects' types.

Research on tablets applications for mobile learning activities is presented in (Kurilovas, 2014).

Kurilovas & Serikoviene (2013) have presented methodology on e-textbooks quality model and evaluation.

The novel method to evaluating the quality of personalised learning scenarios has been presented in (Kurilovas & Zilinskiene, 2012).

5. Intelligent Future School Implementation Vision

Based on Section 3.2, real-life IFS implementation vision is as follows:

1. Collaboration agreements between Vilnius University and (20 pilot) schools on IFS implementation
2. Creating joint expert group on creating interconnections and intelligent agents
3. R&D, creation of technologies and scenarios, and validation at schools
4. Feedback, questionnaires, interviews, data mining
5. Return to (3) based on (4)

6. Conclusion

Future school means personalisation plus intelligence.

Learning personalisation means creating and implementing personalised learning paths based on recommender systems and personal intelligent agents suitable for particular learners according to their personal needs.

Educational intelligence means application of intelligent technologies and methods enabling personalised learning to improve learning quality and efficiency.

Lithuanian IFS project is aimed at implementing both learning personalisation and educational intelligence.

References

Brusilovsky, P., & Millan, E. (2007). User Models for Adaptive Hypermedia and Adaptive Educational Systems. In: P. Brusilovsky, A. Kobsa and W. Neidl (eds.), The Adaptive Web: Methods and Strategies of Web Personalization (pp. 3-53). Lecture Notes in Computer Science, Vol. 4321, Springer.

CCL (Creative Classrooms Lab, CCL) project website (LLP, 2013-2015). Available at: http://creative.eun.org/

Coffield, F., Moseley, D., Hall, E., & Ecclestone, K. (2004). Learning styles and pedagogy in post-16 learning. A systematic and critical review. Learning and Skills Research Centre, UK

Felder, R. M., & Silverman, L. K. (1988). Learning and teaching styles in engineering education. Engineering Education, 78 (7), 674-681. Preceded by a preface in 2002

Fleming, N.D. (1995). I am different; not dumb. Modes of presentation (V.A.R.K.) in the tertiary classroom. In: A. Zelmer (Ed.): Research and development in higher education. Proceedings of the1995 annual conference of the higher education and research development society of Australia (HERDSA), Vol. 18 (pp. 308–313)

Herrmann, N. (1996). The Whole Brain Business Book. McGraw-Hill

Honey, P., & Mumford, A. (2000). The learning styles helper's guide. Maidenhead: Peter Honey Publications Ltd.

iCOPER D3.1 (2009). ICOPER Report on the Standardized Description of Instructional Models. Deliverable 3.1. September, 2009. Available at: http://www.icoper.org/deliverables/ICOPER_D3.1.pdf

iTEC (Innovative Technologies for Engaging Classrooms) project website (7FP, 2010-2014). Available at: http://itec.eun.org

Juskeviciene, A.; Kurilovas, E. (2014). On Recommending Web 2.0 Tools to Personalise Learning. Informatics in Education, Vol. 13 (1), 2014, pp. 17–30.

Kay, J. (2001). Learner Control. User Modeling and User-Adapted Interaction, 11, 111-127

Keefe, J. (1979). Learning style: an overview. NASSP's Student Learning Styles: Diagnosing and Prescribing Programs, 1-17

Kurilovas, E.; Juskeviciene, A.; Bireniene, V. (2015). Research on Mobile Learning Activities Using Tablets. In: Proceedings of the 11th International Conference on Mobile Learning (ML 2015). Madeira, Portugal, March 14–16, 2015, pp. 94–98.

Kurilovas, E.; Juskeviciene, A. (2015). Creation of Web 2.0 Tools Ontology to Improve Learning. Computers in Human Behavior, doi:10.1016/j.chb.2014.10.026

Kurilovas, E.; Kubilinskiene, S.; Dagiene, V. (2014). Web 3.0 – Based Personalisation of Learning Objects in Virtual Learning Environments. Computers in Human Behavior, Vol. 30, 2014, pp. 654–662.
Kurilovas, E. (2014). Research on Tablets Applications for Mobile Learning Activities. Journal of Mobile Multimedia, Vol. 10 (3&4), 2014, pp. 182–193.
Kurilovas, E.; Serikoviene, S. (2013). On E-Textbooks Quality Model and Evaluation Methodology. International Journal of Knowledge Society Research, Vol. 4 (3), 2013, pp. 66–78.
Kurilovas, E.; Zilinskiene, I. (2012). Evaluation of Quality of Personalised Learning Scenarios: An Improved MCEQLS AHP Method. International Journal of Engineering Education, Vol. 28 (6), 2012, pp. 1309–1315.
LSL (Living Schools Lab) project website (7FP, 2012-2014). Available at: http://lsl.eun.org/
LRE APv4.7 (2011). Learning Resource Exchange Metadata Application Profile v.4.7. Available at: http://lreforschools.eun.org/c/document_library/get_file?p_l_id=10970&folderId=12073&name=DLFE-1.pdf European Schoolnet, October, 2011.
Sangineto, E., Capuano, N., Gaeta, M., & Micarelli, A. (2008). Adaptive course generation through learning styles representation. Journal of Universal Access in the Information Society, 7 (1), 1-23.
Soloman, B., & Felder, R. M. (1998). Index of learning styles questionnaire. Available at: http://www.engr.ncsu.edu/learningstyles/ilsweb.html

Computer science (CS) or information and communication technologies (ICT): the curriculum needs both

Don Passey
Professor of Technology Enhanced Learning
Centre for Technology Enhanced Learning, Department of Educational Research
Lancaster University, Lancaster, UK

ABSTRACT

The subject of computer science (CS) has relatively recently arisen within the curriculum; yet it has been quickly followed by in-depth study and creative application. In 1962, Purdue and Stanford Universities established perhaps the first departments of computer science; the first PhD in computer science was awarded in 1965 by the University of Pennsylvania, while a robotic hand was developed in the same year at the University of Belgrade. Soon after, the subject started to appear in curricula for younger students. Since the 1980s, there have been concerns that computing and technologies should play a major role in school curricula and practice. Ways to bridge school and university curricula have been hotly debated. Up to this present time, in parallel and in ways that some would say have demonstrated mistaken appropriation, a major focus of technologies in the school curriculum has in many countries been on applications of existing technologies into subject practice (both software, such as office applications, and hardware, such as robots and sensors). Through uses of these applications, information and communications technologies (ICT) have focused on activities to support subject and topic learning (across wide age and subject

ranges). Limitations of staffing and resources have in the past been considered reasons for choice between ICT or CS curricula. Very recently, the concern for including computers in the curriculum (certainly in England and Australia, for example) has shifted to a much greater focus on computing and CS, more concerned with the uses of and development of programming and problem-solving. Across the world, policy makers at national, regional and local levels are concerned about this shift: whether the shift should be made; how it can be made; and how it can be made effective for teachers and learners. The limitations that once existed appear to have gone; yet key questions remain. This is a time of great opportunity, for building CS and ICT into curricula to support long-term needs. But, how can effective policies, that enable future generations, consider the vital concerns that go beyond what might be regarded as a simple move from ICT to CS? How can we integrate e-learning and e-ducation practices to support a new culture of learning? These key questions are explored here, and conclusions argue for a focus on future practices rather than on poorly-reflected decisions.

KEYWORDS

School curricula, computing, CS, ICT, national policy, economic need, future employment.

1. INTRODUCTION

Computers and technologies that have computing facilities now have a history in education, and their place in education is widely established. The subject of computer science (CS) has, however, only relatively recently arisen within the curriculum; yet it has been quickly followed by a very wide range of in-depth study and creative application. In 1962, Purdue and Stanford Universities established perhaps the first departments of computer science; the first Doctor of Philosophy (PhD) degree in computer science was awarded in 1965 by the University of Pennsylvania, while a robotic hand was developed in the same year at the University of Belgrade. The introduction of computers into schools started after only a very short time interval; they can often be traced back to the 1980s, when single computing machines, initially running programs from tape cassettes, were introduced into schools in a number of countries across the world. Since that time, computing technologies have become increasingly diverse, both in terms of the facilities they offer (for example, being able to run programs from a hard disk, being able to access resources across the world via the Internet, being able to run and play video games, or being able to locate a geographical position and find directions to another location), and size and mobility (for example, using handheld and mobile devices such as mobile telephones, laptop machines, desktop machines, or large display facilities). It is now increasingly common for individual teachers or learners to possess more than one computing device of their own (perhaps a mobile telephone, a laptop, a Moving Picture Experts Group Layer-3 (MP3) player, and a games console, for example).

The original concerns of policy makers (largely at a national rather than regional or local level) when introducing computers into schools in the 1980s were not focused so much on how computing facilities could support subject or topic learning more widely, but were concerned much more with how teachers and learners could experience computers and computer facilities so that they might come to understand more about those that they would find in future employment situations (Passey, 2014). Although this was the key reason for computers being introduced into schools in England at that time, government agencies, research institutions and educational advisors and practitioners quickly saw opportunities and ways for computers to support subject and topic learning that would go beyond the field and subject of computing, computer science (CS) and programming. From the 1980s onwards, software programs were developed that were designed to enhance learning opportunities in classrooms, across subject areas, for example in mathematics, language and science, and this form of development and trend concerned with subject-supporting resources has continued to this present time, to the extent that many rich resources are now accessible to teachers and learners, not only within their own local areas, but from worldwide resources.

This paper is not so much concerned with this shift in focus from early intentions to more recent intentions, but is concerned fundamentally with the current discussion about and focus for school curricula on computing, CS and programming. However, it is perhaps salient to highlight the fact that the contemporary concerns about a focus on computing and CS are not new. However, it can be argued that the context in which this concern is now being discussed is different from that when it happened previously, in the 1980s. This paper will highlight the current reasons for these discussions and concerns, it will consider arguments for a shift to computing, CS and programming, but will highlight some fundamental issues and offer some recommendations about the nature of the curriculum, if the outcomes of that curriculum are to be effective in meeting the needs of learners and their future employment prospects.

2. THE PAST AND THE PRESENT

Most schools and teachers using computing technologies are concerned currently with how these facilities can be integrated into subject and topic teaching, and how their deployment can support learning. Teachers are concerned, for example, with how their learners might gain greater understanding through the teacher's uses of interactive whiteboards, or how the teacher can engage learners in reflective learning through appropriate feedback in electronic form. This focus is concerned with applications of existing computing technology facilities (both software and hardware), rather than a focus on using the underlying computing facilities themselves, and how they might be developed and used through programming or networking to solve problems. To date, the focus of many school curricula has been on: applications in subject and topic curricula; and developing uses of existing software or hardware within information and communication technologies (ICT) curricula.

Very recently, there have been discussions that have raised a fundamental issue: school curricula are not focusing adequately on computing, CS and learner uses of computing that will provide for adequate future needs. These discussions have led to some national curricula (such as those within Australia and England), now requiring a focus on computing and CS rather than on ICT (ACARA, 2013; DFE, 2013). This shift in focus and a shift towards mandatory requirements for schools and teachers to focus on computing and CS appears to be based on six main arguments, which are outlined briefly here.

There is an **economic argument**. It is argued that education should support learners in engaging through a curriculum that is most likely to support a future economy, where young people are able to meet the needs of current and future jobs and their skill requirements. Livingstone and Hope (2011), in a report on the future of the games and visual effects industries in the United Kingdom (UK), highlighted the dire need for more young people to become interested in and aware of prospects that are available to them, involving computing within this field, if these industries are to continue to develop and be fulfilled in terms of employee numbers and skills in the future.

There is an **organisational argument**. It is clear that industries and institutions are increasingly engaging and employing learning technologists to support their own individual local needs, to develop computing facilities that meet their specific organisational requirements. For example, universities and university departments are increasingly employing learning technologists, who are employed to develop and handle learning management systems to enable teachers and students to use online access and to engage in online learning that is managed and administered electronically. This trend is developing and increasing in business and industry too; and there is every reason to believe that such a trend will continue rather than wane over the next 20 years.

There is a **community argument**. That is, computing facilities are increasingly being and will increasingly be used not only by individuals for social purposes but also by 'communities', whether these are business and industry communities, or social communities, based on local government or local community groups, or indeed more widely scattered community groups with shared interests. Activities undertaken by these community groups will increase the need for some individuals to have and to use computing and CS skills to support not only themselves but also others across their community groups for specific shared purposes. Take, for example, the way that some 'older generation groups' are now becoming linked and engaging in uses of computing technologies to communicate with each other, and to take online courses that meet their own needs and interests (see, for example, the University of the Third Age Australia, with registration at local as well as national level) (U3A, n.d.).

There is an **educational argument**. Elements of computing continue to develop, and it is not possible to see an end-point to these developments. With new technological developments and new areas of application being opened up, there is a clear argument that education should appropriately support and fulfil these needs. The provision of a CS curriculum offers this form of provision. Additionally, it has become well recognised that CS and computing enable certain skills to be developed, and indeed that the disciplines are based upon certain fundamental skills and competencies. Skills such as problem solving, collaboration, creativity and logical thinking are often stated as outcomes for those engaging in computer science activities (Kay, 1991; McCormack and d'Inverno, 2012).

There is a **learning argument**. Current and new facilities require users to have technical, operational and application skills and competencies if they are to use and apply such facilities to support themselves and others. With computing technologies becoming increasingly ubiquitous, it can be argued that younger as well as older users should have an increasing understanding of, and capabilities to use, the full range of computing facilities that exist, whether these facilities are accessed through programming, or through application. The European Union has identified, for example, digital skills that all citizens should have, if they are to engage fully and effectively with uses of digital technologies (Ferrari, 2013). Part of these skills are concerned with computing and CS (for example, 'apply settings, programme modification, programme applications, software, devices, to understand the principles of programming, to understand what is behind a programme').

There is a **learner argument**. It can be argued that learners should be enabled to engage not only in what are considered to be generic areas of future need (such as numeracy and literacy), but also in areas that interest them. Computing or CS is an area that is known to engage and interest some learners (Passey, 2012), and it can be argued that for those individuals their engagement in this field should come at a time in their lives when they can potentially see ways in which that interest might shape their future as well as their immediate needs. Arguments for inclusion of computing and CS in school curricula to support learner interest from the age of 5 years is not uncommon (ACARA, 2013; DFE, 2013).

3. DO THE SAME ARGUMENTS APPLY TO ICT?

The **economic argument** that relates to ICT is different from that relating to CS, but is just as powerful. There is a clear need for people to be able to use ICT, and there is an increasing requirement for jobs generally to use ICT. Being able to use ICT is becoming a common need, for employees as well as citizens. Even going the Dartford Tunnel (a tunnel beneath the River Thames, when travelling on the notorious M25 motorway around London) now requires payment not at a barrier, but by a deposit of funds that each individual has to set up through an online account. Even plumbers, for example, now often take sophisticated technology equipment with them when they are undertaking specific tasks, as well as when they are creating bills for the work they have done, when taking payments with credit cards, or for monitoring their work times.

The **organisational argument** is just as strong. Many tasks when undertaking project or team work require those involved to have understandings of, often, quite a range of different technologies. Take, for example, the broadcasting industry, requiring those involved to have knowledge in a range of different areas to come together, in video creation, video editing, music and art forms, speaking to camera, and using interviewing skills, for example. Broadcast news programmes require the integration of these forms of skills, as demonstrated by the annual project run by the British Broadcasting Corporation (BBC) for teams of young people who work together in schools and develop a news broadcast. They capture news from their local community, produce weather and news reports, and integrate these into a programme that is then published on the BBC website. From this wide range of programmes, some are integrated into regional and national radio and television news. An evaluation of this project (Passey and Gillen, 2009) highlighted the need for teams of young people to have, to develop and to share skills where they are using and applying ICT. As the report found from learners themselves that they developed a range of skills that required uses of ICT:

> *... abilities to write an article for an audience, take pictures using a range of media, create ideas for news stories, negotiate a point with others, work hard in contributing to group endeavour, and meet deadlines. Interestingly, students indicated no significant change in their abilities to produce a video and an audio story, whereas teachers felt they had improved. (p.7)*

The **community argument** is similarly strong. Communities (local groups as well as specific but more dispersed groups who have shared interests in fields such as environmental concerns) increasingly want to develop web sites and web resources, to enable an ongoing continuity of their own community sharing but also to support dissemination of their interests and resources to a wider public. Having individuals who have knowledge of how to apply ICT facilities to undertake these forms of activity is increasingly in demand; and such needs are likely to increase over time rather than to decrease.

The **educational argument** is also clear. ICT continues to develop. Passey (1989) charted shifts in developments with ICT at that time, at a range of different levels. At one level, for example, recognisable software shifts were arising about every 18 months. These sorts of shifts have not disappeared; with increasingly powerful technologies, these shifts need to be understood by those who will experience them in their futures, as well as having an understanding of what their implications are for users. There is a clear argument that educating about and for change with ICT should be a necessary element of a school curriculum.

The **learning argument** is clear. Learning how to use current facilities enables a range of benefits, including the bringing of authenticity to a curriculum, and supporting subject and topic learning. Much has been researched in this area, and although some researchers are sceptical of enhancements brought about by the technologies themselves (for example, Higgins, Xiao and Katsipataki, 2012), a recent review of the literature (Passey, 2013) indicates that there are very many ways that ICT can support elements of learning, and specific learners, often through appropriate mediation by teachers, counsellors and parents, as well as through the application of the technologies by the individual alone.

The **learner argument** is also strong. Many learners wish to develop ICT skills to support their uses and applications in educational as well as social arenas. Many learners use ICT to undertake aspects of learning, but the oft-quoted concept of 'digital natives' (Prensky, 2001) does not necessarily infer and mean that learners are aware of how to use ICT for purposes other than those they practice regularly. Indeed, many teachers report that this is the case, and they offer a strong argument to focus on how ICT applications can be better used to support educational endeavour, rather than any focus being on more general use.

4. ARE THE UK AND AUSTRALIA ALONE IN CONSIDERING THESE ARGUMENTS?

Recent national reports from members of the International Federation of Information Processing (IFIP) community suggest that the UK and Australia are indeed not alone in considering shifts with the balance of CS and ICT within the curriculum. Whether the arguments above are a part of that concern in any specific country is not clear, however. What can be stated is that, during May and June 2015, nine national reports on education and technology have separately identified how these countries (nine in total, spread across three continents) are currently considering CS and ICT within the compulsory education curriculum. While there is no single approach being identified in these reports across these nine countries, it is clear that policy and curriculum concerns regarding the inclusion of CS and ICT are being raised internationally. Table 1 offers a view of how the nine countries are currently considering and approaching CS and ICT in terms of provision within the compulsory education sector. The sources of the individual national reports (from national IFIP representatives to the technical committee on education), are shown in the table also.

Country	CS	ICT	Source
Austria	Mandatory ICT/Informatics Education is only in grade 9 (age 14 years)	There is a Digital Competences Standard for grades 5 to 8, and for grades 9 to 12	Futschek, 2015
Finland	In secondary schools, there are ICT driving license courses and quite a number of specialised courses like programming or numerical mathematics, and robotics has become very popular	In primary schools, learners should become comfortable with the computer and some common software. In secondary schools, the curricula are defined in very general terms	Koivisto, 2015
France	Computer science has been an elective subject for students in the scientific stream in general high schools. The May 2015 plan includes "coding" for elementary schools, "programming" for all students in middle schools (how and by which teachers is still to be defined, but within interdisciplinary activities), and there will be an elective course in computer science for all students (not only those in a scientific stream as now) and for the three levels	There is existing ICT use in schools, but it may not be generalised. In elementary schools, ICT is included in the whole curriculum	Grandbastien, 2015

Country	CS	ICT	Source
Ireland	In primary schools, activities are incorporated into the timetable informally by 'champion' teachers, including those programming with Scratch. In secondary schools, computing activities are mainly in the 4th Year (optional year – 15/16 year olds), including coding clubs and competitions, e.g. Coderdojo, Google Call to Code, and the ICS Skills 4-module Computing Curriculum	In primary schools, activities are incorporated into the timetable informally by 'champion' teachers, including digital storytelling using Microsoft (MS) PowerPoint and other free tools, Skype with other schools, and use of Edmodo and other virtual learning spaces to create class blogs. In secondary schools, computing activities are mainly in the 4th Year (optional year – 15/16 year olds), including digital skills programmes, e.g. ECDL, and competitions in STEM areas, e.g. F1 in Schools, Roboslam, Young Scientist	Leahy, 2015
Italy	For computer science education, a policy is present for specialised education (e.g. upper secondary technical schools). A national optional initiative ("*coding the future*") has been launched with the aim of introducing primary school pupils to basic computer science concepts through coding	At primary school level ICT is used occasionally to teach some curricular topics or to motivate and engage pupils (e.g. digital games, etc.). Many secondary schools promote the use of ICT for teaching a large variety of disciplines but results and impact are very varied	Bottino, 2015
Japan	No special course for learning informatics or ICT is stated officially at the elementary stage. Lower secondary schools have informatics and programming curricula as a part of 'Technology and Home Economics', which is one of the compulsory subjects determined by the Ministry	Regarding elementary schools, use of ICT to improve pupils' learning is recommended in the 'General Provision' of the Ministry Courses of Study for elementary schools. Upper secondary schools have a common subject of 'Information', which is a compulsory subject consisting of the 2 elective subcategories, 'Information Study for Participating in Community' and 'Information Study by Scientific Approach'	Saito, 2015
Lithuania	In secondary schools, a large number of ICT learning objectives are included in central steering documents, which include less common objectives such as programming skills and knowledge of computer hardware	ICT is recommended as a general tool across subjects in primary schools. In secondary schools, ICT is taught as a separate subject and also integrated into other subjects in secondary schools. The aim of the separate ICT subject is to develop students' information and technological competences. It is taught from basic grades (5th), and continued in secondary schools and gymnasia	Dagienė, 2015
Republic of Korea	Basics and how to use algorithms, programming, problem solving, representing information, computer networks and computer ethics are taught as an elective course within the curriculum revised in 2009, but new national curricula with a compulsory course from 2017 will be introduced	Upper classes of elementary schools are taught basic methods for using word processing, emailing, Windows and the internet, multimedia, adopting computer applications in other subjects, etc.	Kimn, 2015

Country	CS	ICT	Source
South Africa	Computer science, informatics or ICT taught or used in secondary schools focuses on programming in Java and Delphi, but there is pressure to drop Java in favour of Delphi. There is a large focus on database development	A National Curriculum and Assessment Policy Statement for ICT for Grades R to 12 was introduced in 2012	van Niekerk, 2015

Table 1: Current approaches to CS and ICT in the compulsory school sector in nine countries

As can be seen from the details in Table 1, both CS and ICT are included in many curricula and curriculum practices in compulsory education sectors in these nine countries. In some countries, CS is now being introduced for younger learners, while in other countries ICT continues to be developed alongside the introduction of CS. It is also clear that in some countries, policy decisions are not universally accepted by educators, who see different arguments and concerns from those who are making substantive decisions. In other countries, working groups or consultation groups involving interested parties in computer societies and those in policy groups have been established, and these are leading in some cases to outcomes that range from curriculum documentation to project developments or even technology developments to support the compulsory education sector.

5. MAKING THE CURRICULUM EFFECTIVE

If the arguments, concerns and practices above are accepted, then it is clear that some shift towards CS within a school curriculum is desirable, and while this is being developed in a number of countries, this is happening in different ways. A key question is: how to do this effectively, so that schools, teachers and learners are involved in practices that support current and future needs. Effective use and outcome is likely to require an understanding of the arguments on which this shift is based, and how to consider and address the needs of each element of those arguments.

5.1 The economic argument

The **economic argument** will require an understanding of how both CS and ICT are affecting employment and economies, and how jobs are increasingly using CS and ICT. Whether this understanding can be developed from local, regional, national and international perspectives is a question that should clearly be debated. There is some evidence that is accessible about job changes (such as that from the Bureau of Labor Statistics, 2012), but how such evidence is made accessible to schools, teachers and learners is an aspect that is likely to need a much greater level of discussion and development. The United States (US) data shows employment areas that have the most likely growth up to the year 2022 (Bureau of Labor Statistics, 2012): industrial-organisational psychologists; personal care aides; home health aides; insulation workers; interpreters and translators; diagnostic medical sonographers; bricklayers and tillers; occupational therapy assistants; genetic counsellors; physical therapist assistants; physical therapist aides; skincare specialists, etc. But, whether this form of evidence can usefully be used, or accessible within a local area, or at a national level, and how this relates to computing and CS, is not at all clear.

Skill or job shortages tend to reported rather than predictions of job growth areas. In the UK, there are concerns with ICT skills shortages (e-Skills, 2012). As this e-Skills report states, ‘Replacement demand will generate an additional 321,000 job openings in the sector which in addition to the 50,000 jobs created by growth means there is a total requirement of 371,000 between 2010 and 2020. ... Future skill needs in the sector can be grouped into five areas: security skills, business skills, technology specific skills, interpersonal skills and analytical skills' (p.v). While in England it was estimated that an additional 745,000 ICT workers would be needed between 2013 and 2017 (according to BIS, 2013), almost half of employers reported in 2011 that they had encountered difficulties in finding suitable applicants (e-Skills, 2012). The low numbers of women employed in the ICT sector is highlighted as an issue (and presumably, if addressed, providing a possible solution). The number of women in ICT employment is identified as being far less than the number of men; in 2013, women accounted for 16% of the ICT employment number in England (e-skills and BCS, 2014). As the e-skills and BCS report states, 'By 2013, of 1,129,000 people working as IT specialists in the UK, less than one in six (16%) were women', 'Of the 753,000 people working in the IT sector at this time, just one in five (20%) were women', 'In 2013, within the IT sector itself little more than one in ten (11%) IT specialists were women' (p.7). The report goes on to say that the proportion of women 'in Higher Education in 2013, ...made up just 12% of applicants and 13% of acceptances' and in secondary education, 'females accounted for just 6.5% of those taking Computing A-Level', while 'The proportion of females who sat an IT related GCSE in 2013 was 44%' (p.7). These data suggest that there is a major decline in commitment to (and perhaps interest in) IT beyond GCSE (the national examinations at age 16 years) level.

As can be seen from these figures and evidence, an economic argument is not based entirely upon the sole need to increase the number of employees with interest in and expertise in CS. It is based on a need for the development of individuals who can take up their interests in both CS and ICT.

5.2 The organisational argument

To address the **organisational argument** schools will require an understanding of how CS and ICT are used and integrated into practices in a range of organisations that go beyond 'the programmer in their bedroom or garage'. In higher education, for example, the University and Colleges Information Systems Association (UCISA) recent surveys (UCISA 2010, 2012, 2014) show that institutions in the UK are increasingly employing professionals for CS- and ICT-related jobs with a variety of job titles – learning technologists, e-learning officers, e-learning advisers, and e-learning staff developers. Fifty-four out of 91 institutions participating in the UCISA 2010 survey reported having a form of learning technology support unit (with a team of people) while 56 had an educational development unit that provided support. Additionally, support was reported to be provided by IT support units (rather than by individuals), which were most commonly (66 responses) reported in terms of central units providing IT or ICT support. The UCISA 2014 survey indicated a rise in the number of learning technologists who provided that support:

> *There has been an overall increase in the number of learning technologists both within and outside central units …Despite the challenging economic climate and budgetary pressures, which have led just under half the number of responding institutions to restructure or change existing TEL support roles, 34 institutions reported that they had actually increased staffing levels for TEL since the last survey and 38 institutions foresee staff increases in the future. (p.13)*

The UCISA survey (2010) reported that some 11 members of learning technology staff were employed on average by each of the institutions participating in the survey, and that most of these were located in support units, working with and alongside others. Certainly learning technologists in universities are not a completely homogenous group; individuals can be asked to undertake quite different tasks, which can range across the CS and ICT arenas. Nevertheless, many learning technologists need to work with others, even if they are involved in CS or programming practices. Programming data base or web site applications, for example, requires an understanding of how such facilities will be used, and by whom. Concerning working with others, in some other cases, learning technologists need to take strategic roles, but again, this requires working in groups and with others, rather than in isolation.

The fact that CS skills are now increasingly used not alone, but within teams and groups, in the case of video games development, for example (Passey, 2012), means that schools, teachers and learners should consider whether and how to develop CS within team work or group situations rather than skills being developed in isolation, solely individually. In the study mentioned (Passey, 2012), where the organisation of the activities for young people in schools was based on advice from a leading developer in the video game production field, teams of learners were asked to develop video game levels. The video game creator who advised the schools in setting up their teams to undertake these activities identified the range of individuals involved in the original video game creation: a story lead (to create dialogue, text and scripting); an art lead (to create models and textures); a sound lead (to create speech and special effects); a creative director; and a lead programmer. When schools set up their teams for the video game level development project, individuals worked collaboratively and co-operatively, with their different skills and strengths being deployed and shared across the team. It was clear that those focusing on CS skills did not do this in isolation; they were integrally involved with the team. In these activities the entirety of soft skills deployed and developed were measured, through self-reported levels of those skills before, during and after completion of the project. Individual skill sets that were involved, and that developed further for those learners across the period of the project were: thinking skills; problem solving skills; researching skills; generating ideas; identifying solutions; making skills; evaluating skills; communicating skills; scripting skills; story boarding skills; sequencing skills; logical thinking skills; artistic skills; team working; planning skills; and leadership skills. It was clear from this project that CS skills were being used in an integrated way, and that having these skills on their own, developed in isolation, would not only have provided a false view of how the industry organises team working to include those individuals who contribute CS skills, but would also not allow the skills to be easily or efficiently integrated into the entirety of the design and production of the outcome. Some studies have looked at how this need might be addressed, in the context of learners working in classrooms in pairs, for example (Johnson, 2014).

The organisational argument is concerned with developing shared and team approaches as much as individual skills. From a curriculum perspective, therefore, the ways that those with CS and ICT skills can work together and develop further, and the ways that those within both the CS and ICT arenas can have opportunities to work in groups, in pairs, individually, on projects as well as on specific skills, is a clear need if the practices of the compulsory education sector are to meet the needs of the future.

5.3 The community argument

Addressing the **community argument** will require an understanding of how CS and ICT skills can be deployed within community-based situations. There are examples of initiatives where schools in the Netherlands, for example, enable engagement of their learners with external research issues that are identified by industry and community groups (reported in Passey, 2013). This form of practice enables the learners to deploy problem-solving approaches, some involving levels of CS and ICT integration.

Within the compulsory education sector, other approaches have already been piloted, exploring ways that different age groups might work with others in the wider community. In the UK, a recent example involved the setting up of a project within a primary school, where fathers were encouraged to come into school to work with their children on developing a Lego Mindstorms robot, and then to undertake a series of activities involving programming the robot. The project leaders found that fathers who came into the school had not previously had contact with the school, yet engaged fully with their children in these activities (run during an afternoon session, one session a week for 4 weeks, each lasting about 2 hours). The fathers reported that their reasons for participating were concerned with their interests in 'making' (often arising from their much earlier experiences); many of them had made models when they were young, perhaps using Meccano or Lego, and while programming was new to many of them, their interest in the technical making aspects was sufficient for them to feel that they could cope with taking on some additional skills that they were not familiar to them, especially as they could do this within a supportive environment and with their own children.

This form of project is not only concerned with development of CS and ICT skills, it is concerned also with a potential relationship between formal (classroom), non-formal (interest group) and informal (home) learning, and relationships of an intergenerational nature. While it has been suggested that there can be a 'digital disconnect' between creative uses of ICT and media that occur at home and the more traditional approaches that might be adopted at school (Furlong and Davies, 2012), it is clear also that the ways that teachers can handle and manage activities within classrooms with a group of 30 learners is a factor that needs to be considered in this respect from a management perspective. The project above involved just 4 parents and 4 children, working in pairs. But with a class of 30 children, this is a model that might not be easy for a teacher to emulate (even having the space available to accommodate 30 pairs of children and fathers working together). However, if there are ways that the teacher can take advantage of projects that support classroom practices, then this is clearly of potential benefit. Passey and Gillen (2009), for example, reported how project-based approaches using ICT enabled some learners to better understand why they were learning aspects of literacy and numeracy, and how they could apply them in authentic situations.

The potential for enabling this balance of practice – taking advantage of formal, non-formal and informal learning activities to build into a wider entirety – is being made more accessible through technology developments. For example, a current technology used by learners in schools in the UK, for programming and control, is the Raspberry Pi. These units are used by learners outside as well as inside schools, and some parents are taking more interest in how they might work with their children in using these within their own family home settings. A recent conference session, for example, highlighted how a parent had worked with their child, programming a Raspberry Pi to capture video footage of a nesting bird and the hatching of its chicks.

Taking this further, the BBC is now developing, with a consortium including ARM, Microsoft and Samsung, a new device called Micro Bits, which aims to put 'control in the hands of the children' (CAS, 2015). The intention is to equip all year 7 learners in the UK (about 750,000 learners, 11 years of age) with this device in September 2015. As the Computing at School (CAS) article says, 'Putting the kit in the hands of the children will help engage parents too', since the devices will be 'owned by children'. Teachers, parents and peers will all be able to benefit from access to and use of these devices, but clearly benefits that would bring together the opportunities when learners engage in formal, non-formal and informal activities are likely to be ultimately important, since separating them might not allow the same learning paths for their development.

An important aspect of this Micro Bits initiative is concerned with an intention to support the addressing of what has become known as the 'digital divide', which has been used to describe and identify the different experiences that learners might have when they are in different social backgrounds. In this respect, unequal access to new technology, where home access to a computer has been positively associated with higher levels of educational attainment (Chowdry, Crawford and Goodman, 2010), is an aspect that this initiative seeks to accommodate at least, if not address.

In terms of the community argument, again both CS and ICT arenas need to be considered. In both cases there are good reasons why a curriculum should include elements of CS and ICT, for wider and longer-term community involvement and development. However, managing this within the school in order to gain maximum benefit for learners will require time and effort to be given to the exploration of approaches that can enable formal, non-formal and informal learning activities to be developed appropriately, and considered as an entirety.

5.4 The educational argument

Addressing the **educational argument** will require an understanding of how CS and ICT can be integrated into curricula at school and subject levels. Many curricula are developed in ways that lead to formal classroom level practices; but as argued above, CS and ICT require consideration of how curricula can be accommodated that can lead to non-formal and informal practices as well as formal ones. The new Australian curriculum states that learners should develop 'knowledge, understanding and skills... individually and collaboratively' (ACARA, 2013). Individual learning can certainly be organised in formal ways, where learners have access to an individual desktop machine, perhaps. But for collaborative endeavour, it should be possible also for learners to be able to work in non-formal (groups like clubs or societies, to focus on specific interests) or informal situations (where they might use more mobile or flexible access).

At the same time, schools will need to be aware of local possibilities if they are to be able to develop practices through links with non-formal and informal partners (businesses, agencies, community groups or parents, for example). Monitoring and understanding shifts in local labour markets will also be an important element for schools to locate. There are initiatives in the UK that seek to support in this way. The Tech Partnership (2014) and CAS (2015) both provide links and support for schools, the former focusing more on present and future labour and employment, while the latter (with 10 centres of excellence across England) focuses more on CS projects and practices. It will be important that schools can create links with these groups, in order to both be aware of local opportunities, but also to gain from development opportunities. Schools will also need to consider how the curriculum might support CS and ICT involvement that is not biased towards certain groups, such as boys rather than girls. However, it should also be recognised that while the number of girls going into CS and ICT jobs are low by comparison to boys, girls' jobs tend to involve more 'soft' skills in communication while boys' jobs tend to involve more technical skills (Glover and Guerrier, 2010). Kirkup (2011) states that vocational education and training for ICT jobs might themselves have developed gender- and class-biased occupations, with girls being encouraged to explore areas of 'soft' ICT skills more, so that they end up with what are regarded as employment with lower level skills. The situation in some other countries is also not dissimilar. In Germany, for example, Kirkup (2011) reports that the proportion of girls in electronics technician training is only 2.5% and in information technology specialist training is only 4.7%. The Tech Partnership (2014) in the UK is supporting schools with possible approaches intended to address these issues. Employers, including Hewlett Packard, British Telecom, Oracle and the National Grid, are supporting an initiative called TechFuture Girls, freely available, designed as after-school clubs to 'provide industry-backed challenges for girls aged 10-14. These teach skills such as coding, cyber security, data management and video editing through activities based on girls' interests, including music, sport or fashion' (p.14). Other initiatives run by the Tech Partnership include TechFuture Classroom, which:

> *...delivers curriculum resources for students based on real life projects from industry. These support computing and computer science qualifications including GCSEs and A-levels, and are provided complete with lesson plans and mark schemes for teachers. 810 teachers in 540 schools have already taken advantage of this investment, with over 5,500 schools students having undertaken new industry-relevant learning as a result. This is an order of magnitude greater than the original plan of 850 for the period. (p.14)*

Additionally, the Tech Partnership's TechFuture Careers resources are reported to have been used by over 35,000 students, and 200 volunteers from industry have gone into schools to support a TechFuture Ambassadors programme. A further initiative, TechFuture Teachers, aims to 'bring the power of industry collaboration to the benefit of teachers, with work shadowing, weekly webinars from industry, and other opportunities for professional development' (p.14).

From an educational argument perspective, schools certainly need to take on board a range of key issues, and consider ways that they can locally take approaches that match contexts and needs of their learners. Overall, the need for teacher development in terms of any shift towards CS is clear; the fact that teaching practices have been developed through an ICT perspective in the past does not mean that teachers will be able to naturally or easily focus on CS needs and approaches as well.

5.5 The learning argument

Addressing the **learning argument** will require an understanding of the CS and ICT skills that should be taught and should be learned. It is easy to identify these in terms of programming; it is also essential that these skills are considered from the point of view of their context, with appropriate associated soft skills. The new curriculum in Australia (ACARA, 2013) states that it:

> *aims to develop the knowledge, understanding and skills to ensure that, individually and collaboratively, students: are creative, innovative and enterprising when using traditional, contemporary and emerging technologies, and understand how technologies have developed over time; effectively and responsibly select and manipulate appropriate technologies, resources, materials, data, systems, tools and equipment when designing and creating products, services, environments and digital solutions; critique and evaluate technologies processes to identify and create solutions to a range of problems or opportunities; investigate, design, plan, manage, create, produce and evaluate technologies solutions; and engage confidently with technologies and make informed, ethical and sustainable decisions about technologies for preferred futures including personal health and wellbeing, recreation, everyday life, the world of work and enterprise, and the environment. (p.2)*

Although it does not explicitly indicate the need for learners to consider associated soft skills, it is clear that 'creating products, services, environments and digital solutions' requires a clear focus on audience, which might well (or perhaps should) involve discussion and collaboration with users so that their needs and requirements are understood and fulfilled. By contrast, the aims of the new national curriculum in England can be interpreted at a much more individual learner level, meaning that associated soft skills might well be less likely to be considered:

> *to ensure that all pupils: can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation; can analyse problems in computational terms, and have repeated*

practical experience of writing computer programs in order to solve such problems; can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems; and are responsible, competent, confident and creative users of information and communication technology. (n.p.)

While CS and ICT provide opportunities for learning involving higher order levels of thinking and skills, the challenges that these subjects are stated to offer has not been followed by a wide uptake by learners for courses when these are optional. Rather than an uptake of the subject, in the UK there has been a declining uptake of ICT-related subjects both at General Certificate of Secondary Education (GCSE) (at age 16 years) and at Advanced- (A-level (at 18 years). According to statistics produced by the Joint Council of Qualifications (2014), the number of learners taking A-level qualifications in computing and ICT fell by 43% between 2003 and 2011. Similarly, girls taking IT-related courses fell from 47% in 2012 to 44% in 2013 for GCSEs and from 8% in 2012 to 6.5% in 2013 for A-levels (e-Skills and BCS, 2014). The additional role of social background as an influencing factor here has also been questioned through research; for example, both ICT use and ICT literacy levels have been found to be low on average in learners from disadvantaged backgrounds (The Prince's Trust, 2014).

A factor explored in one study, learner perceptions of computing careers, indicated that these were generally regarded to be poor (McEwan and McConnell, 2013). This has been discussed too by teachers, who have stated that they feel that the value of teaching ICT focusing on IT skills and digital literacy is poor by comparison to the value associated with teaching computing. With ICT teaching described as being 'dull and unchallenging', teachers and educators concerned with these poor perceptions have argued that computing should be adopted more strongly, with its more highly-regarded associated creative and problem-solving approaches (Royal Society, 2012).

While the learning argument is strong, there is currently much more limited research that has looked at the value and outcomes of learning from a CS perspective than from an ICT perspective. While it is clear that the approach teachers take will be crucially important, there is nevertheless a need for a range of research arenas to be developed, to explore features and factors that might maximise the outcomes for learning and for learners.

5.6 The learner argument

Addressing the **learner argument** will require a balance, if the curriculum is to enable learners to develop their own interests in CS- as well as ICT-based practices. Indeed, whether it is essential for all learners to have highly-developed CS skills is not at all clear. What is clear is that learners are enabled to gain what might be regarded as 'life skills' and to take forward their interests, so that CS is provided as an opportunity for all, but that those who have particular interests are enabled to take these interests as far as they are able. The Australian Curriculum, Assessment and Reporting Authority (2013) propose to do this by mandatory integration of CS and ICT up to Year 8 (age 13 to 14 years), with learners choosing optional choices in Years 9 and 10 (age 14 to 16 years):

The Australian Curriculum: Technologies Foundation to Year 10 is written on the assumption that all students from Foundation to Year 8 will study two subjects: Design and Technologies and Digital Technologies. At Years 9 to 10, the Australian Curriculum: Technologies is written on the assumption that school authorities will decide whether students can choose to continue in one or both subjects and/or if technologies specialisations that do not duplicate these subjects will be offered. (p.3)

However, an additional point for any curriculum development to consider fundamentally is the fact that it is reported (not uncommonly) by lecturers that students do not find computing to be an easy subject; students often say that it is ‘hard’. There may be different reasons for this, and perhaps the reasons are different for different individuals, and for some there might indeed be multiple reasons, while for others there might be more singular reasons. For example, the need for learned knowledge to be applied or transferred, or for the need to abstract knowledge and practice, or not understanding the underlying principles, may all be possible reasons for this reported difficulty. Indeed, the current curriculum for key stages 1 and 2 (5- to 11-year-old children) in England (DFE, 2013) if anything underlines these very issues, in stating that:

> *A high-quality computing education equips pupils to use computational thinking and creativity to understand and change the world. Computing has deep links with mathematics, science, and design and technology, and provides insights into both natural and artificial systems. The core of computing is computer science, in which pupils are taught the principles of information and computation, how digital systems work, and how to put this knowledge to use through programming. (n.p.)*

How this is handled for all learners, rather than those who select a computing course, therefore, is itself a critical question. In the England curriculum the programmes of study indicate what pupils should be taught, but they do not indicate how, or how to address what are known to be issues for learners. The success of this development, therefore, would appear to be likely to be based at least to some extent upon the abilities of teachers to address learner and learning concerns, not just teaching needs.

It is also true to say that lessons from the past need to be heeded. For example, an evaluation of a scheme in 2005 to provide after-school clubs for girls to engage them in computing (Fuller et al., 2013) suggested that this had reinforced existing gender stereotypes and expectations, concluding that the initiative was ‘unlikely to have a significant or sustained impact on what remains an occupational and subject area divided by gender’ (p.499). Learning more about the factors that engage different learner groups in CS and ICT is clearly important. Schools are best placed to understand their learners; however, evidence of reinforcement of stereotypes and practices suggests that careful review and monitoring is needed in this area if schools are to be successful in shaping future interests in CS and ICT.

6. THE FUTURE AND PROJECTED NEEDS

If it is accepted that the six arguments above constitute positive reasons for curriculum change, then it is important to explore to what extent the curriculum can, and indeed already has, considered these arguments and addressed them appropriately. It might be all too easy to say that the curriculum should shift from focusing on ICT to focusing on computing and CS, but whether this might take for granted what this implies for the school, the teacher and the learner also needs to be questioned.

The economic argument implies that schools, teachers and learners will not just recognise the fact that employment will in the future involve the need for more CS and ICT skills, but should enable them to understand where those skills might be needed, and how they are used and applied within employment situations. ***Does a school have access to knowledge about the ways that CS and ICT are being used and developed in employment situations, and what future needs might arise?***

The organisational argument implies that schools, teachers and learners understand how CS skills are currently used within organisations, and what this means in terms of the organisation of lessons and activities to enable skills to be developed in a way that matches future as well as current employment needs. ***Does a school understand how CS, computing and ICT skills are deployed and managed in***

organisations, and do they have facilities to undertake team work or group work activities of this form?

The community argument implies that schools, teachers and learners understand the contexts in which CS and ICT will be used and deployed. School provision is often of a formal nature; the community argument is based on a non-formal or informal, rather than a formal approach, however. ***Can a school manage and support activities that are undertaken in non-formal or informal situations, linking with community or organisations to engage with their needs through problem solving and creative solutions?***

The educational argument implies that schools, teachers and learners have access to the facilities that will enable educational outcomes to be realised. These facilities clearly concern not just computing facilities, but also the facilities that teachers can bring to the classroom, and the activities that learners will engage in. Australia is changing its curriculum, so that all pupils are taught two subjects up to Year 8 (age 13 to 14 years): Design and Technologies; and Digital Technologies. In Years 9 and 10 (age 14 to 16 years), pupils will be able to choose to take Technologies as a subject. In England, the change of curriculum to Computing requires all pupils to be taught the subject from Year 1 to Year 11 (from 5 to 16 years of age). ***Does a school have the flexibility to support a curriculum that can provide activities for all learners across certain age ranges, but offer elected courses for those beyond those age ranges?***

The learning argument implies that schools, teachers and learners are gaining skills and competencies that are of value to them in the future as well as in the present. An entire shift from ICT to CS would mean that skills and competencies that are gained from ICT might well be lost. Clearly this suggests that a balance of shift is needed, rather than a move away from one to another. ***Does a school have the facilities to enable teachers to access and use technologies to support both an ICT focus and a CS focus?***

The learner argument implies that schools, teachers and learners are concerned with a curriculum that supports engagement with practices of interest for the future as well as for the present. How learners can be supported in engaging with CS, which also balances provision for ICT, is perhaps the key question. ***Does a school enable its learners to engage at times when their interest might be stimulated in CS or computing or ICT?***

Balance of the formal, non-formal and informal appears to be crucial to the future success of CS and ICT in the compulsory education sector. There is a need to develop individual skills and to practice these within a safe environment (through formal learning activities), but there is a need to develop practices that enable sharing to happen and strengths and skills to be developed in pairs and teams through collaboration and project approachess (through non-formal learning activities), as well as the need to develop practices that solve and address real-life problems in family or community settings (through informal learning activities). In this respect, a recent study (Johnson, 2014), which explored how pairs of learners could develop CS skills with games authoring software, highlights well the need to consider very carefully how to conceptualise and manage this balance, and concludes by saying of the learners and the management of their learning that:

> *The wide range in outcomes further suggests that constructionist approaches are not suitable for all learners, especially those who need more guidance and structure. While most pupils in this study had an above average ability profile ..., they did not all display independent learning behaviours or make use of the sources of support made available to them, and this may account for the variation in the games produced. ...their success seemed to have as much to do with their*

willingness to learn independently as to do with their cognitive ability. This variability in pupils' readiness to learn independently may also reflect the extent to which they had or had not encountered similar project-based activities in other areas of the curriculum.

Constructionist approaches may also not be well-suited to some elements of game authoring. Some aspects of learning, such as the development of graphics software skills, or the learning of programming concepts need, at this level, to be formally taught if they are to be successfully used by all - for these areas of learning, learning by doing and experimentation alone appear not to be sufficient. Pupils also need to be guided to complete tasks which are not immediately popular, such as planning the game program and object interactions. (p.252)

7. CONCLUSION

That there is discussion that school curricula now adopt CS is clear, and there are strong arguments for taking this shift forward. However, in doing so, schools, teachers and learners should not lose the vital and important components that could make the difference between this adoption being successful and it being unsuccessful.

Adopting CS should not exclude the need to integrate ICT across a wider school curriculum. CS should be concerned as much with group work and team work, with concerns for associated soft skills, and with audience needs, as it is with programming in isolation. CS activities should include those that consider how to integrate problem-solving approaches rather than just offer didactic programming activities. ICT should consider how non-formal and informal activities can enhance learning, rather than the entire curriculum being reliant on uses in formal situations and contexts. In this way, the arguments that are made for the inclusion of a CS curriculum may well then be met. Fulfilling these needs may not in itself be simple, but the outcomes when these are successfully addressed are likely to then be fulfilling for learners and teachers alike.

ACKNOWLEDGEMENT

The author would like to thank Jari Koivisto for the opportunity to develop and present this paper. Jari's interest in, feedback on and translation of an original paper from English to Finnish are all very gratefully acknowledged. Thanks also go to the IFIP TC3 national representatives who submitted national education and technology reports that provided an important international perspective for this paper. Lastly, my thanks to Claire Johnson, whose PhD thesis provided invaluable, further insight into this topic.

REFERENCES

ACARA, 2013. *Draft Australian Curriculum: Technologies*. Accessed 5 January 2015 at: http://consultation.australiancurriculum.edu.au/Static/docs/Technologies/Draft%20Australian%20Curriculum%20Technologies%20-%20February%202013.pdf

Bottino, R., 2015. *National Report on Education and Technology: Italy*. Accessible at: http://ifip-education.ning.com/page/national-reports-on-education-and-technology

Bureau of Labor Statistics, 2012. *Economic News Release, Employment Projections: 2010-2020 Summary.* United States Department of Labor, Washington, DC.

Chowdry, H., Crawford, C. and Goodman, A., 2010. *The Role of Attitudes and Behaviours in Explaining Socio-Economic Differences in Attainment at Age 16: IFS Working Paper 10/15.* Institute of Fiscal Studies, London.

Computing at School, 2015. *Switchedon: Computing at School Newsletter Summer 2015.* CAS and BCS, London.

Dagiené, V., 2015. *National Report on Education and Technology: Lithuania*. Accessible at: http://ifip-education.ning.com/page/national-reports-on-education-and-technology

DFE, 2013. *Statutory guidance – National curriculum in England: computing programmes of study*. Accessed 5 January 2015 at: https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study

e-Skills UK and the BCS, 2014. *The Women in IT Scorecard,* London and Swindon. Accessed 6 June 2015 at: http://policy.bcs.org/content/women-it-scorecard-it-profession-numbers

Ferrari, A., 2013. *DIGICOMP: A Framework for Developing and Understanding Digital Competence in Europe*. Institute for Prospective Technological Studies, Seville, Spain. Accessed 5 January 2015 at: http://ftp.jrc.es/EURdoc/JRC83167.pdf

Fuller A., Turbin, J. and Johnston, B., 2013. Computer Club for Girls: An initiative in search of the right problem? *Gender and Education*, 25, 4, 499–514.

Furlong, A. and Davies, C., 2011. Young people, new technologies and learning at home: taking context seriously. *Oxford Review of Education, 38*, 1, 45-62.

Glover, J. and Guerrier, Y., 2010. Women in hybrid roles in IT employment: A return to 'nimble' fingers?' *Journal of Technology Management & Innovation*, 5, 1, 85-94.

Grandbastien, M., 2015. *National Report on Education and Technology: France*. Accessible at: http://ifip-education.ning.com/page/national-reports-on-education-and-technology

Higgins, S., Xiao, ZM. and Katsipataki, M., 2012. *The Impact of Digital Technology on Learning: A Summary for the Education Endowment Foundation - Full Report*. Durham University, Durham, and Education Endowment Foundation, London.

Johnson, C., 2014. 'I liked it but it made you think too much': A case study of computer game authoring in the Key Stage 3 ICT curriculum. PhD thesis, University of East Anglia, Norwich, UK.

Joint Council of Qualifications, 2014. *Examination Results*. Accessed 29 June 2015 at: http://www.jcq.org.uk/examinationresults/

Kay, D. S., 1991. Computer interaction: Debugging the problems. In R. J. Sternberg and P. A. Frensch (eds.). *Complex problem solving: Principles and mechanisms* (pp. 317-340). Lawrence Erlbaum Associates, Hillsdale, NJ.

Kimn, H.J., 2015. *National Report on Education and Technology: Republic of Korea*. Accessible at: http://ifip-education.ning.com/page/national-reports-on-education-and-technology

Kirkup, G., 2011. Preparing women for dead-end jobs? Vocational education and training (VET) for Information and Communication Technology (ICT) jobs'. *International Journal of Gender, Science and Technology, 3,* 2.

Koivisto, J., 2015. *National Report on Education and Technology: Finland*. Accessible at: http://ifip-education.ning.com/page/national-reports-on-education-and-technology

Kutschek, G., 2015. *National Report on Education and Technology: Austria*. Accessible at: http://ifip-education.ning.com/page/national-reports-on-education-and-technology

Leahy, D., 2015. *National Report on Education and Technology: Ireland*. Accessible at: http://ifip-education.ning.com/page/national-reports-on-education-and-technology

Livingstone, I. and Hope, A., 2011. *Next Gen. Transforming the UK into the world's leading talent hub for the video games and visual effects industries: A Review*. Nesta, London.
McEwan, T. and McConnell, A., 2013. Young people's perceptions of computing careers. In *Proceedings of the 43rd Annual Frontiers in Education Conference,* pp. 1597-1603. IEEE, Piscataway, NJ.
McCormack, J. and d'Inverno, M. (eds.), 2012. *Computers and Creativity.* Springer Verlag, Heidelberg, Germany.
Passey, D. and Gillen, J., 2009. *BBC News School Report 2008/2009: Independent Evaluation*. BBC, London.
Passey, D., 2012. *Independent evaluation of the Little Big Planet 2 project in Wolverhampton's Local Education Partnership schools: Outcomes and impacts – Final report*. Lancaster University, Lancaster.
Passey, D., 2013. *Inclusive technology enhanced learning: Overcoming Cognitive, Physical, Emotional and Geographic Challenges*. Routledge, New York, NY.
Passey, D., 2014. Early uses of computers in schools in the United Kingdom: shaping factors and influencing direction. In A. Tatnall and B. Davey (eds.). *Reflections on the History of Computers in Education: Using Computers and Teaching about Computing in Schools from the late 1970s to the early 1990s.* Springer, Heidelberg, Germany.
Prensky, M., 2001. Digital natives, digital Immigrants, *On the Horizon,* 9, 5, 1-6
Royal Society, 2012. *Shutdown or restart? The way forward for computing in UK schools*. Accessed 29 June 2015 at: https://royalsociety.org/education/policy/computing-in-schools/report/
Saito, T., 2015. *National Report on Education and Technology: Japan*. Accessible at: http://ifip-education.ning.com/page/national-reports-on-education-and-technology
The Princes Trust, 2014. *Digital Literacy Survey 2013*. Accessed 29 June 2015 at: http://www.princestrust.org.uk/about_the_trust/what_we_do/research/digital_literacy_research_2013.aspx
The Tech Partnership, 2015. Tech Partnership Progress update April 2014 – March 2015. The Tech Partnership, London.
UCISA, 2010. *Survey of Technology Enhanced Learning for higher education in the UK*. Accessed 29 June 2015 at: http://www.ucisa.ac.uk/groups/ssg/~/media/groups/ssg/surveys/TEL%20survey%202010_FINAL.ashx
UCISA, 2012. *Survey of Technology Enhanced Learning for higher education in the UK*. Accessed 29 June 2015 at: https://www.ucisa.ac.uk/~/media/groups/ssg/surveys/TEL_survey_2012_with%20Apps_final.ashx
UCISA, 2014. *Survey of Technology Enhanced Learning for higher education in the UK*. Accessed 29 June 2015 at: https://www.ucisa.ac.uk/~/media/groups/dsdg/TEL%20Survey%202014_29Sep2014.ashx
U3A Online, n.d. *U3A Online*. Accessed 5 January 2015 at: http://www.u3aonline.org.au/
Van Niekerk, J., 2015. *National Report on Education and Technology: South Africa*. Accessible at: http://ifip-education.ning.com/page/national-reports-on-education-and-technology

FULL PAPERS

The 1:1 classroom – for what purpose?

Bent B. Andresen, bba@edu.au.dk
Aarhus University, Department of Education, Denmark

Abstract

This article presents research findings regarding the benefits of the 1:1 classroom in Danish schools (grades1-9).One of these benefits is that, by using wireless digital devices in the classroom, students can develop ICT literacy. The main theme of this article is the complementary relationship between the ability to use written language – that is, reading and writing – and digital literacy. In particular, this article addresses the following research question: In which ways can appropriate use of digital tools and media increase the student's desire to become digitally literate and promote his/her learning?

Keywords

1:1 classrooms, learning objectives, digital literacy, learning outcome.

INTRODUCTION

"By 2014, all students should have access to a computer (PC, laptop, tablet, etc.) in their lessons" in Danish public schools (Agency for Digitization, 2013). Denmark is the first country in the world to have a deliberate policy based on the "bring your own device"principle (Søby, 2013), where students are invited to bring their own digital equipment into the classroom. However, since public schools cannot demand that students bring their own tablet or similar device into school, the 1:1 solution is achieved in two ways: 1) by providing all students witha digital deviceor 2) by providing some students with a digitaldeviceand allowing others bring their own to use on the school network.

In general, Danish primary and secondary schools do not have a strategy thatrequires all students to always bring their own devices. However, when schoolspromote the 'bring your own device' policy, they include virtually all devices that are not technologically obsolete.So, rather than setting strict requirements, they encourage students to bring what they have. The device simply needs a web browser and a text editor. If the device is missing a spreadsheet or another specific function, this is installed on theschool laptops for the students to borrow.

This article presents findings from an empirical study that is currently in progress in public schools in Denmark. The study uses a mixed-method design. It is based on interviews with teachers and students, classroom observations, and evaluations of students' skills.The scientific research question is: In which ways can the appropriate use of digital tools and media increase the student's desire to become digitally literate and promote his/her learning?

BENEFITS OF 1:1 INITIATIVES

Danish children are among the youngest internet users. On average, they start to use the internet regularly from the age of 7, whilstin the EU, the average child begins to use the internet regularly from the age of 9(Haddon & Livingstone, 2012). Globally, Danish teenagersdevelop some of the highest computer and information competences (Bundsgaard, Pettersson, & Puck, 2014, p. 51). During school days, studentsfrom grades 1 to 10 use digital deviceswhen appropriate to read, write, watch and listen,in the broadest sense of these terms. According to some teachers, this leads to greater flexibility than before and minimises the start-up phase of lessons (Box 1).

> “For a start – and compared to the way it was, when you had to go down and fetch computers – it is definitely more efficient for students to have their own tablets.
>
> “It is obviously easier to use a tablet at the beginning because it’s on all the time, so there isn’t the same logon or waiting time as with a PC” (Andresen, 2014).

Box 1

Teachers appreciate thisincreased flexibility because it allows them to specify different learning objectives for individual students (Box 2).

> “It is an advantage when you are working with different objectives for different students, i.e. differentiated teaching. When the class is working on projects there are things all students need to cover but there are also different things they can do. If they manage it, you don’t have to collect a lot of material because all of this is there on their tablets. They can go online, write, and watch a video – they can do it all. They can instantly move on to the next thing without having to wait for the teacher to do something first” (Andresen, 2014)

Box 2

The teachers help the students to define their missions. In this context, the notion of*mission* is used to designatethe student’s efforts to finish or improve a piece of work or to start a new piece of work (Hattie, 2009). A possible sequence is first to acknowledge the student’s efforts, then to offer some constructive criticism, and then to give more recognition. As the teachers cannot see what is going on in student’s mind, they also decide how the student should demonstrate his/her understanding (Gardner, 2000). According to some teachers, the 1:1 initiative leads to improved self-directed learning (Box 3).

> “An obvious advantage in many hand-in subjects is that teachers can give students different things to do, so they don’t all have to produce the same two pages. Some can produce a radio montage, some can produce a book ..., and some can make a stop-motion film. The teachers can tell the weaker students that there are alternatives to the two pages of writing and that they can make a montage or something else to hand in. That’s not to say that these students can’t produce a piece of written work later on, but the choice is suddenly very wide and this is an obvious advantage of working with tablets”.
>
> “The result is more creative when students produce something on their tablets: they have made timelines, animated films and mind maps on their tablets; some of them have also made a calendar wheel. The results are very varied and more creative than work with paper and pencil” (Andresen, 2014).

Box 3

The student’s outcome can be increased if the teacher combinesface-to-face and online learning. In general, older students derive academic benefit from a blend of face-to-face and online activities (Means, Toyama, Murphy, Bakia, & Jones, 2009). When the teacher’s planning is inspired by e-learning methods, the following question arises: how should students work with the digital tools and media to achieve the desired goals? The learning objectives and the content define the framework – not the other way round.

When students use tablets as sharing tools, it is generally easier to comment on their draft and finished products (Box 4). Therefore, the benefits of the 1:1 classroom include the enhanced distribution of the student's drafts and final products as well as improved access to provide and receive feedback.

> "Teachers have 'corrected' students' exercises and style in this way for many years, so it is nothing new, but it has become easier to comment via online documents. It's easier to work in an online document, because the teachers don't have to download it first before they can mark it, and then have to upload it again afterwards" (Andresen, 2014).

Box 4

In some cases, students receive comments while they are still completing their work. The student can then incorporate and learn from these comments before submitting the finished product. There is evidence to suggest that this type of proactive feedback is one of the measures that has the greatest bearing on student learning (Black & William, 1998). One way of expressing how various educational measures affect student achievement is the 'effect size'.When the teacher evaluates and comments on the student's work whilst he/she is still completing itthe learning effect is more than double the average for other types of educational measures (Hattie, 2009).

Another benefit of 1:1 classroom is that it promotes a climate of confidentiality in the classroom. In general, the teacher does not judge the student for making mistakes, but instead views mistakes as a necessary part of learning and appreciates the student's progress towards his/her individual goal. Consequently, with 1:1 classrooms, individual students are not measured against other students but against themselves; that is, in the context of their on-going and previous work. There is evidence to suggest that this kind of classroom management fosters independent learning (Nordenbo, 2011).

DIGITAL LITERACY STANDARDS

Students are often described as 'digital natives', because they have had access to ICT all their lives (Tapscott, 1998). However, they are also referred to as 'digital naives', because they are partly self-taught and not especially ICT-aware in all cases (Hargittai, 2012). As a consequence, schools have to set the standards.

Ten years ago, an expert review indicated that standards are seldom used in public schools in Denmark: "There appear to be few available standards other than a teacher's own cumulative experience or the collective view of his or her colleagues" (Mortimore, David-Evans, Laukkanen, & Valijarvi, 2004). However,it has since become clear that learning goals are used to some extent in teaching in schools (EVA, 2012). This includes objectives within the area of digital literacy.

Recently, explicit goals for the development of competences, knowledge, and skills in this area have been systematically applied in Denmark. These objectives should not be confused with goals that describe what the teaching aims to achieve; digital learning objectives state what the student should have learnt at any given stage. From 2014/15, such objectives are being introduced that state what students should have attained by the end of various grade in public school (Danish Ministry of Education, 2014a). Some of the objectives for student digital development (related to the subject Danish) are shown in Box 5 below:

After 9th grade, students should know about authors and genres on the internet and about various stages of searching for information. In terms of their skills the aim is that they should be able to-

– make a critical assessment of expert-generated and user-created content

– plan and execute all phases of a search for information

– perform a focused and critical search for information.

After 9th grade, students should also have skills in choosing ICT in relation to dialogue situations. Moreover, they should to be able to discuss the importance of digital technology in their own lives and for society, and ethical questions concerning communication on the Internet and other digital communication facilities (Danish Ministry of Education, 2014b).

Box 5

When addressing citizenship safety, for example, teachers can use the dilemmas associated with going online as a point of departure. In essence, ethics concerns the difference between what people*can* do and what they*actually* do. As such, when students expose themselves or others on the Internet, they candevelop an awareness that they should not do everything they can do.

FLUENT DIGITAL READERS

Each public school is responsible for ensuring the quality of the education in accordance with public school aims. For example, aftercompleting second grade, the student should be able not just to read but also to apply simple pre-reading strategies. These and additional objectives, shown in Box6, should be viewed in conjunction with the municipal aimto increase the student's procurement of digital learning resources.

After 2nd grade, the goal is for students to know about simple pre-reading strategies and be able to apply these strategies, which can benefit their reading on the touch screen. Another goal is that they should know about the structure of websites and be able to find texts by navigating around age-appropriate websites.

After 4th grade, students should know about strategies to read and understand multimodal texts. They should also know about digital profiles, digital communication, and digital footprints. Then there is the ability to operate in a virtual universe.

After 6th grade, the students should have an understanding of techniques for image and full-text searches and the ability to perform such searches. In addition, they should have the ability to assess the relevance of search results on results pages (Danish Ministry of Education, 2014b).

Box 6

As a result of the 1:1 initiative, itis becoming easier for the teacherto vary learning resources and, in turn, to take account of the student's individual learning needs. Digital *resources*and*media*should not be confused (Kress, 2003). A learning resource could be a picture on a wall, on a canvas, or in an e-book. This is the same resource, but in three different media.The 1:1 classroom provides increased access to digital media.Consequently, all students have texts and other materials within arm's reach – the distance to the nearest screen.

Sincestudentsuse the Internet as a distribution medium, they beginsome lessons by decidingfor themselves whichdigital book they would like to read. Provided the wireless connectionworks properly, they can begin the process quickly and are soon able to choose their own books for independent reading by subject or degree of difficulty (Andresen, 2011a). It only takes a moment before the students are immersed in their texts. Observations in 1:1 classrooms indicate that the students concentrate relatively well during reading activities (Andresen, 2011a). An important benefit of the 1:1 classroom is thus extended time-on-task in ‘reading groups’.

Teachers in the 1st grade noted that students of both sexes are highly motivated and hardworking when they log into the portal to find books for independent reading (Andresen, 2011b). Some students selectdigital books with a relatively low readability index, i.e. 3–5, while others opt for books with higher scores. They also choose books based on subjects such as sport, humour, thrillers, love, friendship and adventure. Selectingbooks by subject or readability index generally works well, and students who are instructed to choose books with a certain readability index are self-sufficient in their reading because the chosen texts have a suitable degree of difficulty (Andresen, 2011a). When theyencounter a word they do not already know, they can simply click on it and the computer reads it aloud. Students who require it can also have the whole text read aloud.

In this study, there is no basis for comparison in the form of a control group. Instead, other students at the same stage can act as a national control group. In a standardised test (called OS64) to assess skills in the decoding and reading comprehension of single words, students achieve the following results: in one class, 87% are in the top group of readers, which is above the national average, and, in the other class, 100% are in the top group of readers (Andresen, 2011c).

FLUENT DIGITAL WRITERS

As already mentioned, in the 1:1 classroom, the student’s digital fluency is factored into his/hersubject-specific activities. In general, students use digital devices (for example, tablets and laptops) when working with the multimodal representation of texts, numbers, figures, drawings, photos and sounds. Some objectives concerningthese typesof activities are shown in Box 7.

> After 2nd grade, students should be able to use simple word processing functions and also know about formatting functions. In addition, they should have an understanding of electronic communication in text, images, and sound and the ability to use ICT for everyday communication. They should also understand the relationship between senders and recipients in this communication.
>
> After 6th grade, students should know about cooperation facilities and the potential and pitfalls of communication on the Internet. They should also have the ability to knowledge-share and collaborate on the Internet as well as to evaluate the effects of statements made over the Internet (Danish Ministry of Education, 2014b).

Box 7

In order to meetthese objectives, each student should ideally have access to a digital device with a separate keyboard. However, the primary tool for some students is a tablet without a separatekeyboard. If they do not have access to a separatekeyboard, they can only touch one key at a time. Accordingly, their input becomes both slower and less automated than when they use responsive keys on a separatekeyboard. Therefore, in order to meetthe objective of learning to write, it is arguably better to select a digital device with a separate keyboard (Box 8).

After 2nd grade, the aim is also for students to know about the layout of the keyboard and have the ability to type lowercase and uppercase letters. This could be a practical reason for schools to buy tablets with keyboards and to buy separate keyboards for the tablets they have already.

After 4th grade, one goal is that students should be able to use a keyboard and a word processing program competently procured (Danish Ministry of Education, 2014b).

Box 8

When using a tablet as a 'lean-back device' or to complete smaller writing exercises at the elementary level, students often do not mind that a virtual keyboard takes up space on the screen (Maagaard, 2012). However, it may be an 'ergonomic disaster' to use a tablet without a separatekeyboard as a 'lean-forward device'. While typing on the on-screen keyboard, it may be difficult for students to retain an overview of longer texts or find their way around learning resources or exercises on the screen. Someolder students say it is awkward to work with textbooks and exercises and also tofind space for an on-screen keyboard (Box 9).

"You can *display* documents and PowerPoints, but you can't *edit* them because the keyboard takes up too much of the tablet screen" (Andresen, 2012).

Box 9

A specific purpose of the 1:1 classroom relates to students with reading and writing difficulties. The aim is to avoid marginalising students who find it difficult to handle situations where they have to read and write in order to learn. Regarding students who are slow for their age at reading and writing, there is evidence to suggest that they benefit from predictive text and text-to-speech. As described in the guidance material from the Danish Ministry of Education, all elementary schools are therefore advised to implement these functions (Andresen, 2009).

The first Danish development project at the elementary level to test the daily use of computers for predictive text and text-to-speech was carried out in five schools. At each of these schools, every 3rd grade student used a laptop for reading and writing. The extent of the two categories 'slow reader, many errors' and 'fast reader, poor understanding' was almost the same in the five 3rd grade classes as in Denmark as a whole. However, at all five schools,the students' results in a standardised test (called SL60) to access skills in sentence reading were higher than both the national average and the municipal average (Andresen, 2007).

DISTRACTED DIGITAL LEARNERS

Any innovation will generally have both intended and unintended consequences (Rogers, 2003). An undesirableside effect of the 1:1 classroom is that students can be distracted.In some case, students'go on'social media so often that it takes up almost all of their time (Box 10).

“She’s on Facebook the whole time. Can’t work out how much time she spends on it: Goes in just to take a look; writes if she has done something, like hanging out in the café.”

“Checks for three minutes, then puts it away. She does it the whole time, except when she’s running or sleeping, or singing.”

“Switches on and reads it all: Switches off for five minutes; switches on again and reads updates.”

“He’s on Facebook for five minutes at a time. After ten minutes, he’s on again.”

“He’s on Facebook the whole time – and he means, the whole time. He never turns his computer off. There is always a window open with Facebook on it. And sometimes writes stuff himself.”(Andresen, 2013b)

Box 10

Since the first Danish Schools Act was passed 200 years ago, students have always been able to find ways of amusing themselves when they were bored in class. However,nowadays,in order to distract themselves, students simply need to look at the screen in front of them. It would be unfounded to offer a simple causal explanation for this (such as ‘students get distracted *because* they use tablets and laptops’). In many cases, the real reason for this problem is a lack of motivation and reduced effort, which means that students are bored in the lessons and seek distraction. Consequently, every 1:1 classroom has students who require help to negotiatethe many tempting digital platforms on offer.

In some cases,distraction may be a symptom of the students being so dazzled by the digital narratives that they spend a lot of time playing, checking, and liking updates on the Internet. In other cases, they may lack concentration because they are afraid of missing something on social networks. In some cases, there is also a risk that students become so involved in gaming universes that they develop negative feelings or become aggressive and hostile towards other people. In all these cases, the non-academic use of digital devices can cause students to become distracted and reduce their time-on-task.

If pedagogical measures are to be viable, teachers have to analyse and understand such challenges. In a Danish municipalityin which schools provide iPads to every student, manyteachers have developed an analytical practice thatenables them to understand and reduce educational challenges in the 1:1 learning environment (Andresen, 2013a). They evaluate and improve the impact of the use of tablets on the student’s motivation and effort. After all, the investment in tablets has had a positive effect. 85% of teachers think that the introduction of tablets motivates students to learn (Gammelby, 2012). Likewise, students themselves generally claim that, with tablets, their school activities are easier, more exciting, and less tedious (Maagaard, 2012, p. 94).

CONCLUSION

This article reports findings from an empirical study on 1:1 classrooms. It aims to address the following research question: in which ways can appropriate use of digital tools and media increase the student’s desire to become digitally literate and promote his/her learning?There is evidence to suggest that digital tools and media can promote student learning because the students are suitably challenged. Since every student uses a tablet or a similar device for learning purposes, it is, in principle, possible to achieve more time-on-task with less time wasted than before. However, this rests on the assumption that the wireless connection functions properly and that the students acquire digital literacy.

In many cases, it is an advantage for students to use tablets and similar devices to accesslearning resources with multimodal content. They are often capable of adjusting the content to their current learning needs. Using digital devicesalso has a positive effect on the student's desire to read and learn and on his/her development in general. However, this assumes that there are set targets not just for the student's reading ability but alsothe student's academic grasp of digital learning resources.

It may also be an advantage to use tablets and laptops for practical and creative activities; for example, demonstrating the understanding of academic topics and concepts by choosing multimodal representations and offering more flexibility to thebreadth and depth of the content. Moreover, digital technology may be usedin the classroom as an all-round tool for knowledge-sharing and cooperation,for dialogue between students and teachers, as well as for teachers to comment on the student's work.In practice, students can both increase and decrease their attainment in 1:1 classrooms; this depends on whether and to what extent they learn to navigate the many options and benefit from the most versatile dialogue and sharing toolsthat schools have ever had.

The students' need for extra support changes when they have a tablet or laptop in their hands. Some need less support because they are more self-sufficient at school. Other students may need more support because they lack the basic literacy for their age. For example, they need to hear words or texts read aloud, use predictive text, receive help correcting their spelling and grammatical errors and working with multimodal representations. Some students in this category may lose self-esteem in relation to schoolwork. There are often pedagogical solutions to such problems. However, this assumes that the teacher analyses the educational challenges in order to understand why such challenges arise. A prerequisite for fulfillingthe technological potential is a mind-set that focuses on day-to-day progress and adopts an analytical view of the progress (if any) that is made.

In order to implement the 1:1 classroom successfully, some of the school budget will have to be spent on the purchase of tabletsor laptops for those students who cannot bring their own, up-to-date, ultra-mobile devices into school. However, a larger part of the school budget will have to be spent on enhancing the skills of teachers and allowing them the necessary time to develop an analytical approach to the challenges they face in promoting individual student learning inthe 1:1 classroom.

REFERENCES

Agency for Digitization. (2013). *Folkeskolen skal udfordre den digitale generation*[Public schools need to challenge the digital generation]. Copenhagen: Danish Agency for Digitization. Accessed on April 30, 2015at www: www.digst.dk/Digitaliseringsstrategi/Digitaliseringsstrategiens-initiativer/Folkeskolen-skal-udfordre-den-digitale-generation.

Andresen, B. B. (2007). *Bæredygtig læse- og skriveudvikling. Forskning i tilknytning til projekt 'Computerstøttet undervisning på 3. årg.'* [Effective development in reading and writing; research in connection with the project 'Computer-based teaching in 3rd grade']. Skanderborg: Skanderborg Kommune. Accessed on April 30, 2015 at www: www.skanderborg.dk/Files/Filer/Born_Unge/Itstoettetuv/bentb3klskb.pdf.

Andresen, B. B. (2009). *Den lyttende læser – eksempler på it som støtte til læseusikre elever på folkeskolens mellemtrin* [The listening reader – examples of IT as an aid to unconfident readers in the intermediate classes at public school]. Copenhagen: Danish Ministry of Education. Accessed on April 30, 2015at www: http://pub.uvm.dk/2009/lyttendelaeser/den_lyttende_laeser.pdf.

Andresen, B. B. (2011a). *Tilgang til at fremme elevernes læsning* [Access to promote students' learning]. Accessed on April 30, 2015at www:http://hedskole.blogspot.dk/2011/06/tilgang-til-at-fremme-elevernes-lsning.html.

Andresen, B. B. (2011b).*Erfaringer med en digital bogportal (Superbog.dk)* [Experience with a digital book portal (Superbog.dk)]. Accessed on April 30, 2015 at www:http://hedskole.blogspot.dk/2011/06/erfaringer-med-en-digital-bogportal.html.
Andresen, B. B. (2011c). *Er brugen af it en fordel for elevernes læsepraksis?* [Does the use of IT benefit students' learning?] Accessed on April 30, 2015at www:http://hedskole.dk/digital-laesepraksis/.
Andresen, B. B. (2012). *Evaluering af projekt: Fremtidens undervisningsmidler – e-bøger på skolebiblioteket*. [Project evaluation: teaching materials of the future – e-books in the school library]. Taastrup: Høje-Taastrup Kommune.
Andresen, B. B. (2013a). Transferevaluering af pædagogisk efteruddannelse [Transfer evaluation of teacher training].*CEPRA-Striben* [Journal of the Center for Evaluation in Practice], No. 14, 2013.
Andresen, B. B. (2013b). *Tweens i samfund 3.0 – Roskildeundersøgelsen på fritidsområdet* [Tweens in Society 3.0 – the Roskilde study of leisure activities]. Roskilde: Roskilde Municipality.
Andresen, B.B. (2014): *1:1-skolen – tablet eller tab-let?* Frederiksberg: Frederiksberg Kommune. Accessed on April 30, 2015 at www: www.frederiksberg.dk/~/media/Forside/Politik-og-demokrati/Politikker-og-strategier/Boern/Skolernes%20IT-strategi/Tablet%20eller%20tab%20let.ashx.
Black, P. & Wiliam, D. (1998). Inside the black box: raising standards through classroom assessment. *Phi Delta Kappan*, Vol. 80.
Bundsgaard, J., Pettersson, M., & Puck, M. R. (2014). *Digitale kompetencer* [Digital competencies]. Aarhus: Aarhus Universitetsforlag.
Danish Ministry of Education. (2014a). *Forenkling af Fælles Mål*[Simplification of the Common Objectives]. Accessed on February 1, 2014 at www: http://www.uvm.dk/~/media/UVM/Filer/Udd/Folke/PDF13/Faelles%20Maal/130923%20Master%20til%20praecisering%20og%20forenkling%20af%20Faelles%20Maal.pdf.
Danish Ministry of Education. (2014b). *Forenklede Fælles Mål i faget Dansk* [Simplified Common Objectives for Danish]. Accessed on April 30, 2015 at www: http://www.emu.dk/sites/default/files/F%C3%A6lles%20M%C3%A5l%20for%20faget%20dansk_1.pdf.
EVA. (2012). *Fælles Mål i folkeskolen* [Common Objectives in public schools] Copenhagen: Danish Evaluation Institute, EVA.
Gammelby, L. (2012). *Sammen skaber vi udfordrende læringsmiljøer med plads til fællesskaber, fornyelse og fordybelse* [Together we can create challenging learning environments with space for communities, renewal, and exploration]. Accessed on April 30, 2015 at www: www.oddernettet.dk/dokumenter/Nutidens_digitale_skole_1_1.pdf.
Gardner, H. (2000). *Disciplin og dannelse – betydningen af det sande, det smukke og det gode* [Discipline and education – the good, the bad, and the ugly]. Copenhagen: Gyldendal Education.
Haddon, L. & Livingstone, S. (2012) *EU Kids Online: national perspectives.* London: The London School of Economics and Political Science. Accessed on April 30, 2015at www:http://eprints.lse.ac.uk/46878/.
Hargittai, E. (2012). *Digital Natives or Digital Naïves? Internet Skills among members of the "Net Generation".* Accessed on April 30, 2015at www: http://epresence.univ-paris3.fr/1/watch/204.aspx.
Hattie, J. A. C. (2009). Visible Learning. A Synthesis of Over 800 Meta-Analyses Relating to Achievement. Abingdon: Routledge.
Kress, G. (2003). *Literacy in the new media age.* Oxford: Routledge.
Maagaard, M. (2012). *iPads som redskab for læring* [iPads as a learning tool]. Research thesis at Aarhus University.
Means, B., Toyama, Y., Murphy, R., Bakia, M., & Jones, K. (2009). *Evaluation of Evidence-Based Practices in Online Learning. A Meta-Analysis and Review of*

Online Learning Studies. Washington, D.C.: U.S. Department of Education. Accessed on April 30, 2015at www: http://www2.ed.gov/rschstat/eval/tech/evidence-based-practices/finalreport.pdf.
Mortimore, P., David-Evans, M., Laukkanen, R., & Valijarvi, J. (2004). *OECD-rapport om grundskolen i Danmark* [OECD review of public schools in Denmark]. Danish School Agency discussion papers no. 5. Copenhagen: Danish Ministry of Education. Accessed on March 10, 2007 at: http://pub.uvm.dk/2004/oecd/index.html.
Nordenbo, S. E. (2011). Forskning i klasserumsledelse [Research into classroom management]. KvaN 90. Accessed on February 1, 2014 at www: http://samples.pubhub.dk/9788790066970.pdf.
Rogers, E. M. (2003). *Diffusion of Innovations*. Fifth Edition. New York: Free Press.
Søby, M. (2013). Editorial: Synergies for better learning – where are we now? *Nordic Journal of Digital Literacy No. 01–02.*
Tapscott, D. (1998). *Growing up digital.* New York: McGraw-Hill.

Bent B. Andresen, Ph.D., is associate professor at the Aarhus University, Department of Education. His research interests include teaching, learning, and the application of digital technology in educational settings. In particular, they include digital literacy and citizenship. He has conducted research funded by, among others, the Danish Government, UNESCO, and European Commission, and he is member of international research networks under the auspices of IFIP and ASEM.

Automatic analysis of students' solving process in programming exercises

Aivar Annamaa, *aivar.annamaa@ut.ee*
University of Tartu, Estonia

Annika Hansalu, *anzza8i8@ut.ee*
University of Tartu, Estonia

Eno Tonisson, *eno.tonisson@ut.ee*
University of Tartu, Estonia

Abstract

In most cases student programming exercises are assessed based solely on the final program and a teacher does not know how students actually arrived at the solution or what steps they completed during the process. In this paper we review different approaches to observe how students solve their programming exercises and see how integrated development environment Thonny can be used to observe exactly what novice programmers do while programming. This is followed by a discussion of the benefits of (automatic) analysis of students' solving process.

Keywords

Computer science education, automatic analyses of solving process, novice programmers

INTRODUCTION

Solving of programming exercises has a title role in computer science education. Usually, the students' work is evaluated on the grounds of a completed program. The program could be assessed by the system for automatic assessment, which is almost inevitable in case of a large number of students. Human evaluation of program code is used, especially in case of beginner courses, in order to give more useful and specific feedback, which could be impossible with automatic assessment. In addition to the final program, observation of the solving process is also very important to help rationalise solving, save time, increase motivation, etc. The teacher could, in principle, observe the solving process by standing next to the computer when solving takes place in a classroom. This would be impossible if solving is processed outside of classroom and complicated in case of a larger number of students.

This paper is focussed on automatic analysis of the students' solving process in programming exercises. The aim is to outline approaches that have been used for analysing the solving process. The approaches differ by granularity of data, for example. It is remarkable that the solving process is also (or even especially) interesting and useful in software engineering, not only for educational reasons. After the section on data collection, a brief introduction on potential benefits of such analysis for educational purposes is presented. It is closely related to different types of learners (like stoppers, tinkerers and movers (Perkins et al., 1989)). The innovative part of the paper introduces the logs from a new educational integrated development environment (IDE) Thonny (Annamaa, 2015). The conclusive section describes some possible future works as well.

DATA ON THE SOLVING PROCESS

An analysis of the programming process must start with data collection. The most straightforward approach would be recording the screen and possibly also the subject's body language and voice for later analysis. This approach combined with subsequent manual video analysis, or letting an expert directly monitor and comment on the process, is very flexible but also very expensive.

It would be more scalable to collect process data in a structured form allowing automatic analysis. In software engineering research, the analysis is usually based on source code snapshots extracted from version control systems (VCS). This makes data collection very simple, as using version control is a standard practice in professional context and therefore it is not necessary to set up a separate data collection mechanism. However, as Negara et al have explained (Negara 2012), many interesting aspects of code evolution get lost when we analyse only code snapshots at commit points.

Relying on conventional use of version control systems is even more problematic in the context of programming education, because requiring that students use a VCS can create an excessive cognitive load. Furthermore, beginner programming exercises are usually too small to be analysed on a scale where code commits usually mark the completion of a feature or a bug fix. For these reasons, the analysis of educational programming process is usually supported by a submission system, which may allow several submissions for the same task, for example Web-CAT (Edwards, 2009). More granular and diverse data can be collected by specifically designed or instrumented programming environments, which gather code snapshots on each compile or run.

Vihavainen et al show that in many cases even snapshots collected on each compile or run may be insufficient for getting realistic picture of the programming process, and it makes sense to collect detailed info about all relevant user actions, including key strokes and mouse presses, together with their time stamps (Vihavainen, 2014). If necessary, code snapshots can be constructed from these low level events for arbitrary points in time. There exist several such programming environments or IDE add-ons, for example, Fluorite for Eclipse (Yoon, 2013). In this paper we are mostly interested in these kinds of systems.

POSSIBLE BENEFITS

One could collect data on the solving process but it is important that the data supports improvement of learning and teaching. It is possible to describe different types of novice programmers. Perkins et al (1989) have divided them into three different learner types based on their problem solving strategy: stoppers, tinkerers and movers. Stoppers are students who when faced with a problem tend to give up faster and ask for help rather than trying to solve the problem themselves first. Tinkerers are students who solve problems by experimenting and making small changes in the code while hoping to get the code working. Movers on the other hand are students who move towards the right solution. They have a certain idea for the solution and they are not afraid to try different approaches if the first one does not take them closer to solving the problem.

Cardell-Oliver (2011) has noticed on her students that students who are stoppers do not tolerate too much negative feedback on their work because, if they get too much negativity at once, they tend to turn into non-starters. On the other hand, because of their problem solving strategies, tinkerers and movers benefit the most from detailed feedback. When tinkerers see that they are getting fewer errors it probably means that they are on the right track.

Other authors (Housseini et al, 2014) have found that the classification of students into stoppers, movers and tinkerers by Perkins et al is not enough. They concluded that it would be more accurate to divide student problem solving strategies into four groups: builders, massagers, reducers, and strugglers. Builders are students who constantly add new concepts to their code and by doing that improve correctness of their code. Massagers are similar to builders but they have periods where they only do small code changes without adding or removing any new concepts. Reducers are students who in the beginning take a completed code from somewhere (i.e., from a previously solved exercise) and start removing unnecessary concepts. They remove things until they get the right solution. Strugglers are the students who struggle with their code and make all kinds of changes in their code but they tend to have not enough knowledge to get their code working or find the mistakes and fix them.

Analyses of logs could provide valuable information for more specific classification of novice programmers' solving style. For example, it is possible to observe a particular learner for longer time and get information about persistency of behaviours and impact of feedback. The ultimate task of teaching – provide as tailored feedback as possible – could also be accomplished better with the help of (hopefully automatic) analysis of logs. At first glance, even only awareness of his or her possible weaknesses in the solving process could help a student (and a teacher).

THONNY LOGS

Thonny is a new Python IDE developed in the University of Tartu, designed for learning and teaching programming. Besides program editing and execution capabilities, its most prominent feature is support for program animation – the user can easily step through the execution of the program and follow the changes in the program's runtime state (including global and local variables and call stack).

In order to gain better insight into the solving process, we made Thonny log all interesting events that happen in its window (although we might not need them all because we are still researching which data will give us the needed information). For each usage session Thonny creates a log file containing descriptions of the actions performed by the user. These actions include loading and saving files, modifications to the program text (paste can be distinguished from typed text, for example), program executions, writes into and reads from the program's standard streams, stepping commands, losing and gaining the focus of Thonny window, etc. Each action gets recorded together with its time stamp. The collected information can be used to replay the whole process of program construction and the activities in the shell. For this Thonny provides a separate window where one can choose a log file and see the events replayed at selected speed.

Thonny has been used for one semester in the Programming (Computer Science 1) course at the University of Tartu. The course had 280 participants; half of them were first-year computer science students while the other half consisted mostly of students from related fields (mathematics, statistics). Its program animation features were initially used only in the lectures for demonstrating Python's run time behaviour, but many students chose to use Thonny also on their own for solving exercises in labs and at home.

Before a midterm examination we offered our students extra credit if they solved the programming exercises in Thonny and sent us the log files describing their actions during the midterm. We got logs from 44 students. For proof of the concept we created a small summary of the logs and from this data we learned, for example, that

- a student who used deletion commands two times more often and undo command 10 times more often than students in average, produced one of the best solutions;
- one third of the program executions generated error messages, one third of the errors were syntax errors;

- one of the top students got error messages for 90% of his program executions, another top student for 20%;
- only 14 students had used Python shell for executing statements or evaluating expressions;
- 27 students used the program animation features, 8 of them invoked more than 1,000 animation steps during 100 minutes.

Besides analysis of numeric summaries, we created a visualization of all users' editing and program execution actions on a unified timeline. An extract from this is shown in Figure 1.

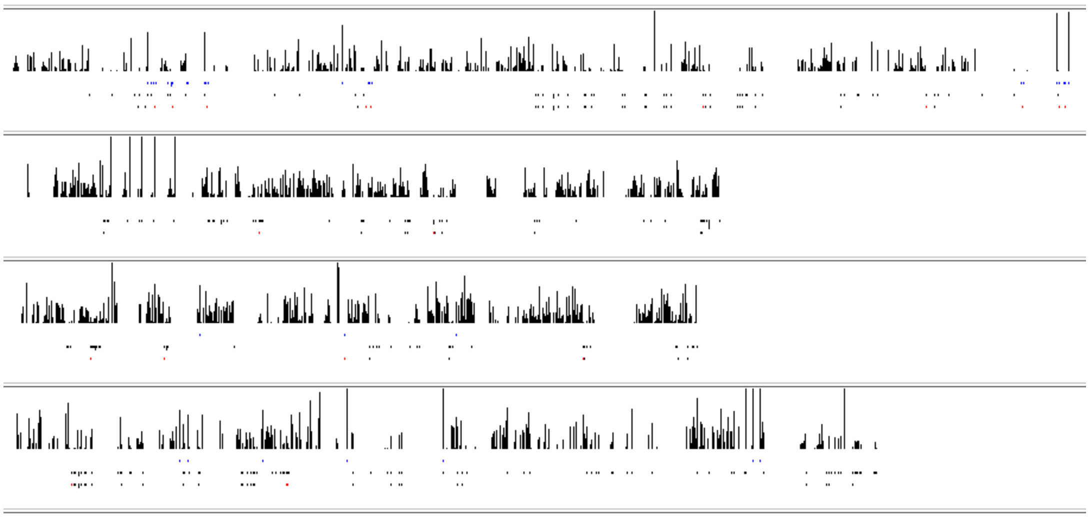


Figure 1: Visualization of solving processes

Different lanes depict the actions of different students (from top to bottom) on a linear time scale (left to right). Height of the black bars corresponds to the number of characters entered during a given time slice; small blue bars indicate the number of pastes (note that one student did not paste text at all). Two bottom rows on each lane show the number of program runs and the result of the run (a dot in the lowest row indicates a syntax or runtime error). We hoped to find some distinctive similarities in the action patterns of stronger or weaker students but instead we found that two very similar action patterns can result in a very high or a very low grade.

After finding some curious cases in our numeric summaries (for example, a well performing student was using the “Undo” command 10 times more often than other students), which were not explained by the action pattern visualization, we replayed the respective logs in Thonny, i.e., we observed how the actual text appeared in the editors and the shell. In most cases this helped us understand why data was different from average (e.g. the student mentioned previously was using undo to get rid of recently entered words with typos).

CONCLUSION

Former studies and our experiences both show that in the context of programming education it is worthwhile to collect fine grained data about learners’ actions during the programming process, and in principle this provides us with the same opportunities for giving feedback as commenting video logs or direct observation. At the same time, fine grained data probably create better opportunities for automatic analysis, for example, based on data mining. Some authors have proposed using data mining techniques to uncover the higher level meaning of a sequence of low level user actions, but this remains a difficult problem. It would be of great help, if learners could easily mark the points in time, at which they completed one micro task (e.g., renaming a variable). It is not difficult to provide a keyboard shortcut for this, but most users likely need extra motivation for using this shortcut often enough to be useful. One possible reward for this key press could be saving the current snapshot of the code into a local history, which can be revised when necessary.

In addition to assessment of the learners' final program, (automatic) analysis of the solving process creates additional benefits for more adequate feedback and evaluation. Considering that in some cases students never submit their work (Vihavainen 2014), it would make sense to keep submission of logs separate from submission of the solutions for grading. It would not be the main purpose but analysis of the solving process could still help to identify illegal attempts in examinations, for example...

It is worth considering how to integrate analysis of programming logs more directly into the teaching and learning process. One approach would be integrating process analysis into automatic feedback systems, which usually analyse only the end result of a programming session. Such a system could, for example, warn the learner if it detects a possibly ineffective working pattern. Process data could be used in gamification – for example, the learner who writes a correct solution with very few deletes and corrections would receive a "Steady hand" badge. In labs, process data could be streamed into a visualization on the teacher's screen to make it easier to identify the students who need help.

We are convinced that automatic analyses of students' solving processes could provide various opportunities that have not been very thoroughly studied at this moment. Furthermore, Thonny seems to be a suitable environment for future experiments (and can be improved if necessary).Of course it needs to be specified which log data is useful to us. One way is to analyse what was the student doing before receiving an error message and how he/she responded to that. Also we have already experimented a little how to give students more specific instructions for solving their exercises and finding their mistakesbased on their programming logs.

This research was supported by the European Union through the European Regional Development Fund.

REFERENCES

Annamaa, A., (2015), Source code and installers of Thonny IDE, https://bitbucket.org/plas/thonny/

Cardell-Oliver, R. (2011), How can software metrics help novice programmers? In Proceedings of the Thirteenth Australasian Computing Education Conference-Volume 114 (pp. 55-62). Australian Computer Society, Inc.

Edwards S. H., Snyder J., Pérez-Quiñones M. A., Allevato A., Kim D., Tretola B., 2009, Comparing effective and ineffective behaviors of student programmers, ICER'09

Hosseini, R., Vihavainen, A., Brusilovsky, P. (2014), Exploring Problem Solving Paths in a Java Programming Course, University of Sussex

Negara, S., Vakilian, M., Chen, N., Johnson, R. E., Dig, D. (2012), Is it dangerous to use version control histories to study source code evolution? ECOOP'12

Negara, S., Codoban, M., Dig, D., Johnson, R. E. (2014), Mining fine-grained code changes to detect unknown change patterns, ICSE'14

Perkins, D., Hancock, C., Hobbs, R., Martin, F. & Simmons, R. (1989), Conditions of learning in novice programming, New Jersey

Vihavainen, A., Luukkainen, M., Ihantola, P. (2014), Analysis of source code snapshot granularity levels, SIGITE'14

Yoon, Y., Myers, B.A., Koo, S. 2013, Visualization of Fine-Grained Code Change History.

AivarAnnamaa was born in Tartu, Estonia in 1979. From 2001 till 2008 he worked as a programmer. At 2008 he started PhD in Computer Science in University of Tartu and is currently working as teaching assistant. His research interests are related to programming languages and teaching.

Annika Hansalu was born in Kuressaare, Estonia in 1989. She is going to graduate as a mathematics and informatics teacher in 2015. Previous two years she has been teaching programming to beginners in University of Tartu and also one year in Tartu Tamme Gymnasium.

Eno Tonisson was born in Tartu, Estonia in 1969. He is a lecturer in the Institute of Computer Science of the University of Tartu. He graduated as a mathematics teacher in 1992 and received his master of science degree in mathematics in 1996 from the University of Tartu. His current research themes include didactics of programming; use of computer algebra systems in mathematics education; career choices of students of computer science and information technology.

He has worked as a mathematics teacher of secondary school for 8 years.

Eno Tõnisson started working at the University of Tartu in 1994.

CS High School Curriculum – A Tale of Two Countries

Tamar Benaya, *tamar@openu.ac.il*
The Open University of Israel, Faculty of Mathematics and Computer Science, Israel

Valentina Dagienė, *valentina.dagiene@mii.vu.lt*
Vilnius University Institute of Mathematics and Informatics, Lithuania

Ela Zur, *ela@openu.ac.il*
The Open University of Israel, Faculty of Mathematics and Computer Science, Israel

Abstract

Computer Science (CS) education in high schools is entering the fifth decade of its existence. Israel is one of the first countries which started to offer CS courses in high schools in the middle of the 1970s. Many European countries jumped in this process a decade later – Lithuania is among them. Both countries put a lot of effort in developing CS curricula and establishing assessment (maturity examinations). Nowadays there has been considerable activities surrounding CS education on all levels therefore we suggest to take a look at the experience of these two countries. In this paper we focus on the CS curriculum for high schools both in Israel and in Lithuania followed by a description of the final exams, some statistics regarding students' participation and achievements as well as exam evaluation. We sum up with a discussion comparing both curricula and some insights.

Keywords

Computer Science Education, K-12, Computer Science Curricula, Assessment.

INTRODUCTION

At present, all evidence points to a significant boom in Computer Science (CS) education at the high school level. This boom is most clearly manifested in the shifting education attention from information technology (IT) to CS (or Computing, or Informatics as it is called in many European countries) and in increasing articles devoted to the questions of CS education in schools.

In today's world, and even more as we move towards an ever more computing-intensive world, being familiar with CS is as critical to every citizen as being familiar with traditional scientific disciplines. To be prepared for the jobs of the 21st century, students must not only be digitally literate but also understand key concepts of informatics. Therefore the emergence of CS as a subject is more and more undoubted.

Education policy makers are becoming inspired by the challenges posed by the CS Teacher Association in USA (Seehorn et al., 2011) and Computing At School in the UK (CAS Working Group, 2012;The Royal Society, 2012). The new Computing curriculum in UK (CAS Working Group, 2012) puts the subject on an entirely new footing, as the "fourth science" at school. It offers new opportunities for professional development for teachers and better education for students.

So, in the last four decades, there has been considerable activity surrounding CS curricula on all levels: beginning with ACM Curriculum Committee on CS (ACM Curriculum, 1968) through Computing Curricula 1991 (Tuckeret al., 1991) and up to Computing Curricula 2013 (Joint IEEE, 2013). Notable is the high-school curriculum designed by the special ACM task force (Merritet al., 1994), and in particular the new K-12 curriculum (Tuckeret al., 2003). The goal of the new K-12 curriculum (2nd edition), was to create a 4-level curriculum that could be widely disseminated, accessible to every high school student in the US. Its aim was to enable every CS student to understand the nature of the field and the place of CS in the modern world. Students need to understand that CS combines theoretical principles and application skills. They need to be capable of algorithmic thinking and of solving problems in other subject areas and in other areas of their lives.

In light of the recommendations presented above, we can see that different countries developed unique curriculums for high school CS education, see for example ACM Transactions on Computing Education (TOCE) special issue on Computing Education in (K-12) Schools (*ACM Transactions, 2014)*. Although much had been previously written about CS education at schools, we would like to share Israel's and Lithuania's long experience in teaching CS at high school. The paper deals with the main issues of CS education and pointsin the following directions: (i) CS curricula. (ii) Assessment and examination. (iii) Discussion and lessons learned.

THE HIGH SCHOOL CS CURRICULUM

Teaching CS inIsraeli schools was offered since mid-1970s and it included a solid detailed course in basic programming and several electives (Gal-Ezer et al., 1995). A new committee was formed in 1990 in order to decide on the general topics and principles and to prepare detailed syllabi for a new curriculum. "The program's emphasis is on the basics of algorithms, and although it teaches programming it does so only as a means for getting a computer to carry out an algorithm" (Gal-Ezer, Harel, 1999). The committee that developed the CS high school curriculum included computer scientists who were also involved in various kinds of educational activities, experienced CS high school teachers, and CS education professionals from the ministry of education (Gal-Ezer, Harel, 1999). The current program committee continuously updates the CS high school curriculum which is based on the foundations laid by the 1990 committee, for example shifting, from procedural to Object Oriented languages.

In Lithuania, the directive to teach informatics came from Moscow in 1985. An outstanding scientist of this time,A. Yershov foresaw as early as 1980 that computers would cause worldwide changes and that Russia could supply schools with theoretical knowledge of CS. Lithuaniahad a good understanding of the situation and experience in the area: there had been a school for young programmers at the Institute of Mathematics and Informatics since 1981 (Dagys et al., 2006). Work of Lithuanianresearchers in the field of the methodology of programming was well known in the Soviet Union. Lithuania runs weekly programs on national TV on teaching CS – half is dedicated to an introduction of computing concepts and the other half is designed to teach how to write algorithms and code. Plenty of textbooks on teaching algorithms and programming based on attractive tasks were developed. Methodology on teaching CS at secondary school level was strong and well designed by Lithuanian's researchers.

The Israeli Case

In Israeli high schools, every student must study at least one subject in depth, in addition to general studies which include Mathematics, English, History, Literature etc. The highest level of studies is the 5-point (as opposed to 3 or 4-point) program, each point representing 90 class hours. CS is one of the subjects that high school students can select to study in depth. The CS program starts in 10th or 11th grade depending on the high school.

The Israeli CS high school curriculum was designed in the early 1990s and first implemented in 1995. One particular principle underlying the curriculum is the interleaving of theoretical principles with application skills. This interleaving notion is specifically termed in the Israeli curriculum as the "zipper approach" (Gal-Ezer et al., 1995). The curriculum has two versions, a 3 and a 5-point version.

The 3-point program includes two mandatory core units (180 hours), *Foundation of Computer Science 1 and 2* (denoted by FCS1 and FCS2), which present the foundations of algorithmic thinking and programming. The third unit (90 hours), *Second Paradigm or Application*, introduces the students to a second programming paradigm or application. This unit has several alternatives for the second paradigm such as: logic programming, functional programming and low-level programming; and several alternatives for the application area such as: Internet programming, computer graphics and information systems.

The 5-point program is intended for more advanced students. It includes the 3-point version and the fourth mandatory unit, *Data Structure*s (90 hours), is an extension of the first two units (FCS1 and FCS2) and it concentrates on data structures and abstract data types. The fifth unit *Theory* (90 hours) can be chosen from several alternatives such as: object oriented programming, computational models and operation research. A detailed description of the program is given in (Gal-Ezer, Harel, 1999; Gal-Ezer et al., 1995).

The Lithuanian Case

In Lithuania CS is named by Informatics (*informatika*). Teaching informatics started with programming. The goal of teaching programming is problem solving transfer, also programming is the best way to build a language for instructing (communicating with) a machine. A. Cohen and B. Haberman have declared CS as a language of technology (Cohen, Haberman, 2007) and we totally agree with that.

A significant role in designing methodology for teaching programming has been played by the scientists of Lithuania. Already in the end of 1970s, a school students' education in programming by using postal services was drafted. Established in 1981, the Young Programmer's School by Correspondence was a unique school for high school students to learn programming (Dagys et al., 2006). The lessons of programming were published in the biggest daily newspaper (!) of Lithuania. They took nearly half a page of the newspaper a few times per month for a number of years (1979-1983). There was also a program on LithuanianTV – a half hour a week for teaching algorithms and programming. The activity of the Young Programmer's School was one of the first examples concerning informatics and had a strong impact on many phenomena related to informatics' teaching, such as development of contests and Olympiads in Informatics (Dagienė, 2006).

In 1985-1986 Informatics was declared as a compulsory subject in high schools of the Soviet Union (in Lithuania also). The first textbook "The basics of Informatics and Computing Techniques" was written by famous Russian informatics professors. While translating to Lithuanianlanguage we extended by adding a chapter on Pascal programming language which had not been recognized at this time inRussia.

Informatics curriculum was aimed at developing algorithms, thinking skills, abstraction and automation of solving tasks (now we can recognize computational thinking but this trend has appeared ten years later). We created a few hundred interesting and attractive tasks on learning programming; most of them were connected with student activities and connected with other science subjects. The tasks were translated and used in many high schools in XX.

As a part of the Education Reform in 1997, the Informatics core curriculum went through a major revision and it was expanded from teaching two years to four years (in total 136 hours) with more focus on application and the processing of information (mainly, text processing).

In regard with the changed role of the IT as well as with the needs of pupils and school communities, the curricula of all subjects were substantially revised and renewed in 2005: subject title “Informatics” was changed to “Information Technologies”.

However, the 34-hours module on introducing algorithms and programminghas been implementedfor grades 9 or 10 in high school. The course is aimed at summarizing and systematizing students’ knowledge on algorithms and drawing attention to their application and programming (Table 1).

Grades 9–10	Basic topics
Elements of algorithms and programming	Conception of algorithm, ways of writing; Programming languages, compilers; Preparation of algorithms, coding and running the program; Dialog between program and user; Entering and output of data, printing formats; Main actions of algorithms: assignment, loop; Simple data types; Stages of program development; Control data and correctness of program; Programming style and culture; Simplest algorithms and their programming;

Table 1:Lithuania - the optional module on programming for grades 9-10

The IT course for grades 11-12 in high school is being essentially revised. Several optional modules mostly oriented to the requirements for study courses in higher educational institutions are being developed. Developing algorithms and learning programming is one of the optional modules (Table 2).

Grades 11–12	Basic topics
Algorithms	Data I/O; Calculation of sum, product, quantity, and arithmetic mean, search of maximal (minimal) value(s); Logical expressions; Ability to modify algorithms according to the particular data structures; Boolean data types; Relational Operators; Logical operands.
Text files	Creating, Editing, Reading, Writing and Closing files; EOL and EOF; Enumerated types; Sub-ranges; Operations on ordinal types.
Procedure and functions	Procedure & function, Parameters & arguments; Standard math procedures and functions; Procedures and functions related to files.
Arrays	One dimensional arrays, array assignment; Using loops; Searching; Sorting; Arrays & Subprograms; Two-dimensional arrays, Reading, writing and manipulating.
Strings	Fields in a record; Array of records; Characters and strings: I/O, comparison, division, combination, sorting; Char and string arrays.
Records	Processing records; Nested records; Record variants.
Programming technique	Programming steps: writing, coding, debugging, testing, improving; Programming style and code alignment; Effectiveness; Program documentations; Arrangement of dialogs.

Table 2:Lithuania - the optional module on programming for grades 11-12

Today, one of the challenges of theCS curriculum is to catch up to the new technologies, to go deeper in its understanding and adjust that to rapidly changing markets and users’ expectations. Moreover, as a starting point, the initial preparation of pupils must also be taken into account.

Since 2005, the main attention in Lithuanianschools is being paid to satisfy user’s needs and to develop computer literacy. Teaching of the basics of informatics as a mandatory part has become quite poor. Students get familiar with the basic knowledge on informatics in grade 5 or 6, when they have a *Logo* or *Scratch* course and in grades 9 and 10 with focus on understanding simple algorithms and coding. The teaching process in Lithuaniadepends very closely on the knowledge and activeness of the teachers themselves. However the optional modules on programming and related topics are available in high school and in some schools also in lower grades. Especially learning coding became more and more popular among pupils with focus on web design and programming of mobile devices.

Comparison and lessons may be learned

Main principles and components of the CS curricula in both countries are similar: more than 2/3 of CS curricula in both countries are devoted to algorithms and programming. In particular, programming can now be interpreted as a component of an emerging new form of literacy (Vee, 2013); as a tool to conceive and create things, to develop creativity (Resnicket al., 2009); as a way for children to widen their experience and experiment with their own ideas (following, in a sense, Papert'sMindstorms' perspective).

Teaching programming has been designed very carefully in both countries: from developing teaching resources, exercises, handbooks, computer tools to teacher training and deep connection between educators and researchers.As we have noticed teaching programming has been focused on CS concepts and building understanding. Pupils are asked to recognize and use variables, data types and data structures, understand and apply control statements: assignment, condition, repetition and procedures. Rather than increasing various CS concepts both countries chose the way “Less is More”: learning less concepts but making more activities and practise.

Beside CS curricula at schools both countries have a long tradition of suggesting to pupils many different outreach activities in the CS field, especially programming clubs and contests such as Olympiads (International Olympiad, 2014) and Bebras (International Contest, 2014).

In the next section we describe the matriculation and final exams both in Israel and Lithuania.

MATRICULATION/FINAL EXAMS

The IsraeliCase

All high school students are required to take matriculation exams in the main subjects studied in high school. The Israeli matriculation exams are similar to the American AP exams in that they are external nationwide exams. Internal high school exams are used to prepare students for the national matriculation exams. The final grade in the subject tested is calculated as the average of the matriculation exam and an internal grade which is based on the internal exam and the student performance throughout the year. It is important that teachers will be familiar with the matriculation exams in order to prepare their students in the best way possible (Drysdaleet al., 2005). The questions in the matriculation exams, like the AP exams, should test the intended concepts accurately, unambiguously, and without bias (Huntet al., 2002).

The content of the matriculation exam of FCS1 and FCS2 reflects the foundations of algorithmic thinking and programming. The duration of the exam is three hours. The exam is divided into three sections according to the Bloom taxonomy.

- The first section contains 5 mandatory ten point questions which test basic skills such as knowledge and comprehension.
- The second section includes 3 fifteen point questions which the students are required to answer two of them. The questions in this section are application questions which require the students to solve problems to new situations by applying acquired knowledge. The questions in this section may require writing a small program, or writing a sub-program and demonstrating its use or tracing a given program. This section requires the use of sequential and/or nested patterns.

- The third section includes 2 twenty point questions from which the students are required to answer one. The questions in this section require analytic and synthesis skills. This section requires writing a complete program which includes: defining appropriate sub-tasks, defining main variables and data structures and implementing the code including documentation.

For the third unit *Second Paradigm or Application* the students are required to prepare a project according to the units' topic and requirements. The students present their project to external examiners who are usually CS high school teachers from other schools. The students must defend and run their projects and answer questions posed by the examiner.

The fourth and fifth unit (*Data Structure*s and *Theory*) have a combined 3 hour exam. The first part deals with *Data Structure*s and the second part deals with the *Theory* unit. In each part the students are presented with four questions from which they must select two.

The Lithuanian Case

Exams in school cause contradictory feelings. No doubt that having a maturity exam increases the value of the subject. School students and teachers often give more respect to exams than to the process of learning. Besides, it is better for pupils to have more choice for choosing exams. In 1995, the informatics maturity exam was developed. Informatics as a separate subject was taught for many years in Lithuanianschools thus to establish the maturity exam in informatics was a natural process. Discussion on informatics exams has been presented in (Blonkis, Dagienė, 2008).

The main goal of the Informatics exam is to encourage students to take interest in programming. The demand for programmers is considerable. Programming as a creative process is being comprehended by learning to write programs from one's as early as possible youth upwards. Algorithmic and structural thinking skills greatly influence the conception of the exact sciences.

The results of the Informatics exam are being recognized when choosing studies of informatics or informatics-related specialties at Lithuanianuniversities. Those, who pass the Informatics exam successfully, have wider possibilities to enter CS-related studies in higher education. At the same time it checks whether student have the aptitude for studying informatics: there are many first year students who quit their studies since they find programming too hard to understand and an uninviting occupation for themselves.

Lithuania's maturity programming exam is an interesting use case of semi-automatic evaluation. Research on exam data demonstrated that this approach is rather effective and still provides good quality evaluation. However this type of evaluation is still not very popular among CS teachers and the outcome of this use case can be rather interesting for the community.

The Informatics exam consists of two parts: the larger part (75%) is allocated to programming, while the rest (25%) concerns the issues of computer literacy. The programming part consists of test (25%) and two practical tasks (50%). The aim of the programming test is to examine the level of students' knowledge and understanding of the tools required in programming (elements of programming language, data types and structures, control structures, basic algorithms).

The Informatics exam focuses on: knowledge and understanding – 30%, skills – 30% and problem solving – 40%. The problems are oriented towards the selection of data structures and application of basic algorithms to work with the created data structures.

In Lithuanian schools, each subject's exam has its own curriculum, which is more concrete than the general subject's curriculum. The curriculum of Informatics exam closely corresponds to the content of the programming module. Three main fields are emphasized: algorithms, data types and structures, and constructs of a programming language (Table 3).

The main attention is being paid to the abilities to choose the proper data types and data structures, also to the implementation of the algorithms and developing the programs.

<table>
<tr><th>Algorithms</th><th>Data structures</th><th>Programming language (Pascal)</th></tr>
<tr><td>Calculation of sums, product, quantity, and average; Search of max/min value;
Data I/O; Sorting: Modify algorithms according to particular data structures.</td><td>Integer, real, char, Boolean, and string;
Text file; One-dimensional array; Record; Creating simple data structures.</td><td>Program structure; Documentation; Variables; Assignment; Relational & Logical operations; If statement; Loops; Compound statement; Procedure & function; parameters & arguments; Standard math procedures & functions; Procedures & functions related to files.</td></tr>
<tr><td colspan="3">Programming environment. Technology of structural (procedural) programming. Testing. Program documentations. Arrangement of dialogs. Program writing (style)</td></tr>
</table>

Table 3: Components of curriculum of Informatics exam

An exam is not the best way of teaching students, – it seems to be late. We have noticed something different. The students who intend to take the programming exam choose the programming module a year before and try to follow the exam model while studying. In other words, if a lot of attention is paid to writing programs, if there are many tasks of algorithms and data structure selection in the exam, the students pay much attention to the mentioned points while learning. Therefore, the exam performs an educational function.

The following section presents some statistics regarding students' participation and achievements and exam evaluation.

SOME STATISTICS AND EVALUATION

The Israeli Case

The data in this section was taken from the Publications of the Israeli Central Bureau of Statistics and from the Science and Technology Department in the Ministry of Education (Publications of Central Bureau, 2013; Science and Technology Office, 2014).

The following statistics refer to the percentage and grades of students studying CS in 2012:

The percentage of students studying the 5-point CS program in high school is 10.4% and the percentage of students studying only the 3-point CS program in high school is 5%.

- 97% of the students who took the CS matriculation exams passed the exams. The percentage of females who passed these exams is slightly higher than the percentage of males who passed the exams.
- The average final grade of *FCS1* and *FCS2* examwas 88, while the average final grade of the *Data Structure*s and *Theory* exam was 80.
- The Theory exam was distributed as follows:
- 67% selected *Computational Models* and achieved an average grade of 81.1
- 12% selected *Object Oriented Programming with C#* and achieved an average grade of 81.1

- 8.4% selected *Object Oriented Programming with Java* and achieved an average grade of 80.6
- 10.4% selected *Computer Systems* and achieved an average grade of 76
- 2% selected *operation research* and achieved an average grade of 82.5

The Lithuanian Case

The practical part of the Informatics consists of two tasks – students have to write programs for the given problems. The practical tasks constitute 50% of all points. The main aim is to examine the students' ability to master independently the stages of programming activities, i.e. to move from the formulation of the task to the final result.

Obviously, the contest system from olympiads can be useful, but it cannot be used without significant changes. A new automated evaluation system with all the requirements met was developed. Application of the evaluation operates with packages of solutions. Each solution must be compiled, and then run with several data sets. The answers provided for all these data sets must be compared with the correct one.

The exam may be approached in two ways: on the one hand, it is the evaluation of the results achieved by a student; on the other hand, it could heighten the motivation to learn. Both must be considered when planning the exam. The exam should be prepared so that it measures the competences needed for further studies in CS. The exam is based on the optional module of the basics of programming which consists of four parts: 1) introduction – basic elements of programming; 2) data structures; 3) developing algorithms; 4) testing and debugging programs.

Students should demonstrate understanding of existing code (Lister et al., 2004). According to many years of experience, the exam has settled structure: 30% is allocated to knowledge and understanding skills, and the rest – to problem solving. The problems are oriented towards the selection of data structures and application of basic algorithms to work with the developed data structures.

In the practical part students have to write programs for the given two tasks. The main aim is to examine the students' ability to master the stages of programming activities independently. The first task is intended to examine the students' abilities to write programs of the difficulty described in educational standards. The abilities of students to use the procedures or functions as well as basic data types, to realize the algorithms for work with data structures as well as the abilities to manage with input and output in text files are examined. The second task is intended to examine the students' understanding and abilities to implement data structures. The core of the task is to develop the appropriate structures of records together with arrays. The abilities to input data from a text file to array of records, to perform operations by implementing the analysed algorithms, and to present the results in a text file are being examined.

Evaluation of the programs submitted to the exam is a very important issue. The National Examination Centre has made a decision to create an automatic evaluation system with all the requirements met. The system consists of several modules responsible for the evaluation of different aspects such as evaluation of the programming style. The development still continues, as the main rules of the exam change step-by-step and new ideas arise for better evaluation (Table 4). One of the latest ideas is to integrate open question answer testing in the same system, by adding C++ as a possibility for the programming part.

Application of the evaluation operates with packages of solutions. Each solution must be processed as follows: it must be compiled, and then it must be run with several data sets. The answers provided for all these data sets must be compared with the correct ones.

Parts or program evaluation	% of points
Testing. Automatic evaluation.	80
Data structures, data reading, actions of calculation, printing of results. ***Evaluated only if*** results of at least one test are incorrect.	80
Obligatory requirements to the program (procedures & functions for single actions are indicated), programming technology, and style.	20

Table 4: Evaluation of the program development

The evaluator team tries to evaluate the solutions positively. This means that students get points for their effort. For example, correct input/output routines can be assessed by several points. Also, some points can be gained for dividing the program to subroutines, for using complex data structures like the array or record, for writing good comments, for good programming style, etc. These criteria can be easily evaluated by a person, while computer evaluation is not so obvious. This is the reason for manual evaluation of solutions.

The first tasks are easier therefore a larger number of students attempt to solve them. The second tasks are intended to examine the students' understanding and abilities of implementation of record data type. The core of the task is to develop the appropriate structures of records together with arrays. The abilities to input data from text files to arrays containing elements of the record type, to perform operations by implementing the analysed algorithms and to present the results in a text file are examined. The operations are to be performed only with numerical values. The curriculum does not require operations with character strings, only reading and derivation of such strings are applied.

In order to get maturity certificate students should pass a compulsory mother tongue exam and at least two optional exams. Informatics exam has quite good number of participants, over two thousands (in comparison, Chemistry and Physics exams have around three thousands students each). The number of failed students varies from 2% to 17% (Table 5).

Students	2011	2012	2013	2014
Attended	1871	1830	2328	2268
Past	97.69%	92.73%	82.43%	92.21%

Table 5: Informatics exam in 2011-2013

SUMMARY AND DISCUSSION

Promoting curricular change, shifting the focus from knowledge as a set of content and especially from technical knowledge to knowledge as integration of process and skills, including culture, language, etc. is a very difficult task. The changing global context due to the impact of ICT is redefining the type of literacy and skills that are needed. Such skills are not only technical but also cognitive and they involve high-order thinking. The importance of new skills has started to receive considerable political interest throughout Europe (Informatics education, 2013). These are new challenges for researchers to concentrate their attention to this field.

Informatics education in schools does not clear up the myths about CS and most of the students in high schools graduate with no clear answers to the popular statements formulated as "relations": CS = programming, CS = IT (ICT), CS = computer literacy, CS = a tool for studying other subjects, CS ≠ scientific discipline. The White Paper by the CSTA (Stephenson et al., 2005) lists a number of challenges and requirements that must be met if we want to succeed in bridging the gaps in education and improve education in informatics as a computer science discipline:

- students should acquire a broad overview of informatics;
- informatics instruction should focus on problem solving and algorithmic thinking;
- informatics should be taught independently of application software, programming languages and environments;
- informatics should be taught using real-world problems;
- informatics education should provide a solid background for the professional use of computers in other disciplines.

Bringing informatics to schools through curriculum in a formal track is quite important, however it is necessary to support the informal ways of introducing pupils to informatics. The most famous informal way to introduce informatics can be contests and Olympiads on programming.

As we can see from the above sections, each country developed a unique curriculum for high school CS education. Israel introduced CS in high school in the mid-1970s while in Lithuania Informatics was introduced a few years later.

Both curricula are continuously being developed and updated according to recommendation of important interest groups and international organizations such as ACM, CSTA and UNESCO. In both countries there is a strong involvement of researchers from the academia in curriculum development and implementation.

Both countries focus on the fundamentals of algorithms and programming and data structures. In Israel it is implemented in the first 2 points Foundation of Computer Science 1 and 2 and in the fourth point Data Structures. In Lithuania it is implemented in the optional module on programming for high school. Israel's program includes additional modules including an additional paradigm or application and a theoretical unit.

REFERENCES

ACM Curriculum Committee on Computer Science.Curriculum ‘68 recommendations for academic programs in Computer Science. Comm. Assoc. Comput. Mach 11 (1968).

ACM Transactions on Computing Education (TOCE), special issue on Computing Education in (K-12) Schools, Volume 14 Issue 2, June 2014 (2014).

Blonskis, J., Dagienė, V.:Analysis of students' developed programs at the maturity exams in information technologies. *LNCS*, 5090, 204—215 (2008).

CAS Working Group. A Curriculum Framework for Computer Science and Information Technology:http://www.computingatschool.org.uk/data/uploads/Curriculum%20Framework%20for%20CS%20and%20IT.pdf (2012)

Cohen, A., Haberman, B.: Computer Science: A Language of Technology. ACM SIGCSE Bull., 39, 4, 65-69 (2007).

Dagienė, V.: The Road of Informatics. Vilnius, TEV (2006).

Dagys, V., Dagienė, V., Grigas, G.: Teaching Algorithms and Programming by Distance: Quarter Century‘s Activity in Lithuania. In: *Proc. of the 2nd Int. Conference on Informatics in Secondary Schools: Evolution and Perspectives.*Vilnius, 7-11 Nov, 402-412 (2006).

Drysdale, S. et al.: The year in review. Changes and lessons learned in the design and implementation of the AP CS exam in Java. Proc of the 36th SIGCSE Technical Symposium on Computer Science Education, 323-324 (2005).

Gal-Ezer, J. & Harel, D.: Curriculum and course syllabi for a high-school computer science program. *Computer Science Education, 9*(2), 114-147 (1999).

Gal-Ezer, J. et al.: A high-school program in computer science. *Computer, 28*(10), 73-80(1995).

Hunt, F. et al.: How to develop and grade an exam for 20,000 students (or maybe just 200 or 20). Proc of the 33rd SIGCSE Technical Symposium on Computer Science Education, 285-286 (2002).
Informatics education: Europe cannot afford to miss the boat. Report of the joint Informatics Europe & ACM Europe Working Group on Informatics Education http://www.informatics-europe.org/images/documents/informatics-education-europe-report.pdf (2013).
International Contest on Informatics and Computer Fluency “Bebras”: http://bebras.org (2014).
International Olympiad in Informatics: http://www.ioinformatics.org (2014).
Joint IEEE Computing Society/ACM Task Force on Computing Curricula. 2013. Final Report.http://www.acm.org/education/CS2013-final-report.pdf (2013).
Lister, R. et al., L.: A multi-national study of reading and tracing skills in novice programmers. In WG Reports from ITiCSE (Leeds, UK, June 28-30, 2004).ACM, New York, NY, 119–150 (2004).
Merrit, S. et al.: ACM model high school computer science curriculum. ACM, New York (1994).
Publications of the Israeli Central Bureau of Statistics at: http://www.cbs.gov.il/reader/shnaton/shnatonh_new.htm?CYear=2013&Vol=64 (2013).
Resnick, M. et al.: Scratch: Programming for All. Communications of the ACM, 52, 60-67 (2009).
Science and Technology Office in the Ministry of Education:http://cms.education.gov.il/EducationCMS/UNITS/MadaTech/csit (2014).
Seehorn, D., et al.: CSTA K–12 Computer Science Standards Revised (2011).
Stephenson C. et al.: The New Education Imperative: Improving High School Computer Science Education. Final Report of the CSTA Curriculum Improvement Task Force, CSTA, ACM (2005).
The Royal Society. Shut down or restart? The way forward for computing in UK schools.The Royal Society (2012).
Tucker, A. et al.: A model curriculum for K–12 Computer Science. Final report of the ACM K-12 Task Force Curriculum Committee:http://csta.acm.org/Curriculum/sub/k12final1022.pdf (2003).
Tucker, A. et al.: Computing Curricula 1991: A Summary of the ACM/IEEE-CS Joint curriculum Task Force Report. Comm. Assoc. Comput. Mach. 34, 69-84 (1991).
Vee, A.: Understanding Computer Programming as a Literacy. Literacy in Composition Studies, 1(2), 42-64 (2013).

Tamar Benaya holds a M.Sc. in Computer Science from Tel-Aviv University. She is a faculty Member of the Computer Science Department at The Open University of Israel. She designed and developed several advanced undergraduate Computer Science courses and workshops, and she serves as a course coordinator of several courses. She also supervises student projects. She is a lecturer of Computer Science courses at The Open University of Israel. Her research interests include Distance Education, Collaborative Learning, Computer Science Education, Computer Science Pedagogy and Object Oriented Programming.

Valentina Dagienė is a professor at the Vilnius University Institute of Mathematics and Informatics and head of the Department of Informatics Methodology. She has published over 150 research papers and the same number of methodological works, has written more than 60 textbooks in the field of informatics and information technologies for secondary schools. She has been working in various expert groups and work groups, organizing the Olympiads in Informatics among students.

Ela Zur is involved in the Israel IOI project since 1997, and repeatedly served as a deputy leader. She holds a PhD Degree in Computer Science Education from Tel-Aviv University. She is a faculty member of the Computer Science Department at the Open University of Israel. She designed and developed several advanced undergraduate Computer Science courses and workshops, and currently serves as a course coordinator of several courses. Her research interests include Distance Education, Collaborative Learning, Computer Science Education, Computer Science Pedagogy, Teacher Preparation and Certification and Object Oriented Programming.

CS-Oriented Robot-Based GLOs Adaptation through the Content Specialization and Generation

Kristina Bespalova, *kristina.bespalova@ktu.lt*
Kaunas University of Technology, Lithuania

Renata Burbaitė, *renata.burbaite@ktu.lt*
Kaunas University of Technology, Lithuania

Vytautas Štuikys, *vytautas.stuikys@ktu.lt*
Kaunas University of Technology, Lithuania

Abstract

Generative learning objects (GLOs) being reusable items in terms of generative capabilities may also offer new opportunities to create individual and highly adaptable content for learning processes. The paper introduces the stage-based GLOs adaptation to create opportunities for adaptive personalized learning. The approach can be treated as adaptation-with-specialization. This framework is, in fact, the modified paradigm known in Computer Science (CS) research as the program partial evaluation (specialization) technique to support adaptation and reuse. We describe and evaluate the approach from the user point of view to teach CS topics. We also provide a case study using the adapted content (LO) within the LEGO NXT learning environment.

Keywords

Meta-programming-based GLO, LO adaptation through specialization, robot-based CS teaching and learning

INTRODUCTION AND RELATED WORK

The learning object (LO) defines the course-independent learning content aiming at supporting interoperability and reusability in the domain. Among multiple ideas and approaches, the generative learning objects (GLOs) proposed by Leeder et al. (2004) and Morales et al. (2005) play a significant role. GLOs are defined as "*an articulated and executable learning design that produces a class of learning objects*" (CETL, 2009). GLOs being reusable and executable items (programs and meta-programs in our case (Burbaitė et al., 2013) may offer also new opportunities to create the individual and highly adaptable learning content.

The adaptability problem is broadly discussed in the literature. There are many attributes to characterize the problem such as the context, learner's profile, capabilities of a system used, etc. There is also a variety of factors influencing its understanding (e.g. content representation forms, cognitive aspects, structure and model of LO, etc.). As a result, one can meet a diversity of related terms in the literature to characterize the problem: *adaptive learning*, *personalized learning* (Butoianu et al., 2010), *adaptable LO, personalized LO* (Brady et al., 2008), *adaptive granularity* (Man & Jin, 2010), *adaptive learning path* (Bargel et al., 2012), etc.

In e-learning, *adaptation* is thought of as the customization of the system "to the cognitive characteristics of the students and implies the study and conjunction of *technical and pedagogical aspects*" (Ruiz et al., 2008). The paper (Bednarik, et al., 2005) defines adaptation as "the adjustments in an educational environment aiming to (1) *accommodate learners' needs*, *goals, abilities*, and *knowledge,* (2) *provide appropriate interaction*, and (3) *personalize the content*".

Adaptation is a reuse-based activity within an educational environment aiming at changing the structure, the functionality (behaviour) of an item (LO), or both so that the predefined objectives or requirements of the context can be fulfilled. The personalised learning content has been shown to increase the learners' interest, comprehension and hence their learning success (Triantafillou et al., 2004).

Meta-programming-based GLOs (Burbaitė et al., 2013) due to the use of meta-programming as generative technology may also offer new opportunities to create individual and highly reusable content for learning processes. The main goal of reusability is also to adapt the teaching content to the context of use. For that, we need to have a framework enabling to connect reuse issues with the educational context in order we could be able first to specialize the GLOs, and then having the specialized GLO, to consider the adaptability problem in some well-defined manner.

In our case, the adaptation task is driven by the specialization process of the GLO as a higher-level program, i.e. meta-program. In general, program specialization, also known as partial evaluation, has shown its worth as a realistic program transformation paradigm. This paradigm makes it possible to automatically transform a program into a specialized version, according to the context of use (Taha, 2004). Program specialization also partial evaluation (Jones et al., 1993) also relates to *stage programming* and *meta-programming*. Shortly it can be summarized as *multi-stage* programming, i.e. the development of programs in several different stages. Taha was the first to provide a formal description for a multi-stage programming language. Staging is a program transformation that involves reorganizing the program execution into stages. We apply these concepts in our approach.

The paper's contribution is the *pedagogically sound* stage-based adaptation through specialization/generation of the specialized GLOs to support user-guided adaptation of the CS teaching content within the educational robot environment. The paper's structure is as follows. Section 2 presents the GLO adaptation task. Section 3 outlines processes to solve the task. Section 4 provides a case study along with adaptation paths for active learning to teach CS within the educational LEGO NXT (Castledine & Chalmers, 2011) robot environment. Section 5 evaluates the results and presents conclusion.

ADAPTATION TASK USING GLO AND EDUCATIONAL ROBOTS

We are able to formulate the adaptation task in our context using GLO and NXT robots as a part of the educational environment as follows. Given the educational environment that includes the following components: *i*) *specialized* GLO (meaning stage-based representation (Štuikys et al., 2014), further GLO_S) oriented to using NXT robots; *ii*) PHP processor to interpret the GLO; *iii*) RobotC programming environment and *iv*) ready-for-use NXT hardware. The task is to *initiate* and *perform* the user-guided multi-stage *processes* that include: *i*) the pre-programmed content (i.e. specialized GLO) *adaptation* to user's (i.e. teacher's and learner's) needs through selecting the adequate parameter values; *ii*) monitoring and evaluating the result of adaptation through the feedback; *iii*) adapting the intermediate result to the robot's environment and *iv*) creating *active learning* through discussions and data exchange.

GLO_S is a specialized version of the original GLO designed for reuse. The original GLO implements a large scale of e-learning variability that may include pedagogical, social, technological and content variability (Burbaitė et al., 2013). We express all kinds of the variability uniformly through parameters and their possible values. The parameter space predefines the reusable variants of LOs derivable from the original GLO. As the number of LO variants may be very large, it is difficult to generate the *needed* LO from the original GLO. Specialization for adaptation enables to tackle this problem.

Specialization is the structural transformation of the original GLO into its specialized form without affecting of the overall functionality of the original GLO. The aim of this transformation is to make possible the pre-programmed user-guided adaptation of a GLO when used. The specialization process (devised as a combination of (Burbaitė et al., 2013), (Štuikys et al., 2014) and (Futamura, 1999)) results in creating the multi-stage executable specification GLOS that is coded as the k-stage heterogeneous meta-program. In Figure 1(b), we can see the structure of the specification. Specialization of GLOs by staging enables to flexibly (automatically) prepare the content for the different contexts of use. To do that automatically, we have developed the specialization tool MP-ReTool (Bespalova et al., 2015) (read as "meta-program refactoring tool") that transforms GLO coded in PHP into the *k*-stage representation (i.e. specialized GLO_S).

Content adaptation is the user-guided process that includes user's actions and automatic processing by the tool. The user views (exams) the given interface of GLO_S so that to recognize and supply his/her context parameter values. Then the automatic processing follows yielding more *specialized variants* to support needs for adaptation. Content adaptation is a part of the whole learning process being included "surface learning" (along with its feedback) and "deep learning" (along with its feedback).

In the paper (Houghton, 2004), surface learning is defined as "accepting new facts and ideas uncritically and attempting to store them as isolated, unconnected items"; deep learning is defined as "examining new facts and ideas critically, and tying them into existing cognitive structures and making numerous links between ideas". Active learning, as defined by (CRLT, 2014), "is a process whereby students engage in activities, such as reading, writing, discussion, or problem solving that promote analysis, synthesis, and evaluation of class content". The use of educational robots promotes active learning due to the possibility of combining cooperative learning, problem-based learning, the use of case studies and feedbacks.

PROCESSES TO SOLVE THE ADAPTATION TASK

Now we are able to present our approach to solving the adaptation task in more detail. In Figure 1(a), we outline the approach schematically as a multiple process with different kind of adaptation and feedbacks. There are three kinds of adaptation scenarios: i) stage-based, ii) technological and iii) adaptation at the active learning phase.

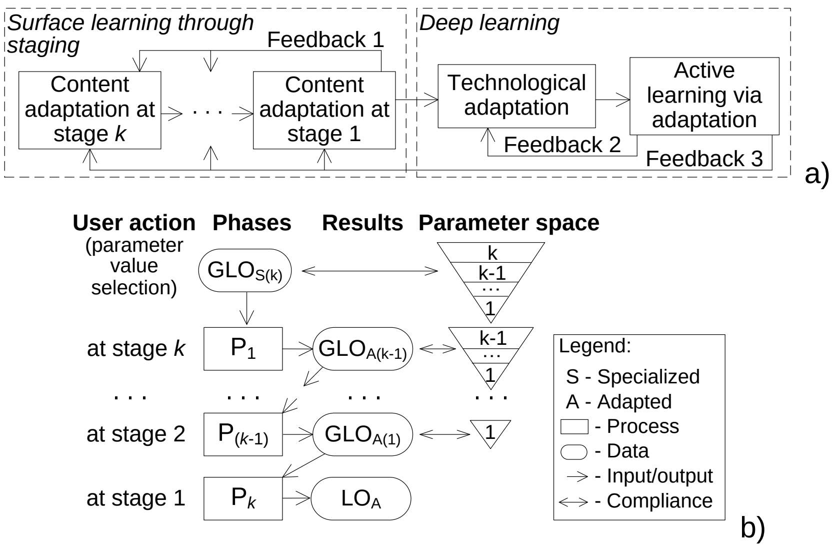


Figure 1: Adaptation scenarios (a) and stage-based adaptation sub-processes (b) ((Štuikys, 2015) © Copyright 2015 by Springer)

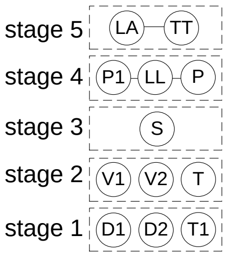
According to given definition, the stage-based content adaptation is surface learning because the user selects the parameter values (see Figure 1(b)) as *"isolated, unconnected items*" (see also the user interface in Figure 3). We present the overall stage-based adaptation in Figure 1(b). Here, the user actions are combined with the automatic processing phases (P_1,…, P_k) performed at each stage by the PHP processor. The result of the processing at a higher stage is the lower-level specialized GLO (denoted, e.g. as $GLO_{A(i)}$). The phase P_k yields the concrete LO_A, i.e. the result of adaptation.

Within each stage, stage-based adaptation is automatic. It is the user-guided process running within the meta-language environment (PHP processor in our case). The higher stages are for the teacher. The lower stages are for learners. The adaptation process as surface learning may follow two modes. In mode 1, there is no feedback. The process goes through phases (stages) resulting in narrowing the space of variants smoothly (see Figure 1(b)). In mode 2, it is possible to return to the previous stages through Feedback 1 and 3 for selecting the other parameter values for adaptation, if the previous values do not satisfy the user's needs. The tool that performs a specialization ensures the functionality of mode 2.

By technological adaptation we mean the compilation of the adapted LO_A (i.e. robot's control program (CP)) and uploading it into the robot's flash memory. After that, the robot is ready to solve the prescribed task and learners are able to monitor the robot's actions, evaluate the characteristics comparing them with selected parameters. The learners are able to analyze the CP, to investigate the correspondence among the abstract parameter values (those that were previously defined at the staging process) with the physical characteristics of the robot's actions. What is most important is the possibility to change the CP by the *short* feedback 2 (meaning the change of CP and its recompilation and reloading), or by the *deep* feedback 3 (meaning the selection of the other parameter values at the stage-based adaptation). In fact, surface learning is the user-guided *content preparation-adaptation* through gradual staging and feedback among stages (i.e. user-oriented parameter selection). The adapted content and possible feedbacks enable to happen the active (deep) learning. However, we are able to describe the whole adaptation process in detail through a case study.

A CASE STUDY: ADAPTATION PATHS OF ACTIVE LEARNING

The aim of this case study is to demonstrate the adaptation process of the real learning task using NXT robot environment (Castledine & Chalmers, 2011) and reveal more practical details on the surface and deep learning through GLO adaptation. We have selected the "Ornament drawing by robot" task. The learning objective was to teach loops and nested loops written in RobotC (2007). In Figure 2, we present the model of the task "Ornament drawing by robot".



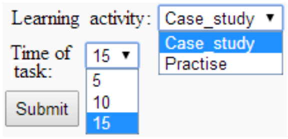
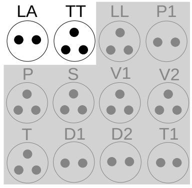
Legend. LA – learning activity: Case study- CT (given by *Teacher*); Practise – PS (done by *Learner*); **TT** – time for the task solution in min; **LL** – learner's previous knowledge level (Beginner-BG; Intermediate – IT; Advanced – AD); **P1** – number of ornaments; **P** – number of ornament's parts; **S** – selected motors (AB, BC, AC); **V1**, **V2**– *drawing velocity* of motors (pen on the paper); **T** – robot's drawing time; **D1, D2** – *moving velocity* of motors (pen over the paper); **T1** – robot's moving time.

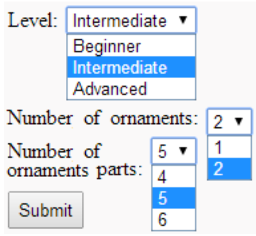
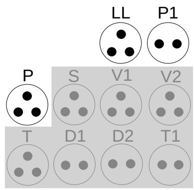
Figure 2: The 5-stage models of the task "Ornament drawing by robot" ((Štuikys, 2015) © Copyright 2015 by Springer)

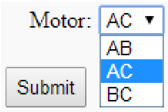
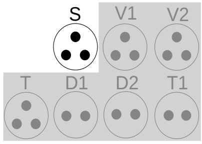
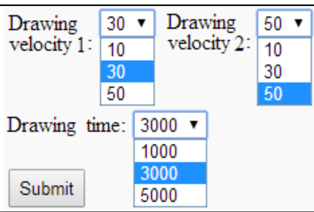
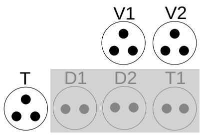
The model contains 5 stages as a result of specialization of the original GLO (Štuikys et al., 2014. The dependent parameters LA and TT represent the teacher's context (are at stage 5). The parameters LL (the pure learner's context), P1 and P (the pure content parameters) (stage 4 in this model). The rest parameters are pure technological parameters (representing the content in the case of using robots). They can be placed into one stage, however, due to the large number of parameters, there might be difficult to ensure a flexible adaptation. Technological parameters are permuted among stages according to the task semantic. In our case, the motors should be evaluated first (at stage 3), then the robot's drawing velocity of motors and drawing time (at stage 2) and, finally, the idle velocity of motors and robot's moving time (at stage 1).

A more detailed analysis of the model is presented in Figure 3. The submitted values by the user initiate the process, and the processing tool creates the intermediate result of the adaptation at the adequate stage. The hidden parameters are not evaluated at the current stage. Note that the result of stages 2 and 3 are not shown in Figure 3.

Stage	User's view for actions	Stage model	State of the GLO after processing phase

5			//Learning activity: Case_study
			//Time for solving task: 15 min
			task main(){
			
			}

4			//Intermediate level
			//Number of ornaments 2
			//Number of ornament parts 5
			task main(){
			for(int i=0; i<2; i++){
			for(int j=0; j<5; j++){
			}
			}
			}

3			...
2			...

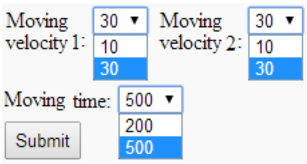
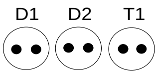
1			//Adapted LO in RobotC (CP)
			task main(){
			for(int i=0; i<2; i++){
			//Pen's descending (in bold)
			motor[motorB]=50;
			wait1Msec(100);
			motor[motorB]=0;
			for(int j=0; j<5; j++){
			motor[motorA]=30;
			motor[motorC]=50;
			wait1Msec(3000);
			motor[motorA]=-30;
			motor[motorC]=0;//rotation
			wait1Msec(3000); }
			//Pen's ascending (in bold)
			motor[motorB]=-50;
			wait1Msec(100);
			motor[motorB]=0;
			motor[motorA]=30;
			motor[motorC]=30;
			wait1Msec(500);
			}
			}

Figure 3: Stage-based adaptation without feedback ((Štuikys, 2015)
© Copyright 2015 by Springer)

In Figure 4, we present the results of solving the Ornament drawing task, the Robot's view to run the task.

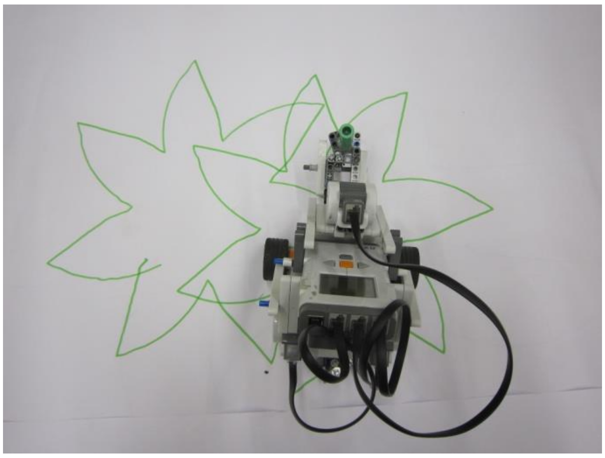


Figure 4: The Ornament drawing task by NXT Robot

At stage 1, we have the LO_A, i.e. two nested loops written in RobotC as the robot's CP (see the source code in Figure 3). The lines in bold are commands to define the robot's action caused by the task specificity, i.e. motor B serves for controlling the pen's state ("over paper", "on paper"), while motors A and C are responsible for moving. Then the robot's CP is to be compiled into the executable code and downloaded into the robot's memory. Next, the robot performs the pre-programmed actions. Robot's actions enable involving the learner into deep learning through the series of activities. First, the learner is able to follow and observe what is going on the robot's action space. As there are a few robots within the classroom with the slightly different CPs, learners can communicate among themselves. It is possible to measure/evaluate visually (roughly) the speed of robots, to observe how robots actions are changing from the drawing state to the idle state. Students are able not only to observe the task solution in action, but also to compare the results.

Even more, it is possible to provide the experimentation and research, for example, to estimate the dependency among CP parameters and robot's actions. For this purpose, of course, learners need to go through feedbacks and repeat the learning path accordingly depending on the new scenarios (see Figure s 1 (a) and 5).

We present a possible scenario of adaptation-based learning that combines both surface and deep learning into the whole process with the deep feedbacks. The scenario (see Figure 5) includes the technological adaptation (i.e. the compilation phase after executing generation at stage 1). The multiple feedbacks are possible on the stage-based adaptation process because the learner is able not only to perform the adaptation by selecting the parameter values, but also by seeing the result of stage-based processing (see Figure 3). The visual monitoring of the result may cause the need for selecting the other values and repeating the process through the feedbacks FB1.

Active learning starts when the robot executes the prescribed task according to the completely adapted CP. The learner's activity is to follow the robot's action, to remember what parameters have been chosen previously, to reason about on how they correspond to what is seen on the drawn picture. In the whole, the learner has the possibility to make the reflection on what is going on the robot's action scene. After that, the learner can evaluate the result of the solved task. There is the possibility to change the CP manually (if some inconsistency was observed) and repeat the process by feedback FB2. Even more, it is possible to provide a wide scale experimentation and research using the deep feedback FB3. Taking into account all possible feedbacks, there is a great deal of learning paths to accommodate the learner's adaptation preferences in gaining knowledge not only to construct gradually (in step-by-step manner) the nested loops, but also to be convinced on how the CP constructs are transformed into physical entities such as velocity, time, idle move, operating move to draw line fragments, etc.

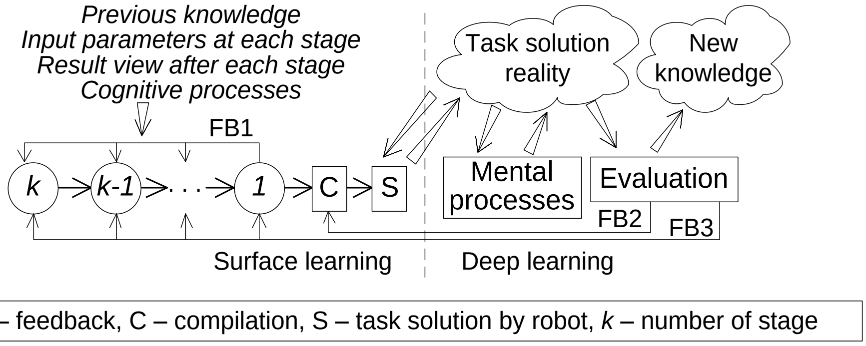
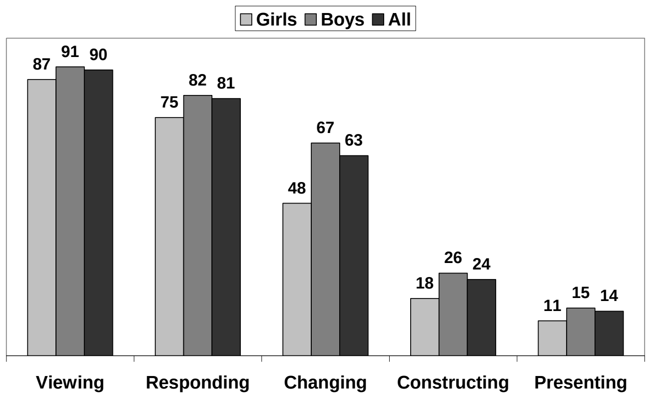


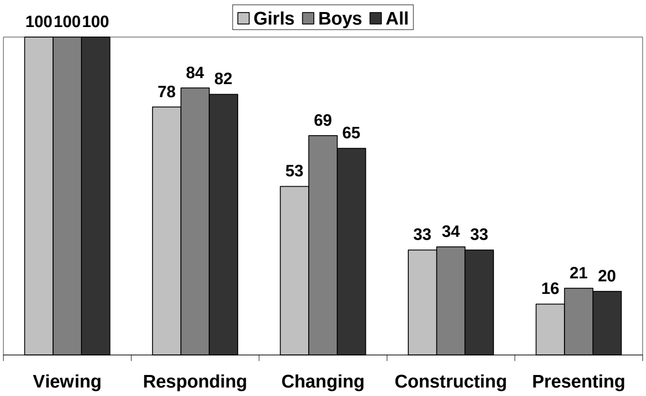
Figure 5: Adaptation-based scenario for active learning ((Štuikys, 2015) © Copyright 2015 by Springer)

EVALUATION AND CONCLUSION

Currently, the approach was tested on three representative tasks (NXT Robot Calibration, Line follower, Ornament's design) with multiple variants. The obtained result has shown its practical value. We have presented a possible scenario of adaptation-based learning that combines both surface and deep learning into the whole active learning. The proposed methodology has been evaluated by the students' engagement levels (suggested by Urquiza-Fuentes and Velázquez-Iturbide (2009) through the following cognitive processes (ordered by cognitive deepness from the lowest to the highest): *viewing*, *responding*, *changing*, *constructing* and *presenting*. In our case, viewing means the student's passive monitoring of robot's actions; responding means the student's ability to answer a question, or the formulation of a question for the teacher; changing means the student's ability to modify the control program directly; constructing means the conscious adaptation (use the feedbacks); finally, presenting means the student's ability to explain, to discuss with the teacher or other students on the topics and to present the obtained and researched results either orally, or in the written form. The pedagogical evaluation enables to conclude that GLOs are most effective at the viewing, constructing and presenting levels. The statistics obtained through experimental research over 3 years (2011-2014) shows the increase of learning improvement from 6 to 15 percent (see Figure 6).



a)



b)

Figure 6: Student engagement levels (2011 to 2014, 186 students: 141 boys, 45 girls): a) not using GLOs; b) using GLOs

The teacher's task is to identify the length of the cognition process path for each student, i.e. to identify how many students are able to go through all cognitive processes and whose failed (in which point and why).

REFERENCES

Bargel, B. A., Schröck, J., Szentes, D., & Roller, W. (2012). Using Learning Maps for Visualization of Adaptive Learning Path Components. *International Journal of Computer Information Systems and Industrial Management Applications*, *4*(1), 228-235.

Bednarik, R., Moreno, A., Myller, N., & Sutinen, E. (2005, July). Smart program visualization technologies: Planning a next step. In *Advanced Learning Technologies, 2005. ICALT 2005. Fifth IEEE International Conference on* (pp. 717-721). IEEE.

Bespalova, K., Burbaitė, R., & Štuikys, V. (2015). MP-ReTool. Retrieved January 15, 2015, from http://proin.ktu.lt/metaprogram/MP-ReTool/

Brady, A., Conlan, O., Wade, V., & Dagger, D. (2008, January). Supporting users in creating pedagogically sound personalised learning objects. In*Adaptive Hypermedia and Adaptive Web-Based Systems* (pp. 52-61). Springer Berlin Heidelberg.

Burbaitė, R., Damaševičius, R., & Štuikys, V. (2013). Teaching of Computer Science Topics Using Meta-Programming-Based GLOs and LEGO Robots.*Informatics in Education-An International Journal*, (Vol12_1), 125-142.

Butoianu, V., Vidal, P., Verbert, K., Duval, E., & Broisin, J. (2010). User Context and Personalized Learning: a Federation of Contextualized Attention Metadata. *J. UCS*, *16*(16), 2252-2271.

Castledine, A. R., & Chalmers, C. (2011). LEGO Robotics: An Authentic Problem Solving Tool?. *Design and Technology Education*, *16*(3), 19-27.

CETL reusable learning objects. (2009). Retrieved January 15, 2015, from www.rlo-cetl.ac.uk/whatwedo/glos/whatareglos.php

CRLT. (2014). Center for Research on Learning and Teaching, Retrieved November 22, 2014, from www.crlt.umich.edu/tstrategies/tsal

Futamura, Y. (1999). Partial evaluation of computation process--an approach to a compiler-compiler. *Higher-Order and Symbolic Computation*, *12*(4), 381-391.

Houghton, W. (2004). *Engineering subject centre guide: Learning and teaching theory for engineering academics*. © Higher Education Academy Engineering Subject Centre, Loughborough University.

Jones, N. D., Gomard, C. K., & Sestoft, P. (1993). *Partial evaluation and automatic program generation*. Peter Sestoft.

Leeder, D., Boyle, T., Morales, R., Wharrad, H., & Garrud, P. (2004). To boldly GLO-towards the next generation of Learning Objects. In *World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education* (Vol. 2004, No. 1, pp. 28-33).
Man, H., & Jin, Q. (2010, April). Putting adaptive granularity and rich context into learning objects. In *Information Technology Based Higher Education and Training (ITHET), 2010 9th International Conference on* (pp. 140-145). IEEE.
Morales, R., Leeder, D., & Boyle, T. (2005, June). A case in the design of generative learning objects (GLOs): applied statistical methods. In *World Conference on Educational Multimedia, Hypermedia and Telecommunications* (Vol. 2005, No. 1, pp. 2091-2097).
RobotC – Improved Movement. (2007). Robotics Academy.
Ruiz, M. D. P. P., Díaz, M. J. F., Soler, F. O., & Pérez, J. R. P. (2008). Adaptation in current e-learning systems. *Computer Standards & Interfaces*,*30*(1), 62-70.
Štuikys, V., Bespalova, K., & Burbaitė, R. (2014). Generative Learning Object (GLO) Specialization: Teacher's and Learner's View. In *Information and Software Technologies* (pp. 291-301). Springer International Publishing.
Štuikys, V. (2015). Smart Learning Objects for the Smart Education in Computer Science: Theory, Methodology and Robot-Based Implementation. Springer. 2015.
Taha, W. (2004). A gentle introduction to multi-stage programming. In *Domain-Specific Program Generation* (pp. 30-50). Springer Berlin Heidelberg.
Triantafillou, E., Pomportsis, A., Demetriadis, S., & Georgiadou, E. (2004). The value of adaptivity based on cognitive style: an empirical study. *British Journal of Educational Technology*, *35*(1), 95-106.
Urquiza-Fuentes, J., & Velázquez-Iturbide, J. Á. (2009). Pedagogical effectiveness of engagement levels–a survey of successful experiences.*Electronic Notes in Theoretical Computer Science*, *224*, 169-178.

Kristina Bespalova is a PhD student at Software Engineering Department of Kaunas University of Technology (KTU). Her research interests are e-learning and formal methods in computer science education.

Renata Burbaitė PhD of informatics at Software Engineering Department of Kaunas University of Technology (KTU) and a teacher of Computer Science (CS, Informatics) at Gymnasium. She also provides teaching of students in Informatics at KTU. Her research interests relate to the e-learning domain that includes the use of meta-programming-based Generative Learning Objects and Educational Robots to teach CS topics. She is a member of Interest Group of Informatics Teachers.

Vytautas Štuikys is a Professor of Software Engineering Department at Kaunas University of Technology and holds the degree of Habilitate Doctor of Science (1992). His research interests include system design methodologies based on reuse and automatic program generation and transformation, and their applications in e-learning. He is author of about 100 papers in this area. Currently he is a head of research group at KTU and a supervisor of three PhD students. He is a member of ACM and IEEE

Developing Computational Thinking Skills through the Literacy from Scratch project, an International Collaboration

Miroslava Cernochova, miroslava.cernochova@pedf.cuni.cz
Charles University in Prague, Faculty of Education, Czech Republic

Mark Dorling, mark.dorling@computingatschool.org.uk
Computing at School, United Kingdom of Great Britain and Northern Ireland

Lawrence Williams, lawrencewilliams@mirandanet.ac.uk
United Kingdom of Great Britain and Northern Ireland

Abstract

This paper builds on the growing international success of the Computing project, Literacy from Scratch, showing how computational thinking skills (mainly algorithmic thinking at this early stage) can be developed through this creative, cross-curricular project. However, while it provides an excellent starting point for the development of computational thinking in schools, there is now a clear need to show teachers how computational thinking can be incrementally developed through this project, and underpinned by research. Accordingly, this paper looks at the practicalities of the project itself, and suggests ways in which the Progression Pathways (an assessment framework with a set of incremental skills, concepts, and principles) set out in England by Computing at School (CAS), and researched and developed in the Czech Republic, can begin to be mapped.

Keywords

Computing, programming, creativity, cross-curricular learning, literacy, Scratch, story-telling, the pupil´s concept of a computer, assessment framework, creation as a cognitive process

INTRODUCTION

In many countries across the world, new attention is being given to computing and computational thinking in schools. In England, this is being driven by a new government curriculum initiative, called Computing. By contrast, some countries, like the Czech Republic, are not waiting for governmental instructions, but are attempting to introduce computational thinking into their curriculum, quite independently. A Computing project, which has successfully engaged school pupils and teacher trainers in both of these countries, is called Literacy from Scratch ([Williams, Černochová, 2013]; [Demo, Williams, 2014]; [Williams, Černochová, Demo, Younie, 2014]). It is an example of a very fruitful international collaboration between primary and secondary schools, and teacher training institutions in England and the Czech Republic. While the concept of developing computational thinking is the same in both countries, there have been different approaches to the pedagogy and research methods underpinning these relatively early stages of the project.

1. CASE STUDY IN THE UK

The teaching of the new subject, Computing, as it is now called in England, as a part of the National Curriculum, has only been undertaken since September 2014. Prior to this, Computing, which includes aspects of ICT, e-safety, as well as computer programming, was taught in some schools as a lunch-time activity, or in after-school clubs.

Some of us are of a generation that remembers, through concrete experience, the wonderful cross-curricular, collaborative, and creative projects undertaken in our Primary schools (age 5 to 11), a process sadly destroyed by the introduction of the National Curriculum in the 1990s. By using Scratch, a graphical programming language, as the tool, however, there was the opportunity to replace this lost teaching model, while simultaneously introducing the new Computing curriculum. This is an exciting prospect [Williams, 2015].

The opening of the National Curriculum states, “A high-quality Computing education equips pupils to develop computational thinking and creativity, in order to understand and change the world.” Here is a reminder of what UK pupils, aged 5 and 6 (Key Stage 1), must be able to do:

- Understand what algorithms are, how they are implemented as programs on digital devices, and that programs execute by following precise and unambiguous instructions
- Create and debug simple programs
- Use logical reasoning to predict the behaviour of simple programs?
- Use technology purposefully to create, organise, store, manipulate and retrieve digital content?

The starting point for the journey into Computing, therefore, is clear.

Teachers who have adopted the Literacy from Scratch teaching and learning model, in the UK and abroad, have been amazed at how exciting the project can be, and, perhaps more importantly, how they can so easily develop creative writing, art, music, and Computing skills, all at the same time. However, this project, Literacy from Scratch, was devised originally by a teacher of English, rather than of Computing, and while it certainly works extremely well in getting former ICT and other teachers “off the mark” in Computing, and using Scratch to meet the demands of the new National Curriculum, the project simultaneously has to be able to encourage pupils to develop their computer programming skills in a broadly incremental way. This necessary progression requires a more rigorous Computing approach, and for this we turn to Computing at School’s National CPD Co-ordinator, Mark Dorling. He has, with colleagues, developed Progression Pathways, a thorough “map” of the key Computing concepts that schools in England should aim to develop. The web of Progression Pathways has had over 20,000 downloads, and is fast becoming the “gold standard” in English schools, as it was included in the Department for Education (DfE) and Microsoft funded QuickStart Computing project.

1.1 Concepts of Computational Thinking and Computing

The next section of this paper explores how the Literacy from Scratch project can help to meet these incremental requirements.

The Literacy from Scratch project and Progression Pathways

This project was conceived as a cross-curricular response to the new government Computing initiative. As such, it covers all of the requirements for the first level of Progression Pathways, which includes the three main elements: Computer Science (CS), Information Technology (IT) and Digital Literacy (DL). The main focus for English schools, which already teach IT and DL, is therefore on Computer Science.

This, is turn, is divided into six main teaching areas: Algorithms, Programming and Development, Data and Data Representation, Hardware and Processing, Communication and Networks and Information Technology (Figure 1).

For the purposes of this section of the paper, analysis is confined to the first three of these pathways, as shown above.

Literacy from Scratch covers all six aspects of the curriculum requirements for Computing at a starting level. This level is coloured in pink on the grid (see on-line for the colour coding) of which three (Algorithms, Programming and Development, and Data and Data Representation) are shown. There is no statutory guidance on assessment, and the school inspectorate is not keen on levels. Therefore, use of colours allows teachers to assign any arbitrary value to colour, and focus on progression instead. The Progression Pathways statements have computational thinking opportunities mapped to them, in order to support teachers in identifying, recording, and auditing which computational thinking techniques they have been using. Mark Dorling and others have developed this CAS computational thinking framework. It is also worth mentioning that the CT Framework is at the core of the DfE and Microsoft funded QuickStart Computing project. There is therefore a clear structure and coherence behind the whole process.

Pupil Progression	Algorithms	Programming & Development	Data & Data Representation
	• Understands what an algorithm is and is able to express simple linear (non-branching) algorithms symbolically. (AL) • Understands that computers need precise instructions. (AL) • Demonstrates care and precision to avoid errors. (AL)	• Knows that users can develop their own programs, and can • demonstrate this by creating a simple program in an environment that does not rely on text e.g. programmable robots etc. (AL) • Executes, checks and changes programs. (AL) • Understands that programs execute by following precise instructions. (AL)	• Recognises that digital content can be represented in many forms. (AB) (GE) • Distinguishes between some of these forms and can explain the different ways that they communicate information. (AB)
	• Understands that algorithms are implemented on digital devices as programs.(AL) • Designs simple algorithms using loops, and selection i.e. if statements. (AL) • Uses logical reasoning to predict outcomes. (AL) • Detects and corrects errors i.e. debugging, in algorithms. (AL)	• Uses arithmetic operators, if statements, and loops, within programs. (AL) • Uses logical reasoning to predict the behaviour of programs. (AL) • Detects and corrects simple semantic errors i.e. debugging, in programs. (AL)	• Recognises different types of data: text, number. (AB) (GE) • Appreciates that programs can work with different types of data. (GE) • Recognises that data can be structured in tables to make it useful. (AB) (DE)
	• Designs solutions (algorithms) that use repetition and two-way selection i.e. if, then and else. (AL) • Uses diagrams to express solutions. (AB) • Uses logical reasoning to predict outputs, showing an awareness of inputs. (AL)	• Creates programs that implement algorithms to achieve given goals. (AL) • Declares and assigns variables. (AB) • Uses post-tested loop e.g. 'until', and a sequence of selection statements in programs, including an if, then and else statement. (AL)	• Understands the difference between data and information. (AB) • Knows why sorting data in a flat file can improve searching for information. (EV) • Uses filters or can perform single criteria searches for information.(AL)
	• Shows an awareness of tasks best completed by humans or computers. (EV) • Designs solutions by decomposing a problem and creates a sub-solution for each of these parts. (DE) (AL) (AB) • Recognises that different solutions exist for the same problem. (AL) (AB)	• Understands the difference between, and appropriately uses if and if, then and else statements. (AL) • Uses a variable and relational operators within a loop to govern termination. (AL) (GE) • Designs, writes and debugs modular programs using procedures. (AL) (DE) (AB) (GE) • Knows that a procedure can be used to hide the detail with sub-solution. (AL) (DE) (AB) (GE)	• Performs more complex searches for information e.g. using Boolean and relational operators. (AL) (GE) (EV) • Analyses and evaluates data and information, and recognises that poor quality data leads to unreliable results, and inaccurate conclusions. (AL) (EV)

Figure 1 Part of Progression Pathways

This means that by adopting and developing Literacy from Scratch in their classrooms, many teachers have made a useful start to the new Computing curriculum, even if only at a (very manageable) first level.

1.2 Research methods

In England, as there is only just starting to develop Computing as a new curriculum subject, from September 2014, there can, necessarily, be little evidence of progression, as yet. Research is currently confined to Action Research, as shown on the web site: www.literacyfromscratch.org.uk.

At the present, in this project there are used qualitative methods mainly based on interviews with teachers and pupils involved in, on monitoring pupil´s activities and on assessment of pupil´s outcomes.

1.3 Activities of pupils and teachers

But how are teachers to move their pupils' levels from level one (pink) to level two (yellow) and on to level three (orange), and beyond? How is progression to be managed? Within the creative concepts of the project, can further progress realistically be made? Here are some activities that may be included to extend computational thinking skills:

Loops and repetition (i.e. level two, yellow)

Loops can be developed in a number of ways. First, there is a very simple, cross-curricular way, through music. In order to create a musical background to the story, or a song within it, or a dance at the end of it, the Sounds section of Scratch can be a very useful ally. A song like the traditional "Good King Wenceslas", for example, has a repeated phrase.

Why repeat writing out the coding when a simple loop (or repeat) can be used, as above? This, incidentally, was coded by a seven-year old pupil, who has clearly met the simplest algorithmic thinking statements (AL) already. More importantly, progression to a higher level in Computing is driven by adding music to the project.

But all sorts of animation effects can also be used as a stimulus for developing computational thinking. Children love to see effects in action, and so, for example, making an animated bat (which can be hand-drawn in an art lesson, or borrowed from the Scratch pool of animals) can be made to "catch an insect" (again drawn or borrowed), and class discussion, a very important aspect of computational thinking, fixes this in the mind of the pupil. This animation, can, in turn be developed as a mathematical concept, by using the x and y axes as a background. The bat can travel around the lines of a square until it touches an insect. (Prediction: What will happen if…? can also be added into the mix very easily.) Here, higher level thinking is also being driven by the needs of the actual story line, and its animation elements.

Subjects that might be included in the story-writing

These narratives can also be developed through a focus on topics such as the environment, and tied into UK primary curriculum concepts such as mini-beasts, and predator and prey, and linked with personal and social education (care for others) all linked through the Computing curriculum, the mechanism by which these stories unfold on the screen. Was there ever a better time to introduce computer science concepts into the curriculum?

1.4 Summary of the case study

Literacy for Scratch has engaged pupils in Computing from the age of 3 up to the age of 14, (and up to age 16 in Italy, through the work of G. Barbara Demo), and can do so in increasingly challenging ways. The project supports the development of creative narrative work, as well as cross-curricular thinking, by combining Literacy, Computing, Art, and Music. When allied also to subject material such as science (through science-fiction stories) or the environment, or social and moral issues (through stories about bullying, for example), it is clear that there is huge potential for cross-curricular teaching and learning, brought about through the development of the Computing curriculum.

2. CASE STUDY IN THE CZECH REPUBLIC

Unlike England, Australia, or Lithuania, the Czech Republic has not discussed computational thinking in the context of school education. Nevertheless, in 2014 this concept was integrated into a document Strategy for digital education for 2020 approved by the Czech Government on November 2014. Some Czech schools, inspired by experiences from abroad, are examining ways, and seeking to develop new approaches to ICT in education, and have started to concentrate on algorithmic thinking development. The project, Literacy from Scratch, is one example of how to start this international initiative.

2.1 Concepts of Computational Thinking and Computing

The starting point for a case study "Tell your story and try to program it", designed within the framework of Literacy from Scratch, is based on several facts in the context of the Czech Republic:

- *The position of ICT as a subject, and especially topics of algorithms and programming in the curricular document "Framework Educational Program for Basic Education" (FEP BE):*

The aims of ICT as a subject in the curricular document FEP BE [RVP ZV, 2013, p. 32] do not include the requirement to develop the algorithmic thinking of pupils, nor to dedicate any attention to their programing skills. These two concepts – algorithmic thinking and programming – are not mentioned and specified either in the curriculum, or as learning outcomes of particular themes of ICT in Basic education for pupils aged 6-15 [RVP ZV, 2013, pp. 33-34], neither in a proposal of Standards for ICT as a school subject [Brdička et al., 2013].

ICT as a subject in [RVP ZV, 2013] is focussed mainly on ICT user skills development, to be able to work with a computer and to search for information from different digital resources, including the Internet, and to publish and present them in different digital formats.

- Algorithms and programming in school practice:

Extensive research carried out among Czech ICT teachers at lower secondary education, in 2013, discovered, and at the same time confirmed, that the teaching of ICT as a school subject is focused primarily on work with the Internet (searching information, e-mail communication) and on fundamental user skills to use main SW (text-processing, spreadsheet graphics applications, SW for doing digital presentations) [Rambousek et al., 2013].

It was discovered that only a very few Czech schools in the Republic have been dedicated over a long period to algorithmic thinking and programming development ([Rambousek et al, 2013]; [Černochová, 2010]). This was reinforced also in interviews with 73 part-time students of MA degree study of ICT at the Faculty of Education in Prague in 2012-14, who teach ICT as a school subject in Basic schools. The basics of programming are implemented in school education only by ICT teachers who have a technical professional background with a focus on computer science, IT, or who at some time worked as a computer programmer.

- Theoretical basis for teaching approaches to pupil learning:

The theoretical basis for teaching approaches and to knowledge processes, more and more emphasises the importance of active and engaged learning, especially the importance of creating different types of artefacts, including mental artefacts. "Creation and cognitive process are (inter)related more than teaching practice assumes" [Slavík et al., 2014, p. 110]. And the process of computer program creation in schools, based on an appropriate educational programming environment, can play an important role in pupil learning, and in acquiring different literacies, such as mathematical literacy, reading literacy, linguistic or visual literacy, and including digital literacy.

Most Czech pupils (aged 6 to 15) do not come across any programming outside their school activities. This young generation has no idea about how it is possible for them to use a computer, and other digital devices, to carry out so many miscellaneous social activities (such as playing games, social communication, publishing, and searching for information). Unfortunately, pupils' experiences with tablets amplify this fact. Work with tablets differs from a work with an ordinary PC. Pupils understand a computer as a tool for work with SW, as a tool for downloading programs, or browsing photos, for communication on Facebook, or searching for information on the Internet.

- The first steps in a governmental strategy for education in the Czech Republic:

Although there is no intensive discussion among teachers, teacher educators, researchers and policy-makers in the Czech Republic about the need to reform the existing curricular documents, and to implement into them the basics of Informatics or Computer Science, the incorporation of the requirement to develop computational thinking of pupils and their teachers into the governmental document Strategy for digital education for 2010 [Strategie digitálního vzdělávání do 2020, 2014, p. 46] was nonetheless achieved in 2014. The document was approved by the Czech Government on 12.11.2014. It seems to be an indication that the Czech Republic could be engaged very soon in the introduction of key Informatics/Computer Science concepts into school education (including pedagogical approaches to algorithmic thinking, and programming skills development) into curriculum and university teacher education and teacher continuing professional development. At present, the Czech Ministry of Labour and Social Affairs has started to collaborate with the Czech Ministry of Education, Youth and Sport about questions related to digital literacy [MPSV MŠMT, 2015].

2.2 Research methods used in “Tell your story and try to program it”

From September 2014 to January 2015 a research project was developed called, “Tell your story and try to program it” in Scratch, with two groups of pupils aged in 11 to 22, one group was formed of 22 pupils of Grade 6 at a Basic school and second one was formed of 14 pupils in the eighth-year gymnasium in Prague.

Lessons were organised regularly, 1 hour per week, in both schools (the Basic school and eight-year gymnasium) managed by the same ICT teacher in the framework of the compulsory ICT subject. The ICT teacher published her lesson plans, study materials for pupils and tutorials on her web pages (http://scratch.sandofky.cz). Teaching at both schools took place in computer labs, with data projection, and in a good working atmosphere.

The project “Tell your story”, was aimed not to research a teaching methodology for programming in Scratch for beginners, but rather, primarily, to start inquiry into the nature of pupils’ ideas about computers. The project strove to develop a complex of fundamental literacies (reading, visual, digital, linguistic, mathematical, and musical) with a special accent on cross-curricular links. On no account was there any aim to compare ICT knowledge and skills between two groups of pupils, that is, between Basic school pupils and eight-year gymnasium pupils.

Target group

The group from the Basic school was formed by 22 pupils (10 male, 12 female). The curriculum of this Basic school is focussed on pupil creativity development, and on creative teaching approaches to learning. Pupils of this Basic school are used to approaching tasks creatively not only in artistically-oriented subjects, but also in other subjects, including science. If pupils do not understand what they learn they show it by their behaviour. They were very spontaneous in asking questions, they enjoyed speaking, but sometimes they had problems in narrating stories. Some of them had a problem in concentrating on their work. Algorithms and programming are not included in the school curricular programme. Nevertheless, some pupils of Grade 6 did some programming in Scratch in the school year 2013/14. ICT as a school subject at this Basic school is focussed on creative work on computer, not on technological principles about how computers function.

The group of eight-year gymnasium was formed of 14 pupils (12 male, 2 female). The pupils were accepted to study at this selective school according to their success in their entrance examination. In ICT they are used to apply simple algorithmic procedures and logical thinking. They did not know Scratch before the start of the project “Tell your story”.

Research question

The research question concentrated on which ideas about computers the pupils should begin work with, in order for them to have an idea about how a computer works. Do pupils have any idea of how they will be able to program a computer, with the aim to fulfil their dreams and to ask their computer to do what they wish? Will the pupils’ ideas about what a computer can do, and why it can do what it is able to do, change in any way if pupils spend some lessons working in a programming environment and thinking about how to instruct the computer to do what pupils want? Will pupils’ first programming experiences have any impact on their ideas about what they can do with a computer?

Research methods

To resolve this research question, we applied qualitative research methods, a systematic observation, questionnaire method, a test with matching tasks, a test with short answers, essay, interviews with pupils, analytic and synthetic methods for evaluation of pupils’ outcomes.

2.3 Activities of pupils and teachers

At the beginning of the project the pupils were given a questionnaire with the aim of seeing their initial ideas about a computer.

From the questionnaires, we discovered that the pupils in Grade 6 from Basic school learn how to manipulate and use computers primarily from their siblings, from their parents and from other members of their family, and in many cases by themselves without any help. In spite of the fact that pupils spend at home a relatively a large amount of time using computers or another digital device, the questionnaire, unsurprisingly, did not substantiate pupils having any profound knowledge about computers, or any clear idea about how computers work. They do not think at all about how and why a computer works, and 75% of them stated they had no idea about this problem. For some pupils, a computer works *“accordingly to technology. It is a box full of ideas and thoughts.” “A computer functions thanks to a graphics card and to other components, but I did not think about it.” “According to my idea, it contains some components in which information is stored. And using wires it is transmitted to a monitor.”*

Answers to the questions, “How can you communicate with a computer? How can you tell it what you want from it?” pointed towards what the pupils of the Grade 6 mainly do with their computers:

- the pupils search for information on the Internet:

I write it to a search engine – I write to my computer what I am searching for, and it finds it – I write it to my computer – I write something, then I use ENTER and press OK – I move mouse on an icon (for example Google) and using keyboard I write what I need to know (for example a recipe how to bake a cake) or you can play games – it is handled by a mouse or keyboard.

- the pupils work with SW applications:

Using a keyboard and a mouse I click on what I need – I do not understand very much the question, but either I write it to my computer on a keyboard or I click by a mouse – Using a keyboard I write what I need and using a mouse I click on icons or items)

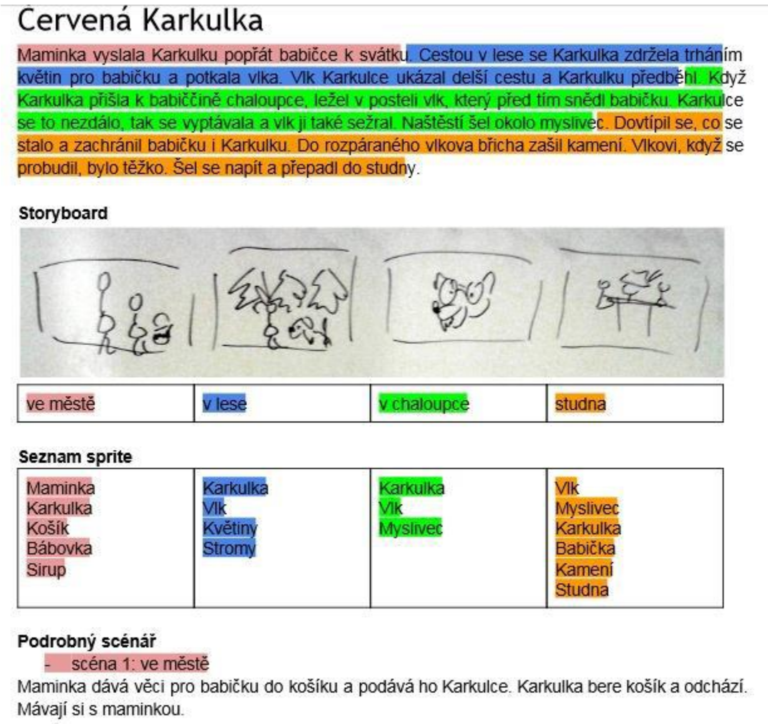
The pupils of the Year 1 at Gymnasium do not spend so much time using game consoles, and playing games as pupils of Grade 6 at the basic school. Their answers in the questionnaire demonstrate they have clearer and more complex ideas about how and why a computer functions, following a series of classroom lesson on hardware and software:

A computer can work due to electricity – it transmits a set of 1 and 0 which are displayed on a monitor or stored in a computer memory and its other elements. There will be presented a picture on a monitor, in memory cards there are stored all data, and a processor operates and controls all processes. – In a computer inside there is a hard disk where are stored all information about everything in a "binary language" which operates with 0 and 1 only, everything is recorded in 0 and 1. By pressing A (B, C, ...) on a keyboard a signal passes in a binary form through a silicon board, and a graphics card to a monitor where it is displayed in a form recorded for this key.

The pupils from the gymnasium think a person can communicate with a computer that s/he *"puts some things using a mouse or a keyboard into various programs, but if you want to program your own program you must formulate a complicated formula in a programming langue – to create the programme." Some of them answered "The question is very simple – we put data and information into a computer through a keyboard, mouse, microphone ... (a touch monitor?)"*. Understandably, there are some pupils who, similar to pupils of Grade 6 in their answers, mentioned „using a keyboard and mouse". In pupils from the gymnasium opinions were: *"a computer is able to do only what is being programmed for a computer. A computer may do only two things, but it can also do 5000 things or more – it depends what a person programmed the computer what to do." "A computer knows to do many things, but it needs programs. A computer is used for calculating numbers, communication, entertainment, recording, writing, editing, processing photos, videos or sounds." Some pupils compared a computer to "a robot – it works according to our instructions, but it has no own opinions." Another student wrote "we cannot compare a computer to anything. Each its form (calculator, mobile, etc.) is specific. A computer is an unique object."*

PHASE 1: Teamwork and scenario development

Lessons of programming in Scratch in "Tell your story ..." were based on pupil teamwork. Pupils formed two-members teams. Each team proposed its own story which at the start had to describe in a written form as an essay; using four colour markers in each essay pupils had to mark out four phases (scene) of the story which related to the same background (stage). Then each team presented a main idea of its story to all class verbally. After then, using worksheets teams developed a structured storyboard with four scenes (Figure 2). They had to name each scene separately. Using a table for each scene pupils wrote a list of sprites. Finally, they again told the story in a written form in details structured by each scene.



Červená Karkulka

Maminka vyslala Karkulku popřát babičce k svátku. Cestou v lese se Karkulka zdržela trháním květin pro babičku a potkala vlka. Vlk Karkulce ukázal delší cestu a Karkulku předběhl. Když Karkulka přišla k babiččině chaloupce, ležel v posteli vlk, který před tím snědl babičku. Karkulce se to nezdálo, tak se vyptávala a vlk ji také sežral. Naštěstí šel okolo myslivec. Dovtípil se, co se stalo a zachránil babičku i Karkulku. Do rozpáraného vlkova břicha zašil kamení. Vlkovi, když se probudil, bylo těžko. Šel se napít a přepadl do studny.

Storyboard

ve městě	v lese	v chaloupce	studna

Seznam sprite

Maminka Karkulka Košík Bábovka Sirup	Karkulka Vlk Květiny Stromy	Karkulka Vlk Myslivec	Vlk Myslivec Karkulka Babička Kamení Studna

Podrobný scénář

- scéna 1: ve městě

Maminka dává věci pro babičku do košíku a podává ho Karkulce. Karkulka bere košík a odchází. Mávají si s maminkou.

Figure 2 Example of structured scenario for a story „Little Red Riding Hood“

Pupils could draw inspiration from an example based on a Czech popular song „Skákal pes přes oves“. This example (Figure 3) was used as a teaching material to teach pupils to program in Scratch[1]. Firstly, under teacher guidance, the pupils learnt how to analyse the program of this story, all they had a study material including a program in Scratch. Its melody which is very well known to each child helped pupils to understand a structure of the Scratch program.

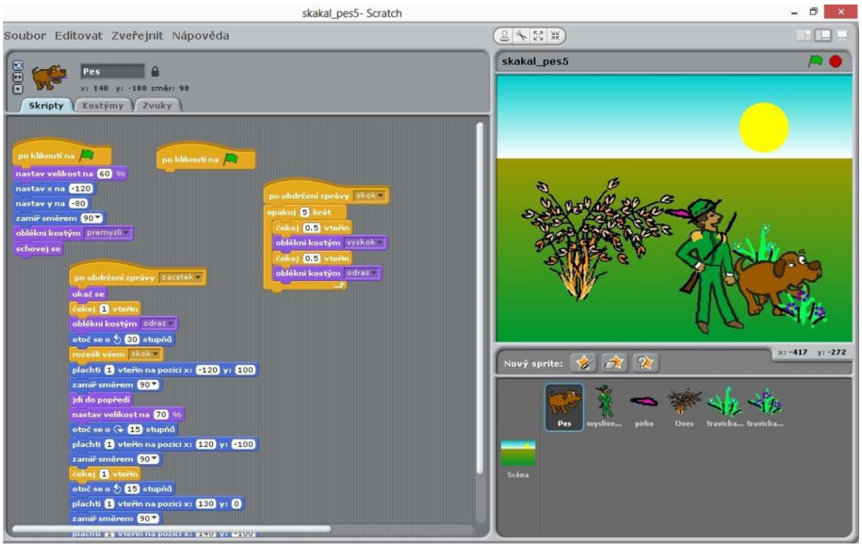


Figure 3 Example of a story based on the Czech song "Skákal pes přes oves"

PHASE 2: Teamwork in programming stories

Pupils were not allowed to use ready-made graphics and sprites available on the Internet. They had to design original graphics. The majority of pupils designed all

[1]http://scratch.sandofky.cz/wp-content/uploads/lekce/02_Scratch_podrobnynavod.pdf

graphics in Scratch editor. Only one team scanned their own hand made pictures. In spite of all pupils tried to draw pictures using a graphic tablet, in the end nobody used it for their stories in Scratch.

Pupils of Grade 6 programmed in Scratch 12 different stories. For us it was very important that the pupils were interested in their teamwork because they programmed their stories which they had thought themselves. The progress of each team was assessed continuously: each team obtained in a printed form information and recommendation what would be better to do or change, how to solve some situations in their program (Figure 4). Pupils had to read these recommendations and instructions very carefully.

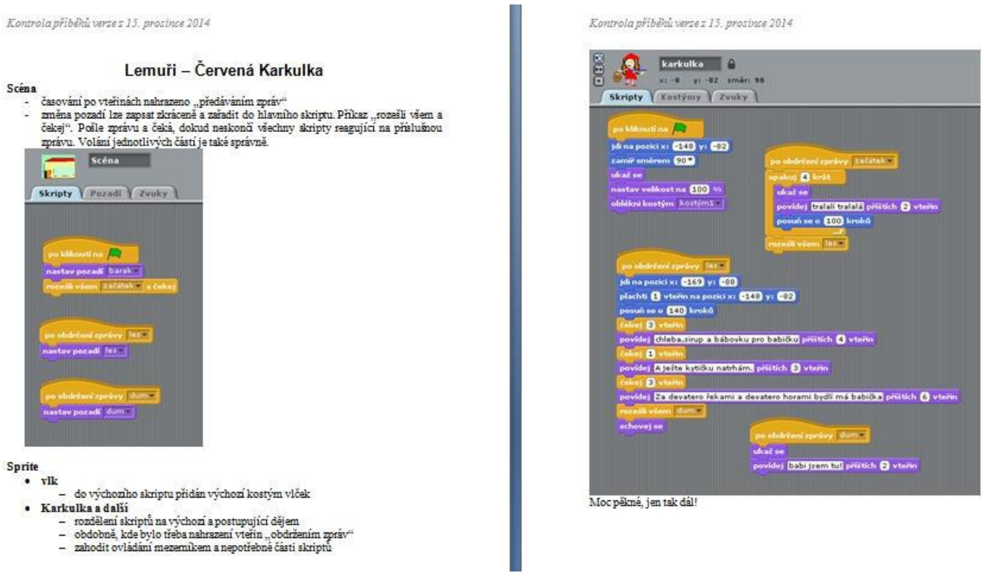


Figure 4 Recommendation from a teacher for a team

PHASE 3: Explanation of key concepts related to programming

We exploited some experiences which pupils step-by-step have acquired in Scratch to explain to them some key concepts (algorithm, program, sequence of command, input, output, variable, cycle, branching, debugging) in a context of specific examples in particular situations (Figure 5).

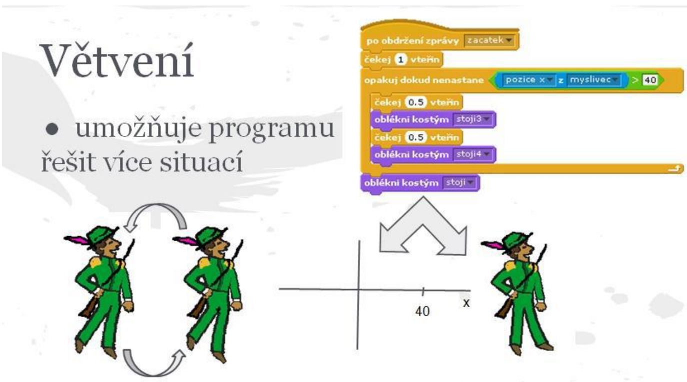


Figure 5 Explanation of a concept of branching using a situation from a song Skákal pes přes oves

PHASE 4: What have pupils learnt?

At the end of the project, the pupils answered a final questionnaire and analysed a program in Scratch. The pupils of Grade 6 showed us how deeply and profoundly embedded were the experiences that they acquired from searching on the Internet, and from using a computer as a tool for communication, as well as altering their perceptions about computers. Activities in Scratch clearly contributed to the improvement of the ideas of pupils of eight-year gymnasium, about how a computer can function.

CONCLUSION

Literacy from Scratch as a classroom project for introducing pupils to computational thinking is succeeding because it allows non-specialist teachers (or rather teachers with other skills!) to begin working on Computing through a constructivist model. But it also allows for conceptual progression. Much still needs to be done to develop computational thinking further, of course, but, in Literacy from Scratch, we now have a creative and cross-curricular teaching and learning framework within which pupils and teachers can develop understanding and skills in a constructivist way, one which can be easily mapped against Progression Pathways and the computational thinking framework, and that is being effectively supported by research.

REFERENCES

BRDIČKA, B. et al. (2013) *Standardy pro základní vzdělávání. Informační a komunikační technologie.* Pracovní verze z 30. 4. 2013. Zpracováno dle upraveného RVP ZV platného od 1. 9. 2013.

ČERNOCHOVÁ, M. (2010) Teaching practice from the perspective of ICT student teachers at the Faculty of Education, Charles University in Prague. In *Key Competencies in the Knowledge Society.* IFIP TC 3 International Conference, KCKS 2010. Held Part of WCC 2010 Brisbane, Australia, September 2010, Proceedings. (Eds.) N. Reynolds, M. Turcsányi-Szabó. IFIP AICT 324. ISSN 1868-4238. Springer, Printed in Germany. pp. 44-55.

DEMO, G. B., WILLIAMS, L. (2014) The Many Facets of Scratch. In *Proceedings of the ISSEP Conference 2014*. Istanbul, September 2014.

DORLING, M. (2015) *QuickStart Computing A CPD toolkit for secondary teachers UK*. Department for Education/Microsoft, 2015.

MPSV, MŠMT (2015) *Strategie digitální gramotnosti ČR na období 2015 až 2020.* MPSV : Praha, leden 2015.

RAMBOUSEK, V. et al. (2013). *Rozvoj informačně technologických kompetencí na základních školách.* Praha: Česká technika - nakladatelství ČVUT.

RVP ZV (2013) *Úplné znění upraveného RVP ZV.* Verze platná k 1.9.2013. Příloha č. 2 k Opatření ministra školství, mládeže a tělovýchovy, kterým se mění Rámcový vzdělávací program pro základní vzdělávání, č.j. MSMT-2647/2013-210. Praha, 2013.

SLAVÍK, J. et al. (2014) *Tvorba jako způsob poznávání.* Karolinum : Praha, 2014.

MŠMT (2014) *Strategie vzdělávací politiky České republiky do roku 2020.* MŠMT : Praha, 2014.

WILLIAMS, L., ČERNOCHOVÁ, M. (2013). Literacy from Scratch. In *Proceedings of the 10th IFIP World Conference on Computers in Education*. Torun, July 2-5, 2013, WCCE 2013. Copernicus University, Torun, PL, pp 17-27.

WILLIAMS, L., ČERNOCHOVÁ, M., DEMO, G. B., YOUNIE, S. (2014) A Working Model for Teacher Training in Computing through the Literacy from Scratch Project. In: *Key Competencies in ICT and Informatics. Implications and Issues for Educational Professionals and Management.* IFIP Advances in Information and Communication Technology Volume 444, 2014, pp 25-33. Springer Berlin : Heidelberg, 2014.

WILLIAMS, L. (Ed) (2015) *Introducing Computing: A guide for teachers.* Routledge : London, 2015.

Learning History: a gamified activity for mobile devices

Sónia Cruz, *soniacruz@braga.ucp.pt*
Catholic University of Portugal, Braga, Portugal

Ana Amélia A. Carvalho, *anaameliac@fpce.uc.pt*
University of Coimbra, Portugal

Inês Araújo, *inesaraujo@fpce.uc.pt*
University of Coimbra, Portugal

Abstract

This paper presents a gamified activity designed to teach a Portuguese historical event from the 20th century to 6th grade students. This gamified activity is based on data from an ongoing project carried out in Portugal concerning the games most played by students. In our paper we analyse the data related to the students in the 2nd cycle of the Portuguese Educational System (n = 508), collected through a questionnaire. We present the results obtained as well as learning principles and game mechanics identified. Finally, we describe the game that is being developed (according to the game mechanics identified) related to a Portuguese historical event - the Implementation of the Republic in 1910.

Keywords

Mobile Learning, Mobile Devices, Games, Gamification

INTRODUCTION

Although ubiquitous learning has been pointed to as the future of education for more than a decade, it was only with the recent spread of various mobile devices in everyday life that several researchers awoke to the importance of using those devices in the educational field.

> "In recent years, mobile learning has undergone a significant transformation due to rapidly growing ownership of smartphones and tablets, accompanied by the proliferation of apps, social networks and mobile-friendly open access resources" (Kukulska-Hulme, 2014, p.12).

Games, which have been applied in education for a long time, have also benefited from these advances in technology and from the portability of multiple devices (Jeong& Kim, 2007). Consequently, it is very common to see today's young people spending their free time playing. And they do it with an enthusiasm they do not show in other school activities. In fact, the possibility that the player may take a certain game with him/her and play it anywhere - because he/she makes the mobile phone an extension of his/her cognition - is quite attractive.

Being aware that games allied with mobile technologies can enhance motivation, we believe they can be used as a powerful mechanism for teaching formal contents, as they are fun and learning occurs in every move without the player even paying much attention (Prensky, 2010).

In this paper, we are going to set out a proposal for a gamified activity for learning History, based on previous research about students' game preferences. For a better understanding, we will present the research conducted and identify students' game preferences, as well as analyse their comprehension of the characteristics each game must have so that they can keep playing it.

RESEARCH

A project carried out in Portugal aims to combine games and mobile devices in learning contexts. This still ongoing project, named "From Games to Interactive Activities for Mobile Learning", sought to identify the games Portuguese students play most, especially on mobile devices. Students from the 5th grade of Basic Education up to Master's degree level completed a survey, with a sample size of 2303 players of mobile games. Overall, results from the study indicated that i) students' preferences change according to their educational level and gender, ii) female students show more interest in casual and simple games, while male students prefer games that involve planning, strategy and cooperative work, iii) females prefer to play alone, while males prefer to play in online communities and iv) older players value the games with impressive graphic effects that engage them in virtual environments (Carvalho&Araújo, 2014).

Within the scope of this project, after the analysis of the favourite games and the identification of the learning principles and the game mechanics, the research team is now designing gamified activities for mobile devices aimed at students of all educational levels and in formal learning situations, which will be evaluated in real learning contexts.

Therefore, in this paper, we begin by presenting the data obtained related to 5th and 6th grade students (N = 508).

1. THE GAMES MOST PLAYED BY 2ND CYCLE STUDENTS

The first phase of the project aimed to identify the most played games, as well as students' gaming habits on mobile devices. To achieve these purposes, we carried out a survey (Babbie, 2003) online in Portugal between May and November 2013. The data collection instrument was divided into four dimensions, namely i) Characterisation of the students; ii) Characterisation of their gaming habits; iii) Preferences about games and iv) Preferences for learning school contents through games. The survey was validated by experts in the field and approved by the Educational Department (DireçãoGeral de Educação) of Portugal.

Sample characterisation

Out of the 649 respondents, 508 were players on mobile devices, of whom 58.7% were males and 41.3% were females (see Table 1), with an average age of 11.2 years. In this study, we corroborate findings related to players' gender already presented in other studies (Lucas & Sherry, 2004; Simons, Bernaards& Slinger, 2012) indicating that male students are more likely to play than female students (see table 1).

Educational Level	Respondents			Respondents playing games			
	Total	Gender	f	Total	Gender	f	%
2nd cycle	649	Male	367	508	Male	298	58.7
		Female	282		Female	210	41.3

Table 1: Number of answers obtained from 2nd cycle students

When asked about the mobile devices on which they played, the respondents indicated the laptop (78.7%), followed by the mobile phone (63.8%), the PSP (46.7%), the tablet (44.7%), the Nintendo 3DS (24.2%) and the smartphone (21.3%). Regarding the mobile device they use most to play their favourite game, the laptop is the device indicated by 45.3% of the respondents, followed by the mobile phone (14.4%), the tablet and the PSP (13.4%), the smartphone (5.1%) and the Nintendo 3DS (2.4%). 6.0% didn't mention which devicethey used to play their favourite game.

Gaming habits

When asked about the time they devote to games, 41.1% of the respondents indicated a range of 1 to 5 hours a week, 34.3% less than an hour and 13.4% between 6 and 10 hours. 4.7% of the respondents indicated 11 to 20 hours and also 4.7 indicated more than 20 hours per week (1.8% of the respondents did not answer this question). On average, respondents play 4.4 hours per week; male respondents play an average of 5.5 hours and their female peers 2.8 hours.

With respect to the respondents' preference for game partners, the majority of the respondents indicated that they prefer to play alone (57.1%). 42.9% prefer to play with others online, 2.4% with friends and / or acquaintances, 23,8% with colleagues 13.4 % with siblings, 12.4 % with other family relatives, 10.0% with strangers and 2.6% with parents.

Preferences about games

Regarding the games most played by the respondents, we identified 141 games (Carvalho et al., 2014).

Taking into account the great diversity of the games identified by the respondents (N= 508), we present the 5 games most commonly indicated by the students from the 2nd cycle (n=161).

Thus, the 5 most played games correspond to 31.7% of the sample that was part of the project mentioned above. Data highlight that the Pro Evolution Soccer option was only identified by the male students and that Pou and Subway Surfers were predominantly female students' options (see table 2).

Game	Male (n=119)		Female (n=42)	
	f	%	f	%
Grand Theft Auto (GTA)	39	32.8	1	2.4
Pro Evolution Soccer (PES)	26	21.8	0	0.0
Subway Surfers	16	13.4	15	35.7
Minecraft	17	14.3	5	2.4
Counter Strike (CS)	15	12.6	3	7.1
Pou	6	5.1	22	52.4

Table 2: *Top 5* favourite games, distribution by gender (n=161)

As our sample is mainly male, typically male games stand out, characterisable as violent (GTA, CS), sport (PES) and sandbox (Minecraft) (Terlecki et al., 2010) (see table 2). These highly competitive games require effort and take several hours to complete. It is important to note that some of the games mentioned are not suitably age rated for the students that make up our sample, whose preferences are therefore, at variance with the European standards (PEGI 16 and 18). In addition, 2.3% of the boys play games rated 16, and 23.2% of male and 4.3% of female regularly play games rated 18+ (PEGI18) (Carvalho et al., 2014, p.31).

These data indicate that the major differences are in the gender-related preferences. While male students prefer longer games, which sometimes involve teamwork and group cooperation, female students prefer faster games, which they can play alone without interacting with other players (Carvalho et al., 2014, p.30). We found that the students also give different answers in relation to the themes of the games. The boys prefer action and adventure (GTA), war (CS) or football (as PES, FIFA) games, while girls prefer games about everyday life, where they can take care of an "animal" (Pou) or people (Sims), or where they can test some skills such as speed of reaction (Super Mario and Subway Surfers) (idem) (see table 3).

Position	Male (n=128)	Female (n=80)

1st	Grand Theft Auto (*Rockstar Games*) (30.5%)	Pou (*Zakeh*) (27.5%)
2nd	Pro Evolution Soccer (*Konami*) (20.3%)	Super Mario (*Nintendo*) (20.0%)
3rd	Minecraft (*Mojang*) (13.3%)	Subway Surfers (*Kiloo Games & Sybo Games*) (18.8%)
4th	Subway Surfers (*Kiloo Games & Sybo Games*) (12.5%)	The Sims (*Electronic Arts*) (17.5%)
5th	Counter Strike (*Valve Software*) (11.7%) FIFA (*EA Sports*) (11.7%)	Stardoll (*Stardoll*) (16.3%)

Table 3: *Top 5* favourite games according to gender (n=208)

Regarding the time spent by students playing each game, we noted that the most common answer is 1 to 5 hours per week (47.2%), followed by less than an hour a week (27.3%) and finally between 6 and 10 hours a week (13.7%). Few respondents marked over 11 hours a week (10.6%).

In the questionnaire the students (n=508) were asked to express their opinion about the difficulty level of the games they play most. The most chosen level is Moderate (40.8%), Easy is chosen by 22.4% and Very Easy by 16.9%. Hard and Very Hard where chosen by 12.2% and 7.7% respectively.

They were asked to indicate the importance of some characteristics of the game for keeping them playing. The differences between male and female students are noteworthy.

Males (n=298) mentioned that these features are very important to them: graphic effects and animations (79.9%), gameplay (78.2%), characters (74.2%), being able to play with others (72.8%), being able to improve one's score (71.1%), the game having many levels (70.8%), scenarios (69.5%), a long game (67.8%), friends playing the same game (64.4%), being able to play online with others (63.4%), storytelling (57.0%) and sounds (55.7%).

On the other hand, females (n=210) indicated that the characteristics most important for keeping them playing were: characters (79.5%), the game having many levels (72.4%), being able to improve one's score (70.5%), graphic effects and animations (70.0%).We conclude, therefore, that the male gender values more the graphics, the gameplay, the characters, being able to play with others, being able to improve one's score, the game having many levels, and the scenarios while the female gender highlights the game characters, the game having many levels, being able to improve one's score, and graphic effects and animations as the factors that make them keep playing.

Learning course content through games: students' preferences

We asked in the questionnaire if the students would like to use games to learn course content, and most of them answered yes (84.6%). Those who answered positively (n=430), were asked to choose the type of game they would like to use in class. Adventure (66.0%), Action (59.1%) and Sport (50.5%) were the types most commonly chosen.

2. LEARNING PRINCIPLES AND GAME MECHANICS IDENTIFIED IN THE GAMES MOST PLAYED

After having analysedthe games theyplaymost, based on the 36 learning principles (Gee, 2003), we identified some of these principles as common to the games most played: i) Psychosocial Moratorium; ii) Committed Learning Principle; iii) Amplification of Input Principle; iv) Achievement Principle; v) Practice Principle; vi) Regime of Competence; vii) Intuitive Knowledge and viii) Affinity Group Principle.

The reasons mentioned by the students for liking the games most playedare the opportunities to do violent actions, to drive a car or to fight criminals. These dangerous and morally reprehensible activities are carried out in a safe virtual environment where it is possible to take risks without real consequences (*Psychosocial Moratorium* - by Gee, 2003).

Sport games, mainly football, are a typical boys' activity (Blakemore, Berenbaum, &Liben, 2008; Cherney& London, 2006; Sin, Talib, Norishah, Ishak, &Baki, 2014; Terlecki et al., 2010; Williams, Consalvo, Caplan, & Yee, 2009), for this reason it is easy to understand that in the list of the most played games there is one football game (PES).

On the other hand, Minecraft is a game with its own characteristics where adventure and the ability to build something are very appealing ingredients for male players.

Casual games like Pou and Subway Surfers are more appealing for female players. These are games with short matches, highly rewarding without major punishments; their aim is to collect as many coins as possible to purchase items for avatars, which enables social interaction by sharing through Facebook, SMS or email. This social interaction is mentioned by Terlecki et al. (2010) as important for female players.

However, if we look for common characteristics in these games we can identify that in all of them it is necessary to fulfill tasks - these can be repetitive - (Pou, PES, Subway Surfers) or to complete missions (GTA, CS, Minecraft). Most of these tasks are feasible in short time periods and rewardedas they are performed in the game. Carrying out these tasks in a regular manner makes it possible to progress in the game and to develop the avatar by customising it or acquiring items. It is the *Committed Learning Principle* (Gee, 2003) when students apply their time on a regular basis to doing the tasks that make the avatar development possible, thus feeling a commitment towards this virtual identity.

In all the games we find the *Amplification of Input Principle* (Gee, 2003), because the tasks are set in a simple way that produces feedback without a need to actually do them as in the real world. For instance, a football match can be completed between 4 and 12 minutes and not in 90 minutes as in the real world.

Some of the tasks are repetitive and this makes it possible to improve the player's capabilities (*Practice Principle; Regime of Competence; Intuitive Knowledge* - Gee, 2003). The player repeats similar actions in the game improving his/her capabilities, by driving vehicles in GTA, by playing soccer matches in PES or by running on a railway line (*Practice Principle*). Even if the player cannot complete a task at the first attempt he/she feels that he/she is capable of accomplishing it by trying again and again, improving his/her performance (*Regime of Competence*). For instance when a player helps Jake, the avatar, to run on a railway line jumping and moving between the tracks in every attempt he/she can go further and feels that he/she is capable of doing even better the next time around. Here, practice and experience allow the player an intuitive and tacit knowledge that is highly valued between players who share an interest in the same game (*Intuitive Knowledge*).

Another common characteristic is social interaction with other players, throughsharing items (Pou, Subway Surfers), or by using a multiplayer mode (GTA, PES, CS and Minecraft): here we find the *Affinity Group Principle* (Gee, 2003). The player shares the same interest over the same game, enjoys sharing his/her achievements because the other players understand and value them, but it is also possible to learn from the experience of other players or to teach others.

We also identified the common gamification mechanism that can be seen in the games most played (Manrique, 2013):

- World (The whole space where the gamified system takes place);
- Avatar (A representation of the person in the game);
- Customisation (The possibility of personalising our character);
- Equipment (The items that the character wears. A source of power);
- Currency (Any kind of virtual currency that creates an economic market);
- Quest (A mission with an objective that leads to rewards);
- Tutorial (Learning tool to develop the player's skills in that game).

GAMIFIED ACTIVITY

Drawing on the above analysis, an interactive activity was created that we describe below.

Planning

The game has been designed according to two prerequisites: i) allow the formal learning of contents, ii) be a creative game that would attract students' attention in order to make them learn. The mechanics identified in students preferred games were taken into consideration in the design of the game. After the definition of the subject under study, we elaborated a storyboard.

Gameplay

This game is under development for Android system, as this is the mobile operating system most used by the Portuguese population. To interact with the game, the player will need to touch the screen to select the answer he/she thinks to be the right one, or use the drag and drop function. When the player starts the game, he/she will see how the avatar (a journalist) lived from 1890 to 1911.The player must help this avatar to write the best news to be published in the newspaper, by selecting the correct words or choosing the most impressive photos. The game can be regarded as a mission, i.e., helping a journalist to write about the historical events that were part of the history of the republican revolution in Portugal.

Subject theme and Course Content

The History discipline was chosen to make a gamified activity because in the project team we have a History teacher.The course content selected was the Republic implantation in Portugal; this content is delivered inthe 6th grade.

The designed game can be used in two different ways. The students play the game out of the classroom, but the teacher knows the students' results by checking the report in an Excel page. It can also be used in the classroom to introduce the new subject or to reinforce a subject already studied.

Theme

In the late 19th century, the Portuguese monarchy faced great difficulties. The Republican Party took advantage of the discredit of the monarchy to attempt to overthrow it. To achieve this purpose, the leaders of that party organized several riots. In 1908, King Carlos I and his heir-apparent, Prince Royal Luís Filipe, were assassinated by revolutionaries. D. Manuel became king of Portugal, struggling with several difficulties. However, he failed to solve the serious economic, social and political crisis that was raging in the kingdom. On 5th October 1910, a revolution established the republic in Portugal with the support of secret associations. After the revolution a provisional government was created, new national symbols were adopted, elections were called and a constitution was drawn up. The governments of the 1st Portuguese Republic also took several measures to develop education and tackle the difficulties faced by workers. In spite of having failed to solve the kingdom's problems, this historical event is remarkable and is regarded as a crucial aspect to make children and young people aware of the need to value the History of their country, as a way of preserving national memory and identity.

Student's role

The students log into the game. The first time they log in, they are supposed to watch a video that will present them the role they are to play in the game - a journalist in the year of 1890. This journalist's job is to investigate facts / events in order to write news for the newspaper he works for in Lisbon. The game starts when the journalist is in the newspaper office and he is asked to write a news article for the next day's edition. The director tells him that the more the newspaper sells, the higher his salary will be (coins).

Then, we present the interactive activities designed (see table 4).

Year	Episodes	Challenges
1890	1	The player (journalist) sees a news article with some blank spaces to be filled. To complete the article, he/she must use the notes he/she has taken and fill in the gaps with historical information.
1908	1	The player (journalist) is supposed to cover photographically the arrival of the royal family when the regicide happens. The player must choose one of the photographs to illustrate the news article for the next day's newspaper. He/she has to put together a jigsaw puzzle to get the whole picture.
	2	The player is challenged to find some members of a secret society. He must go through an avenue with a code name and look for a person who is able to give him/her a countersign. When the player identifies that person, he/she will be invited to be part of a secret organization. However, to be accepted, he/she must find other members of that organization.
	3	In a café, the journalist listens to a member of the Republican Party, who is stating his political convictions. The player will be faced with several items related to the republican ideals; he must tick the correct ones.
1910	1	The journalist starts a private conversation with a republican and convinces him that only a coup could implant the republic. Then, he shares the plans of Carbonaria (the secret society) with him. The player's task is to drag the names of the places to the correct places in the map. When he/she does it correctly, he/she will be provided with some historical information about each place.
	2	The player is faced with a map displaying the most important places during the revolution. The player must click on the images of those places and read the information provided. At the end, the players are supposed to complete a quiz about these key points.

	3	The journalist must write a new article to inform the country about the republican victory and the new national symbols (anthem, currency, flag). The player should see the national flag uncoloured. Then, he/he must use a brush to tick the respective colours in the flag (some historical explanations will be provided). Then, the player will see the musical score of the Portuguese Anthem and he/she is supposed to drag the title and the verses of the anthem to the correct spaces. At the end, the player listens to the chorus of the anthem. As soon as the player completes the task, he/she will be able to see the coin counter and to notice that the "real" has replaced by the "escudo".
1911	1	The journalist (years later) organizes a news article to present the measures taken by the republicans. These measures are provided to students in categories, but in a randomly way. The player must match the different measures to the area they apply to.

Table 4: Game Activities

It should be noted that the tasks required from the players are distributed by episodes within a historical period with notable dates (1890, 1908, 1910, 1911). These episodes require a greater involvement of the student according to the increased complexity of the issues analysed. The tasks performed allow the player to unlock the time frames.

According to the reward system used, following the game mechanics identified above, the player gets a specific number of coins for each right answer. To pass to the next level, the student needs to have gained a fixed amount of coins. If the answer is wrong, the player does not get the respective coin but can progress forward till the end of that episode. At the end, the player is given the chance to recover the coins he did not win, by performing the task once again.

Implementation

The design of the game is still ongoing but it will be tested by the end of the school year; therefore we are planning to implement it at school between April and June 2015.

CONCLUSION

Mobile learning may facilitate the implementation of a student-centred learning approach, promoting students' learning through searching online, games, reflection and development of critical thinking.

Our project findings suggest that students are receptive to using games in course activities, to learn school subjects. They mentioned that they would prefer the Adventure (66.0%), Action (59.1%) and Sport (50.5%) types of games to use in class.

Looking at the games most played we find that male players prefer long games that require more effort from the player and have multiplayer functions. Female players prefer shorter games that they can play alone.

When planning gamified activities it is important to allow the repetition of tasks, accumulating any kind of reward that can be exchanged for collectibles, the value of which may depend on the provision held. The repetition of tasks allows for the development of specific skills or motor dexterity in the student. Another factor to consider is the existence of avatars which the player can identify with and customise.

We are building a game for mobile devices in order to promote the learning of an important topic in History, Republic implantation in Portugal, based on the learning principles proposed by Gee (2003) and game mechanics identified in the games we have analysed. We will test the game that we are developing between April and June 2015.

ACKNOWLEDGEMENT

This research project is funded by FEDER through the ProgramaOperacionalFactores de Competitividade - COMPETE and by national funds through FCT - Fundação para a Ciência e a Tecnologia - PTDC/CPECED/118337/2010.

REFERENCES

Babbie, E. (1997). *Survey Research Methods*. Belmont, California: Wadsworth.

Blakemore, J. E. O., Berenbaum, S. A., &Liben, L. S. (2008). *Gender development.* Washington: Hogrefe& Huber Publishers.

Carvalho, A. A. A.; Araújo, I.C.; Zagalo, N.; Gomes, T.; Barros, C.; Moura, A.; Cruz, S. (2014) Os jogos mais jogados pelos alunos do Ensino Básico ao Ensino Superior. In Carvalho, A. A. A.; Cruz, S.; Marques, C. G.; Moura, A.; Santos, I. (orgs.). *Atas do 2.º Encontro sobre Jogos e Mobile Learning* (pp.23-37). Braga: CIEd. Retrieved from : http://hdl.handle.net/10316/25740

Carvalho, A. C. & Araújo, I. (2014). Jogos Digitais Que os Estudantes Portugueses Jogam: Diferenças de Género. In Rocha, A.; Fonseca, D.; Redondo, E.; Reis, L. P. & Cota, M. P. (Eds*). Sistemas y Tecnologías de Información: Actas de la 9ª Conferencia Ibérica de Sistemas y Tecnologías de Información*, Volume 1 (pp. 969-974), Barcelona: AISTI. Retrieved from: http://www.aisti.eu/cisti2014/index.php/pt/proceedings

Cherney, I. D., & London, K. (2006). Gender-linked Differences in the Toys, Television Shows, Computer Games, and Outdoor Activities of 5- to 13-year-old Children. *Sex Roles*, 54(9-10), 717–726. doi:10.1007/s11199-006-9037-8

Gee, J. P. (2003). What Video Games have to teach us about learning and literacy. USA: Palgrave Macmillan.

Jeong, E., & Kim, D., Definitions, Key Characteristics, and Generations of Mobile Games. In Taniar, D. (Ed.), *Encyclopedia of Mobile Computing and Commerce* (2007), 185 – 189.

Kukulska-Hulme, A. (2014).Mobile, Wearable, Companionable: Emerging technological challenges and incentives for learning. In Carvalho, A. A. A., Cruz, S., Marques, C. G., Moura, A.; & Santos, I. (orgs.) (2014). *Atas do 2.º Encontro* sobre Jogos e Mobile Learning. Braga: CIEd.

Lucas, K., & Sherry, J. L. (2004). Sex Differences in Video Game Play: A Communication-Based Explanation. *Communication Research*, 31(5), 499–523. Retrieved from: http://digitalcommons.unl.edu/commstudiespapers/20/

Manrique, V. (2013). 35 Inspiring Game Examples for Gamification Mechanics. Epic Win Blog. Retrieved from http://www.epicwinblog.net/2013/06/35-inspiring-game-mechanics-examples.html

Prensky, M. (2010). *Teaching digital natives: Partnering for real learning*. Thousand Oaks: Corwin Press.

Simons, M., Bernaards, C., & Slinger, J. (2012). Active gaming in Dutch adolescents: a descriptive study. *International Journal of Behavioral Nutrition and Physical Activity,* 9(118), 1–9. Retrieved from: http://www.ijbnpa.org/content/9/1/118

Sin, N. M., Talib, O., Norishah, T. P., Ishak, A. A., &Baki, R. (2014). Male Students and Digital Game: Reason, Motivation and Feeling. *International Journal of Information and Education Technology*, 4(1), 6–11. doi:10.7763/IJIET.2014.V4.359

Terlecki, M., Brown, J., Harner-Steciw, L., Irvin-Hannum, J., Marchetto-Ryan, N., Ruhl, L., & Wiggins, J. (2010). Sex Differences and Similarities in Video Game Experience, Preferences, and Self-Efficacy: Implications for the Gaming Industry. *Current Psychology*, 30(1), 22–33. doi:10.1007/s12144-010-9095-5

Williams, D., Consalvo, M., Caplan, S., & Yee, N. (2009). Looking for Gender: Gender Roles and BehaviorsAmong Online Gamers. *Journal of Communication*, 59(4), 700–725. doi:10.1111/j.1460-2466.2009.01453.x

Sónia Cruz is a lecturer at the Catholic University of Portugal (Braga - Portugal) and a researcher in the Centre of Philosophical and Humanistic Studies at the Catholic University of Portugal (Braga - Portugal). She has developed a model to guide students from being critical consumers to producers online. Her latest researches are concerned with the integration of ICT in learning including gamified activities.

Ana A. Carvalho is a lecturer and a researcher in the Faculty of Psychology and Education Sciences at University of Coimbra, in Portugal. She developed a model to promote case analysis and reflection online based on Cognitive Flexibility Theory. Her latest project is about games and interactive activities in mobile-learning. She is an IFIP TC3 member, WG 3.3.

Inês Araújo is a PhD Student of Educational Technology and Communication in Faculty of Psychology and Education Sciences in University of Coimbra (FPCEUC). Her professional path goes from development and evaluation of educational software, teachers and adults training in ICT's. She is a Research Assistant in a project named "From Games to Mobile-Learning Interactive Activities" developed by FPCE UC.

Professional field involvement in ICT curricula at the Dutch UaS

Hans Frederik, j.a.frederik@vu.nl
VU University Amsterdam, Netherlands

Abstract

This paper analyses Dutch accreditation reports in the field of information and communication technology (ICT) to assess the degree of reported involvement of the professional field in the curricula of universities of applied sciences. Qualitative content analysis of the reports of all the ICT programs in the country indicates loose coupling in reporting on mechanisms of interaction. This means that whereas the involvement of the professional field is strongly suggestive at the strategic levels, there is an underrepresentation of university-industry interaction on operational levels which suggests the need to explore the real interaction taking place between the professional field and the universities of applied sciences, and possible implications for the future of the profession.

Keywords:

universities of applied sciences, accreditation, professional field, loose coupling, means-ends decoupling, university industry interaction.

INTRODUCTION

Increasingly, policymakers are interested in the relationship between higher education and the industry (Teichler, 2007). Since the discourse on the role of higher education has become more market oriented (see; Slaughter and Rhoades, 2004; Sam and van der Sijde, 2014), higher education institutions started paying more attention to the industry and its involvement in shaping concrete learning outcomes (Bennenworth and Jongbloed, 2010; Leisyte, et. al., 2013). In particular, the interaction between higher education and the industry is the topic of study for the last 30 years that primarily concentrated on research commercialization, and analysing the potential impact of market oriented behaviour on institutional structures of higher education (see Slaughter and Leslie, 1997; Slaughter and Rhoades, 2004). Some scholars are critical to these new developments in higher education and its influences on traditional higher education structures (Awbery, 2002; Lynch, 2006; Washburn, 2005; Alajoutsijarvi, Juusola and Siltaoja, 2013) as well as roles of academia in changed higher education settings emphasizing increasing collaboration with companies (e.g. Hazelkorn and Moynihan, 2010). On the other hand, the shift towards a market oriented higher education and growing industrial stakeholder involvement does not have to mean that universities were forced to displace their traditional activities (see Ylijoki, 2003). Industry collaboration is for instance beneficial for institutional growth, transformation or evolution (Marginson and van der Wende, 2007).

This paper intends to contribute to an ever-growing body of literature analysing the involvement of industry as a stakeholders in higher education, but it goes a step further in exploring the inter-linkage. First, it focuses on universities of applied sciences (UaS) in the Netherlands and second, it uses accreditation reports to explore formal discourse of the UaS on the interaction of the professional field and practice. Accreditation reports are built according to a standard format in the Netherlands controlled by the Dutch-Flemish Accreditation Organisation (NVAO 2011). They report on programme profile, learning outcomes and assessment, yet the higher education departments and/or teaching staff “make sense” (Weick et.al., 2005) of each program, and structure activities accordingly to their goals and preferences.

Their responses to external influences also vary (Pfefer and Salancik, 1978) and are presented in accreditation report as illustrative. By analysing the accreditation reports we should be able to explore the interaction with the industry on strategic level and on operational level in ICT programs at Dutch UaS. **The analysis intends to evaluate the coupling on both strategic and operational levels (Weick 1976; Bromley and Powell, 2012) of the ICT programs as reported in the accreditation**. Coupling implies the extent of interconnectedness of elements within the system, their dependency on each other on the one hand, and the independent development or autonomy on the other.

We focus on universities of applied sciences where linkages with the professional field, although part and parcel of the institutional tissue (Huisman, 2008), have been rather unexplored, and in particular in the context of the accreditation process. Their interaction with companies has intensified over the years with the introduction of the official role of performing research. In particular evidence of curricular innovation, emphasizing problem-oriented and practice-based learning, and growing social and economic landscape entrenchment can be observed (Hasanefendic, Heitor and Horta, 2015). In this light, it is interesting to explore how the interaction with companies is structured and what are the implications for curricular program development.

Our study reveals that universities of applied sciences are more coupled with the industrial field at strategic levels than at operational levels. Formal criteria of collaboration according to Dutch quality regulation system (NVAO, 2011) are thus met. In the case of reporting on collaboration with companies on operational level, outputs of interaction are “obscure” (Bromley and Powell, 2012). Therefore, concrete implications of the involvement of industry in the curricula and the very learning process are hard to determine from accreditation reports and yields further, more empirical, investigation.

The following section will introduce theoretical framing for analysis and explore the implications for this research. Then we focus on the method of analysis and explain the selection of qualitative content analysis to understand the coupling dimensions as discerned from the accreditation report. This section is followed by results and a discussion where we model our summations. Conclusion provides an overview of our most relevant findings and indicates lines for future research.

Loosely (de)coupled systems: Strategic and operational level of involvement of industry in the curriculum

Majority of studies analysing higher education-industry collaboration has been undertaken at operational levels (Keep, 2012), or by analysing concrete outputs and implications, rather than focusing on formal compliance to policies in place which stimulate such interaction. This makes sense as policy outcomes are most visible when we look at organizational practices. The collaboration between the two “worlds”, should however not be restricted to just the operational level. The nature of the two types of collaboration (strategic and operational collaboration) and the mechanisms involved presupposes a kind of “coupling” between the two types (e.g. Weick, 1976) and it is the coupling between these two levels in the accreditation reports that is the central topic of this study.

Coupling theory was introduced by Weick in 1976 and later Orton and Weick (1990) to explain the behaviour of independent units embedded within larger systems. They showed that actions within these independent units may have little or no effect to the other unit or even the overall system (Gilmore et.al., 1999). This phenomena is called *loose coupling* (). The basic thought is that, unlike tight coupling which presupposes highly integrated and responsive to each other systems (Orton and Weick, 1990), loose coupling indicates that the system is less robust and units are free to adjust accordingly to change without causing transformation to the entire system. Literature has substantiated proof of the existence of loosely coupled systems, either within organizations or outside organizations creating interdependent partnerships (Sharp, 2009), where misalignments are present (Soh and Sia, 2004). Such literature always emphasizes the process of mutual adaptation towards some form of eventual alignment (see Berente, 2009). Less has literature focused on the concrete benefits of loose coupling (e.g. Ravasi and Verona, 2001) or advantages of such systems to organizational environment and effectiveness (e.g. Dubois and Gadde, 2002). It is envisioned in this work to address this juxtaposition as well in terms of industry collaboration in curricular programs on strategic and operational levels.

METHODOLOGY

Our analysis draws on systematic comparison of accreditation reports conducted in 2009 - 2012. It is substantiated with our experiences and observations as either researchers or professionals in the field of higher education and quality assurance, especially within the Dutch context. Therefore, this research approach is depleted with observations and discussions with professionals in the accreditation field, ICT field and from university of applied sciences setting.

The overall data collection represents the 53 accreditation reports on ICT program curricula across 22 universities of applied sciences in the Netherlands in 2012. The program field of information and communication technologies can be divided based on a particular curricular focus; either on (technical) computer science (n=15), business information technology & management (n=18) and information technology (n=20). The field of ICT has been growing in importance in the Dutch context in past couple of decades (Cucchiarini, Daelemans and Strik, 2001; den Adel, Blauw and Entzinger, 2003) and research in the field of ICT has been gaining on prominence (Frederik, 2013). Still, each year ICT industry shows a demand of over 10.000 professionals which points to a lack of critical mass in the field (van Ruud, 2001). Shortage of people trained in the ICT field could become potential disadvantage for the economic development and hamper international competitiveness of the Dutch ICT sector (den Adel, Blauw and Entzinger, 2003). It is in this context that the quality of the overall educational curricula is sought and in particular with close industry collaboration and participation of companies in practice based and problem oriented learning.

In the Netherlands, the responsibility for quality assurance lies within the higher education institutions. They assure that their programs are periodically evaluated by an independent Review and Assessment Agency (VBI) and accredited as official degrees by the NVAO, the Dutch-Flemish Accreditation Organization (Scheele, Limbach and Rijcke, 2006). The accreditation decision is based on the report that the external evaluation body sends to the NVAO and they are publicly available. The external evaluation body is always composed of one student representative, one professional and one higher education representative besides official Chairman and Secretary positions (NVAO 2011). Industry therefore has their representative in the quality assurance processes in the Dutch context. Furthermore, it is expected that the views of industry and employers are built into the quality assurance management, and that they partake a role in program formulation and definition of concrete learning outcomes (Kolster and Westerheijden, 2014).

In order to systematically interpret meaning from the accreditation reports we developed categories for analysis (Mayring, 2000) which served as reference during the process of content data synthesis (van Dijk, 1980). These have been developed from existing literature (see Davey et. al., 2011) and in close consideration of the very content and structure of accreditation reports. Categories follow mechanisms through which university-business collaboration is transparent, developed by Davey et. al. (2011). Table 1 provides an overview of the key mechanisms of collaboration and specifies their operationalization. All 53 reports were analysed according to the concepts and mechanisms described in Table 1. We used trigger words (vocabulary on university-business collaboration) to allocate text/content to selected category. Whenever a word was encountered in the content it would be flagged and the text allocated to the category. The work was done in Excel and the flags were manually checked and double checked for validity of the content allocated to categories.

Mechanisms for coupling on the *strategic* level	***Governance***	**Professionals from IT industry field in Boards and Committees in universities of applied sciences**
	Curriculum evaluation	IT industry involvement in regular (e.g. annual) evaluation of the curriculum (quality management)
	Curriculum design	IT industry involvement in curriculum design
	Lifelong learning	IT industry and lifelong learning
Mechanisms for coupling on the *operational* level	*Mobility*	Frequency of temporary or permanent assignments offered to students and teachers in collaboration with IT industry.
	Collaboration in R&D	Impact of the collaboration with IT industry on curricular development.
	Commercialization of R&D results	Impact of commercialization of R&D projects on the curriculum involvement of industry.
	Entrepreneurship	Integration of entrepreneurship into the IT curriculum and the role of the IT industry in this (via collaboration).

Table 1 – Mechanisms of university-business collaboration and their operationalization. Source: adapted from Davey et.al. 2011.

RESULTS

Coupling on the strategic level

Table 2 shows the results of qualitative content analysis of 53 accreditation reports and the mechanisms for coupling on the strategic level. Coupling with the industry takes place most frequently on the governance levels, which includes, “business representatives involved in education decision making boards” (Davey et.al., 2011:10).

	Mechanisms for coupling on the strategic level										
	Total			**Information Science**			**Business IT & management**			**Computer Science**	
	n=53			n=20			n=18			n=15	
	n	%		N	%		n	%		N	%
Governance	47	91%		18	90%		16	89%		13	87%
Curriculum Development	49	94%		18	90%		16	89%		15	100%
Curriculum Evaluation	38	72%		14	70%		12	67%		12	80%
Lifelong learning	40	74%		16	80%		12	67%		12	80%
		83%			83%			78%			87%

Table 2 – Mechanisms of coupling on the strategic level and percentage of reporting in total and by field of the ICT program

For example, the Fontys UaS appoints ICT professionals in the Advisory Council, the Professional Committee and the Board of External Experts. Similarly, Saxion UaS and UaS Arnhem – Nijmegen, have a Professional Committee which advises on the position of the industry in education. At UaS Leiden professionals are represented in the education Advisory Committee and the same goes for NCOI Nederland . These findings indicate that there is a strong presence of ICT industry stakeholder representatives at managerial levels in universities of applied sciences in the Netherlands. These representations in Boards, Councils or Committees are also important in the curriculum evaluation, as between 67% and 80% of the programs organise regular meetings with them to discuss the results of evaluations considering the (degree of) involvement of the professional field, or send out a (bi)annual survey for evaluation (e.g. UaS Zeeland, Vlissingen) to industrial stakeholders.

Another mechanism for coupling on strategic or management levels is the curriculum development. In other words, there is a strong influence of the industrial stakeholder on the design of courses, modules, minors and internships/apprenticeships. These are project based and practice oriented where company collaboration is particularly fostered. For instance, at UaS Amsterdam within the course "Social Smart City" each group of students works on a unique case for an external partner. In Fontys UaS ICT students participate in problem solving activities with regional companies of which fifty are already integrated as partners. Most UaS seem to encourage participation of external stakeholders in their ICT courses as tools to building a learning environment that stimulates project based and problem oriented practices.

Both the involvement of the professionals from the ICT field in the governance and through course programs relying on practice based research and problem solving activities at companies ensures that curricula is depleted with references from PROFESSIONAL FIELD which suggests a higher likelihood to impact curricular development. It follows that, collaboration with the (regional) industry plays a role in skill development and modernisation of practices of teaching and learning thus contributing to curricular innovation processes.

Coupling on the operational level

Table 3 exemplifies the coupling of "UaS" with industry on operational levels. The results indicate a relatively low percentage of reporting on the type and purpose of collaborative activities. According to the accreditation report only 19% of the programs used mobility to report involvement with the professional field. For example Saxion UaS has arrangements with some companies to exchange professionals. An employee thus becomes a lecturer at Saxion for a year, while at the same time one of the lecturers works for the company.

In accreditation reports, details on collaboration in R&D, such as contract research, R&D consulting, cooperation in innovation, informal and personal networks, joint publications, are rarely provided and only included for 9% of the programs. One of the programs which reports on the collaboration in more practical terms is at the ICT program at Leiden UaS. They specify that industry is involved via external projects, internships and graduation, guest lectures and other forms of cooperation such as consulting.

Avans UaS also specifies that their lecturers from the field of Automation set up knowledge networks. Professionals from the field and companies participate in these knowledge networks, which enables transfer of professional or field knowledge to the curriculum. Additionally, knowledge networks are also a basis for projects with companies in which students actively participate.

	Mechanisms for coupling on the operational level							
	Total		**Information Science**		**Business IT & management**		**Computer Science**	
	n=53		n=20		n=18		n=15	
	n	%	N	%	n	%	N	%
Mobility	10	19%	4	20%	4	22%	2	13%
Collaboration in R&D	5	9%	3	15%	1	6%	2	13%
Commercialization of R&D results	5	9%	3	15%	1	6%	1	7%
Entrepreneurship	5	9%	3	15%	1	6%	1	7%
		15%		22%		13%		13%

Table 3 – Mechanisms of coupling on the operational level and percentage of reporting in total and by field of the ICT program

As a mechanism for coupling with the industry at operational levels, entrepreneurship is rarely mentioned. However, the reports mention that some ICT programs stimulate students entrepreneurial capabilities. For instance at the Hague UaS, regular ICT programme is combined with running a business. Students are thus also independent entrepreneurs.

DISCUSSION AND CONCLUSION

This paper explored the coupling of the universities of applied sciences and the industry as discerned from accreditation reports on ICT programs in the Netherlands. Drawing on Weick (1976) to exemplify loose coupling we have two important implications for the way industry is involved in the ICT curriculum. First, our results indicate that on both the strategic and the operational level, there are mechanisms in place to couple the industry with the curricula. The 53 programs all use different combinations of mechanisms to ensure the coupling yet the strengths of the coupling vary (Weick, 1976; De Caluwe, 2012). Figure 1 specifies the relationship between the universities of applied sciences and the industry on strategic and operational levels by observed mechanisms of interaction. According to the accreditation reports, the involvement of industry is most prominent at governance or strategic levels which suggest a high degree of commitment and compliance to formal requirements. Dutch Accreditation Process specifies that industry collaboration in governance is an obligatory element for positive evaluation (NVAO, 2011). This may be problematic as accountability is limited to few representatives from the industry which occupy positions in Advisory Boards. It cannot clearly show in what ways does the industry actually exert influence over the curricula. In accreditation reports, we therefore have a symbolic compliance to national regulation whereas the goals of industry interaction are not clearly reported or are considered “obscure” (Bromley and Powell, 2012).

At the same time, the analysis suggests that the coupling with the industry is low on operational or practical levels. This leads us to conclude that this is either due to a lack of interaction or an underrepresentation of real practice. The operational level shows loose coupling (Weick, 1976) with industry, or the collaboration with industry is not as tightly coupled as on strategic (governance) levels.

Future studies should address the interaction with companies at more practical levels and by conducting in depth interviews with lecturers and industrial stakeholders which participate in curricular activies. At the same time, research is needed to understand how well are the representatives of the industry able to capture the essential requirements and company needs in training the labour force. Exploring the mechanisms by which industry collaboration can be better entrenched with the teaching and research programmes and evaluation procedures which would incorporate the perspective of companies in the field or industry should be the next important step if adequately skilled workforce is to be provided to ever changing labour markets.

REFERENCES

Alajoutsijärvi, K., Juusola, K. and Siltaoja, M. (2013). Academic capitalism hits the fan: the birth acamanic capitalism”, in Virpi Malin, Murphy Jonathan and Siltaoja Mario (eds.), Getting Things Done, Dialogues in Critical Management Studies, Vol. 2, Emerald Group Publishing Limited, 91-121.

Awbery, S. M. (2002). Making the “invisible hand” Visible: The Case for Dialogue about Academic Capitalism. National Center for Research on Teacher Learning, East Lansing, MI.

Bennenworth, P. and Jongbloed, B.W. (2010). Who matters to universities? A stakeholder perspective on humanities, arts and social sciences valorization. Higher Education 59, 567-588.

Berente, N. (2009). Conflicting Institutional Logics and the Loose Coupling of Practice with NASA’s Enterprise Information System. Doctoral Degree. Case Western Reserve University.

Bromley, P., & Powell, W. W. (2012). From smoke and mirrors to walking the talk: Decoupling in the contemporary world. The Academy of Management Annals, 6(1), 483-530.

Cucchiarini, C., Daelemans, W. and Strik, H. (2001). Strengthening the Dutch Human Language Technology Infrastructure. ELRA newsletter, 6(4), 3-7.

de Caluwé, L. (2012). Losjes gekoppelde systemen. Het gedachtengoed van Karl Weick. In Ruijters, Manon and Simons, R.J. (eds.), Canon van het leren. 50 concepten en hun grondleggers. Kluwer:Deventer.

den Adel, M., Blauw, W. and Entzinger, H. (2003). Report of the ICT sector in the Netherlands. Draft of PEMINT project. Erasmus University Rotterdam.

Davey, T., Baaken, T., Galan Muros, V., and Meerman, A. (2011). The State of European University-Business Cooperation. EU:Brussels.

Dijk van, T.A. (1980). Macrostructures. Erlbaum: Hillsdale, N.J.

Frederik (2013). Waar blijft de afstemming, over de relatie IT-onderwijs en beroepenveld, Informatie 55/2, maart 2013 (38 – 43).

Gilmore, T.N., Hirschhorn, L. and Kelly, M. (1999). Challenges of Leading and Planning in Higher Education. CFAR: Philadelphia and Cambridge, US.

Hasanefendic, Heitor and Horta (2015; forthcoming). The changing patterns of non-university higher education and the challenge to train the labor force: implications for

science policy in Portugal (submitted to Technology Forecasting and Social Change, 2015)
Hazelkorn, E. and Moynihan, A. (2010). Transforming academic practice: human resources challenges. in Kyvik, S. and Lepori, B. (eds.), The Research Mission of Higher Education Institutions Outside the University Sector, Springer, Dordrecht, 77 - 97.
Huisman, J. (2008). Shifting boundaries in higher education: Dutch Hogescholen on the move. In Taylor. J.S. et.al. (eds.), Non-University Higher Education in Europe, 147 - 167.
Keep, E. (2012). Education and industry: taking two steps back and reflecting. Journal of Education and Work, 25 (4), 357 – 379.
Leisyte, L., Westerheijden, D.F. and Epping, E. and Faber, M. and Weert, E. de (2013) Stakeholders' involvement in quality assurance in higher education. Presented at EU Life Long Learning Programme project IBAR closing conference, Prague, November 10-12, 10-11-2013 - 12-11-2013, Prague.
Lynch, K. (2006). Neo-liberalism and marketisation: the implications for higher education. European Educational Research Journal, 5 (1), 1-17.
Marginson, S. and van der, Wende M. (2007). Globalisation and Higher Education, Working Paper, No. 8. OECD, Paris.
Mayring, P. (2000). Qualitative content analysis. Forum: Qualitative Social Research, 1 (2).
NVAO. (2011). Assessment frameworks for the higher education accreditation system. Nederlands-Vlaamse AccreditatieOrganisatie.
Orton, D. and Weick. K.E. (1990). Loosely coupled system: A reconceptualization. Academy of Management Review, 15, 2, 203–.
Scheele, K., Limbach, R. and de Rijcke, F, (2006). Accreditation in International Perspective. The Inspectorate of Education in the Netherlands. Utrecht.
Sharp,C. (2009) Theoretical and practical application of loose coupling: a study of criminal justice agencies in the state of Florida. The Southwest Journal of Criminal Justice, 6 (1), 45-58.
Slaughter, S., and Rhoades, G. (2004). Academic Capitalism and the New Economy: Markets, State and Higher Education. Baltimore, MD: Johns Hopkins University Press.
Teichler, U. (2007). Does higher education matter? Lessons from a comparative graduate survey. European Journal of Education, 42 (1), 11-34.
van Ruud, D. (2001). Iets Met Computers: Over Beroepsvormiing van de Informaticus. Katholieke Universities Nijmegen.
Washburn, J. (2005). The Corporate Corruption of Higher Education. Basic Books: New York.
Weick, K. E. (1976). Educational organizations as loosely coupled systems. Administrative Science Quarterly, 1-19
Ylijoki, O, H. (2003). Entangled in academic capitalism? A case study on chaning ideals and practices of university research. Higher Education, 45 (3), 307-335.

“What’s the weather like today?”: A computer game to develop algorithmic thinking and problem solving skills of primary school pupils

Hasan Gürbüz, gurbuzhsn@gmail.com
Istanbul University, Informatics Department

Bengisu Evlioğlu, bngsevlgl@hotmail.com
Istanbul University, Informatics Department, Turkey

Tuğçe Açıkgöz, tgc.ackgz@gmail.com
Istanbul University, Informatics Department, Turkey

Talat Demir, td.demir@gmail.com
Istanbul University, Informatics Department, Turkey

Kerem Hancı, kerem_hanci@hotmail.com
Istanbul University, Informatics Department, Turkey

Hulusi Gülseçen, hgulsecen@istanbul.edu.tr
Istanbul University, Astronomy and Space Department, Turkey

Sevinç Gülseçen, gulsecen@istanbul.edu.tr
Istanbul University, Informatics Department, Turkey

Abstract

The effect of the Information Technologies on the development of algorithmic thinking skills of primary school pupils is growing rapidly. This paper examines the effect of a weather forecast game developed with various programming languages on reasoning skills of children. The aim of this computer-based game is to make children develop reasonable algorithms and create criteria by using simple visual and linguistic templates, and establish relationships between these criteria by making forecasts. So that the pupils build forecast models related to algorithmic criteria by playing a game and develop algorithmic thinking skills, and have analytic information about everyday life weather events. At the development stage of this game, maximum importance has been attached to make it interesting and visual. Repetitive steps and instructions have been avoided. This game has been aimed to design an educative way to develop their reasoning and establishing logical relationship skills. An experiment was conducted with primary school pupils between the age of 8-10 to test their algorithmic thinking skills. The results of measurements before and after playing the weather forecast game show an increase of algorithmic thinking and problem solving skills of pupils.

Keywords

Problem Solving Skills, Algorithmic Thinking, Weather Forecast, Computer Game, Primary School

INTRODUCTION

Nowadays, the usage of computer-based learning technologies become increasingly common in schools. Lectures, discussions and problem solving can be achieved more easily over these technologies. Students are able to do their homeworks easily with a tablet computer and have the opportunity to benefit from the unlimited benefits offered by information technologies such as computer games, which are every child’s passion.

It is well known that computer games could be effectively used to develop reasoning skills of children. Computer games may be considered as entertainment tools, but they are also very effective learning ones (Prensky, 2005).

The generation born in the digital age, so-called digital natives (Prensky, 2001), use the constantly changing and evolving technology in all areas of their lives. Game playing is on the top of these areas. This situation could allow the creation of more effective games to make learning funnier and permanent and to develop the algorithmic thinking and problem solving skills of children (Prensky, 2005).

According to Özkan (2009), an algorithm is a commands sequence that defines a problem to be solved, that all of its steps are definable, that comprises the consecutive steps that designed to solve the problem by revealing all these steps in a meaningful way. Considering these properties, it can be seen that people use algorithms in their daily lives. According to Futschek (2006), algorithmic thinking is a pool of abilities that are connected to construct and understand algorithms. Cooper et al. (2000) indicated that many undergraduates, especially computer science students are not capable to think algorithmic, that they are not able to develop a formal step-wise algorithm to solve a given problem. In this case, learning algorithmic thinking has to start in the early years of a child’s life, and according to the child's learning capacity (Rushan and Sajid 2012), should be developed at school or home environment.

There are some programming environments like Alice (Cooper et al. 2000), Logo (Papert 1980), Baltie and Scratch (Maloney et al. 2004), suitable for very young children. These environments allow the manipulation of virtual objects with a wide variety of commands. Tim the Train (Futschek and Moschitz 2011) provides the suitable environment to children to develop their algorithmic thinking skills with tangible objects and a variety of interesting tasks. (Bottino et al. 2011) present the relationship between playing digital mind games and the development of reasoning skills, by evaluating the school performance of primary school pupils. Kula and Erdem (2005) also evaluate the development of arithmetic skills by testing elementary school students with an educational computer game; Add’em Up. Many studies on this subject, as evidenced by different perspectives, show that educational games help to develop algorithmic thinking and problem solving skills of children.

A problem can be expressed as an obstacle to anyone that wants to reach the target. In everyday life, people are faced with many situations to solve problems. Accordingly, problem solving is not only the analysis and solution of mathematical problems, but also a set of activities aimed to eliminate the difficulties encountered by individual’s everyday life.

D’Zurilla and Goldfriend (1971) gather the problem solving process under four major skills: Problem definition and formulation; Generation of alternative solutions; Decision making; Solution implementation and verification.

By constructing problem solving methods as a five steps model, IDEAL has generated a problem solving model (Bransford et al. 1998). These five steps are as follows: Identify problems and opportunities; Define goals and represent the problem; Explore possible strategies; Anticipate outcomes and act; Look back and learn.

In the Creative Problem Solving Model (Isaksen and Treffinger, 1985) creative thinking is described as making and communicating connections to: think of many possibilities; think and experience in various ways and use different points of view; think of new and unusual possibilities; guide in generating and selecting alternatives. Critical thinking is described as analyzing and developing possibilities to: compare and contrast many ideas; improve and refine ideas; make effective decisions and judgments; provide a sound foundation for effective action. These definitions are used in a six-stage, problem-solving process; Mess finding, Data Finding, Problem Finding, Idea Finding, Solution Finding, Acceptance Finding (Canedo, 1990).

According to Balcı (2007), there are three basic properties for problem solving. These are: Problem is a challenge for an individual; It is a situation that an individual needs to solve; Individual has previously encountered with the problem and he/she has no preparation to solve it.

Larkin (1980) specifies that students can learn to solve problems, thus problem solving must be a part of school education. Whereas formal education stands on abstract informations that cannot be transferred to everyday life. Problem solving has been learned since childhood and problem solving skills are developed at school. Considering education as a problem solving process, the students are expected to be good problem solvers (Serin et al. 2010).

To observe problem solving development in primary school pupils, problem solving therapy, named ADAPT can be exemplified (D'Zurilla and Nezu, 2010). It consists of five steps as Attitude, Define, Alternatives, Predict and Try out. Attitude stands for an individual's positive and optimistic attitude before solving a problem. Define stands for the individual's achievable goals by defining the problem in a realistic way. Alternatives stand for some alternative ways to solve the problem. Predict stands for the individual's best choice in alternative ways, by predicting positive and negative sides of these ways. The last one, Try out includes the individual's implementation of the result in his/her everyday life (Totan and Kabasakal, 2012).

General problem solving methods are as follows: To guess; To find solutions and answers to problems; To use the trial and error method; To do modelling, To reach the analysis by listing, grouping, classifying; To search patterns among the information given; To use hypotheses and assumptions; To make the backward solution of the problem.

Process skills used to solve a problem and to reach the solution can be explained under four titles: Reasoning can be described as making inferences based on logic rules, modelling algorithms and using these models in problem solving. Inductive reasoning skills are needed to produce descriptive knowledge, deductive reasoning skills are needed to produce functional knowledge and reductive reasoning skills are needed to produce systematic knowledge (Apaydın and Taş, 2010). Algorithmic thinking refers to the process of reaching a solution by establishing a relationship between abstract and concrete objects with sequential processing cycle intended for a solution. Associating connections: For the main purpose of solving problems, it is necessary to establish all abstract and concrete relationship between objects, sets, entities and models. Association is a modelling process that allows to build relationships to find possible solutions between objects, sets and entities. Communication is the skill of establishing relations between symbols and abstract, intuitive algorithmic language. Communication skill determines expressing algorithmic thoughts by using concrete models, forms, images, graphics and tables. It provides solving problems by using daily language with algorithms and symbols.

Computer games activities may be useful to improve reasoning, communication and problem solving skills of primary and secondary school students. These activities provide students with opportunities to activate and promote reasoning strategies such as analogy, generalisation, progressive and critical thinking a debugging based on visualisation and empirical inference (Kaur and Toh, 2012).

The purpose of this study is to examine the effects of educational game based on learning in primary school students and to investigate the relationship between educational games and problem solving skills. A meteorological forecasts game was developed and an experiment was conducted with primary group (Bebras.org) pupils.

METHODS

Design and development of the weather forecast game

The aim of this computer based game is to develop the problem solving abilities of the students aged between 8-10. Assistance has been made by a teacher, a psychologist, a pedagogue and an information technologies expert. So that a weather forecast game suitable for primary group students has been designed. This game has been developed by using different computer programming languages. Weather forecast game was designed enjoyable and suitable for the cognitive development level of the students. Weather forecast game is converted entirely from numeric form into categorical form because the aim was to measure the effects of students' reasoning skills on communication and association skills. It is aimed to observe whether these skills are developed or not. During the design phase, the algorithms are developed by a weather forecast expert.

Forecast parameters consist of 4 divisions and 3*4*3*4=144 variables are determined. According to the results of these variables 6 forecast outcomes are determined. Parameters are;

- Solar conditions: Sunny, Few Cloudy, Mostly Cloudy
- Temperature**:** Cold, Warm, Hot, Very Hot
- Humidity: Low Level of Humidity Under %40, Middle Level of Humidity Between %40-%70, High Level of Humidity Above %70
- Wind**:** Low Level of Wind, Middle Level of Wind, High Level of Wind, Storm

1.SOLAR CONDITIONS	2. TEMPERATURE	3. HUMIDITY	4. WIND
a. ÇOK BULUTLU	d. ÇOK SICAK	h. NEM YÜKSEK %70 VE ÜSTÜ	k. FIRTINALI
b. AZ BULUTLU	e. SICAK	i. NEM ORTA %40-%70	l. KUVVETLİ RÜZGAR
c. GÜNEŞLİ	f. ILIK	j. NEM DÜŞÜK %40 VE ALTI	m. RÜZGAR ORTA KUVVETTE
	g. SOĞUK		n. HAFİF RÜZGARLI

Table 1: Weather forecast game algorithm table

Outcomes are determined according to the variables in Table 1. Assistance in the analysis of these outcomes has been made by a forecast expert. These outcomes are: Open air, Rainy, Snowy, Lightning, Mist and Storm.

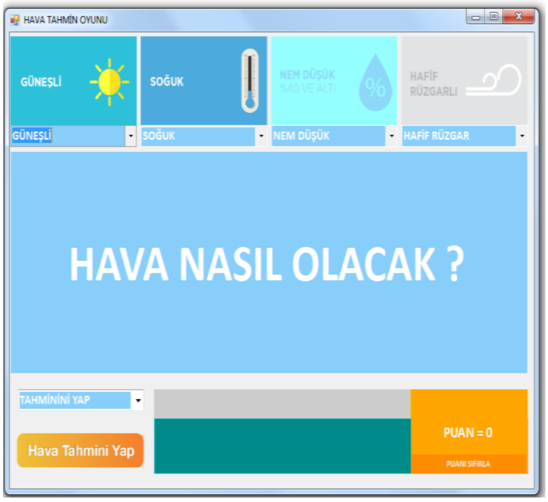


Table 2: Weather forecast game screen design Table 3: En example of forecast result according to the WFG

Weather condition parameters are located visually as in the Table 2. Forecasts can be made by selecting the weather condition scenarios with arrows. Students can do their forecasts with the help of the forecast part. And by the weather forecast part, it is decided whether the student's forecast is consistent with the real forecast and a score is given. The game wants students to make forecasts by modelling from 3*4*3*4 possible situations. To make forecasts, it is necessary for students to come through by using problem solving methods. To achieve this it is also crucial for students to move on with association, communication, modelling, and reasoning. For example, does it rain when it is sunny, cold; there is a low level of humidity and a low level of wind. Definitely students will answer this question by using reasoning and algorithmic thinking skills. In algorithmic thinking there will be a cycle to reach definite conclusions. For example;

Sunny	Cold	High level of humidity	Low level of wind	Result: Foggy
Few cloudy	Cold	High level of humidity	Low level of wind	Result: Foggy
Mostly cloudy	Cold	High level of humidity	Low level of wind	Result: Snowy

Table 4: An example of weather condition algorithm cycle

As in the Table 3 and 4 a forecast result is seen. It is chosen mostly cloudy from solar conditions division, warm from temperature division, high level of humidity from humidity division and low level of wind from the wind division. According to the results of this algorithm, the weather forecast result is 'rainy'.

Research group

The study is carried out with the primary group students visiting the meteorology office in March 2015. The global ability test is applied to the 2nd, 3rd and 4th grade students.

Data collecting tools and the limitations

Questions are prepared over analytical, verbal abilities and logic of models according to the global test method. Tests are applied before and after the weather forecast game. An education expert and a pedagogue helped in the accusation of the global ability test and the weather forecast game. The first three questions of the test aimed to measure reasoning, 9th and the 10th questions aimed to measure algorithmic thinking, 4th, 5th and the 6th questions aimed to measure association, 7th and 8th questions aimed to measure communication and problem solving skills. The aim of this test is to measure whether the weather forecast game has an effect on the students’ problem solving skills, algorithmic thinking, reasoning, association and communication skills or not. Global ability test consists of 10 questions and it is applied to 15 students from each group. The results obtained before the game and after the game have been categorised in a table as TRUE and FALSE. These categorical data have been transformed into a code to analyze if there is a meaningful result or not. The results are analyzed with the SPSS 16.0 statistics programme. WFG is applied with a small number of students. It is considered more useful to apply and evaluate this test to a large number of students from different schools.

The analyze of data

Problem solving									
Reasoning			Associating			Communication		Algorithmic thinking	
1.Q	2.Q	3.Q	4.Q	5.Q	6.Q	7.Q	8.Q	9.Q	10.Q

Table 5: The distribution of questions according to the problem solving

The test consists of 10 questions and aimed to develop students’ problem solving skills. Test has been applied for the first time before the game is played. By making students play WFG, the effects of the game on students’ problem solving abilities are investigated. Test has been applied for the second time after the game is played. According to the test results, true answers are categorised as TRUE and false answers are categorised as FALSE. Coded data are analyzed in the SPSS 16.0 programme.

Nonparametric tests are applied because of the assumption of nominal coded data are not distributed normally. Sample count is formed by N=15 person for every group. By nonparametric Friedman test used to compare more than two paired sample mass, the equality of averages for all answers are tested to see if there is a meaningful difference.

Answers Before/ after	answer1	answer2	answer3	answer4	answer5	answer6	answer7	answer8	answer9	answer10	answer1s	answer2s	answer3s	answer4s	answer5s	answer6s	answer7s	answer8s	answer9s	answer10s
Mean Rank	8,77	6,77	8,77	7,43	6,77	7,43	8,77	7,43	8,10	7,43	12,77	12,10	14,10	12,77	14,10	14,77	12,77	12,10	14,10	12,77

Test Statistics[a]	
N	15
Chi-Square	97,745
df	19
Asymp. Sig.	,000
a.Friedman Test	

Table 6: The average ranking for primary group by Friedman test

As it is seen in Table 6, for primary group, it has been determined that the averages are not equal for all paired answers. Asymp. Sig = 0,000 shows that for the multiple comparisons of the answers, the averages are not equal. Although hypothesis H0 is based on the equality of the averages of paired answers for the primary group, hypothesis H1 is based on the opposite (not equal). Because of Asymp. Sig $P<0.05$ hypothesis H1 is accepted and for all the answers of primary group pupils, a meaningful statistical difference is observed. That’s why for the first and last tests, pairwise comparison tests like Wilcoxon, Sign, McNemar, Marginal and Homogeneity nonparametric tests are applied for all the answers.

FINDINGS

The test results applied to students are evaluated by the nonparametric statistical relation analyze method. And it is investigated if there is a significant difference between the statistical results of pre-test and final test. The results are attended from the primary group as in Table 8.

Test Statistics (Z Asymp.Sig. (2- ailed))	Wilcoxon Signed Ranks Test a.Based on negative ranks.	Sign Test a.Binomial distribution used.	McNemar Test a.Binomial distribution used.	Marginal Homogeneity Test
Answer1 (Before-After)	-2,449[a] ,014	,031[a]	,031[a]	,014
Answer2 (Before-After)	-2,828[a] ,005	,008[a]	,008[a]	,005
Answer3 (Before-After)	-2,828[a] ,005	,008[a]	,008[a]	,005
Answer4 (Before-After)	-2,828[a] ,005	,008[a]	,008[a]	,005
Answer5 (Before-After)	-3,317[a] ,001	,001[a]	,001[a]	,001
Answer6 (Before-After)	-3,317[a] ,001	,001[a]	,001[a]	,001
Answer7 (Before-After)	-2,236[a] ,014	,031[a]	,031[a]	,014
Answer8 (Before-After)	-2,646[a] ,008	,016[a]	,016[a]	,008
Answer9 (Before-After)	-3,000[a] ,003	,004[a]	,004[a]	0,03
Answer10 (Before-After)	-2,828[a] ,005	,008[a]	,008[a]	,005

Table 8: The results of the pre-test and final test relation analysis from the Primary group

As it is seen on Table 8, there is a meaningful statistical difference in the final test results compared to the pre-test results. With 95% confidence, hypothesis H0 is based on the acceptance of there is not a meaningful difference in the test results. Though with 95% confidence, hypothesis H1 is based on the opposite. For the Asymp. Sig. (2-tailed) Wilcoxon, Sign, McNemar, Marginal Homogeneity tests, with 95% reliability scale and $p<0.05$, there is a meaningful statistical difference between the first and last tests. It can be pointed out that the positive effect of the WFG on problem solving skills of the primary group students.

Answers	Pre-test		Final Test	
	True	False	True	False
1	5	10	11	4
2	2	13	10	5
3	5	10	13	2
4	3	12	11	4
5	2	13	13	2
6	3	12	14	1
7	5	10	11	4
8	3	12	10	5
9	4	11	13	2
10	3	12	12	3
Mean	3,5	11,5	11,8	3,2
St. Dev.	1,18	1,18	1,40	1,40

Table 9: Answer distribution of the pre-test and final test from the primary group

According to Table 9, the number of the students answering correctly at the final test increased by 4 times compared to the pre-test answers. From pre-test to final test, true mean increased from 3,5 to 11,8 and false mean decreased from 11,5 to 3,2.

It is determined that WFG created a significant difference in the answers of the first three questions so that this game has a positive effect on the students' reasoning skills. The number of the true answers in the first three questions increased by three times in the final tests. WFG created a significant difference in the answers of the 4th, 5th and the 6th questions so that this game has a positive effect on the students' association and problem solving skills. It is observed that the number of the true answers of the 4th, 5th and the 6th questions increased 3 times in the final tests. Answers of the 7th and the 8th questions created a positive effect on the students' communication and problem solving skills. It is observed that the number of the true answers of the 7th and the 8th questions increased 2,5 times in the final tests. Answers of the 9th and the 10th questions created a positive effect on the students' algorithmic thinking and problem solving skills. It is observed that the number of the true answers of the 9th and the 10th questions increased 2,5 times in the final tests.

CONCLUSION

According to the test results of WFG, a positive effect on students' problem solving abilities is observed. And also a significant difference is observed between the pre-test and final test results. According to the global ability test results, students performed better after playing the WFG.

Reasoning, algorithmic thinking, association and communication skills are considered as problem solving abilities (Futschek and Moschitz, 2011). In the research, it is investigated if WFG has an effect on these skills and if yes, in what way it affects these skills. Considering these four problem solving approaches, it is accepted that WFG created a statistically important difference and this difference created a positive effect after the analysis of test results.

As a result, it is clearly understood from the tables in the 'Findings' part, WFG can improve the students' problem solving abilities by positively effecting students' reasoning, association, communication and algorithmic thinking skills.

The positive effects of the WFG on children should be investigated with a large number of students. By gathering large number of students from different schools, more results should be obtained and the test results should be analyzed again.

REFERENCES

Apaydın Z., and Taş E., Farklı Etkinlik Tiplerinin Öğretmen Adaylarının Akıl Yürütme Becerileri Üzerindeki Etkileri, Journal of Turkish Science Education.Volume 7, Issue 4, December 2010.

Balcı, G., İlköğretim 5. Sınıf Öğrencilerinin Sözel Matematik Prolemlerini Çözme Düzeylerine Göre Düzeylerine Göre Bilişsel Farkındalık Becerilerinin İncelenmesi, Çukurova Üniversitesi, Yüksek Lisans Tezi, 2007.

Baltie home page, http://progopedia.com/language/baltie/

Bottino, R. M., Ott, M., and Tavella, M. (2013). Children's performance with digital mind games and evidence for learning behaviour. In Information Systems, E-learning, and Knowledge Management Research (pp. 235-243).Springer Berlin Heidelberg.

Bransford, John D., Heynes, Ada, F., Stein, Barry, S., Lin, Xiaodong, The IDEAL Workplace: Strategies for Improving Learning, Problem Solving, and Creativity, 1998.

Canedo, M. (1990). The Inventive Thinking Curriculum Project. DIANE Publishing.

Cooper, S., Dann, W., & Pausch, R. (2000, November). Developing algorithmic thinking with Alice. In The Proceedings of ISECON 2000 (Vol. 17, pp. 506-539).

D'Zurilla, T. J., and Goldfried, M. R., 1971. Problem solving and behaviour modification. Journal of abnormal Psychology, 78, 107-126.

D'Zurilla, T. J., and Nezu, A. M., 2010. Problem-solving therapy. In K. S. Dobson (Ed.), handbook of cognitive-behavioral therapies (3rd ed., pp. 197-225). New York: Guilford press.

Futschek, G. (2006). Algorithmic thinking: the key for understanding computer science. In Informatics education–the bridge between using and understanding computers (pp. 159-168). Springer Berlin Heidelberg.

Futschek, G., and Moschitz, J. (2011). Learning algorithmic thinking with tangible objects eases transition to computer programming. In Informatics in Schools. Contributing to 21st Century Education (pp. 155-164). Springer Berlin Heidelberg.

Isaksen and Treffinger, "Creative Problem Solving: The Basic Course (1985) by Isaksen and Treffinger" and published in Buffalo, New York, by Bearly.

Kaur, Berinderjeet and Toh, Tim Lah, Reasoning, Communication and Connections in Mathematics: Yearbook 2012, An introduction.

Kula, A., and Erdem, M., (2005). Öğretimsel bilgisayar oyunlarının temel aritmetik işlem becerilerinin gelişmesine etkisi. Hacettepe Üniversitesi Eğitim Fakültesi Dergisi, 29(29).

Maloney, J., Burd, L., Kafai, Y., Rusk, N., Silverman, B., and Resnick, M. (2004, January). Scratch: a sneak preview [education]. In Creating, Connecting and Collaborating through Computing, 2004. Proceedings. Second International Conference on (pp. 104-109). IEEE.

Özkan, Y., 2009 Programlama Dilleri: C ile Programlama, Alfa Yayınları, İstanbul, ISBN: 9786051061009.

Papert, S. (1980). Mindstorms: Children, computers, and powerful ideas. Basic Books, Inc.

Prensky, M. (2001). Digital natives, digital immigrants part 1. On the horizon,9 (5),1- .

Prensky, M. (2005). Computer games and learning: Digital game-based learning. Handbook of computer game studies, 18, 97-122.

Rushan, Z., and Sajid, M. (2012). Rapidmental Computation System as a Tool for Algorithmic Thinking of Elementary School Students Development. European researcher= Европейский исследователь, 25(7).

Serin, O., Serin, N. B., Saygılı, G. (2010). Developing problem solving inventory for children at the level of primary education (PSIC). Elementary Education Online, 9(2), 446-458.

Totan, T., and Kabasakal, Z. (2012). Problem çözme becerileri eğitiminin ilköğretim altıncı sınıf öğrencilerinin sosyal ve duygusal öğrenme ihtiyaçları ve becerileri üzerine etkisi. *İlköğretim Online*, *11*(3).

Hasan Gürbüz is an MA Student in Informatics Department at Istanbul University.

Tuğçe Açıkgöz is an MA Student in Informatics Department at Istanbul University.

Kerem Hancı is an MA Student in Informatics Department at Istanbul University. He is a software test engineer on banking at the same time.

Sevinç Gülseçen is a Professor and Head of the Department of Informatics at Istanbul University

Bengisu Evlioğlu is an MA Student in Informatics Department at Istanbul University.

Talat Demir is an MA Student in Informatics Department at Istanbul University.

Hulusi Gülseçen is an Associate Professor in Astronomy and Space Science Department at Istanbul University.

LEM-LT: Analytical educational framework for the eCAL program

Seeta Jaikaran-Doe, seetam@utas.edu.au
University of Tasmania, Australia

Abstract

A mixed methods approach was adopted to demonstrate the practical implementation of LEM-LT framework in the evaluation of the eConnect and Learn program (eCAL) (Gopeesingh, 2010) initiated in 2010 in Trinidad and Tobago. The LEM section of the framework was constructed through the lens of the systems level, six design principles and 21st Century skills. The LT section consisted of the alignment of the Levels of Teaching Innovation (LoTi) (Moesrch, 2010) and Technological Pedagogical Content Knowledge (TPACK) (Koehler, Mishra, & Yahya, 2007). Each component of LEM-LT guided the analyses of quantitative or qualitative data. Results confirmed there was a statistically significant difference ($p \leq .001$, two-tailed) between pre-service and in-service teachers' TPACK confidence to integrate ICT for teaching and supporting their students learning with ICT in relation to the six design principles and 21st Century skills. Pre-service teachers had higher mean scores than in-service teachers. In addition, results indicated inadequate provisions in the systems level prevented teachers and students from the full use of the eCAL program. The paper concluded adoption of the eCAL program is still at its early stages, reaching the Infusion level of LoTi with substantial use of pedagogical content knowledge and technological knowledge. This paper focused on the practical implementation of LEM-LT for guiding the analyses relating to the eCAL program and not on a detailed description of the processes involved in the construction of the framework.

Keywords

LEM-LT, eCAL program, education informatics, technological pedagogical content knowledge, Levels of Teaching Innovation Adaptation.

INTRODUCTION

Although the 21st Century has brought a new wave of information, new educational technologies, and different modes of communication, many educators have not yet developed the confidence to integrate these in their teaching and students' learning. A review of the literature has revealed two out of every five students leaving initial teacher education programs at two Universities in Australia had no confidence or just some confidence to use ICT for teaching and learning (Finger, Jamieson-Proctor, & Albion, 2010). Tersptra (2010) concluded in her study that although pre-service teachers use digital technologies in their personal lives on a daily basis, they fail to use this technological knowledge in the preparation of their own teaching. Collins and Weiner (2010) noted that some of the challenges in the education informatics environment are:

> gaps in existing information resources; …scarce resources for the development of concepts, models, theories, and techniques related to the use of information and communication technologies (ICT) for learning and teaching; minimal understanding of how people access and use education information and digital information resources, services, systems, environments, and media for learning.

To fill the existing gap, a conceptual and analytical framework, LEM-LT, was constructed. The framework consists of two major sections. The first is the learning environment model (LEM) and the second is the alignment of LoTi (Moesrch, 2010) with TPACK (Koehler et al., 2007) (LT). The framework is multi-faceted with essential components to enhance teachers' confidence for building 21st Century pedagogical skills and instructional dissemination. In addition, LEM-LT framework provides educators, administrators and other stakeholders with the opportunities to analyse their schools' learning environment and identify the elements necessary to promote a rich, productive and reliable informatics education environment.

RATIONALE AND CONSTRUCTION OF LEM-LT FRAMEWORK

In 2010, the Trinidad and Tobago government initiated the eCAL program which provided free personalised laptop computers for all students transitioning from primary schools (11 to 12 years old) to secondary schools (Gopeesingh, 2010). The Ministry of Education (MOE) anticipated the program will build teachers' confidence to infuse ICT in their pedagogical content practices and support students' use of ICT in the information age. According to the Prime Minister of the country, the program also aimed at developing a technologically capable workforce (Persad-Bissessar, 2010) with appropriate 21st Century skills to participate successfully in the present and future global information economy. The LEM-LT framework has the potential to underpin these goals. It can be used as a conceptual, informative, and an evaluative tool to guide teachers of ways to promote 21st Century teaching and learning. In this study it was used as an evaluative tool to analyse data collected on the eCAL program from pre-service and in-service teachers.

The framework comprised the learning environment model (LEM) which is made up of the systems level, the six design principles, and 21st Century skills. The LT section is a combination of the alignment of LoTi (Moesrch, 2010) and TPACK (Koehler et al., 2007). Factors in the systems level and full application of the six design principles can provide a platform for building teachers' LoTi and TPACK. The components of the LEM-LT framework are illustrated in Figure 1.

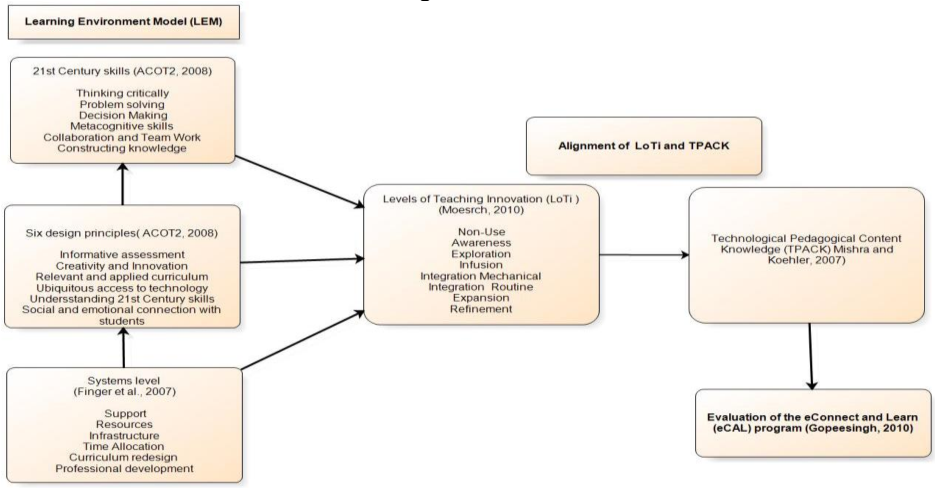


Figure 1. LEM-LT Framework

COMPONENTS OF LEM-LT FRAMEWORK

Systems level

The systems level of LEM-LT comprises six elements: support, resources, infrastructure, time allocation, curriculum redesign and professional training. These elements have the potential to transform the way teachers deliver instructions and create new ways for disengaged students to become engaged learners. *Support* is essential from the Department of Education, administrators, parents, technicians and other stakeholders to enable teachers and students to enhance and develop confidence in the integration of technological devices for teaching and learning (NCATE, 1997) respectively. *Resources*, such as LCD projectors and interactive whiteboards, are necessary to provide visual educational stimulation for students. Access to internet connectivity and the World Wide Web provide open access to information in real time (Finger, Russel, Jamieson-Proctor, & Russel, 2007). *Infrastructure* describes the appropriate physical layout of electrical installations, servers, local networks and wireless routers (Severin & Capota, 2011) to promote proper functioning of educational technological devices. *Time allocation* is important for teacher preparation to *redesign the curriculum*, collaborate and develop strategies to facilitate, and utilise technological pedagogical content knowledge (Mishra & Koehler, 2008). Students too must be given time to complete tasks and develop skills for 21st Century learning. *Professional development* programs are necessary to improve teachers' competencies in teaching and learning, and need to be practiced routinely on a regular basis (Darling-Hammond, Bransford, LePage, Hammerness, & Duffy, 2005).

Six design principles

The six design principles (ACOT2, 2008) are necessary to focus on the dimensions of 21st Century skills for facilitating and enhancing teaching and learning with new and emerging technologies. Teachers and students need to re-educate and reskill themselves for *understanding 21st Century Skills and Outcomes* for the demands of the digital global community. Continuous *Informative Assessment* is integral (Lei, Conway, & Zhao, 2008) to make informed decisions for readjusting instruction and curricular changes for *Relevant and Applied Curriculum*. This construct enables 21st Century skills to facilitate immersion in real life activities, problem-based and project based learning and leverage Web 2.0 and other digital technologies. Its application result in a *Culture of Innovation and Creativity* which acknowledges the synergy that drives today's global economy. This is made possible when students and teachers have *Ubiquitous Access to Technology* since there are numerous free web-based tools for them to use together (Madden & Fox, 2006). They may require all day, everyday access to anywhere, anytime educational technology to empower them to do research, collect information and analyse data. The interaction between educators and students should be done cognitively, *socially and emotionally* to produce a vibrant, successful education informatics environment.

21st Century skills

Provision of the systems level (Finger et al., 2007) and the application of the six design principles will facilitate promotion for the development of 21st Century skills: critical thinking, problem-solving, decision making, metacognitive skills, creativity and innovation, and collaboration and team work. These are necessary skills for teachers and students to deconstruct and reconstruct new knowledge for the challenging information economy.

Technological Pedagogical Content Knowledge (TPACK)

TPACK is an integrative approach to teaching and learning with technology. It is the integration of three knowledge constructs: content knowledge (CK), pedagogical knowledge (PK) and technological knowledge (TK) to produce four new constructs, technological content knowledge (TCK), technological pedagogical knowledge (TPK), pedagogical content knowledge (PCK) and technological pedagogical content knowledge (TPACK).

TCK encompasses how technology can be used by educators to create new representations and transformations for specific content (Mishra & Koehler, 2008). TPK focuses on how pedagogical practices might change as a result of the selection and application of the most suitable technologies for the dissemination of information and completion of tasks. PCK is an understanding of “how particular topics, problems, or issues are organized, represented, and adapted to the diverse interests and abilities of learners, and presented for instruction” (Shulman, 1986, p. 8). TPACK, on the other hand is the knowledge teachers should adopt for integrating technology into their pedagogical practices in specific disciplines, interdisciplinary or multidisciplinary content areas. It requires appropriate technologies to be utilised for building upon existing knowledge and ‘developing new epistemologies or strengthening old ones’ (Mishra and Koehler, 2008 p.10). The LEM-LT framework utilise the seven domains for alignment with the Levels of Teaching Innovation in relation to their specific functions in the evaluation of the eCAL program. The TPACK framework in Figure 2 illustrates the seven domains.

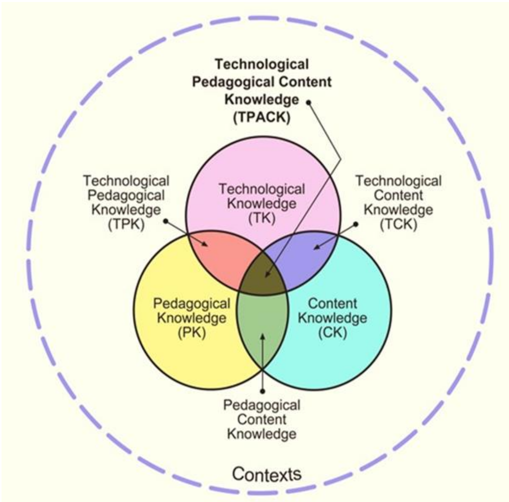


Figure 2. Domains of the TPACK framework (graphic from http://tpack.org)

Levels of Teaching Innovation (LoTi)

The Levels of Teaching Innovation (LoTi) (Moesrch, 2010) was adapted from the Concerns-Based Adoption Model (CBAM) (Hall & Hord, 1987). LoTi is characterised with seven levels which emphasise a constructivist approach (Vygotsky, 1987). Learning principles, such as Bloom’s taxonomy, problem-based learning, value beyond school, differentiated curriculum, constructive methods, authentic and relevant learning in real-time situation, and collaborative tools are characterised in different stages on the LoTi continuum. A description of each level is further outlined in the results section of this paper with the alignment of LoTi and TPACK constructs in Table 2.

PRACTICAL IMPLEMENTATION OF LEM-LT FRAMEWORK

Methodology

A mixed methods approach (Creswell & Clark Plano, 2007) was utilised in this study to provide evidence, understanding, and insight of teachers’ confidence to use the affordances of the eCAL program for teaching and learning; to explore the provisions in the learning environment; and to evaluate the stage the program had reached. The combination of two different and contrasting approaches (quantitative and qualitative) underpinned by LEM-LT informed rich analyses of three research questions (RQ):

RQ 1: How confident were teachers to use ICT to support their students’ use of ICT?

RQ 2: What provision was made in the learning environment to facilitate the successful implementation of the eCAL program?

RQ 3: What stage the eCAL program had reached in relation to the Levels of Teaching Innovation and Technological Pedagogical Content Knowledge?

Data collection took place during September, 2013 to January, 2014 in two phases. Firstly, quantitative data was collected for RQ1 through a validated TPACK survey with Cronbach's Reliability Coefficient = .97 (Jamieson-Proctor et al., 2013). A total of 53 pre-service teachers from two universities and 173 in-service teachers from 12 secondary schools in Trinidad and Tobago responded to the TPACK survey on a Likert scale ranging from 1 to 6, with 1 = not confident, 2 = partially confident, 3 = moderately confident, 4 = confident, 5 = very confident, 6 = extremely confident for each item. For the second phase, teachers who participated in the survey, were further invited to take part in semi-structured face-to-face interviews for RQ 2 and RQ 3. A total of 21 in-service teachers and 15 pre-service teachers accepted the invitation. The responses of pre-service teachers were excluded from the analyses of RQ 2 and RQ 3 because they were not directly involved in the application of the eCAL program, however, analysis of their data was important to underpin the results of RQ 1.

Data collected from the survey were analysed with IBM SPSS to answer RQ 1. Individual independent-samples t-tests were conducted for each of the 24 items. Each item was matched with either one of the relevant six design principles or an appropriate 21st Century skills (See Table 1). This was deliberately done not only to reflect how the LEM-LT framework underpinned the analysis of the survey but also to get an insight of the potential of teachers' TPACK confidence to utilise the six design principles and develop 21st Century skills in their pedagogical practices.

For RQ 2, teachers were asked about the provisions made to implement the eCAL program. For RQ 3, teachers had to describe at least two lessons to illustrate how they used computers and related devices in their teaching. Data were coded and analysed with the assistance of NVivo software. The systems level and the alignment of LoTi and TPACK in LEM-LT also guided the analysis for RQ 2 and RQ 3 respectively. An overview of all the responses for the three research questions were interrogated to present a broader picture for the evaluation of the eCAL program.

RESULTS AND DISCUSSION

Research Question 1

Analysis of the quantitative data indicated there was a statistically significant difference ($p \leq .001$, two tailed) between pre-service and in-service teachers TPACK confidence to use ICT and to support their students' learning with ICT. Pre-service teachers had higher mean scores for each item with the highest ($M = 4.72$, $SD = 0.97$) in Relevant and Applied Curriculum for integrating different digital media to create appropriate projects. In contrast, the highest mean ($M = 3.62$, $SD = 0.96$) scores for in-service teachers was in metacognitive skills for developing competencies in their subject area/s. It is interesting both cohorts had the lowest mean ($M_{pre\text{-}service} = 3.14$, SD = 0.77; $M_{in\text{-}service} = 2.79$, SD = 0.88) scores in Creativity and Innovation for providing motivation for curriculum activities. Examining the mean scores in reference to the Likert scale, pre-service teachers' TPACK confidence for using ICT and supporting their students' use of ICT ranged from moderately confident to almost very confident (M = 3.14 to 4.72) whereas in-service teachers were below moderately confident to below confident (M = 2.79 to 3.62). These results are presented in Table 1.

Although pre-service teachers were not involved in the eCAL program, they (pre-service teachers) were more knowledgeable in the application of the six designed principles and development of 21st Century skills in their teaching and learning. During the interview sessions they described how educational technology and Instructional design courses at the universities helped them to prepare and deliver lesson plans for their teaching practicum. A group of them created an online puzzle on fractions, and many of them used WebQuest, Make me Genius, Wiki, Voki, Mahara, Google Docs, and eBeam.

Table 1 *Results of the individual independent-samples t-tests positioned with the elements of the six design principles/ 21st Century skills of LEM-LT framework (df = 226)*

How confident are you to use ICT to support your students' learning with ICT in the following?	Design principles/ 21st Century skills	Teachers	*M*	*SD*	*t*	*p*
To integrate different digital media to create appropriate projects	Relevant and Applied curriculum	PS IS	4.72 3.45	0.97 1.07	7.62	.001
To engage in sustained involvement with curriculum activities	Relevant and Applied curriculum	PS IS	4.02 3.32	0.95 1.09	4.18	.001
To understand and participate in the changing knowledge economy	Creativity and Innovation	PS IS	4.15 3.22	1.06 1.05	5.62	.001
To provide motivation for curriculum tasks	Creativity and Innovation	PS IS	3.14 2.79	0.77 0.88	2.58	.011
To gather information and communicate with a known audience	Ubiquitous to technology	PS IS	4.68 3.41	1.01 1.13	7.34	.001
To engage in independent learning through access to education at a time, place and pace of their own choosing	Ubiquitous access to technology	PS IS	4.16 3.27	1.05 1.00	5.43	.001
To undertake formative and/or summative assessment	Assessment	PS IS	4.54 3.41	0.70 1.06	8.91	.001
To develop rich understanding about a topic of interest relevant to the curriculum area/s being studied	Metacognitive skills	PS IS	4.40 3.45	0.88 1.01	6.11	.001
To acquire the knowledge, skills, abilities and attitudes to deal with on-going technological change	Understanding 21st Century skills and outcomes	PS IS	4.15 3.29	1.08 1.09	5.04	.001
To demonstrate what they have learned	Constructing knowledge	PS IS	4.25 3.51	1.05 0.93	4.92	.001
To actively construct knowledge that integrates curriculum areas	Constructing knowledge	PS IS	4.32 3.44	0.98 1.02	5.50	.001
To synthesize their knowledge	Constructing knowledge	PS IS	3.96 3.38	1.02 1.05	3.50	.001
To develop understanding of the world	Curriculum redesign	PS IS	4.52 3.48	0.96 1.05	7.60	.001
To facilitate the integration of curriculum areas to construct multidisciplinary knowledge	Curriculum redesign	PS IS	3.91 3.18	0.97 1.12	4.27	.001
To gain intercultural understanding	Social and emotional connection with students	PS IS	4.53 3.33	0.85 1.05	7.60	.001
To acquire awareness of the global implications of ICT-based technologies on society	Thinking critically	PS IS	4.33 3.23	0.99 1.08	6.50	.001
To analyse their knowledge	Thinking critically	PS IS	4.00 3.41	1.02 1.03	3.67	.001
To critically interpret and evaluate the worth of ICT-based content for specific subject area/s	Thinking critically	PS IS	4.00 3.12	1.04 1.14	5.00	.001
		PS	4.40	0.98	5.14	.001

To develop competencies in your subject area/s	Metacognitive skills	IS	3.62	0.96		
To engage in activities of the learning process	Metacognitive skills	PS IS	4.70 3.56	0.95 1.00	7.27	.001
To communicate with others locally and globally	Decision making	PS IS	4.64 3.55	1.09 1.07	6.46	.001
To critically evaluate their own and society's values	Decision making	PS IS	4.11 3.18	1.01 1.09	5.53	.001
To plan and/or manage assigned curriculum projects	Collaboration and team work	PS IS	4.21 3.37	1.07 1.06	5.08	.001
To actively construct their own knowledge in collaboration with their peers and others	Collaboration and team work	PS IS	4.38 3.41	1.10 1.04	5.82	.001

Note: PS represents pre-service teachers and IS represents in-service teachers.
Special Note. The TTF Project was funded by the Australian Government Department of Education, Employment and Workplace Relations (DEEWR) through the ICT Innovation Fund.

Research Question 2

Findings for Research question 2 draws from teachers' responses relating to the systems level of the LEM-LT framework: support, resources, infrastructure, time allocation, and professional development. (Responses for curriculum redesign were not forthcoming from the interviews). The following is an anecdotal report of analyses.

Support and resources

Teachers reported although principals were supportive of technology integration for teaching and learning, they (teachers) were constrained in utilising the resources provided to enhance the eCAL program. According to the school's policy, teachers had to apply for permission to use the laptop computers, digital projectors, scanners, document/digital cameras from their department. By the time the equipment was set up in the classroom a "chunk" of their teaching time was lost. As a result of this 52% of the teachers bought their own equipment, 14% used the "talk and chalk" method, and the others utilise the resources whenever they were available.

Infrastructure

A total of 80% of the teachers articulated the infrastructure was inadequately organised in terms of electrical outlets and internet connectivity. There were less than four electrical outlets for use by teachers and students in each classroom. Additionally access to internet connectivity was a problem. The bandwidth did not have the capacity to accommodate the entire school's population. As a result of this, administrators and ICT technicians provided Internet connectivity and WiFi accessibility in two computer labs, the library, a few classrooms, and an open area, such as the auditorium. These facilities were ineffective for some teachers since it was inconvenient to take their classes and teaching materials, such as science and art, in the designated areas.

Time Allocation

Time allocated for collaboration and planning to discuss ways of improving the infusion of the eCAL program lasted for about an hour and varied from once per week, to once per month, to once per semester in five schools. In contrast, teachers from three schools expressed there was no fixed formal time for collaboration. Their personal planning time was sometimes used for class supervision when a teacher was absent, however, they informally communicated with their peers about what worked well for them in dissemination of instructions with computers and related devices during the recess or lunch period. Teachers from the other two schools reported they had not discussed their successes or failure relating to the eCAL program because their time was focused on the completion of the syllabus and/or involvement with extra-curricular activities.

Professional development

Professional development to enhance integration of the eCAL program was initially organized during the vacation period for all teachers by the Ministry of Education. The timing was inconvenient to many teachers because their vacation period was planned in advance. During the second year of the program, principals attempted to arrange internal workshops conducted by their Information Technology teachers and the ICT technicians. Although there were inconsistencies in the organization and attendance of workshops, 33% of the teachers reported they were introduced to Microsoft Photo Story, eBeam, and Google Docs. They explored websites such as Edmodo, WebQuest, and Pennacool. Additionally teachers independently learned to use Wed 2.0 technologies, accessed online training programs on WWW, and utilise software for their teaching and students' learning with their personal computers.

Although the elements in the systems level of the LEM-LT framework were not fully provided by relevant stakeholders, teachers demonstrated they were willing to propel the eCAL program forward by purchasing their own computers and related devices, attended workshops and professionally educated themselves. On the other hand, according to Rogers (2003), some of the teachers may belong to Late Majority or Laggards, and need to be persuaded before adopting the eCAL program for teaching and learning. Others may adopt the program only if the elements in the systems level are provided.

Research Question 3

Teachers' responses to their use of the eCAL program for their everyday pedagogical practices and instructional dissemination were aligned with TPACK constructs to make informed decision of the stage the program had reached on the Levels of Teaching Innovation.

Table 2. *Positioning the eCal Program on the LEM-LT Framework*

Levels of Teaching Innovation (LoTi)	**Teachers' use of computers and related devices from the eCAL program**	**Teachers' responses in %**	**TPACK Constructs**
Non-use level (0) Instructions ranged from didactic to a collaborative, student-centred learning	No use of computers and digital devices	8%	CK and PK
Awareness - Level 1 Application of digital tools and resources are used for lecture enhancement, curriculum management or as a reward for students' completion of given tasks	Teachers used PowerPoint (PP), videos, digital/document camera, and email, computers for instructional dissemination	42%	CK and PK with the application of simplest form of TK
Exploration - Level 2 Teachers and students use digital tools for extension activities, research and to produce multi-media products	Websites, videos, puzzles and tutorials	18%	Emerging PCK and greater use of TK than in Level 1
Infusion – Level 3 Changes emerge as teachers shift to a more inductive, scientific inquiry	Video, PP, spread sheet , Word Processing, hyperlinks, simulation, and Word Press	16%	Substantial use of PCK and TK.

approach. Students use digital tools and resources for completing tasks.			
Routine - Integration Mechanical, Level 4A Students are engaged in exploring real world issues and solving authentic problems using digital tools and resources. Teachers rely on pre-package materials and internal/ external professional assistance from others.	Pennacool.com and Sanako Lab 100 External software from local Ministries Video for visual arts Webinar for professional development Blogging	12%	Full use of PCK and emerging TCK and TPK
Integration Routine, Level 4B Teachers are within their comfort zone in technology integration for inquiry-based models and learner-centred strategies	A Video was made by students in geography class on the structure and cave formation An advertisement was design for a virtual media company In English class	4%	Development of TCK and TPK; and emerging TPACK
Expansion – Level 5 Teachers and students are more efficient with complex digital tools for transforming teaching and learning respectively.	No tasks were completed at this level	0%	Full use of PCK, TPK, TCK. Greater development of TPACK
Refinement- Level 6 No longer is there a division between instruction and digital tools and resources.	No tasks were completed at this level	0%	Full use of TPACK

An inspection of Research Question 3

Teachers' responses to their use of the eCAL program for their everyday pedagogical practices and instructional dissemination were aligned with TPACK constructs to make informed decision of the stage the program had reached on the Levels of Teaching Innovation.

Table 2 revealed a total of 49 responses were made. The majority of responses, 42%, were identified at the Awareness level which emphasized the lower cognitive skills of Bloom's taxonomy (Anderson, 2006): understanding, remembering and applying skills. As the Levels of Teaching Innovation became more complex, requiring higher order cognitive skills, and implementation of PCK, TPK, TCK and TPACK, teachers' responses became fewer. Although two responses were provided at Level 4B, there was an absence of learner-centred strategies, constructivist, problem-based models of teaching, metacognitive skills, creativity and innovation as well as critical thinking and problem solving. Therefore it was pertinent to rate the teachers' use of computers and related devices of the eCAL program at Level 3, Infusion, where teachers made substantial use of PCK and TK.

CONCLUSION

LEM-LT is constructed with strong theoretical underpinnings. Activity theory (Engeström, 1999) demonstrates the potential of the six design principles to guide educators to plan, disseminate, evaluate and redesign instruction to facilitate teaching and learning activities for the development of 21st Century skills. This theory is also supported by the Social and Learning Development Theory (Vygotsky, 1987). The More Knowledgeable Other, teachers and peers, can guide the transfer of knowledge to foster the development of cognitive and metacognitive skills in the multicultural social learning environment of the classrooms. Because of the explosion of knowledge via the World Wide Web and the internet, curriculum designers and teachers need to carefully select content from Relevant and Applied Curriculum to promote 21st Century skills. How they disseminate concepts for students to remember, understand, apply, analyse, evaluate and create new knowledge for the 21st Century should reflect Constructivist Learning Theory (Anderson, 2006).

LEM-LT framework guided analyses of the three research questions. Positioning the six design principles and 21st Century skills with the alignment of LoTi and TPACK provided a broader picture of the findings related to the eCAL program. Analyses of the interview data set reflected a lack of structural planning before the program was introduced. Poor provision of the elements in the systems level prevented in-service teachers to make optimum use of the program. Hence their TPACK confidence was not at the same level as the pre-service teachers but their (in-service teachers) willingness to purchase their own computers and related devices propelled the eCAL program forward. Alignment of LoTi and TPACK confirmed teachers had reached the Infusion level where changes emerged as teachers shift to a more inductive, scientific inquiry approach as students used digital tools and resources for completion of tasks. An overview of all the results from the three research questions concluded the eCAL program was still at its early stages of adoption.

Currently the LEM-LT framework has the potential to help educators to identify important components, and utilise appropriate pedagogical techniques and skills to guide the successful implementation of new technology programs in the learning environment. So far the framework was used as a platform for the analysis of quantitative and qualitative data, and provided evidence for the conclusion of the stage the eCAL program had reached. The framework is still at its early stages, and needs to be trialled on a larger scale to determine its flaws/effectiveness. This study provided baseline data for future research which can modify and strengthen the framework.

REFERENCES

ACOT2. (2008). *Apple Classrooms ofTomorrow—Today: Learning in the 21st Century*. USA: Becta Publisher

Anderson, L. (2006). *Revised Bloom's taxonomy*. Paper presented at the North Carolina Careerand Technical Education Curriculum Development Training, Raleigh, NC.

Collins, J. W., & Weiner, S. A. (2010). Proposal for the creation of a subdiscipline: Education informatics. *Teachers College Record, 112*(10), 2523-2536.

Creswell, J., & Clark Plano, V. L. (2007). *Designing and conducting mixed methods research*: Thousands Oaks, CA: Sage.

Darling-Hammond, L., Bransford, J., LePage, P., Hammerness, K., & Duffy, H. (2005). *Preparing teachers for a changing world: What teachers should learn and be able to do* San Francisco, CA: Jossey-Bass.

Engeström, Y. (1999). Activity theory and individual and social transformation. In Y. Engeström, R. Miettinen & R.-L. Punamäki (Eds.), *Perspectives on Activity Theory* Cambridge: Cambridge University Press.

Finger, G., Jamieson-Proctor, R., & Albion, P. (2010). *Beyond Pedagogical Content Knowledge: The Importance of TPACK for Informing Preservice Teacher Education in Australia*. Paper presented at the Key competencies in the knowledge society : IFIP TC 3 International Conference, KCKS 2010, held as part of WCC 2010, Brisbane, Australia.

Finger, G., Russel, G., Jamieson-Proctor, R., & Russel, N. (2007). *Transforming learning with ICT: Making it happen*. Malaysia Pearson, Prentice Hall.

Gopeesingh, T. (2010). *eConnect and Learn Programme Policy*. Trinidad and Tobago: Government Printery.
Hall, G. E., & Hord, S. M. (1987). *Change in Schools:Facilitating the process*. New York State University of New York Press.
Jamieson-Proctor, R., Albion, P., Finger, G., Cavanagh, R., Fitzgerald, R., Bond, T., & Grimbeek, P. (2013). Development of the TTF TPACK Survey Instrument. *Australian Educational Computing 27*(3), 26-35.
Koehler, M., Mishra, P., & Yahya, K. (2007). Tracing the development of teacher knowledge in a design seminar: Integrating content, pedagogy, & technology *Computers and Education 49*(3), 740-762.
Lei, J., Conway, P. F., & Zhao, Y. (2008). *The digital pencil: One to one computing for children*. London: Routledge, Taylor & Francis Group.
Madden, M., & Fox, S. (2006). Riding the waves of “Web 2.0.”. *Backgrounder, Pew Internet and American Life Project, 23*(1).
Moesrch, C. (2010). Levels of Teaching Innovation Framework. *Learning and Leading with Technology, 37*(5), 20-23.
NCATE. (1997). Transforming teacher education through clinical practice: A national strategy to prepare effective teachers. Retrieved from http://www.ncate.org/
Persad-Bissessar, K. (2010). *High level Plenary Meeting of the Sixty-fifth session of the United Nations General Assembly on the Millennium Development Goals*. New York: Permanent Mission of Trinidad and Tobago to the United Nations.
Severin, E., & Capota, C. (2011). One-to-One Laptop Programs in Latin America and the Caribbean: Panorama and Perspectives. *Education Division (SCL/EDU)*. Retrieved from *http://www.iadb.org.*
Tersptra, M. J. (2010). Developing Technological Pedagogical Content Knowledge: Preservice Teachers’ Perceptions of How They Learn to Use Educational Technology in their Teaching. (Doctor of philosophy), Michigan State University, USA.
Vygotsky, L. S. (1987). *Thinking and speech. In R. W. Reiber and A. S. Carton (Eds.), (Trans., N. Minick), The collected works of L. S. Vygotsky. Vol. 1: Problems of General Psychology.* New York, NY Plenum.

On Evaluation of Computational Thinking of Software Engineering Novice Students

Vladimiras Dolgopolovas, *vladimiras.dolgopolovas@mii.vu.lt*
Vilnius University, Institute of Mathematics and Informatics, Lithuania

Tatjana Jevsikova, *tatjana.jevsikova@mii.vu.lt*
Vilnius University, Institute of Mathematics and Informatics, Lithuania

Loreta Savulionienė, *l.savulioniene@eif.viko.lt*
University of Applied Sciences, Faculty of Electronics and Informatics, Lithuania

Valentina Dagienė, *valentina.dagiene@mii.vu.lt*
Vilnius University, Institute of Mathematics and Informatics, Lithuania

Abstract

During past years computational thinking (CT) is being actively promoted through the K-12 curriculum, higher education, contests and other activities. CT skills are important for further students' educational and professional career. The paper presents a study conducted among the first year software engineering students, learning a structured programming (SP) course. As an instrument to measure CT skills, the test consisted of preselected Bebras contest tasks was developed and validated. The correlation between the test resuls and the SP course results was investigated.

Keywords

Computational thinking, Bebras contest, computer science concepts, computer science education, novice programming students, novice software engineering students.

INTRODUCTION

During past years, CT has being actively promoted through the K12 curriculum as a part of computer science (CS) subject or in an integrated way, paying more and more attention to programming and fundamental CS concepts (e.g. Royal Society, 2012). As J. Wing defined, "computational thinking represents a universally applicable attitude and skill set everyone, not just computer scientists, would be eager to learn and use" (Wing, 2006, p. 33). So, it should be educated not only through school curriculum, higher education, and through the initiatives which are accessible for a wide area of participants as well.

Since 2004, Bebras contest, originated in Lithuania (Dagiene, 2006), has become an international initiative whose goal is to promote CS and CT especially among teachers and pupils of all ages, but also to the public at large by extent. The big challenge of Bebras is to organize easily accessible and highly motivating online contests in many countries (Bebras.org, 2015). The Bebras contests consist of a set of tasks in a form of short questions (problems), or quiz. Each Bebras task can both demonstrate an aspect of CS and test the aspects of CT of the participant. These tasks can be solved without prior knowledge about CS, but are clearly related to fundamental CS concepts.

The tasks are developed by an international community of CS experts. The requirements for quality of Bebras tasks have been already discussed (Dagiene, Futchek, 2008; Vanicek, 2014). The operational definition of CT (ISTE, 2015) suggests that the CT is a problem-solving process that includes (but is not limited to) the following characteristics:

- Formulating problems in a way that enables us to use a computer and other tools to help solve them.
- Logically organizing and analyzing data.
- Representing data through abstractions such as models and simulations.
- Automating solutions through algorithmic thinking (a series of ordered steps).
- Identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources.
- Generalizing and transferring this problem solving process to a wide variety of problems.

Bebras tasks involve students into these operations. As we will see in the next sections of this paper, each Bebras task, we have analysed, has at least some components of CT: Abstraction (AB), Decomposition (DE), Algorithmic thinking (AL), Evaluation (EV), and Generalisation (GE). To solve those tasks, students are required to think in and about computer science, discrete structures, computation, data processing, data visualisation, but they also must use algorithmic as well as programming concepts (Dagiene, Stupuriene, 2014).

Prior related research has studied the differences in task solving results of boys and girls (Pėlikis, Dagiene, 2014) and classification of Bebras tasks using Bloom taxonomy (Dagiene, Stupuriene, 2014). Within the scope of this research, we would like to study the CS concepts behind the CT Bebras tasks and the ability of novice software engineering students to use these concepts in a real-life context, which is modelled by the task.

The research presented in this paper is based on authors' practical experience in teaching SP for the first semester software engineering students and authors' solid practical and theoretical experience in developing contest related educational methods (Dagiene, 2006; Dagiene, Futchek, 2008; Dolgopolovas et al., 2014). The SP course is oriented to problem solving and is based mainly on Python programming language. Python as a programming language is widely promoted as an alternative to other programming languages and as the first language to study programming. The reason for this is that Python is positioned as a tool for rapid application development and at the same time as an educational tool, enabling problem solving programming courses to be developed (Dierbach, 2014; Wen et al., 2014). During such a course, novice software engineering students develop their knowledge of CS basics and main concepts, and study the language syntax in parallel.

We could formulate the next research hypothesis: "a well developed problem solving focused SP course develops computational thinking skills as well". A relevant measurement tool is needed, taking into account that CT is a latent trait and it could be implemented during the problem solving process. To test the hypothesis, the test, which is independent of programming language, was developed. The aim of the test is to evaluate students' CT skills and to compare test results with SP course results. The test process had several steps and used a combination of quantitative and qualitative testing approaches. As the first step of the test process, a homogeneous dichotomous test, which uses Bebras contest tasks as the test questions, was developed. Solving the presented tasks, students should employ CT skills, which we intent to measure as a latent trait. The next steps of the test process include: questions requiring students to identify such CS concepts, which were implicitly presented in the tasks of the first test step; solving CS problems by finding coding solutions to the presented tasks. The last two steps of the test process still need to be evaluated and are positioned as further work.

The main **Research Questions** of this paper are:

- How computational thinking skills of novice software engineering students can be evaluated in a programming language independent way?

- What is the relation between novice software engineering student's computational thinking skills and programming course results?

Why this questions are of primary importance for us? We consider computational thinking skills as very important for further students' educational and professional career. Such skills form a basis for students' better understanding of further computer science knowledge and could sufficiently reduce failure and drop out students' rate. In order to answer the above research questions, we conducted a case study where the novice software engineering students who studied SP had to solve the test of preselected Bebras tasks.

RESEARCH METHODOLOGY

The preparation phase of the research consisted of several steps. First, the appropriate Bebras tasks were selected. The main selection criteria were:

- Computational Thinking Concepts. Each task should have at least one well-expresed CT concept.
- Focus on algorithmical thinking. The test is aimed to software engineering students, therefore all of the tasks are realated to data structures, algorithms and their methods.
- Difficulty level. As Bebras contest is mainly addressed to school pupils, and we are going to use them for the first year higher education students studying software engineering, we selected the tasks that were considered as difficult according to international expert evaluations and the contest results.

The second phase included the analysis of the selected Bebras tasks, marking the main computer science concepts, used in them. During the third step, an online quiz for students has been designed. The quiz included 10 selected Bebras tasks. We used quantitative research methods to analyse the data. Test validity has been studied using Item Response Theory.

PARTICIPANTS

65 first year (first semester) software engineering students, studying the SP course, took part in the experiment. The experiment was held with four groups of students, consisting of correspondingly 16, 10, 20 and 19 students. The more detailed structure, including percentage of male and female students, scores in Maths and Computer Science maturity exams, is presented in Table 1.

Total number	Male	Female	Maths maturity exam scores		CS maturity exam scores	
			Score	Students	Score	Students
65	95%	5%	80-100	6.2%	80-100	10.8%
			60-80	12.3%	60-80	16.9%
			40-60	32.3%	40-60	27.7%
			20-40	36.9%	20-40	33.8%
			<20	9.2%	<20	10.8%

Table 1: Basic characteristics of the participants of the experiment

RESULTS

The task for the participating in the study students was to choose the correct answer for the presented Bebras task. Ten tasks were preselected and prepared for the test. All these tasks are CT oriented and include a set of CS concepts within them.

Solving CT Bebras tasks, the average result is 54.2% of correct answers. The detailed structure of the correct answers per task is presented in Table 2.

No.	Computational thinking task name	Correct answers
1	Beaverrail	38.5 %
2	Beaver's log factory	33.8 %
3	Bob's Best strategy	56.9 %
4	0X	41.5 %
5	Water supply	63.1 %
6	Collecting candies	73.8 %
7	Beaver in his canoe	86.2 %
8	Constructive Beaver	21.5 %
9	Sorting the Sticks	90.8 %
10	Bebras-city streets	35.4 %

Table 2: The general results of solving computational thinking tasks

Test structure

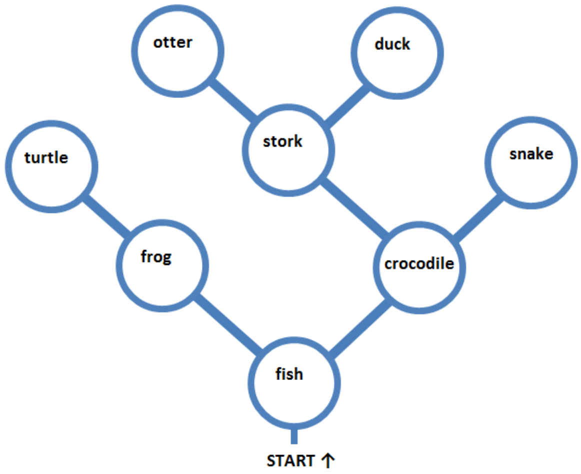
The main concepts, "encoded" inside the gamified tasks, were asked to be identified. These include the main concepts of data structures, algorithms, methods, logical operations, and control structures (Table 3). The concept name is followed by the number, that corresponds to the CT task number, already referenced above in Table 2.

Data structures	Algorithms	Methods	Logics	Control structures
Binary tree (3)	Algorithm (1)	Dynamic programming (6)	Binary logical operation (5)	Loop (1, 4)
Graph (6)	Maximum element search (9)	Binary tree modelling (7)	Disjunction (5)	Function (8)
Logical data (4)	Modified sorting algorithm (9)	Tree traversal (3)	Equivalence (5)	Conditional sentence (1)
Array (6)	Depth-first search (3, 7)	Operation abstraction (8)	Inversion (5)	Conditional loop (9)
List structure (4)	Breadth-first search (7)	Optimization (10)	Conjunction (5)	
Directed graph (2)	Greedy algorithm (10)	Parameterization (8)		
Weighted graph (3)		Automatization (7)		

Table 3: The main concepts, within the computational thinking tasks

The format of this paper restricts us from presenting more extended examples of the test tasks. Here we present an example of one of 10 tasks selected for the study (Table 4).

<table>
<tr>
<td>Beaver in his canoe
Beaver paddles in his canoe on a river that has a number of little lakes.
Beaver likes all lakes of the river and has thought of an algorithm to make sure that he reaches every lake. He knows that at each lake there is a maximum of two rivers that he has not yet seen. If beaver arrives at a lake he decides which river to take with the following rules:
If there are two rivers he has not yet seen, he takes the river on his left hand side. If there is one river which beaver has not yet seen, beaver takes this river.
If beaver has seen all the rivers from a little lake, he paddles his canoe one lake back towards the previous lake.</td>
<td>



</td>
</tr>
<tr>
<td colspan="2">Beaver stops his day of canoeing if he has seen everything and has come back to the start point.
In each little lake beaver sees a different animal. Beaver writes down the animal name when he sees an animal for the first time.
In which order will beaver write down the animals?
Answer
a. fish, frog, crocodile, turtle, stork, snake, otter, duck
b. fish, crocodile, snake, stork, duck, otter, frog, turtle
c. fish, frog, turtle, crocodile, stork, otter, duck, snake
d. fish, frog, turtle</td>
</tr>
</table>

Table 4: Task 7, “Beaver in his canoe”[2]. The correct answer is written in bold

We can identify the main components of CT in this example of the task, as it was already mentioned in the Introduction section of this paper:

- Abstraction (AB): from real objects (lakes, rivers) to abstract objects like binary tree.
- Decomposition (DE): checking the rule, applying to the parts of the tree.
- Algorithmical thinking (AL): the task itself provides an algorithm that should be understood and applied. This task, however, is not an example where students should develop their own algorithm to select correct answer.
- Evaluation (EV): evaluation of all correct decision, evaluation of wrong answer set.
- Generalisation (GE): applying algorithm rule to the whole tree, analysing the result in general.

The main CS concepts included in this task are *binary tree modelling*, *depth-first search*, and *automatization*.

Test validity

Evaluating the test statistics we implement several basic considerations. First, we consider the test as a whole as a homogenous dichotomous test for evaluating students’ implicit ability of computational thinking in the context of the process of solving gamified tasks which implicitly involve cognitive procedures of CT and problem solving. Next, we should make a note on the design and implementation of

[2] The task is authored by Judith Helgers

the test. We use dichotomous scale for measurement of the latent trait, scoring positively for the right solution.

We implement the Rash model for the evaluation and use R environment and eRm package for testing the model [3] [4] (Mair, 2009; De Battisti, 2012). The result of the estimation of the dichotomous model is presented in Figure 1 and Figure 2. The Wald test plot is presented in Figure 3.

```
Results of RM estimation:

Call:  RM(X = ads)

Conditional log-likelihood: -254.7285
Number of iterations: 20
Number of parameters: 9

Item (Category) Difficulty Parameters (eta):
                Q2          Q3        Q4         Q5         Q6         Q7
Estimate 0.9495589 -0.03514585 0.6071324 -0.2976646 -0.8043209 -1.5912333
Std.Err  0.2584529  0.24596351 0.2486839  0.2507688  0.2707939  0.3351865
                Q8         Q9       Q10
Estimate 1.5983880 -2.046030 0.8785908
Std.Err  0.2936692  0.394893 0.2559417

> LRtest(res)

Andersen LR-test:
LR-value: 9.38
Chi-square df: 9
p-value:  0.403

> MLoef(res)

Martin-Loef-Test (split criterion: median)
LR-value: 18.244
Chi-square df: 24
p-value: 0.791
```

```
> Waldtest(res)

Wald test on item level (z-values):

          z-statistic p-value
beta Q1        -1.496   0.135
beta Q2         1.373   0.170
beta Q3         1.572   0.116
beta Q4        -0.884   0.377
beta Q5        -0.780   0.436
beta Q6         0.574   0.566
beta Q7        -0.862   0.389
beta Q8         0.256   0.798
beta Q9         0.796   0.426
beta Q10        0.044   0.965
```

```
Conditional log-likelihood: -254.7285
Number of iterations: 20
Number of parameters: 9

Item (Category) Difficulty Parameters (eta): with 0.95 CI:
    Estimate Std. Error lower CI upper CI
Q2     0.950      0.258    0.443    1.456
Q3    -0.035      0.246   -0.517    0.447
Q4     0.607      0.249    0.120    1.095
Q5    -0.298      0.251   -0.789    0.194
Q6    -0.804      0.271   -1.335   -0.274
Q7    -1.591      0.335   -2.248   -0.934
Q8     1.598      0.294    1.023    2.174
Q9    -2.046      0.395   -2.820   -1.272
Q10    0.879      0.256    0.377    1.380

Item Easiness Parameters (beta) with 0.95 CI:
         Estimate Std. Error lower CI upper CI
beta Q1    -0.741      0.252   -1.234   -0.247
beta Q2    -0.950      0.258   -1.456   -0.443
beta Q3     0.035      0.246   -0.447    0.517
beta Q4    -0.607      0.249   -1.095   -0.120
beta Q5     0.298      0.251   -0.194    0.789
beta Q6     0.804      0.271    0.274    1.335
beta Q7     1.591      0.335    0.934    2.248
beta Q8    -1.598      0.294   -2.174   -1.023
beta Q9     2.046      0.395    1.272    2.820
beta Q10   -0.879      0.256   -1.380   -0.377
```

Figure 1.Dichotomous model. Results of the Rash model estimation, Andersen's Likelihood Ration and Martin-Loef tests. Summary of the Rash model estimation

Figure 1 shows the Rash model estimation for the test items: items easiness parameter, items' standard errors, and items' 95% confidence intervals. The 95% confidence interval for each eta parameter presents if the parameter is significantly different from zero.

We check the fit of Rash model according to Andersen's likelihood ratio test. The mean of raw scores was chosen as the partitioning criterion. The p-value shows that the likelihood ratio test is non-significant and, therefore, the Rash model holds the data.

Martin-Loef test evaluates unidimentionality the two sets of items. As well as Andersen's test, Martin-Loef test confirm the Rash model data. Wald test shows non significant difference for all the test items as is seen from the plot (Figure 3). Items characteristics curves show close to the uniform distribution of the test items (Figure 2).

---

[3] R Development Core Team (2008). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0

[4] R-forge. eRm Project, http://r-forge.r-project.org/projects/erm/

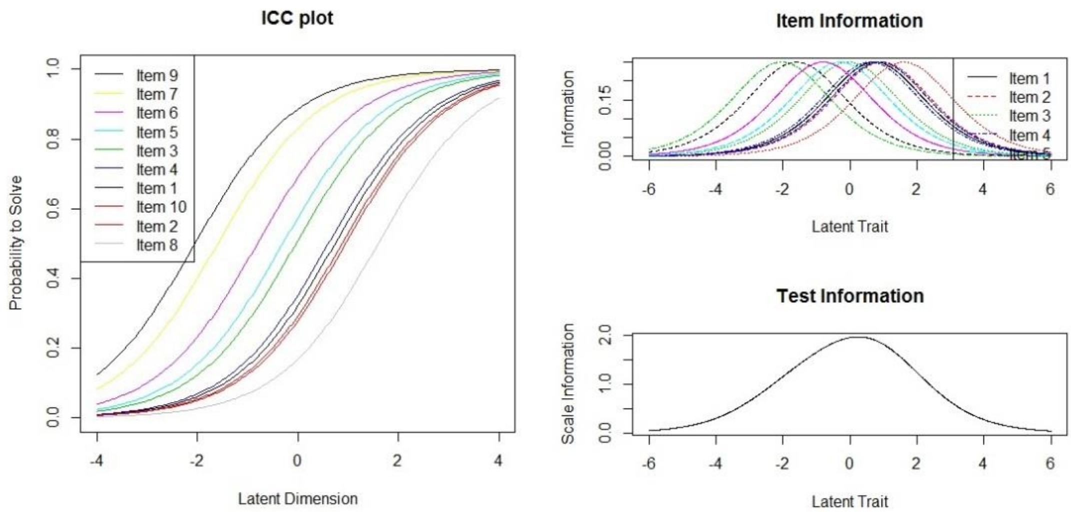


Figure 2. Dichotomous model. Items characteristics curves. Items and test information curves

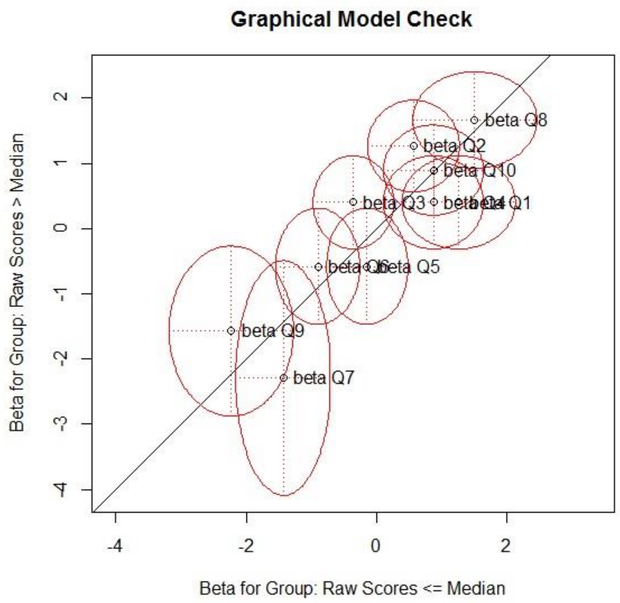


Figure 3. Dichotomous model. The Wald test plot

The test statistics confirms the general validity of the test.

Correlation results

In order to answer our second research question we measure and compare the measured latent abilities with structured programming course results. The positive correlation would indicate the success of the SP course. The test results and the SP course examination results correlation plot is presented in Figure 4.

The plot shows no correlations.

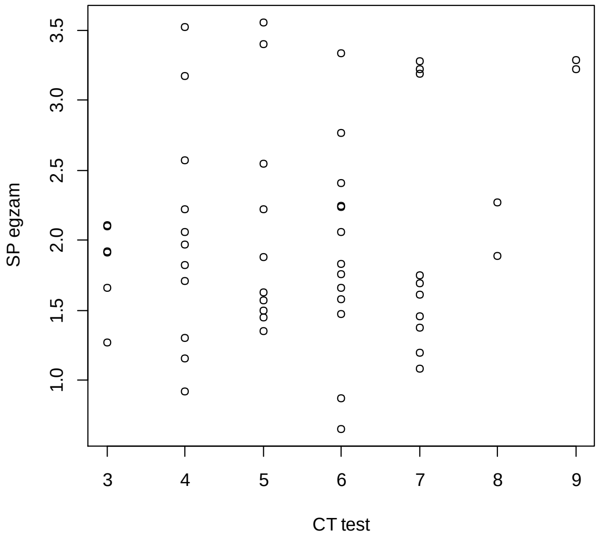


Figure 4. CT test vs SP course exam plot

One of the reasons of no correlation between the test results and course results could be the fact that the course is oriented mostly to study programming language sintax. The course structure and didactical approach should be re-developed to address more problem solving skills.

CONCLUSION

- CT tasks used in this study are designed for a secondary school students. Unless the task were designed to be solved by a secondary school students, it was quite surprising that there were only 54.2% of correct answers while solved by the first year software engineering students. Possibly, the reason for that is not sufficient students' preparation on a school level, not addressing problem solving skills adequately enough.
- The statistical evaluation of the test used in this study, has shown the possible validity of the test as an instrument to evaluate computational thinking.
- In spite of our prediction, the test did not present any correlations between CT skills and SP course. This means that the course needs to be improved. At this point of our research we cannot either confirm or reject the reseach hypothesis we rose in the beginning of this paper. The course improvement strategy could be based on the analysis of the structure of the test tasks, studying underlying CS concepts and shifting the structure of the course and its exam to the test shown the most difficult to solve problems.

DISCUSSION AND FUTURE WORK

A case study, presented in this paper, was aimed to investigate how the problem solving oriented SP course enhances CT skills of novice software engineering students. Such an example of students' activity (game-like CS task with the concepts "encoded" into it), could be used not only to test students' CT abilities, but may be as well useful as a learning activity for CS/software engineering students. The activity by itself may be a high level motivation to learn new CS concepts.

The case study has given some controversial results. This case study may be considered as a first step in a series of the research activities. The next steps of the research would be: to analyze how novice software engineering students identify CS concepts which are implicitly presented inside CT tasks, create game-like computer programs, based on CT tasks, and use the main CS concepts in their programming activities. This would possibly bring more light on the reasons of the problems and limitations identified in this case study.

The results have also pointed out the should be improved problems of the SP course itself. Using programming language independent test, presented in this paper, the influence of different didactical approaches on the improvement of CT skills could be studied and compared.

REFERENCES

Dagienė, V. (2006). Information Technology Contests--Introduction to Computer Science in an Attractive Way. *Informatics in Education-An International Journal*, 5(1), 37-46.

Dagienė, V., & Futschek, G. (2008). Bebras international contest on informatics and computer literacy: Criteria for good tasks. In *Informatics Education-Supporting Computational Thinking* (pp. 19-30). Springer Berlin Heidelberg.

Dagienė, V., Stupurienė, G. (2014) Informatics Education based on Solving Attractive Tasks through a Contest. *Conference KEYCIT 2014*, 51-62.

De Battisti, F., Nicolini, G., & Salini, S. (2012). The Rasch Model. *Modern Analysis of Customer Surveys: with applications using R*, 259-281.

Dierbach, C. (2014). Python as a first programming language. *Journal of Computing Sciences in Colleges*, 29(6), 153-154.

Dierbach, C. (2012). Introduction to Computer Science Using Python: A Computational Problem-solving Focus. *Wiley Publishing*.

Dolgopolovas, V., Dagienė, V., Minkevičius S., and Sakalauskas L. (2014), Python for Scientific Computing Education: Modeling of Queueing Systems, *Scientific Programming*, IOS press, 22(1), 37-51.

ISTE (The International Society for Technology in Education) (2015) Retrieved February 27, 2015, http://www.iste.org/.

Mair, P., Hatzinger, R., and Maier, J.M. (2009). Extended Rasch Modeling: The R Package eRm. PDF-Dateianhang zum Programmpaket eRm.

Pėlikis, E., Dagienė, V. (2014). Informatikos uždavinių sprendimo tyrimas: skirtumai tarp vaikinų ir merginų*. Studentų mokslinė praktika 2014, konferencijos pranešimų santraukos*, I dalis, 89–91. [in Lithuanian]

Royal Society. Shut down or restart: The way forward for computing in UK schools. Retrieved February 27, 2015, http://royalsociety.org, 2012. *Sciences in Colleges*. 29(6), 153–154.

Vaníček, J. (2014). Bebras Informatics Contest: Criteria for Good Tasks Revised. In Informatics in Schools. Teaching and Learning Perspectives , Springer International Publishing, 17-28.

Wen, X., Yang, Z., Wang, Z., and Zhao, M. (2014). Exploration on Cultivating Students' Abilities Based on Python Teaching Practice. *In 2nd International Conference on Teaching and Computational Science*. Atlantis Press.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.

Vladimiras Dolgopolovas received a degree in civil engineering in 1986 and a second degree in mathematics and computer science teaching from Vilnius University. He has participated in research projects on embedded systems, software defined radio and industrial networks. He is a lecturer in Vilnius University of Applied Science and a member of research staff at Institute of Mathematics and Informatics of Vilnius University. His research interests include numerical analysis, operational research, and applications to teaching and learning.

Tatjana Jevsikova is a researcher at the Vilnius University Institute of Mathematics and Informatics as well as an associate professor at the Vilnius University Mathematics and Informatics faculty. She received her PhD in computer science. Her main research interests include informatics education, e-learning, teacher training, software localization, cultural aspects of human-computer interaction. She is an author (or a co-author) of more than 20 scientific papers, several methodological books and dictionaries of computer science terms.

Loreta Savulionienė got bachelor's degree in mathematics in Vilnius University in 1996, master's degree in mathematics in Vilnius University in 1998, and in 2013 a PHD degree in informatics. She is a vice dean of the Faculty of Electronics and Informatics and a lecturer in Vilnius University of Applied Sciences. In 2013 she was awarded a grant for best Lithuanian programming lecturers. Loreta's research interests include computer science, programming education reseach, educational planning and organization of an educational process.

Valentina Dagienė is a professor at the Vilnius University Institute of Mathematics and Informatics and head of the Department of Informatics Methodology. She has published over 150 research papers and the same number of methodological works, has written more than 60 textbooks in the field of informatics and ICT for secondary schools. She has been working in various expert groups and work groups, guiding the activity of a Young Programmer's School, for many years, organizing the Olympiads in Informatics among students.

Computer use in class: the significance of educational framework conditions, attitudes and background characteristics of secondary school teachers on a level of international comparison

Kerstin Drossel, kdrossel@mail.upb.de
Paderborn University, Germany

Birgit Eickelmann, birgit.eickelmann@upb.de
Paderborn University, Germany

Julia Gerick, gerick@ifs.tu-dortmund.de
TU Dortmund University, Germany

Abstract

This paper presents findings based on the IEA-study ICILS 2013 (International Computer and Information Literacy Study, 2010-2014) which investigates the computer and information literacy (CIL) of secondary school students and the contexts in which students develop CIL in 21 countries. In this case, the teaching-related use of digital media by teachers is located as a process on the level of the school and of the class. The use of digital media in schools is associated with the goal of supporting learning processes and improving the quality of education (Voogt & Knezek, 2008). By regression analyses, this contribution focuses on relevant factors for the teachers' use of computers in class in six selected countries. Therefore, the analysis observes educational framework conditions, attitudes and background characteristics of secondary school teachers as an indicator of school quality in the digital age.

Keywords

Computer use, educational teaching and learning conditions, international comparison

INTRODUCTION: THE RELEVANCE OF THE USE OF DIGITAL MEDIA

For more than 40 years, the use of digital media in educational contexts of teaching and learning has been investigated internationally and its scope has been a recurring theme in the context of educational practice (Voogt & Knezek, 2008). Due to the increasing use of information and communication technologies (ICT) and the society's transition towards an information or knowledge society new challenges for schools have emerged (Anderson, 2008). In this context, the use of digital media is associated with the goal of supporting learning processes and improving the quality of education (Voogt & Knezek, 2008). The use of digital media, for instance, can help students to process information or can provide more effective teaching techniques for the teachers (Spiezia, 2010). Moreover, the competent use of digital media is presumed to be a key competence in the 21st century (Fraillon, Schulz & Ainley, 2013; Griffin, McGaw & Care, 2012; Voogt, Erstad, Dede & Mishra, 2013). In the context of an increased digitalization with respect to practically all spheres of life, prominence has therefore been given to a reconsideration of schooling in favor of incorporating information and communication technologies (ICT) into the educational sector (Davis, Eickelmann & Zaka, 2013) in order to provide an enhanced educational career to coming generations of students that will ultimately need to adapt to what is commonly considered the digital age. The promotion of students' skills and their preparation for life through computer-assisted teaching and learning in schools, however, still presents an ambitious task and an extensive challenge for teachers (Voogt & Knezek, 2008). Findings suggest that the perception of ICT as an advantage throughout the school constitutes a prerequisite for its successful introduction (Eickelmann, 2011), most notably with respect to the teachers who take on the role of a mediator when it comes to ICT innovations (Davis et al., 2013). Owston (2003) equally identifies the perceived value of an innovation by teachers as an essential factor in determining the success of ICT incorporation in the educational sector.

For the purpose of analyzing students' computer and information literacy in relation to framework conditions, the IEA carried out the International Computer and Information Literacy Study 2013 (ICILS 2013), which will constitute the data basis of this paper (Fraillon, Ainley, Schulz, Friedman & Gebhardt, 2014). With regression analyses, this contribution compares the use of digital media by teachers across selected countries and investigates the supporting and hindering factors for the use of digital media in school.

THEORETICAL FRAMEWORK

Figure 1 illustrates the basic classification of antecedent and process-related contextual factors and their relationship with computer and information literacy (CIL) as outcome. The model also represents the multilevel structure that influences studenta' learning and includes the following levels: the individual level, the home environment, the school/classroom and the wider community. Along the lines of the ICILS 2013 theoretical framework (Fraillon et al., 2013), the teaching-related use of digital media by secondary school teachers – as examined by the present paper – is located as a process on the level of the school and of the class. This level encompasses all school-related factors. These factors directly influence the students' CIL learning progress and are constrained by the antecedent factors.

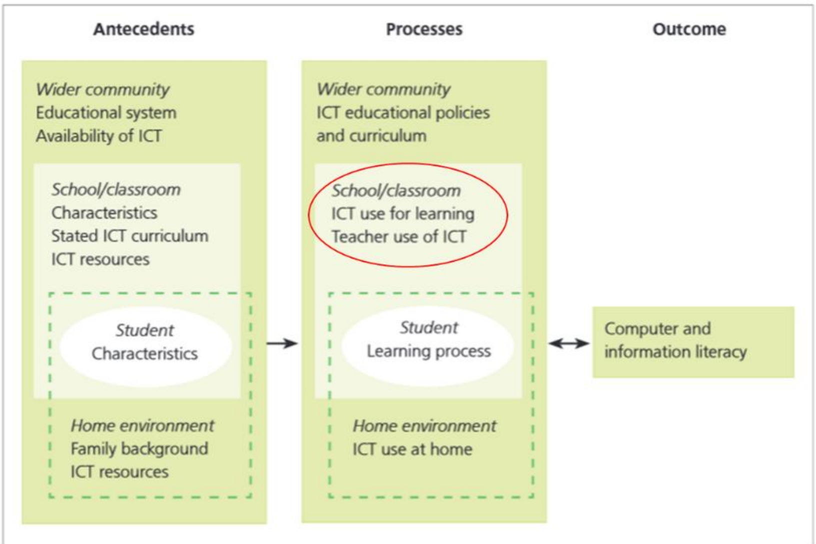


Figure 1: Context for CIL learning and learning outcomes
(Fraillon et al., 2013, p. 26)

FINDINGS FROM PREVIOUS STUDIES: THE USE OF DIGITAL MEDIA AROUND THE WORLD

Numerous studies concerning the primary and secondary levels of education have proven that by international comparison, countries show diverse frequencies of digital media use in class (see i.a. OECD, 2011; Mullis, Martin, Foy & Arora, 2012). In addition, computer-based activities, like looking up ideas and information or practice skills and procedures, differ between subjects, grades and countries (Mullis et al., 2012).

The available studies indicate that both educational teaching and learning conditions influence the dealing with new technologies in the context of teaching and learning processes. One relevant obstacle on the school level for teachers using ICT is a lack of support, especially pedagogical support (Law, Pelgrum & Plomp, 2008). Another relevant factor regarding the use of ICT for learning purposes is the IT infrastructure in secondary schools that has been identified as crucial (Albirini, 2006; Petko, 2012). While most of the studies point out that a modern and adequate ICT infrastructure at school is needed to incorporate computers and new technologies in instructional settings (European Commission, 2013), some countries show a low availability of computers for instruction (Mullis et al., 2012).

Factors located on the individual teacher level, such as age and gender, as well as the perceived benefits of teaching-related digital media use and the self-efficacy of using digital media in class are considered relevant throughout previous research (see Legris, Ingham & Collerette, 2003; Venkatesh, Morris, Davis & Davis, 2003; Spiezia, 2010). At that, teachers' attitude towards the use of ICT for purposes of instruction seems to play a vital role in the success of implementing ICT use at school (see Acker, Buuren, Kreijns & Vermeulen, 2011; Albirini, 2006; Govender & Govender, 2009). While Govender & Govender (2009) equally reveal that a lack of compatibility with the curriculum and insufficient conditions with respect to time constitute obstacles perceived by secondary school teachers regarding the implementation of ICT at school, Albirini (2006) points to the significance of cultural perceptions for the teachers' attitudes, particularly in developing countries with a low density of media availability. Following Davis et al.'s (2013) finding that teachers take on a key role in the implementation of new technologies in the classroom, it seems plausible that obstacles perceived by teachers need to be overcome before ICT use can be improved – that is: the improvement of teachers' attitudes needs to precede the provision of IT equipment (see Albirini, 2006) and as such constitutes a factor of high significance in terms of examining influencing factors with respect to ICT use at secondary schools.

While there is an extensive body of research documenting the importance of the various factors for using ICT at secondary schools however, the previous studies often focus on a selection of factors and in many cases examine these in smaller samples only.

THE USE OF DIGITAL MEDIA IN ICILS 2013

The International Computer and Information Literacy Study 2013 also analyzed the use of digital media in secondary schools but in addition, focused on students' acquisition of computer and information literacy and the ICT learning environment in schools. According to this, it is – for the first time – possible to investigate the use of digital media in secondary schools in relation to a multitude of relevant background variables with a representative sample in many countries (Fraillon et al., 2014). Therefore, more than 60,000 students in grade eight in 21 participating educational systems around the world and their teachers (N= 35.000) were included. In addition, the study collected contextual data from school ICT-coordinators, principals and the national contexts.

With respect to teaching practices, the international report on ICILS 2013 revealed that the use of ICT in the educational context is extensive with three out of five secondary school teachers using ICT at least once per week (Fraillon et al., 2014). The national percentages ranged from a high of 90 percent in Australia to a low of 41 percent in Croatia and Poland. A majority of teachers possesses an experience of two years or more in using ICT (84%), however, not predominantly due to teaching-related computer use (ibid.).

Teachers' experience in using computers for teaching purposes was, on average, moderately strongly associated with the frequency of computer use (Fraillon et al., 2014). The frequent users of computers also had stronger positive views and less negative views about the effects of ICT. In comparison to the infrequent users, they reported better ICT resources and a stronger collaboration with other teachers in using ICT (ibid.). Although teachers generally considered ICT to be beneficial, only two thirds seemed to link ICT to an improved academic performance (ibid.). The teachers' confidence in using ICT, along with their perceived value of ICT, was shown to have a significant impact on its frequency of use and on the emphasis placed on the development of students' CIL (ibid.). With the above-mentioned factors being highly heterogeneous across the various participating countries, it can be concluded that teachers play a key role in the implementation of ICT at school, which is in alignment with Davis et al.'s (2013) finding of teachers as a keystone species in bringing technologies to schools and ultimately to their students, recognizing "the co-evolution of digital technologies and education within the local and global knowledge society of the 21st century" (Davis et al., 2013, p. 439). Fraillon et al. (2014: 255) therefore conclude that "if schools are to develop students' CIL to the greatest possible extent, then teacher expertise in ICT use needs to be augmented, and ICT use needs to be supported by collaborative environments […]".

RESEARCH QUESTION: RELEVANT FACTORS FOR THE USE OF DIGITAL MEDIA IN ICILS 2013

The present paper aims to extensively investigate the effect of perceived educational teaching and learning conditions as well as individual teacher factors on the use of digital media at school in a context of international comparison. On the one hand, the focus will be placed on countries with a high frequency of computer use at school; on the other hand, countries with a low frequency will equally be examined. In addition, the term of a tradition of using ICT in schools and the students' achievement in ICT literacy in ICILS 2013 were involved in the selection of the countries. This paper aims to address the following research question: What effects do IT resources at school and teacher factors have on secondary school teachers' use of computers at school in the selected countries?

THE STUDY: ANALYSIS OF FACTORS FOR SECONDARY SCHOOL TEACHERS' COMPUTER USE AT SCHOOL

Data collection and instruments

In order to answer the research question, data is needed that can provide information about the IT resources in school and individual teacher factors in different countries. This data is available from the IEA-Study ICILS 2013 (Fraillon et al., 2014; Fraillon et al., 2013).

With ICILS 2013, collected data about secondary school teachers' use of computers as well as their attitudes towards the use of computers, their background and the frame conditions are now available. The population for the ICILS 2013 teacher survey was defined as all secondary school teachers teaching regular school subjects to the students in grade eight at each sampled school. The teacher questionnaire comprises questions about the teaching practices with ICT, the attitudes towards the use of ICT in teaching, the participation in professional learning activities related to the use of ICT in teaching and the perceived IT resources at school.

On the basis of the ICILS 2013 teacher data, it shall be investigated for the selected participating countries (1) which educational teaching and learning conditions, (2) which attitude-related traits of the teachers and (3) which further characteristics of the teachers have an impact on the use of computers in class as a prerequisite for school quality and a facilitator of student learning progress in the field of CIL.

In order to analyze educational teaching and learning conditions, we focus on three items, which determine the restraints among the IT resources at school, such as a limited internet access or problems with IT-resources. The teachers′ attitudes are represented with four items and one international index that documents the positive view on using ICT in teaching and learning. The scale was constructed by using the IRT Rasch partial credit model (Rasch, 1960). The scale’s reliability (Cronbach’s Alpha) is .83 on average across the countries. The other items that are used to operationalize the teachers’ attitudes implicate aspects of collaboration between teachers in using ICT, an aspect of ICT self-efficacy and the teachers’ participation in professional development with regard to courses on integrating ICT into teaching and learning. Furthermore, we include three background-variables: the duration of teachers’ experience in using ICT for teaching purposes, their age and their gender.

The results of the selected countries of the study will be discussed against the background of contextual factors on the level of the educational systems, which are available through the so-called *National Context Survey*.

SAMPLE

For the secondary analysis presented in this contribution, IEA-ICILS 2013 teacher data from the background questionnaires is used. A selection of countries is used to detect factors that affect the teachers’ use of computers at school. In total, six countries were selected: the Netherlands, Denmark and Australia represent countries with a high frequency of computer use in secondary schools, a high student achievement in computer and information literacy (between 537 and 542 scale points) and a long tradition of implementing ICT in secondary education. Poland and Germany were selected for a very low use of computers. The students in Poland showed with 537 scale points a very good score, whereas the results for students in Germany conformed with the European ICILS 2013 average (523 and 525 scale points respectively). Finally, Lithuania represents a country with a high level of computer use and with 494 scale points a comparatively low achievement (Fraillon et al., 2014).

METHODS

The question of which indicators can be deemed important for the prediction of computer use by teachers in class will be investigated on the basis of regression analyses. Therefore, the IDB Analyzer (Ver. 3.1) was used, which is an application developed by the IEA DPC to facilitate the analysis of data from IEA's large-scale assessments. In doing so, we take care of the complex sample structure of the ICILS 2013 database.

FINDINGS

The regression model reveals different results for the selected countries of ICILS 2013 (Table 1). Compared to the factors describing the teachers’ attitudes, most of the three indicators concerning the restraints among the IT resources at school in the selected countries do not show a significant effect. The limited connectivity to the internet merely seems to be significant for computer use in Australia. The factor that ICT is not considered a priority for use in teaching is only a significant predictor in Germany and Lithuania. In three countries (Denmark, Poland and Lithuania), the restraint that the school does not have sufficient ICT equipment (e.g. computers) is a relevant factor for the use of computers in instruction.

Concerning the secondary school teachers' attitudes towards ICT in learning processes, merely the *self-assessed competencies* regarding the dealing with new technologies can be deemed significant in all considered countries. If teachers are confident of being able to prepare lessons that involve the use of ICT, this factor can be considered significant across all selected countries. In addition, the *professional development* with regard to courses on integrating ICT presents a significant factor in all selected countries except Denmark. In four out of six countries, the *positive view on using ICT* as well as the *collaboration with colleagues in developing ICT use* plays an important role in the teachers' use of computers in class. In the Netherlands and Lithuania, however, the two factors do not point to a relation with the teachers' use of computers for instructional purposes, which emphasizes that – when it comes to the teachers' attitudes – there are no detectable patterns with regard to the factors' significance considering the intensity of use in the respective countries, the students' CIL competencies and the tradition of implementing ICT literacy. Only for three countries (Denmark, Australia & Germany) a relative importance regarding the collaboration with colleagues to develop ICT-based lessons can be detected.

In consideration of the *background variables*, it becomes apparent that in five out of six countries, the teachers' experience in using ICT for teaching purposes constitutes a significant factor for the explanation of computer use. Only in Denmark, no significant effect could be observed. The respective teacher's gender equally demonstrates a significant effect regarding computer use in class in most of the selected countries, with female teachers showing a higher frequency in the use of computers for instructional purposes in the Netherlands and in Australia and male teachers using computers more frequently in class in Poland and Germany. Age, however, plays a significant role in Germany and Lithuania where younger secondary school teachers use computers in class on a more regular basis. Overall, the teachers' attitudes could be identified to have the most significant impact on the frequency of computer use in most countries. Restraints among the IT resources at school, however, are not linked to the frequency of computer use. At that, no regular pattern could be detected between countries with different intensities of computer use, the students' achievement in CIL or the tradition of implementing digital media in teaching-related contexts. In Australia, for instance, – a country where the majority of teachers uses computers for instructional purposes on a regular basis and where students show above-average CIL competencies – eight out of the eleven factors analyzed show a significant correlation with the frequency of computer use. In the Netherlands, however, – a country which has been selected according to the same criteria – only four of these factors show a correlation. In Germany, where teachers report an infrequent use of computers in class and where students perform averagely compared to the European average, nine factors show a significant correlation.

The fact that the considered factors explain frequency of computer use in class differently is equally reflected in the regression models' explained variance. In the Netherlands, for instance, the model explains the lowest variance (9%) , whereas the model for Germany explains the highest amount of variance (35%).

Restraints among the IT resources at school	Netherlands β	Denmark β	Australia β	Poland β	Germany β	Lithuania β
School does not have sufficient ICT equipment (e.g. computers)[A]	0.03	0.13*	-0.01	0.13*	0.09	0.12*
Limited connectivity to the Internet[A]	0.06	0.04	0.05*	-0.02	-0.03	0.03
ICT is not considered a priority for use in teaching[A]	0.00	-0.05	-0.02	0.00	0.18*	0.09*
Teachers' Attitudes						
Positive view on using ICT[B]	0.00	0.10*	0.14*	0.07*	0.06*	0.04
ICT self-efficacy: Preparing lessons that involve the use of ICT[C]	0.17*	0.19*	0.19*	0.25*	0.24*	0.14*
Professional development with regard to courses on integrating ICT[D]	0.07*	0.02	0.07*	0.05*	0.08*	0.10*
Collaboration with colleagues to develop ICT-based lessons[E]	0.06	0.08*	0.07*	0.02	0.11*	0.01
Collaboration with colleagues to develop ICT use[E]	0.04	0.18*	0.11*	0.13*	0.13*	0.00
Background-variables						
Age[F]	-0.02	0.03	0.03	0.00	0.09*	0.08*
Gender[G]	0.08*	0.00	0.05*	-0.06*	-0.12*	0.04
Teachers' experience in using ICT for teaching purposes[H]	0.15*	0.02	0.10*	0.19*	0.20*	0.17*
R^2	0.09	0.17	0.15	0.24	0.35	0.15

Notes:

β - Regression weight (standardized)

Dependent variable: Frequency of tIcomputer r use during lessons 0 - never; 1 - less than once a month; 2 - at least once a month but not every week; 3 - at least once a week but not every day; 4 - every day

* significant coefficients

[A] 0 - agree; 1 - disagree

[B] International index (M = 50; SD = 10; min. = 9.3; max. = 76.9)

[C] 0 - I don't know how to do this; 1 - I know how to do this

[D] 0 - no; 1 - yes

[E] 0 - disagree; 1 - agree

[F] 0 - 50 years and older; 1 - up to 49 years

[G] 0 - male; 1 - female

[H] 0 - less than two years; 1 - two years and more

Table 1: Regression model to explain secondary school teachers' use of computers during their lessons

CONCLUSIONS

It seems to be well-accepted that ICT as such does not support learning itself but needs to be properly integrated into relevant learning scenarios (Lai, 2008; Law et al., 2008). The results can be considered as hints to important factors, which contribute to the teachers' use of computers in their lessons. Especially the secondary school teachers' ICT self-efficacy – whether they are convinced that they can prepare lessons that involve the use of ICT or not – is an important factor in every country that was selected for the analysis. Other factors of the teachers' attitudes turned out to be equally decisive for the frequency of computer use. The IT resources, however, were not equally significant for the teaching-related use of computers in most countries. This finding gives rise to the assumption that teachers do not use computers in class, despite an optimal technical environment, if they lack the confidence in using it. At this point, it can be deemed desirable for future research to analyze whether professional development in the field of integrating digital media into the context of teaching is related to the teachers' self-efficacy and whether this could constitute a way of promoting the use of digital media in teaching using appropriate measures.

The presented research identified differences in the relevance of the factors in the selected educational systems could not be explained by the three criteria used such as the traditions and policies in implementing new technologies in teaching and learning or the CIL competencies. Therefore, a deeper look into the different systems could grant an insight into why certain factors are relevant in some countries but not in others. Cultural and pedagogical differences as well as other factors like the principals' priorities regarding the facilitation of ICT use could be included. The fact that other factors such as the use of digital media could equally be responsible, is illustrated by the partly small proportions of explained variance.

Overall, further research may want to take a closer look at each of the selected countries analyzed in this paper as this could yield important insights for policymakers and provide knowledge for educational systems. The interpretation of the results further needs to take into account that the latter do not reveal a causality between the respective factors. In this matter, a future longitudinal investigation would be expedient.

REFERENCES

Acker, F. v., Buuren, H. v., Kreijns, K. & Vermeulen, M. (2011). Why teachers use digital learning materials: The role of self-efficacy, subjective norm and attitude. *Education and Information Technologies 18*(3), 495–514.

Albirini, A. (2006). Teachers' attitudes toward information and communication technologies: the case of Syrian EFL teachers. *Computers & Education 47*, 373 – 398.

Anderson, R. (2008). Implications of the information and knowledge society for education. In J. Voogt & G. Knezek (Eds.), *International handbook of information technology in primary and secondary education* (pp. 5–22). New York, NY: Springer.

Davis, N., Eickelmann, B. & Zaka, P. (2013). Restructuring of educational systems in the digital age from a co-evolutionary perspective. *Journal of Computer Assisted Learning 29*, 438–450.

Eickelmann, B. (2011). Supportive and hindering factors to a sustainable implementation of ICT in schools. *Journal for Educational Research Online/Journal Online*, 3(1), 75–103.

Fraillon, J., Schulz, W. & Ainley, J. (2013). *International Computer and Information Study: Assessment Framework*. Amsterdam: International Association for the Evaluation of Educational Achievement (IEA).

Fraillon, J., Ainley, J., Schulz, W., Friedman, T. & Gebhardt, E. (2014). *Preparing for life in a digital age. The IEA International Computer and Information Literacy Study International Report*. Springer.
Griffin, P., McGaw, B. & Care, E. (2012). *Assessment & teaching of 21st century skills.* Dordrecht, the Netherlands: Springer.
Govender, D. & Govender, I. (2009). The Relationship between Information and Communications Technology (ICT) Integration and Teachers' Self-efficacy Beliefs about ICT. *Education as Change 13*(1), 153–165.
Lai, K. W. (2008). ICT supporting the learning process: The premise, reality, and promise. In J. Voogt & G. Knezek (Eds.), *International handbook of information technology in primary and secondary education* (pp. 215–230). New York, NY: Springer.
Law, N., Pelgrum, W. J. & Plomp, T. (Eds.) (2008). *Pedagogy and ICT use in schools around the world: Findings from the IEA SITES 2006 study*. Hong Kong: CERC-Springer.
Legris, P., Ingham, J. & Collerette, P. (2003). Why do people use information technology? A critical review of the technology acceptance model. *Information and Management, 40*(2), 191–204.
Mullis, I. V. S., Martin, M. O., Foy, P. & Arora, A. (2012). *TIMSS 2011 International Results in Mathematics*. Chestnut Hill, MA: Boston College.
OECD (2011). *PISA 2009 Results: Students On Line: Digital Technologies and Performance.* Paris: OECD Publishing.
Owston, R. D. (2003). School context, sustainability, and transferability. In R. B. Kozma (Ed.), *Technology, innovation and educational change. A global perspective* (pp. 125–162). Washington, DC: ISTE.
Petko, D. (2012). Teachers' pedagogical beliefs and their use of digital media in classrooms: Sharpening the focus of the 'will, skill, tool' model and integrating teachers' constructivist orientations. *Computers & Education 58*, 1351–1359.
Rasch, G. (1960). *Probabilistic Models for some Intelligence and Attainment Tests.* Danish Institute for Educational Research, Copenhagen.
Spiezia, V. (2010). Does Computer Use Increase Educational Achievements? Student-level Evidence from PISA. *OECD Journal: Economic Studies 1,* 1–24.
Venkatesh, V., Morris, M.G., Davis, F.D. & Davis, G.B. (2003). User acceptance of information technology: Toward a Unified View. *MIS Quarterly, 27*(3), 425–478.
Voogt, J., Erstad, O., Dede, C. & Mishra, P. (2013). Challenges to learning and schooling in the digital networked world of the 21st century. *Journal of Computer Assisted Learning 29*, 403–413.
Voogt, J. & Knezek, G. (Eds.) (2008). *International handbook of information technology in primary and secondary education.* New York: Springer.

Kerstin Drossel is working as a postdoctoral research assistant in the area of educational science with a focus on large-scale assessments, teacher collaboration and motivation.

Birgit Eickelmann is professor for educational science with a focus on ICT in teaching and learning contexts, especially in schools and teacher education. She is IFIP member of WG 3.3, TC3.

Julia Gerick is working as a postdoctoral research assistant in the area of educational science with a focus on ICT in teaching and learning, school development and school leadership.

Analysing the Skill Gaps of the Graduates of Vocational ICT Programs in Afghanistan

Mohammad Hadi Hedayati, *hedayati@tlu.ee*
Tallinn University, Institute of Informatics, Estonia

Mart Laanpere, *martl@tlu.ee*
Tallinn University, Institute of Informatics, Estonia

Abstract

This study compares the expectations of employers and ICT Vocational institution graduates about job skills leading to differences defined as skill, employability, and perception gaps based on learning from employers in Afghanistan. According to the implementation of the European Qualifications Framework (EQF), the European e-Competence Framework (e-CF) was created for application by ICT service, user and supply companies, for managers and HR departments, for education institutions and training bodies including vocational education, for market watchers and policy makers, and other organizations in public and private sectors. Within e-CF five different areas (PLAN, BUILD, RUN, ENABLE and MANAGE)have been created which are very sufficient to use for ICT Professionals in different sectors. Analysis will be based on e-CF skills in the five categories: Plan, Build, Run, Enable and Manage. The results will identify the gaps for each of the five skill categories. The findings show a huge gap between Institution graduates and the labour market, but the demands of the employers match with the e-CF.

Keywords

ICT Vocational Program, Skill gap, Curriculum Development, Students Alumni

INTRODUCTION

Afghanistan is one of the poorest countries in the world, and has experienced many years of intensive conflicts. Four decades of war and political instability have hindered the development of the infrastructure of the country. The human terrain of the country has taken the most damage and this is especially evident in the educational sector. However, rebuilding has continued in several parts of the country (IRA, 2011).

Although security is the main issue and the people of Afghanistan have suffered a lot including fatalities from suicide attacks, millions of students can attend schools and universities. Currently, more than fifty thousand female and male students are enrolled at vocational institutions in the country. Information Communication Technology (ICT) is the targetfield within these vocational institutions. The vocational institutions are attracting and educating the next generation of Afghan professional workforce.

Alongside this positive development in the vocational education sector, there is a main challenge to the future growth of the ICT vocational education study programs in Afghanistan, which is hindered bythe complex relationship between employers, training institutions, and graduates in the field of ICT. This paper analyses the mismatch between the contents of existing ICT study programs and expectations of both graduates of vocational institutions and their (potential) employers in Afghanistan (Zarman, 2011)

COMPETENCY PROFILES FOR ICT SPECIALISTS IN OTHER COUNTRIES

Recent developments in ICT professional education curricula aremainly focusing on the introduction of studies based on the job market. In the past ten years, a great deal of research has been devoted to competences and qualifications in ICT professions. Some of these studies focus on education and training, examining the profession of computer science and technology and the ways in which this impacts on school guidance to students in vocational institution education. Other research findings focus on graduate students and the labour market, examining skills in working conditions, quality of employment, careers, and work life balance (Agarwal, 2013).

Although ICT is one of the most standardised domains across all national contexts, the competency profile for an ICT specialist is not standardizedat the global level. Many countries have definedICT qualifications and competence models at the national level, but this is not a case in Afghanistan. In developed countries, there is a clear difference between ICT study programs on different levels: while some programs are more theoretical and focus on high-skilled jobs in computer science and engineering, others encompass a wider spectrum of professional ICT skills as practitioner skills. Recent research (Valenduc ,2011)indicatesthat ICT professions may include several occupational groups:

> "*The traditional ICT competency profile is occupations in design and engineering of IT systems, software, and networks; in the development of software applications, programming and coding, functional analysis, quality monitoring, software parameterization; in hardware and software maintenance, assistance to users and help-desk.*
>
> *But the new ICT competency profile is web and multimedia occupations, which combine technical skills in computers, software and networks with communication skills, management skills, planning skills.*
>
> *The ICT professional can be specific fields like medical informatics, scientific informatics, e-commerce platforms, enterprise resource planning systems, supply chain management systems, customer relationships management systems*."

There is a common policy vocabulary and a common set of curricular goals with respect to ICT education atuniversity level. In many countries, there exist separate policy approaches for defining the goals and contents for university-level and vocational ICT education, while some other countries seem to have an integrated approach. There are countries that do not have a specific policy approach to vocational ICT education i.e. Greece, Poland, Portugal and Romania(European Centre for the Development of Vocational Training, 2011)

The importance of ICT for the developing countries economy, cultureand educationis obvious. Impacts of ICT developments on society create an 'information society' and new opportunities and challenges in all areas of work and life have arisen. In particular, this applies to the ICT sector itself, but it is not limited to one sector only (Lu, 2009).Well-skilled staff areneeded to manage business and work processes in the core ICT sector as well as in public organizations and industries in other sectors. There is a growing need to understand, teach and apply the new ICT skills among professions such as e.g. computer repairman, network technician, database expert, e-commerce, fixed and mobile telecommunications specialist etc. There is also an extensive need to possess a range of ICT competences relevant for different job profiles outside of ICT sector. These demands can be met by various study programs atboth higher education and vocational education institutions (Vaquero, Toro, Martín &Aregita, 2009).

The situation regarding ICT experts' supply and demand varies across developing countries. The state of development of the ICT economy and of the national systems of higher education and vocational education affect this situation both in qualitative and quantitative terms. However, recent worldwide problems with the new economy have contributed to more reasonable discussions on the demand and the problems of the ICT labour market (Garrido, Sullivan & Gordon, 2010).

To address the skill gap between expectations of ICT labour market and current study programs offered in vocational schools, the curricula need to be designed in accordance with the skill needs and contents of ICT employment. Though based on the identified ICT skill needs and contents, a didactic and pedagogic reflection is necessary for decisions on an appropriate range and depth of competences of the ICT curriculum.

European policies in vocational ICT education are guided by a set of well-defined job profiles in 14 generic work areas at sub-degree levels. These job profiles have been developed and are constantly revised by representatives of educators, employers and standardization bodies in the field of ICT. Yet, we cannot take it for granted that developing countries outside Europe should just copy the ICT job profiles, qualification standards and study programs from Europe, as the job market needs might be quite different there.

One of the recent frameworks for curriculum development in ICT across Europe is e-Competence Framework. The European countries developed the e-Competence Framework (e-CF) from their varied perspectives, drawing on theirtechnical expertise. The European e-CF version 3.0 provides a reference of forty competences as required and applied at the ICT workplace, using a common language for competences, skills and capability levels that can be understood across Europe.

The e-CF is a component of the European Union's strategy on «e-Skills for the 21st Century». It is also supporting key policy objectives of the «Grand Coalition for Digital Jobs» launched in March 2013. It is promoted as a very useful tool to developdigital skills and the recognition of competences and qualifications across countries, and to foster ICT professionalism in Europe.

As the first sector-specific implementation of the European Qualifications Framework (EQF), the e-CF was created for application by ICT service, user and supply companies, for managers and HR departments, for education institutions and training bodies including vocational education, for market watchers and policy makers, and other organizations in public and private sectors.

Within e-CF version 3.0 five different areas (PLAN, BUILD, RUN, ENABLE and MANAGE)have been created which arevery sufficient to use for ICT Professionals in all industry sectors.

Most of the ICT professionalworkforce in European countriesareemployed byICT demandorganizations spread across multiple sectors.EU-wide agreement on competence standards for ICT professionalsserves as a good model for other countries, but it cannot be directly applied in a cultural contextthat is radically different from well-developed European economies. It clearly requires analysis, adaptation and adjustment to local needs before something similar can be implemented in developing countries such as Afghanistan.

Vocational ICT study programmes in Afghanistan today

The Afghanistan Technical Vocational Institute (ATVI) is the largest public vocational school located in Kabul. ATVI was established through a collaborative effort by national and international affiliates, with contributionsfromthe Afghan Government and Ministry of Education. ATVI is providing education to more than eight hundred male and female students from Kabul and other urban and rural regions in Afghanistan.

The ICT Department is one of the largest departments at ATVI, in parallel with departments such as the Horticulture Department, Business Management Department, Construction Department and Automotive Department. The ICT vocational program at ATVI has a duration of two years. The following table (Table 1) shows the subjects taught infour semesters.

First Semester	Second Semester	Third Semester	Fourth Semester
ICS	Programming Concept	OOP using java	Software Engineering
Operating System	Operating System-II	Visual Programming	Database Programming
Computer Application-I	Computer Network-I	Web Designing	Data Structure
Civics-I	Computer Application-II	Database System	Network Administration
Mathematics-I	System Hardware-II	System Administration	Data Communication
Physics-I	Civics-II	Computer Network-II	System Administration-II
English-I	Mathematics-II	Civic Education-III	Project
	English-II	English-III	

Table 1: Afghanistan Technical Vocational Institute (ATVI) two year ICT Curriculum

- Computer Technology Institute (CTI)

The Small Industry and Food Material Management Institute was established 1970.Recently, due to low student enrolment and changing demands from the job market, this institute started offering an ICT major and changed its name to CTI. The CTI was established in 2006 according to labour market needs. There are now more than 500 alumni. Mr. Bawary, the principle of CTI, states that they train ICT managers and assistants, and the graduates have been hired at banks, mobile and TV companies.

The CTI has copied the curriculum from the Computer Science Faculty of Kabul University. The Faculty consists of three departments, each offers a separate study programme: Software Engineering, Information Technology and Information System. Since the CTI faced lack of human resources, the Information System Department has not yet been established in CTI and Information Systems courses are not included in CTI study programmes.

- Afghan Institute for Technology (AIT)

The Afghanistan Institute for Technology (AIT) was established in 1951, by the government of Afghanistan withsupport from the United States of America (USA). Hundreds of students have received quality technical education which was adequately complemented with training materials and technical expertise provided from its affiliate universities in the United States.

After the occupation by USSR in 1979 the US Cooperation and support to AIT was slowly withdrawn and AIT was stopped until 1993. Also, during the civil war from 1992 to 1996, district third where this institute is located became the frontline of battles leading to enormous damage to the institute.

The Government of Afghanistan commencedrenovation of building, labs and workshops. AIT was reopened in 2002 and admitted students in seven departments

(Architecture, Building Constructions, Machine tools technology, Auto Repairing, Sheet Metal Work, Electricity and Computer Science) with minimum educational facilities under the Ministry of Education, Government of Afghanistan. The AIT still lacksa standard curriculum. The following subjects are offeredin the two year ICT diploma at AIT.

- Fundamental of Databases
- PHP
- ASP.net
- VB.net
- Access Program and Project
- My SQL
- Data Communication
- Computer Network

Although the program of is not designed properly, most of the graduates find jobs due to huge demand on the job market. The alumni report described by one of the instructors of CTI mentioned that the graduates get hiredbypublic and private organizations such as TV companies, banks, ICT support companies, and hospitals. The job market in Afghanistan’s ICT sector is likely to grow also in the future.

EMPIRICAL STUDY: ANALYSING THE SKILL GAPS OF VOCATIONAL ICT GRADUATES FROM THE EMPLOYERS’ PERSPECTIVE

A quick review ofICT programs intechnical vocational education identifiedthat a mixture of methods has been used for seeking employers’ opinions regarding ICT profile and competencyskills required. The method of focus group interview was selected, because it can be effective in qualitative data collection from a clearly defined target group.

The top 20 employers of vocational ICT graduates were identified based on initial consultations with 4 vocational schools. 20 representatives of these employers were invited to participate in this focus group interview, but only nine representatives were able to attend the meeting. Together with the invitation, the following preparatory questions were sent to invitees:

- How many new ICT specialists has your company hired during last three years, to which jobs/positions? How many of them had vocational degree, how many had a university degree?
- What are the ICT-related jobs in your company (e.g. software developer, tester, analyst, system administrator, IT support) and how many of each are you planning to hire within the next three years?
- What kind of induction training is offered to the newly hired ICT specialists in your company, to addressthe skill gaps of new employees?

Based on the analysis of the existing ICT Vocational programs in four Afghan vocational institutes and previous informal discussions with employers, the researcher selected the following questions for the focus group interview:

1. From the employers’ perspective, what competencies are expected of ICT Vocational graduates in the current ICT job market in Afghanistan?
2. What changes do you as employers expect in the objectives, structure and content of ICT Vocational programs in Afghanistan? Should there be more specialization, e.g. focusing separately on planning, developing, managing, selling and supporting IT systems?
3. In which ways are the employers willing and able to contribute to the change in ICT Vocational programs in Afghanistan? E.g. advising on program contents,

offering opportunities for practical placement, arranging summer/winter schools, supervising and evaluating student projects.

4. What should be the main differences between University-level and the Vocational study programs in the ICT domain in Afghanistan?
5. What courses need to be added toexisting ICT Vocational programs, what courses can be removed?
6. What changes are required in the main course contents and the ways of teaching and learning?
7. What other measures canyou suggest to improve the quality of education at ICT Vocational institutions?

The semi-structured interview was carried out for 2 hours; it was recorded and transcribed for the qualitative content analysis.

RESULTS

Responses from employer representatives to the three questions which were attached to theinvitation letter are summarised in Table 2 below.

No	**Employer Name**	**Related Jobs**	**No People Hired**		**Type of Training**	**Future Plan for hiring**
			TVEI	Uni-versity		
1	Mobile communication Company I	software developer, troubleshooting, analyst, system administrator, IT support, Network Engineer	31	51	Network Management and Planning, Programming, Communication Skills, Marketing, MIS	50
2	Mobile communication Company II	software developer, troubleshooting, analyst, system administrator, IT support, Network Engineer, IT Specialist	43	75	Network Management and Planning, Programming, Communication Skills, Marketing, MIS	50
3	Bank	Database Specialist, system administrator, IT Officer	19	32	Network Management and Planning, Programming, Communication Skills, Marketing, Database Development	20
4	Company I	software developer, Database Expert, system administrator, IT support, Network Engineer, IT Technician	3	19	Software Engineer, Network Management and Planning, Programming, Communication Skills, Marketing,	20
5	Ministry of Information Communication Technology	software developer, troubleshooting, analyst, system administrator, IT support, Network Engineer, IT Specialist	41	64	Network Management and Planning, Programming, Communication Skills, Marketing, Software Engineer	150
6	Kabul airport	e-Security specialist, System Administrator, IT Officer	4	9	New Technology Introduction	20
7	TV I	Program editor, IT Manager, IT Support	2	8	Multimedia tools,	10
8	Hospital I	IT Assistant	1	1	-	2
9	AEC (Afghan Election commission)	IT Assistant, System Admin, Database Expert	45	78	Oracle 11G,	50

Table 2: feedback from employers

In addition to the focus group instruments above, we refer to e-CF applying different analytical tools on data from employer interviews. Within e-CF five different competence areas (PLAN, BUILD, RUN, ENABLE and MANAGE) have been created which are sufficient to use for ICT Professionals in different sectors. The analysis of the focus group datasuggest gaps in all five e-CF competence areas. According to the rest of the questions, we examined which skills employers perceived important when hiring ICT graduates (see Table 3).

Plan	**Build**	**Run**	**Enable**	**Manage**
Ability to adapt tochanging technology Technical skill Able to develop business plan Monitoring Innovation Problem solving	Business ethics Professionalism System development Solutions Testing and analysis Problem solving	Can work at Small and intermediate Organization User supports Technical supports Problem solving	discover personal competences based on experience and certification information security strategy ICT quality strategy Digital Marketing	Self confidence Critical thinking Creative thinking Interpersonal skills Leadership skills Problem management Risk Management Relationship management

Table 3: Skill demands identified inemployer interviews

Satisfaction with vocational ICT programs

Employers are not satisfied with the current ICT vocational education in graduate skills, said by one of the mobile company delegation. This shows that a set of courses offered by ICT vocational education institutes do not match the skills demanded by employers.

The employers are likely to perceive soft skills more important than professional skills. However, ICT vocational graduates with limited and weak professional skills are undesirable for employers, said by one of company representative. The focus group results, for instance, show a clear signal to the Problem Solving that is under professional skills. As shown in Table 3, Problem Solving has the largest gap in professional skills.

Need for change

The employers' responses suggest that institutes shouldchange the objectives and the outcomes to match the employers'needs. The employers don't want to offer any extra courses to the graduates. In Afghanistan, because of economic challenges, the employer cannot provide summer or winter schools. So they suggest updating the curriculum according to the labour market.

The employers stated that they don't have too manyproblems when they hire university graduates, but compared to the vocational institution graduates the university graduates demand high salary.

Discussion

The focus group data do not show any relationship between usefulness of the ICT education and ICT workforce and market demands in Afghanistan.

According to the research findings the ICT vocational graduates in the job market are not being provided with the required skills. The students who complete courses from vocational institutions are having on the job training.

There is a guarantee that they will find asuitable ICT related profession, when they graduate from institutions. They don't have many problems in finding a suitable job to develop their career path. Therefore, the vocational training institutes have been able to compile the ICT study programmes in the easiest manner, by merely copying the parts of computers science curricula from Kabul University, where most of the staff members received their university degree. The following courses are common at vocational and university level.

Name of the courses	**ATVI**	**CTI**	**AIT**	**University**
Introduction to Programming	Yes	Yes	Yes	Yes
Introduction to Database	Yes	Yes	Yes	Yes
Introduction to Network	Yes	Yes	Yes	Yes
Operating system	Yes	Yes	Yes	Yes
Computer Architecture	No	No	No	Yes
OOP	Yes	Yes	Yes	Yes
Software Engineering	Yes	Yes	No	Yes

Table 4: Common Courses offered in institution and University

CONCLUSION

The ICT vocational education programmes in Afghanistan annually supply considerable amount of ICT professionals to the labour market. In order to this the state recognized the importance of vocational education and training; the Afghanistan government has spent a large amount of money on vocationalinformation technology education. When considering the above criteria the study of the usefulness of ICT education in vocational education and training sector in Afghanistan, and giving recommendations is a timely and an important requirement in the context of policy making and planning in ICT related training provision TVET sector.

According to the findings, there is no national vocational qualification framework of Afghanistan to modify qualifications.Competencies of the academic staff are the intervening variables that affectthe quality of education as well as to the usefulness of the ICT education in vocational education in Afghanistan. The employer demands match with e-CF which is a standard framework in Europe.

Our study revealed that until today, the vocational training institutes of Afghanistan have designed their ICT study programmes by copying the university curricula, without consulting the employers. The big demand for basic-level ICT specialists has made this practice possible. Yet, the authors suggest that in longer term the needs of employers have to be addressed by the redesigned study programmes, in order to fulfil the responsibility of Technical Vocational Training Institutes in the societal development. Our next steps inspired by this study are aiming at designing the new model of designing vocational ICT curricula that takes into consideration the specific cultural context of Afghanistan and engages employers, alumni, students and staff in the process.

REFERENCES

Agarwal, T. (2013), Vocational education and training programs (VET): An Asian perspective, Asia-Pacific Journal of Cooperative Education, 2013, 14(1), 15-26 http://www.apjce.org/files/APJCE_14_1_15_26.pdf

Dickson, V. (2004). Collaboration plus! The development of an information literacy and communication program. Australian Library Journal, 53 (2), 153-160.
European Centre for the Development of Vocational Training, (2011). Vocational educationand training at higherqualification levels, Research Paper No. 15. publications Office of the European Union, pp. 33-39.
European e-Competence Framework, www.ecompetences.eu
European Qualifications Framework for lifelong learning (EQF), http://www.eqavet.eu/gns/policy-context/european-vet-initiatives/european-qualifications-framework.aspx
Garrido, M., Sullivan, J. and Gordon, A. (2010). Understanding the links between ICT skills training and employability: an analytical framework. In *Proceedings of the 4th ACM/IEEE International Conference on Information and Communication Technologies and Development* (ICTD '10). ACM, New York, NY, USA, Article 15, 10 pages. DOI=10.1145/2369220.2369234 http://doi.acm.org/10.1145/2369220.2369234
Islamic Republic of Afghanistan (2011), Direct communication: Information and replies on VET policy, given to the OECD, (VETquestionnaire), Kabul and Paris. http://www.oecd.org/investment/psd/VocationalEducation.pdf
Jianwei, W. Hongjie, B. and Guoxin, Y. (2011). Exploration and analysis of teaching methods of higher vocational education curricula, volume 108 of Advances in Intelligent and Soft Computing.
Kigongo-Bukenya, I. M. N. (2003). Towards a viable curriculum: A comparative study of curricula at the East African School of Library and Information Science and the departments of Library and Information Studies of the Universities of Wales, Botswana, and Capetown. Education for Information, 21 (2/3), 113-148.
Lu, W. (2009).Ontologyaware Course Management for Curriculum Evolution Process in Higher Education.VERITE 2009/3/13
National Education Strategic Plan for Afghanistan. Retrieved 28.04.2014 from http://www.iiep.unesco.org/fileadmin/user_upload/News_And_Events/pdf/2010/Afghanistan_NESP.pdf
Opel, S. and Brinda, T. (2013). Arguments for contextual teaching with learning fields in vocational IT schools: results of an interview study among IT and CS training companies, Proceedings of the 8th Workshop in Primary and Secondary Computing Education, p.122-131, Aarhus, Denmark
Ryan, R., (2011). Apprenticeship: between theory and practice, school and workplace, Economics of Education Working Paper Series 0064, University of Zurich, Institute for Strategy and Business Economics (ISU)
UNESCO (2004).Improving Access, Equity and Relevance in Technical and Vocational Education and Training, Synthesis Report. UNESCO Asia and Pacific Regional Bureau for Education, UNESCO Bangkok, Thailand
Valenduc, G., (2011). Not a job for life? Women's progression, conversion, anddropoutin ICT professions, International Journal of Gender, Science and Technology, Vol. 3, No.2, p. 484.
Vaquero, J., Toro, C., Martín, J., Aregita, A. (2009).SemanticEnhancement of the Course Curriculum Design Process. In J.D. Velásquez et al. (Eds.): KES 2009, Part I, LNAI 5711, pp. 269–276, 2009.
Zarman, w. (2011). Vocational Education and its Potential Problem, International Seminar on Islamic Education on May 18, 2011, IbnuKhaldun University, Bogor

Mohammad Hadi Hedayati is Vice Chancellor of Kabul University of Afghanistan. He is PhD Student at Tallinn University of Estonia. He has studied his Master Degree at University of Western Cape South Africa.

The Author is also responsible of teaching Information Technology at Kabul University. The main research area is ICT curriculum development.

Mart Laanpere, PhD, is a senior researcher in the Institute of Informatics and head of the Centre for Educational Technology at the Tallinn University, Estonia.

Promoting collaborative programming for introductory programming courses through an “individual work branch and real-time sharing” approach

Yuya Kato, kato@sakailab.info
Graduate School of Informatics, Shizuoka University

Yoshiaki Matsuzawa, matsuzawa@inf.shizuoka.ac.jp
Graduate School of Informatics, Shizuoka University

Sanshiro Sakai, sakai@inf.shizuoka.ac.jp
Graduate School of Informatics, Shizuoka University

Abstract

We have developed a system to support collaborative programming in introductory programming education. Our ideal collaborative programming is described as (1) all members make contributions to the group products on an assumption of skill level differences between members, and (2) a product is not a result of working individually, but the procedure includes activities of reading others’ code, improvement, and expansion. We proposed the model named “individual work branch and real-time sharing” to support collaborative programming. The model has the following features: (A) changes made by someone do not directly affect others’ code, (B) it is always possible to view others’ latest programs, and (C) novice programmers can import others’ programs in a simple way. We implemented the model as workable software, and it was examined using 6 liberal arts students (3 members per group) in our introductory programming course. Qualitative analysis was conducted by using system logs and results of users’ interviews. The results show that the system usability is an acceptable level for novice programmers. We also found that interaction logs between students showed that the system succeeded in promoting collaborative programming through limited situations.

Keywords

Collaborative Learning, Programming education, novice

INTRODUCTION

In introductory programming education, students might be given opportunities to write a program through group collaboration (Matsuura & Aiba, 2003) (Tamada, et al., 2005). In this paper, we call such an activity collaborative programming (details described in section “Definition of Collaborative Programming”).

However, there are some difficulties in managing group tasks in a practical class from a teacher’s viewpoint. In our class, the following phenomena have been observed: (A) most of the program is written by the student who is most skilled in programming in the group (driver); the others (followers) do not write the any of theprogram, (B) group members work in completely separate task divisions (e.g. agroup product of creating a game comprised of small sub-games, each working independently).

There is another problem from the view of the file sharing method. General students in the introductory programming course have limited time to learn configuration management tools (e.g. git, mercurial), and such tools are designed for professionals so the students feel using them is cumbersome.

We conduct collaborative programming as the final project in our introductory programming course. Liberal arts students participated in the project as a compulsory assignment. It was the first experience of collaborative programming for almost all the students. Students formed their groups arbitrarily with the limitation of group membership being three members.

We conducted a survey by using a questionnaire regarding collaborative programming for the 2013 class. Seventy valid responses were collected. The results indicate half of the students indicated that “Merging of source code” and “Work sharing”were perceived as obstacles in collaborative programming. The results indicate 60% of students used external storage units when they shared their code. “Others” included studentswho sharedtheirprogramming side-by-side by using a single computer. Students were asked to rank programming ability in the group, and how much code they had contributed. We measured the correlation between the ranking of programming ability and code contribution. These data show that followers’ code contribution is less than half of drivers’. Followers cannot contribute equally due to their lack of ability. We thought this was due to hesitation involving the main programmer’s work. In other words, in spite of followers having opinions about the program, they hesitate to express them.

For solving these problems, we proposed a new model to promote collaborative programming in introductory programming education by arranging current programming models. Next, the model was examined by implementing the actual tool based on the model. The interaction diagrams for collaborative programming were provided for evaluation of the tool.

RELATED WORK

Many tools to support group programming have been proposed in software engineering, CSCW (Computer Supported Cooperative Work); some of them are proposed for education in CSCL (Computer Supported Collaborative Learning).

Collabode (Goldman, 2011)is one of the latest tools designed for educational usage. Collabode is categorized as a real-time synchronization environment where multiple students can edit a code simultaneously, as similar tools have been proposed in the literature (Vandeventer &Barbour, 2012). Technically the tool was built upon the EtherPad which is the library for developing a simultaneous editing environment for the document. The library is used for the implementation of Google Doc. Saros (Salinger et, al., 2010) is a similar tool built on Eclipse. The tool provides a function to follow the view (scroll and movement of files) of the specific member. The authors claim that the tool is useful in terms of consciousness sharing of the group members. In addition, Saros is a system that can be utilized to review the source code between the group members.

However, real-time synchronization models have been rarely used in professional collaborative programming such as commercial software development or open source projects. Professionals have used SCM (Source Configuration Management) tools for over thirty years including CVS, subversion or Git. Currently Git / Github is the standard tool used to conduct collaborative programming in open source community. Those tools merely support the branch and merge model; however, they do not support real-time sharing even though they are the latest tools.

Currently, there is no clear evidence that a real-time synchronization model and branch and merge model offer advantages for educational usage. In the literature, there is little research which has attempted to show the advantages of the real-time editing environment in education(Brodahl, et al., 2011; Zhou, et al., 2012). There is a report in which a sequential work strategy has an advantage over a simultaneous one on a complex creative task (André et al., 2014). On the other hand, we could not find any articles that report success in introducing SCM to introductory programming education.

In our analysis, the critical problem of the real-time synchronization model is that a student's editing directly affects the group output. For this reason, students cannot edit the source code without having a sense of hesitation. Another disadvantage is that the model requires time-sharing; it does not promote work outside of class collaboration (Zhou, 2012). By contrast, the branch and merge model is flexible for both in terms of editing without hesitation and sharing time. The major problem of the model is it requires a high cognitive load for learning the model and operation thereof, especially once a conflict occurs; it is a difficult task even for professionals.

There is some related research regarding supporting tools for collaborative learning. Aoki et al., 2009 has developed a system to support distributed pair programming. Users are assigned the role (driver or navigator) via the form of pair programming. Only the driver can edit source code. The system has functions to support communication such as annotation and chat. Similar tools to support distributed pair programming have been developed, for example Sangam (Ho, et al., 2004) and XPairtise (Schümmer&Lukosch, 2008). These systems adopt the same model as Saros for sharing one project with a group. Hence, Sangam and XPairtise have the same problem as Saros.

Accordingly, many tools for supporting communication, supporting distributed pair programming, and SCM for professionals, have been proposed in literature. However, the system that novices can use and the support of "collaborative programming" have not been developed yet.

PROPOSAL OF INDIVIDUAL WORK BRANCH AND REAL-TIME SHARING MODEL

We propose an individual work branch and real-time sharing model to support collaborative programming for novices. In this section, we describe the model by comparison between this model and other existing models.

Definition of Collaborative Programming

Our goal is to promote fully collaborative programming based on agile/knowledge creation theory in the introductory educational environment. Specifically, we emphasize solving the two typical problems observed in practical classrooms. One is skill level difference between group members. Although we do not intend the notion that each member should contribute equally in terms of originally produced code size or devotion of time, wet do intend that members share responsibilities in producing their output with the achievement being reached as a result of their collective work. Another finding in our experience was that many groups design their work in isolation from the other members. For example, a group of three members developed software that is comprised of three mini-games; this means that final group product is just a collection of each individual's work. Although the result can appear well organized if each component has quality, the activities not need interaction between group members and responsibilities are never shared. It cannot be called collaborative programming/collaborative work.

Hence, this research defines the following two descriptions as collaborative programming:

collective contribution

All members make contributions to the group products on an assumption of skill level differences between members.

collective interaction

A product is not a result of the work produced in isolation, but that of responsibility sharing. Procedures would be assumed including highly cohesive interaction among members, such as reading others' code, improvements and expansion.

SCM Model

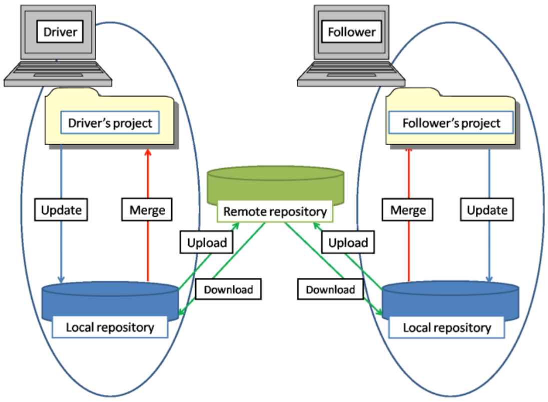


Figure 3: SCM Model

Firstly, we review an existing model implemented in the traditional SCM. We call it the SCM model, and the model is illustrated as shown in Figure 3. According to the figure, the model has the following four characteristics:

1. The user has to "push" in order to reflect local changes to remote.
2. The user has to "fetch" in order to reflect remote contents to local.
3. The user has to solve a conflict when it happens.
4. The user should manage some versions by themselves.

The problem with this model is that the tool requires a high cognitive load for beginners to learn operations, as well as use of terminologies (e.g. push, fetch and merge). We should avoid this as an unnecessary cognitive load prevents students from focusing on important programming work.

Another problem for the model is the leading student's project is not revealed to others until the student intentionally pushes his / her project to the remote repository. As a result, other students might not be able to share the latest version. In this situation, other students have to ask the leading student to push his / her work. However, other students hesitate to ask because they are worried about suspending the leading student's work.

Real-time Sharing and Editing Model

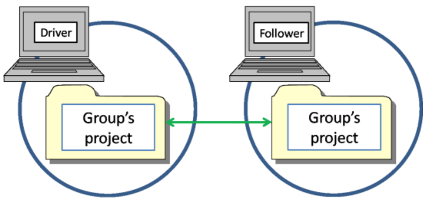


Figure 4: Real-time sharing and editing Model

Secondly, we reviewed another model implemented in the real-time synchronization tool. We call it the "real-time sharing and editing model"; the model is illustrated in Figure **4**. According to the figure, the model has the following characteristics:

1. All members (in a group) share one project.
2. All members have to edit the shared project simultaneously.

3. The shared project is displayed and updated in real-time.

The problem with this model is it is mentally hard for non-leading programmers to edit the code. Regardless of whether a leading programmer is working or not, other members have to think about the possibility of breaking the workable source code, thus making it unworkable. The editing should be done very carefully to assure forward movement in their programming activities.

Individual Work Branch and Real-time Sharing Model

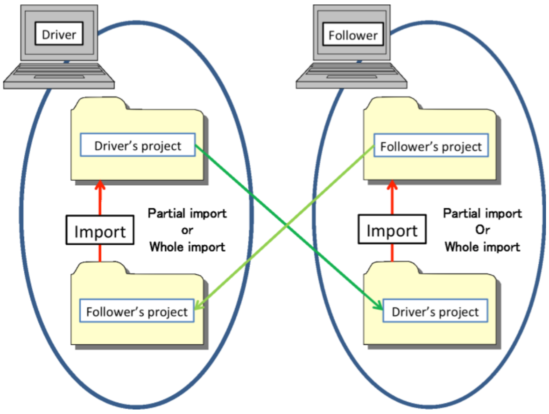


Figure 5: Individual work branch and real-time sharing Model

We propose the individual work branch and real-time sharing model which is designed for novice programmers to encourage collaborative programming; this resolves the problems in the former models. The model is shown in Figure 5. The individual work branch and real-time sharing model has the following characteristics:

1. Every member has his / her own branch.
2. Every member can see the latest version of other members' projects in real-time synchronization. (They cannot edit the others' code directly.)
3. Every member can merge the others' code whenever he / she would like to do so with simple operations.

The aim of the designing feature (1) is to solve the problem of the real-time sharing and editing model. Every single member can hold his / her version, there are no obstacles to proceed with their programming activities. The aim of features (2) and (3) are to solve the problem of the CMS model. Feature (2) allows students to read and import group members' current project without any requests for updating to other group members. Non-leading members do not have to be worried about suspending the leading member's work. Feature (3) solves the high cognitive load problem. Importing incremental differences of files correspond to partial import such as copy and paste.

Hence, we consider that the individual work branch and real-time sharing model supports collaborative programming for novice programmers.

THE PROPOSED SYSTEM: CHECOPRO

Design Objective

We propose CheCoPro (Cheerful Collaborative Programming) as an implementation of the individual work branch and real-time sharing model.

Providing a tool that eliminates the trouble for merging and sharing source code would be considered an approach to solving the problem. We need to provide a mechanism that enables merging of codes by a simple operation that makes it possible for the novice. For file sharing, we need to provide a mechanism without using external tools for file sharing.

We hypothesize that through resolving the above problems, followers can contribute to their group and their work will have higher completion.

Implementation

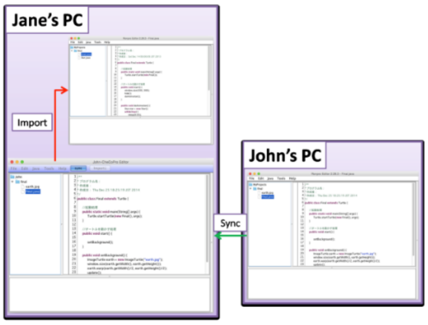


Figure 6: Interface

This system was implemented as a plug-in of the "Ronpro Edirot" which is a Java Code Editor designed for an introductory programming course; we are using it in an actual course. The interface of CheCoPro is shown in Figure 6. Basic functions are synchronization, observation and importing.

Synchronization

We implemented the synchronization function to support the definition (1) of collaborative programming (collective contribution). The user can open any other member's code editor, but cannot edit it directly. The user can toggle the state of others' editors as sync / async. This button also allows one to choose whether one gets the latest file or not. The user can run or import others' programs by changing to async.

Observation

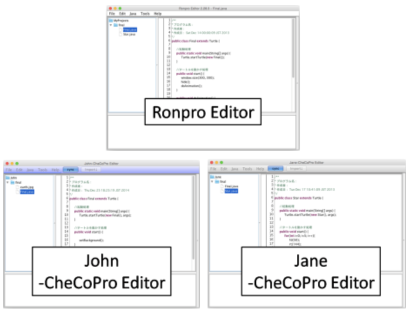


Figure 7: Observing group member's source code

We implemented the function that displays a member's source code in real time for supporting creative extension. Figure 7 shows a user observing his group member's source code. The user writes source code on Ronpro Editor. The other window (CheCoPro Editor) shows members' source code.

Importing

CheCoPro provides two simple ways for merging the others' code to their own branch. One is whole import, another is partial import. The whole import is importing several files at once. A user can choose either importing only Java files or including other files such as image files. There is no special function in CheCoPro to conduct partial import. A partial import is designed to be done manually by copy and paste; a user copies the other's source code from other's editor and pastes it to their own source editor. We believe it is the simplest way for beginners.

EXPERIMENTAL METHOD

We examined CheCoPro in our introductory programming course.

Subjects

We conducted the experimental study using CheCoPro in our introductory programming course which is designed for first year undergraduate students in the liberal arts faculty (Non-CS students). We randomly chose two groups (6 students) where each group had 3 members. Subjects studied a basic knowledge of Java programming over four months in the class (excluding object-oriented programming). None of the subjects had prior experience of collaborative programming.

Examination Description

CheCoPro was used in the final project of our introductory programming course. The period of the final project was 2 weeks, and the task assigned was for them to create theirown software with Java. Subjects were given 20 minutes of instruction for usage of CheCoPro. After finishing the project, we conducted a questionnaire about collaborative programming.

Available Data

Available data is as follows:

	Group A	Group B
Lines of Code	364	2510
Number of Methods	3	31
Number of Resources	20	239
Theme of Product	Congeniality diagnostic	Fighting game
Number of Members	3	3

Table 3: Overview of project

Table 3 shows overview of projects of each group."Methods" means java method. "Resources" means files such as image files or music files. All values of group B are higher than group A's.

The product of group A is a congeniality diagnostic game. It shows a question and choices, then receives input from the user. After these were processed repeatedly, a diagnostic result displayed.

The product of group B is a fighting fame. Characters punch or kick by typing on a keyboard. This game can be played by two people.

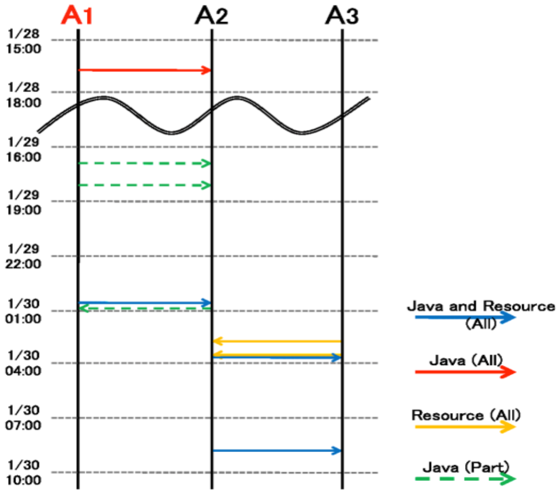
RESULTS

Questionnaire

All six subjects said that they used CheCoPro.

Five of 6 subjects answered that one merit of CheCoPro is exchanging a member's file without the need to use external tools. In free descriptive answers, subjects requested the addition of a chat system because reading members' code is too difficult for novices without previous arrangements or contact.

Members' Interaction



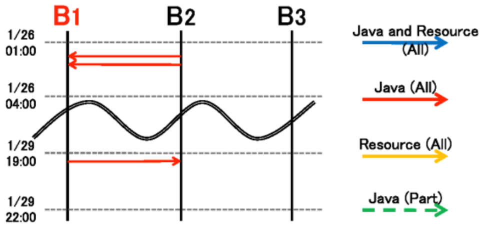


Figure 8: Import flow of Group A Figure 9: Import flow of Group B

Error! Reference source not found. and Figure **7** show the import flow constructed rom the system logs and student interviews. The leader for group A is represented as A_1, and for group B as B_1. Both groups understood how to use CheCoPro because we observed that subjects imported members' code. Group A imported frequently.

In group A, we observed that A_1 wrote the initial source code. A_2 imported A_1's code. After that, A_2 edited the code and ran it. A_2 repeated the operation. Group A produced the congeniality diagnostic game. A_2 created a method to display the diagnostic result. A_2 imported the main method that was made by A_1 and then ran it. It means that A_2 tried to test his method on the main action that was made by A_1. A_3 was not involved in coding, but he provided resources such as image files for his group. Finally, A_3 imported a completed project, then he tested it.

In group B, we observed that B_1wrote the initial code. Group B used not only CheCoPro but also a messenger tool for file sharing. B_1 imported B_2's source code twice. At this time, B_1 and B_2 had the same source code. B_1 copied B_2's code into the newly created file, the purpose of this behavior, we believe, was to keep a backup. B_3 did not participate in the project at all. As a result of the interview, we discovered B_3 had been sick during the project.

DISCUSSION

Collective Contribution

From the result of the questionnaire and observation, there were differences of skill level between members of both groups A and B. All group members contributed in group A. Contributions of each member reflected their skills. About group B; B_2 and B_3 did not contribute to the deliverables. B_2 was not able to contribute to the deliverables, but he read the source code that was written by B_1, and he repeated trial and error. B_2 also tried to edit imported B_1's source code. B_3 could not start his work because of an illness.

According to the above results, we concluded that the individual work branch and real-time sharing model generally worked to support collective contribution.

Collective Interaction

Members of group A used the importing function frequently. Observations indicate the tool succeeded to promote extending other's programs for A_2. Members of group B could see others' source code, but they could not import or extend it.

Accordingly, we concluded that the individual work branch and real-time sharing model worked to support collective interaction.

Lines of code and resources of group A were less than group B's. The product of group A was the congeniality diagnostic game. If user answers a question, the system displays the next question or the result. This software is configured by conditional branches. We considered that followers could read the leader's code because skill level differences between members of group A were smaller than those of group B.

The skill level of the leader of group B was higher than other members, and the product of group B was more difficult than group A's. The source code of group B has as much as 2510 lines, nevertheless there are only a few comments with some variable declarations and methods. Group B made a fighting game. Character images changed by input of user such as punching or kicking. Accordingly, the project of group B has 239 resources. Followers could not understand what kind of images as image files were named indefinitely. In addition, there are no comments.

The source code of group B was written by B_1 alone. B_2 and B_3 could not participate in the project because of illegible source code written by B_1.

Usability

The subjects could understand how to use CheCoPro and they could import member's files with only 20 minutes instructions. By this observation, CheCoPro has enough usability that novice programmers can use it easily.

CONCLUSION

In introductory programming education, students are sometimes provided an opportunity to write a program with a group. It is difficult for novices to do collaborative programming in the present circumstances. We proposed the individual work branch and real-time sharing model, and developed the collaborative programming support tool "CheCoPro" that is available for novices.

We used our introductory programming course for validating the effectiveness of the individual work branch and real-time sharing model. As a result, CheCoPro proved to be usable for novices, and it is true that individual work branch and real-time sharing model can support collaborative programming, but it is limited. Additionally, we have found CheCoPro can provide useful logs for investigating an actual process of collaborative learning done outside the classroom. We will recruit more subjects, and an improved experimental study would be expected to bring the great understanding to collaborative programming.

REFERENCES

André, P., Kraut R. E., &Kittur, A. (2014)Effects of Simultaneous and Sequential Work Structures on Distributed Collaborative Interdependent Tasks. In CHI'14, 139-148.

Aoki, K., Tateiwa, Y., Yamamoto, D., &Takahashi, N. (2009). Implementation of Distributed Pair Programming Exercise System. IEICE Technical Reports, ET, Vol.109, 193-198.

Brodahl, C., Hadjerrouit, S., &Hansen, N. K. (2011). Collaborative Writing with Web 2.0 Technologies: Education Students' Perceptions. Journal of Information Technology Education: Innovations in Practice, Vol. 10, 73-103.

Goldman, M., Little, G., &Miller, R. C. (2011). Real-time collaborative coding in a web IDE. In Proceedings of the 24th annual ACM Symposium on User Interface Software and Technology (UIST), ACM, 155-164.
Ho, C. W., Raha, S., Gehringer, E., &Williams, L. (2004). Sangam: a distributed pair programming plug-in for Eclipse. In eclipse '04: Proceedings of the 2004 OOPSLA workshop on eclipse technology eXchange, 73-77.
Matsuura, S.,& Aiba, A. (2003). A Software Engineering Education by Experimental Software Development Group Work. IPSJ SIG Technical Reports (CE), 13(2002-CE-068), 1-8.
Salinger, S., Oezbek, C., Beecher, K., &Schenk,J. (2010). Saros: An Eclipse Plug-in for Distributed Party Programming. In Proceedings of the 2010 ICSE Workshop on Cooperative and Human Aspects of Soft- ware Engineering, 48-55.
Schümmer, T., & Lukosch, S. (2008) Supporting the social practices of distributed pair programming. In Groupware: Design, Implementation, and Use, volume 5411 of Lecture Notes in Computer Science, 83-98.
Tamada, H., Igaki, H., Hikichi, K., Monden, A.,& Matsumoto, K. (2005). An Analysis of Group Development Processes in a Programming Class. The base of software engineering XII, Japan Society for Software and Technology FOSE2005, 123-128.
Vandeventer, J., &Barbour, B. (2012). CodeWave: A Real-Time, Collaborative IDE for Enhanced Learning in Computer Science. In Proceedings of the 43rd ACM technical symposium on Computer Science Education (SIGCSE '12), ACM, 75-80.
Zhou, W., Simpson, E., &Domizi, D. P. (2012). Google Docs in an Out-of-ClassCollaborative Writing Activity.International Journal of Teaching Learning in Higher Education, 24, 359-375.

Yuya Kato received the bachelor degree in Informatics from University of Shizuoka (Japan) in 2014. He is currently a graduate student of Informatics. His research interests include promoting collaborative programming. Main areas of expertise and interest are supporting collaborative programming in introductory programming courses and analysis of interaction.

Yoshiaki Matsuzawa received his PhD in Media and Governance in 2008 from Keio University, Japan. He is an assistant professor in the Faculty of Informatics at Shizuoka University. His research interests include software-engineering education, educational software development, modeling and simulation of complex systems, and learning sciences.

Sanshiro Sakai received his PhD in engineering in 1984 from Shizuoka University, Japan. He is a professor in the Faculty of Informatics at Shizuoka University. His research interests include computer supported collaborative learning, software development environment and programming learning support system.

Affective and Cognitive Correlates of Cell-phone Based SMS Delivery of Learning: Learner Autonomy, Learner Motivation and Learner Satisfaction

Yaacov J Katz, *yaacov.katz@biu.ac.il*
Michlala – Jerusalem Academic College and Bar-Ilan University, Israel

Abstract

Cell-phone based SMS delivery platforms are gradually being introduced into the educational system, especially at the tertiary level, to enhance academic learning and achievement. Cell-phone based SMS delivery of learning provides students with access to learning materials without being limited by space or time.

The present study is yet another in a series of research studies designed to examine the relationship between cell-phone based SMS delivery and cognitive and affective aspects of learning at the university level. In the present study two groups of first year university students who studied academic English vocabulary for the social sciences in a mandatory14 week long (semester) foundation course were exposed to two different modes of delivery of the vocabulary definitions. The first group of students received weekly lists of vocabulary definitions sent via cell-phone based SMS delivery to their cell-phones and the second group received weekly lists of vocabulary definitions sent by hardcopy snail mail messages to their residential postal addresses. The definitions of the vocabulary definitions studied by both delivery platforms were identical and the students received 30 vocabulary definitions on a weekly basis for a period of 14 weeks. At the end of this period the students in the two groups were tested on a standardized academic English vocabulary achievement test and responded to a questionnaire that examined their levels of learner autonomy, learner motivation and learner satisfaction.

Results of the study indicate no significant differences between students in the cell-phone based SMS delivery and snail mail hardcopy delivery groups on the standardized achievement test. However, there were significant differences between the students in the two delivery groups regarding learner autonomy, learner motivation and learner satisfaction. Students who received their vocabulary definitions via cell-phone based SMS delivery were characterized by significantly higher levels of learner autonomy, learner motivation and learner satisfaction than their counterparts who received vocabulary definitions via snail mail hardcopy delivery. It appears that cell-phone based SMS delivery is positively related to learner autonomy, learner motivation and learner satisfaction which conceivably have a positive relationship with the learning experience. Thus it is suggested that cell-phone based SMS delivery be seriously considered as a viable technological mobile delivery platform in the university learning process.

Keywords

Cell-phone based SMS Delivery, Snail Mail Hardcopy Delivery, Academic Achievement, Learner Autonomy, Learner Motivation, Learner Satisfaction

DISTANCE LEARNING

Distance learning has become an increasingly common solution to university campus overcrowding and student requirements for flexible schedules.According to McKee (2010) distance learning systems are perceived by students as being user-friendly, efficient, flexible and convenient.In addition to the above reasons for adopting distance learning, another major advantage of the introduction of distance learning at the university level is that students may potentially benefit from increased analytical thinking, enhanced learning autonomy and problem solving skills (Katz and Yablon, 2011). Fujioka-Ito (2013) confirmed that distance learning at the university level supplements and enhances traditional classroom-based learning because students are necessarily more active than in face-to-face lectures and express more satisfaction with a learning process that incorporates distance learning.

Ismail et al (2010) confronted the implications of university learning and instruction using distance learning courses. They contended that distance learning has moved formal instruction in these courses from the on-site setting of the university campus to the home of the student. Learning has become significantly more flexible and content sources more accessible. Creating, sharing and knowledge capitalization are all facilitated by distance learning. Wider sources of learning are provided in distance learning courses and worldwide expertise can systematically be brought to the student's desktop.

With the rapid development of distance learning courses for use in university level education, increasingly more research studies have been conducted in an attempt to evaluate different issues related to distance learning. For example Chandra and Watters (2012) indicated that learning physics through the medium of distance learning not only enhanced students' learning outcomes, but also had a positive impact on their attitudes toward the study of physics. Ituma (2011) confirmed that a large percentage of university students who were enrolled in distance learning university courses had positive perceptions of the technology based learning methodology and were in favor of joining additional distance learning courses that supplemented traditional face-to-face classroom instruction. Valaitis et al (2005) found that students who participated in distance learning courses perceived that the methodology increased their learning flexibility and enhanced their ability to process content, and provided access to valuable learning resources. Abdallah (2009) found that distance learning courses contributed to improved quality of students' learning experiences. Students reported positive attitudes toward their distance learning and felt that such learning should be part and parcel of standard learning practice. Delfino et al (2010) confirmed that student teachers who participated in technology based distance learning teacher training courses developed self-regulation of learning which provided them with the opportunity to flexibly cope with their academic assignments.

In summary it may be stated that recent research studies indicate that distance learning can enhance students' learning experiences, can promote students' learning flexibility, can promoter positive attitudes of students toward the learning process and can increase learning efficiency.

MOBILE LEARNING

Mobile learning, a specific aspect of distance learning, offers a learning environment that is especially characterized by flexibility offered to the learner (Katz and Yablon, 2009). Al-Fahad (2009) indicated that mobile learning enhanced Saudi Arabian university students' learning experiences. He found that the most significant advantage of mobile learning is that it can be used anywhere, anytime and promotes learning flexibility. According to Premadasa and Meegama (2013) mobile learning provides more flexibility, mobility, convenience and seamless integration of data access for students than more traditional as well as onlinelearningenvironments. Ducate and Lomicka (2013) reported that mobile learning, which offersstudents unlimited access to resources, is an effective platform for delivery of learning. Yang (2013) confirmed that mobile learning contributes significantly towards a more comprehensive educational environment for learning. Keeand Samsudin (2014) found that learners held positive attitudes towards the use of mobile devices when learning. Learners felt that mobile learning is convenient and easy to use in order to access knowledge. Rui-Ting et al (2014) emphatically stated that as mobile learning has become a key learning channel in the educational system it is critical that researchers as well as practitioners concentrate on developing mobile technology for learning.

CELL-PHONE ASSISTED LANGUAGE LEARNING

One of the latest developments within mobile learning has been the introduction of the use of the cell-phone as a delivery system (Attewell and Savill-Smith, 2004; Prensky, 2005). It should be noted that the use of cell-phones is multi-dimensional and cell phone technology now provides technological possibilities including voice, text, still-camera, video-camera, paging and geo-positioning capabilities. These tools provide a rich variety of platforms that potentially enhance the learning process. Moreover, cell-phone based learning is not bound by space or time and students can choose to engage in learning without almost any limitations (Hassan et al, 2012).

Mobile assisted language learning (MALL) is fast developing as a legitimate learning delivery platform (Godwin-Jones, 2010). As mobile technologies which include hand-held PDAs, tablets, palm computers and cell-phones have evolved technologically, mobile learning is increasingly being used as a delivery platform for language learning (Kukulska-Hulme and Shield, 2008). Godwin-Jones (2011) indicated the effectiveness of cell-phones for language learning and Elwood and McLean (2009) showed that a majority of Cambodian students preferred to receive study materials on cell-phones rather than on personal computers. Stockwell (2008) found that that more than two-thirds of the Japanese learners who participated in his study were interested in using cell-phones for language learning. Kennedy and Levy (2008), Lu (2008), Motallebzadeh and Ganjali (2011) and Zhang et al (2011) confirmed the effectiveness of the use of cell-phone based SMS text messages for language and vocabulary learning. Alemi et al (2012) indicated that cell-phone based SMS delivery of vocabulary had a more significant effect on students' vocabulary retention than traditional instruction based on the use of dictionaries. Hayati et al (2013) reported that students who studied language via cell-phone based SMS delivery, were more enthusiastic about their learning of language idioms and attained higher scores on an achievement test than their counterparts who studied idioms by traditional teaching methods. Chen (2014) confirmed that students who received English vocabulary lessons via cell-phone-based SMS delivery performed significantly better on an achievement test than students who received their vocabulary lessons by way of traditional face-to-face instruction.

AFFECTIVE ATTITUDINAL VARIABLES AND LANGUAGE LEARNING

Learner attribution, learner autonomy, learner control of the learning process, learner creativity, learner curiosity, learning flexibility, learner motivation, learner satisfaction, learner self-confidence, learner self-efficacy, learner self-image, and learner self-esteem are a few some of the major variables known to positively contribute to enhanced language learning delivered via technology based delivery platforms. Cavus and Ibrahim (2009), Katz and Yablon (2009) as well as Katz (2013a; 2013b; 2013c; 2014a; 2014b; 2014c) confirmed the positive association of some or all of the above factors with effective technology based delivery of language learning.

Learner autonomy is considered to be a critically important factor found to enhance effective language learning (Blin, 2004). Shahsavari (2014) reported that results of a research study on learner autonomy and technology based language learning indicated that nearly all the teachers and learners surveyed in the study agreed that learner autonomy is related to more efficient learning. Granic et al (2009) confirmed that learner autonomy is a contributing factor to effective language learning delivered via cell-phone based SMS messaging. Thus an assumption of the present study is that the cell-phone based SMS delivery platform of vocabulary learning facilitates learner autonomy in vocabulary learning.

Learner motivation is also considered to be a vital factor in language learning (Dornyei, 2003). In a comprehensive meta-analysis study on the relationship between learner motivation and study of language, Masgoret and Gardner (2003) indicated that learner motivation is a major factor known to enhance successful language learning. Kiernan and Aizawa (2004), as well as Chinnery (2006), confirmed that cell-phone technology promotes learner motivation in the study of languages.Thus this study will examine the assumption that a cell-phone based SMS delivery platform promotes learner motivation in vocabulary learning.

Learner satisfaction with the learning process is perceived to be an important factor that leads to efficient and effective language learning. Shahriar et al (2011) indicated that the extent of Pakistani students' satisfaction with studying the English language correlated significantly with their study motivation. Chang and Smith (2008) reported that learner satisfaction is an especially important component that contributes to successful learning in distance learning courses. Hoven and Palalas (2011) found high levels of satisfaction among students who studied language via a cell-phone based SMS delivery platform. Mao (2014) reported that in a survey of Chinese students, overall satisfaction with cell-phone based mobile learning was high and that most of the students who participated in the survey declared their inclinations to increase their future utilization of mobile learning for the study of language and other subjects. This research study will examine the hypothesis that a cell-phone based SMS delivery platform will be related to students' satisfaction with vocabulary learning.

METHOD

Sample

The research sample consisted of 116 first year students enrolled in a 14 week semester-long mandatory academic vocabulary foundation course offered at one of the seven chartered universities in Israel. Students came from similar socio-economic backgrounds and all were accepted to the Faculty of Social Sciences at the university after attaining the university acceptance criteria of a mean matriculation grade of at least 80% as well as a mean psychometric university entrance score of 600. The students were randomly assigned to two comparison groups in which they were provided with lists of academic vocabulary definitions via two alternative learning content delivery methods. The first group of 54 students received their vocabulary definitions via a cell-phone based SMS delivery platform to their personal cell-phones and the second group of 62 students received their vocabulary definitions in hardcopy format by way of regular snail mail.

Instruments

Two research questionnaires were administered to the students in this research study. A standardized academic English vocabulary for social sciences achievement test was administered to the participants in order to assess students’ mastery of academic English vocabulary definitions related to their studies in the Faculty of Social Sciences. The test scale ranged from 0-100, the higher grades indicating higher levels of achievement on the test. The second instrument administered to the participants was a 30 item Likert type scale response questionnaire (students responded to a five point scale with 1=totally disagree and 5=totally agree) designed to examine participants' levels of learner autonomy, learner motivation and learner satisfaction. The first factor, learner autonomy, contained ten items (Cronbach α=0.86), the second factor, learner motivation, consisted of ten items (Cronbach α=0:84) and the third factor, learner satisfaction, was made up of ten items (Cronbach α=0.81).

Procedure

Students who studied in the Faculty of Social Sciences, and were enrolled in the mandatory academic English vocabulary foundations course and possessed personal cell-phones with texting and internet capacity were eligible for participation in this study. Following the selection of the students who met the above criteria, they were randomly assigned to the two delivery platform groups. Students in the first group received vocabulary definitions by way of a cell-phone based SMS delivery platform sent to their personal cell-phones. Students in the second group received identical vocabulary definitions in traditional hardcopy format via snail mail sent to their personal residential addresses. The students in the two groups were sent weekly lists that contained concise definitions of the academic English vocabulary studied in the course with each weekly list consisting of 30 new vocabulary definitions delivered via the two respective learning delivery platforms. Thus each of the students received 420 academic English vocabulary definitions during the 14 week long course. On completion of the course the students in the two comparison groups were administered a standardized academic English vocabulary for social sciences achievement test in order to assess their level of knowledge of the 420 vocabulary definitions taught in the course. In addition they were administered the 30-item attitudinal questionnaire which examined their scores on the three attitudinal research factors, namely learner autonomy, learner motivation and learner satisfaction.

RESULTS

The main aim of this study was to examine the efficiency and effectiveness of two different learning delivery platforms, namely cell-phone based SMS delivery and traditional hardcopy delivery. Two research questions were posed in the study: the first examined the acquisition of academic English vocabulary definitions by students and the second investigated students' attitudes related to the two delivery platforms. The mean scores of each of the attitudinal variables were standardized in order to allow for a comparison between the factor scores. Standardized means and standard deviations of students' scores on the achievement test and on the attitudinal factors are presented below.

Factor	Cell-phone based SMS Delivery(N=54)		Snail Mail Hardcopy Delivery (N=62)	
Academic Achievement	M=87.13	SD=8.64	M=86.97	SD=8.19
Learner Autonomy	M=3.85	SD=0.85	M=3.19	SD=0.96
Learner Motivation	M=2.94	SD=1.03	M=2.58	SD=0.80
Learner Satisfaction	M=4.37	SD=0.56	M=3.60	SD=0.78

Table 4: Mean Scores and Standard Deviations of Students in Cell-Phone SMS and Snail Mail Hardcopy Delivery Groups for Learner Achievement, Learner Autonomy, Learner Motivation and Learner Satisfaction

One-way analyses of variance (ANOVA) were conducted in order to investigate intergroup differences on the four research variables. No significant differences were found between students in the cell-phone based SMS and snail mail hardcopy delivery groups for grades attained on the standardized academic English vocabulary achievement test. On the other hand significant differences between the two delivery groups were yielded for learner autonomy [$F(1,114)=14.87$, $p<0.001$, $\eta^2=11.2\%$]; for learner motivation [$F(1,114)=4.44$, $p<0.05$, $\eta^2=1.8\%$]; and for learner satisfaction [$F(1,114)=35.50$, $p<0.001$, $\eta^2=16\%$]. Results of the one-way ANOVA procedures indicated that students in the cell-phone based SMS delivery group attained significantly higher scores on all three attitudinal variables, namely learner autonomy, learner motivation and learner satisfaction than students in the snail mail hardcopy delivery group

DISCUSSION

The research findings of this study indicate that neither cell-phone based SMS delivery nor snail mail hardcopy delivery held any advantage regarding academic achievement on the standardized academic English vocabulary definitions achievement test. Students who studied the 420 vocabulary definitions via cell-phone based SMS delivery sent to their personal cell phones or delivered to them by hardcopy via snail mail attained similar grades on the standardized learner achievement test. Thus it seems that different learning delivery platforms do not necessarily lead to differential academic achievement. Although this result is in direct contradiction to evidence presented by Efendioglu (2012) and Guzeller (2012), it confirms similar results which have indicated that academic achievement is not conditional to type of learning delivery platforms used to facilitate the learning process (Bohlen and Ferratt, 1993; Dyer and Osborne, 1996; Katz, 2013a; 2013b; 2013c; 2014a; 2014b; 2014c; Katz and Yablon, 2009; 2011; 2012).

Additional research results clearly indicate that the two different learning delivery platforms employed in the present study are related to significantly differential levels of learner autonomy, learner motivation and learner satisfaction. Scores attained by students in the cell-phone based SMS delivery group on the attitudinal factors were significantly higher than those of students in the snail mail hardcopy delivery group. It appears from the nature of these results that cell-phone based SMS delivery of learning is linked to higher levels of learner autonomy. This finding confirms similar indications regarding the unique advantages of cell-phone technology in enhancing learner autonomy as reported by researchers such as Granic et al (2009) and Shahsavari (2014). Similarly the research results that indicate that the cell-phone based SMS delivery platform appears to be more significantly related to motivation than delivery by hardcopy sent snail mail delivery confirm the findings of Kiernan and Aizawa (2004) and Chinnery (2006) who suggested that cell-phone learning technology is related to learner motivation. In addition the findings of the present study that cell-phone SMS delivery of learning is more significantly connected to learner satisfaction than snail mail hardcopy delivery strengthen the findings of Hoven and Palalas (2011) and Mao (2014) who indicated that cell-phone based learning leads to an enhancement of learner satisfaction.

CONCLUSION

The results of the present study indicate the potential of cell-phone based SMS delivery of learning as positive delivery platform significantly related to attitudinal variables such as learner autonomy, learner motivation and learner satisfaction. It should be noted that no relationship was found between the two delivery platforms and academic achievement related to the study of vocabulary definitions. Further studies need to be conducted so as to further explore the possible relationship between delivery SMS and other technology based learning content delivery platforms and academic achievement. However, even if no significant relationship is found between the delivery platforms and academic achievement, the fact that a significant relationship exists between the cell-phone SMS delivery platform and attitudinal variables such as learner autonomy, learner motivation and learner satisfaction as well other similar attitudinal variables as reported in earlier studies, indicates the potential cell-phone based SMS delivery has for the university learning process. In addition, attitudinal traits such as learner autonomy, learner motivation and learner satisfaction may, in the long run, positively contribute to a positive frame of mind of learners, which in itself could be conducive to academic achievement Thus cell-phone based SMS delivery of learning should be universally considered as a legitimate and positive learning delivery platform at the university level and should be given serious consideration as a viable alternative or additive to the more traditional learning delivery platforms widely used in tertiary education.

REFERENCES

Abdallah, S. (2009). Learning with online activities: What do students think about their experience? *International Journal of Web-Based Learning and Teaching Technologies, 4(2)*, 1-25.

Alemi, M., Sarab, M. R. A. and Lari, Z. (2012). Successful learning of academic word list via MALL (Mobile Assisted Language Learning). International Education Studies, 5(6), 99-109.

Al-Fahad, F.N. (2009). Students' attitudes and perceptions towards the effectiveness of mobile learning in King Saud University, Saudi Arabia. *TOJET, 8(2),* 111-119.

Attewell, J. and Savill-Smith, C. (2004). *Learning with mobile devices: Research and development*. London: Learning and Skills Development Agency.

Blin, F. (2004). CALL and the development of learner autonomy: towards an activity-theoretical perspective. *ReCALL, 16(2),* 377-95.

Bohlen, G.A. and Ferratt, T.W. (1993*).* The effect of learning style and method of instruction on the achievement, efficiency and satisfaction of end-users learning computer software. In M.R. Tanniru (Ed.), *Proceedings of the 1993 Conference on Computer Personnel Research (pp. 273-283). New York: Association* for Computing Machinery.

Cavus, N. & Ibrahim, D. (2009). M-Learning: An experiment in using SMS to support learning new English language words*. British Journal of Educational Technology, 40(1),* 78-91.

Chandra, V. and Watters, J.J. (2012). Re-thinking physics teaching with web-based learning. *Computers and Education, 58(1),* 631-640.

Chang, S.H.H. and Smith, R. A. (2008). Effectiveness of personal interaction in a learner centered paradigm distance education class based on student satisfaction. *Journal of Research on Technology in Education, 40(4),* 407-426.

Chen, C. (2014). Using concept mapping instruction in mobile phone to learning English vocabulary. Creative Education, 5(1), 4-6.

Chinnery, G. M. (2006). Emerging technologies going to the MALL: Mobile assisted language learning. *Language Learningand Technology, 10(1),* 9-16.

Delfino, M.; Dettori, G. and Persico, D. (2010). An online course fostering self-regulation of trainee teachers. *Psicothema, 22(2),* 299-305.

Dornyei, Z. (2003). Attitudes, orientations, and motivations in language learning: advances in theory, research, and applications. *Language Learning, 53(S1),* 3-32.

Ducate, L. and Lomicka, L. (2013). Going mobile: Language learning with an iPod touch in intermediate French and German classes. *Foreign Language Annals, 46(3),* 445-468.

Dyer, E.J. and Osborne, E. (1996). Effects of teaching approach on achievement of agricultural education students with varying learning styles. *Journal of Agricultural Education, 37(3),* 43-51.

Efendioglu, A. (2012). Courseware development model (CDM): The effects of CDM on primary school pre-service teachers' achievements and attitudes. *Computers and Education, 59(2),* 687-700.

Elwood, J. and MacLean, G. (2009). ICT Usage and Student Perceptions in Cambodia and Japan. International Journal of Emerging Technologies and Society, 7(2), 65 - 82.

Fujioka-Ito, N. (2013). Designing a curriculum for a distance learning class: An example of a first-year Japanese course. *Theory and Practice in Language Studies*, *3(10),* 1717-1725.

Godwin-Jones, R. (2010). Emerging technologies from memory palaces to spacing algorithms: approaches to second language vocabulary learning. *Language, Learningand Technology, 14(2),* 4-11.
Godwin-Jones, R. (2011). Emerging technologies: Mobile apps for language learning. *Language, Learningand Technology, 15(2),* 2-11.
Granic, A., Cukusic, M. and Walker, R. (2009), M-learning in a Europe-wide network of schools. *Educational Media International, 46(3),* 167-84.
Guzeller, C.O. (2012). The effect of web-based portfolio use on academic achievement and retention. *Asia Pacific Education Review*, *13(3),* 457-464.
Hassan, M., Hamdan, Z. and Al-Sadi, J. (2012). A new mobile learning content design process. *International Journal of Academic Research, 4(1),* 23-28.
Hayati, A., Jalilifar, A. and Mashhadi, A. (2013). Using Short Message Service (SMS) to teach English idioms to EFL students. *British Journal of Educational Technology, 44(1),* 66-81.
Hoven, D. and Palalas, A. (2011). (Re)conceptualizing design approaches for mobile language learning. *CALICO Journal, 28(3),* 699-720.
Ismail, I., Idrus, R.M. and Johari, S.S.M. (2010). Acceptance on mobile learning via SMS: a Rasch model analysis. *International Journal of Interactive Mobile Technologies, 4(2)*, 10-16.
Ituma, A. (2011). An evaluation of students' perceptions and engagement with e-learning components in a campus based university. *Active Learning in Higher Education, 12(1)*, 57-68.
Katz, Y.J. (2013a) University students' attitudes toward cell-phone based learning. *EURODL*, *(special issue on the best research papers presented at EDEN Conferences 2012)*, 92-105.
Katz, Y.J. (2013b) Concept learning via SMS delivery at the university level. In M.F. Paulsen and A. Szucs (Eds.) *The joy of learning*: *enhancing learning experience, improving learning quality - Proceedings of EDEN 2013* (pp. 125-134) .Budapest: European Distance Education Network.
Katz, Y.J. (2013c). SMS-based learning in tertiary education: achievement and attitudinal outcomes. In M.B. Nunes and M. McPherson (Eds.) *Proceedings of the international conference: IADIS e-Learning 2013* (pp. 118-125). Lisbon, Portugal: IADIS Press.
Katz, Y.J. (2014a) The historical relationship between affective variables and ICT based learning and instruction and achievement in the Israeli school system. In A. Tatnall and W. Davey (Eds.) *Reflections on the history of computers in education: early use of computers and teaching about computing in schools* (pp.324-338). Heidelberg: Springer-Verlag.
Katz Y.J. (2014b). Cognitive and affective aspects of SMS based learning at the university level. *Athens Journal of Education*, *1(3),* 259-269.
Katz, Y.J. (2014c). Mobile learning delivery via social networks: what platforms do first-year university students prefer? In A.M. Teixeira and A Szucs (Eds.) *Challenges for research into open and distance learning: doing things better - doing better things - Proceedings of EDEN 2014* (pp. 249-256). Budapest: European Distance Education Network.
Katz, Y.J. and Yablon, Y.B. (2009). Mobile learning: a major e-learning platform. In A. Szucs (Ed.). *New technology platforms for learning revisited* (pp. 121-126). Budapest: European Distance Education Network.
Katz, Y.J. and Yablon, Y.B. (2011). Affect and digital learning at the university level. *Campus Wide Information Systems*, *28(2),* 114-123.
Katz, Y.J. and Yablon, Y.B. (2012). Acquiring vocabulary at the university level: a comparison of three learning strategies. In F. Doyran, (Ed.). *Research on teacher education and training* (pp. 267-276). Athens: Athens Institute for Education and Research.

Kee, C.L. and Samsudin, Z. (2014). Mobile devices: Toys or learning tools for 21st century teenagers? *TOJET, 13(3),* 107-122.

Kennedy, C., and Levy, M. (2008). L'italiano al telefonino: Using SMS to support beginners' language learning. *ReCALL, 20(3),* 315-330.

Kiernan, P.J., and Aizawa, K. (2004). Cell phones in task based learning: Are cell phones useful language learning tools? *ReCALL, 16(1),* 71-84.

Kukulska-Hulme, A., and Shield, L. (2008). An overview of mobile assisted language learning: From content delivery to supported collaboration and interaction. *ReCALL, 20(03),* 271-289.

Lu, M. (2008). Effectiveness of vocabulary learning via mobile phone. *Journal of Computer Assisted Learning*, 24(6), 515-525.

Mao, C. (2014). Research on undergraduate students' usage satisfaction of mobile learning. Creative Education 5(8), 614-618.

Masgoret, A.M. and Gardner, R.C. (2003). Attitudes, motivation, and second language learning: A meta-analysis of studies conducted by Gardner and associates. *Language Learning*, 53(S1): 167-210.

McKee,T. (2010). Thirty years of distance education: Personal reflections, *International Review of Research in Open and Distance Learning,* 11(2), 100-108.

Motallebzadeh, K., and Ganjali, R. (2011). SMS: tool for L2 vocabulary retention and reading comprehension ability. *Journal of Language Teaching and Research, 2(5),* 1111-1115.

Premadasa, H.K.S. and Meegama, R.G.N. (2013). Mobile learning environment with short messaging service. Campus-Wide Information Systems, 30(2), 106-123.

Prensky, M. (2005). Listen to the natives. *Educational Leadership, 63(4),* 8-13.

Rui-Ting, H., Chia-Hua H.,Tang, T.W. and Tsung-Cheng, L.(2014). Exploring the moderating role of perceived flexibility advantages in mobile learning continuance intention (MLCI). International Review of Research in Open and Distance Learning, 15(3), 140-156.

Shahriar, A., Pathan, H., Mari, M.A. and Umrani, T. (2011). The extent of satisfaction on the key factors that affect learner motivation. *International Journal of Academic Research in Business and Social Sciences, 1(3),* 96-108.

Shahsavari, S.(2014). Efficiency, feasibility and desirability of learner autonomy based on teachers. and learners' point of views.*Theory and Practice in Language Studies 4(2),* 271-280.

Stockwell G. (2008) Investigating learner preparedness for and usage patterns of mobile learning. ReCALL, 20(3), 253-270.

Valaitis, R.K., Sword, W.A., Jones, B. and Hodges, A. (2005). Problem-based learning online: perceptions of health science students. *Advances in Health Sciences Education,10(3)*, 231-252.

Yang, J. (2013). Mobile assisted language learning: Review of the recent applications of emerging mobile technologies. English Language Teaching, 6(7), 19-25.

Zhang, H., Song, W., and Burston, J. (2011). Reexamining the effectiveness of vocabulary learning via mobile phones. *TOJET, 10(3),* 203-214.

Yaacov J Katz is Professor Emeritus at Bar-Ilan University and President of Michlala – Jerusalem Academic College. He served as Professor of Education at Bar-Ilan University and as Chief Pedagogic Officer of the Israel Ministry of Education. He specializes in attitudinal research on technology in education as well as in the study of religiosity and values. He has published numerous scientific papers and edited volumes on the above topics.

Modernization and integration of IT education in Ukraine to international and Europe educational environment: problems and perspectives

Tetiana Kovaliuk, tetyana.kovalyuk@gmail.com
National Technical University of Ukraine «Kiev Polytechnic Institute»

Olena Chaikovska, lena@knukim.edu.ua
Kyiv National University of Culture and Arts

Abstract

The article provides an overview of projects on the new list of IT subjects in curriculum of Ukrainian Universities in accordance to ISCE 2013 and the Law of Ukraine "On Higher Education". Based on analysis of the current state of IT industry and IT education in Ukraine, the new list of IT specializations in general subject "Information & Communication Systems" is proposed. The list proposed is consistent with international standards, principles of Bologna process and Europe framework of ICT competencies and qualifications.

Keywords

Computing Curricula, IT-education, IT graduates specializations, IT industry, professional IT standards, academic mobility

INTRODUCTION

Higher education reform is based on implementation of New Law "On Higher Education", development of Strategy of reforming of Higher Education in Ukraine by 2020 and Concept of Ukraine Education Development for the period 2015 – 2025 [5].

We would like to draw your attention to the Concept of Ukraine Education Development for the period 2015-2025 as it is one of the core sources in the higher educational reform. According to the project of the concept of Education Development in Ukraine for the period 2015 – 2025, there's a plan to move to the new list of specializations for all degrees in Higher Education Institutions as of 2016. This new list of specializations is planned to be as close to the International Standard Classification of Education as possible and number of specializations is planned to be limited to 60 – 80. The article provides overview of the list of specializations in Information and Communication Technology field. This list is based on the Computing Curricula 2001 – 2005 Recommendation. The list was discussed and defined during the meetings and round table discussions with representatives of the leading Ukrainian and International IT Companies, such as Epam Systems, Global Logic, Luxoft, Data Art. After multiple consultations, the list of IT specializations is recommended to the Ministry of Education and Science of Ukraine. This list is reviewed in the article.

Besides the project of the concept of Education Development sets objective to develop and implement within 2015 – 2017 the new generation of standards in Higher Education in accordance with the new amendments of the Law of Ukraine "On Higher Education". Currently Ukrainian Universities follow State Higher Education Standards on Computer Science, Computer Engineering and Program Engineering which were implemented during 2009 – 2011. These Standards are based on the documents of Computing Curricula 2001 – 2005, on the objective to gain deep fundamental knowledge in Mathematics and Physics and to obtain practical professional skills in operational and scientific establishments.

As for the higher educational reform itself, on the one hand the reform is welcomed and understood: higher education establishments are granted status of real autonomy, level of academic mobility is growing, and content of higher education is being updated in accordance with needs of the economy and integration of Ukraine with European educational and economic environment. On the other hand the process of change of list of qualifications and fields of education in accordance with International Standard Classification of Education (ISED-F-2013) is being debated a lot. It is particularly true of Information and Communication Technologies field. As of February 2015 few projects of national standard classification of education are proposed (please see addendum A). Unfortunately, majority of the projects are not consistent with current perception of strategy of IT education development which should be in line with perspectives of IT industry development and recommendation of international documents. That is why absence of Information and Communication Technologies field in some projects is unacceptable taking into account the current state of development of information society when the role of information and communication technologies is dominant.

DISCUSSION

Development indicators and personnel problems in IT Industry of Ukraine

IT is the biggest and most developed segment of innovative economy in Ukraine, which means actively growing high value added export channels. IT Industry in Ukraine is constantly moving forward and IT may be one of the most transparent areas of economy which does not require government subsidies and grants. Today Ukraine is the world's fourth country with the biggest number of certified IT specialists (after USA, India and Russia). Ukraine is in top 30 locations preferred for ordering software development. By the end of 2014 IT outsourcing world market exceeded 300 billion dollars. This includes 2 billion in Ukraine. According to the information of the State Agency on Electronic Regulation of Ukraine, there are 66,5 thousands entities of different forms of property in telecom and IT spheres [6]. 13% of average number of employees in Service Industries are ICT employees. Based on "Innovation Ukraine" project, it is planned to create 100 thousand new IT jobs by 2020. In case of successful completion of this project, total revenue of Ukraine from IT export to USA and EU will be more than 10 billion dollars. Despite the fact that IT job is the most prestigious in Ukraine, there's a deficit of qualified professionals. Prognosis for 2015 is deficit of 170 thousand IT specialists. Annual increase of 15,5 thousands is not able to cover the market demand [2].

Moreover, according to the project of the concept of Education Development in Ukraine for the period 2015 – 2025, there's a plan to harmonize professional education content and new professional standards which will be competence based, developed in collaboration with professional communities and associations of employers by 2018. Currently Standards for 5 professions have been developed in Ukraine:

- Information Systems Specialist
- Software development Specialist
- Project Manager in IT area
- Product Manager in IT area
- Information Recourses Specialist

Representatives of such IT Companies, Organizations and Universities as Epam Systems, Global Logic, Skyline Software, Ukrainian public organization «Council on Competitiveness Industry Information and Communication Technologies in Ukraine» and National Technical University of Ukraine «Kyiv Polytechnic Institute» contributed to development of the Professional Standards.

The Professional Standards will be one of the sources for development of the State Education Standards for IT qualifications in terms of defining typical professional tasks, special professional and instrumental competencies, basic knowledge and skills. Together with Computing Curricula CC2001/CC2005, Computer Science Curricula 2013 Recommendations of International IT Associations and Communities, such as ACM, IEEE, AIS, AITP, IT Professional Standards will contribute to bringing graduates' qualifications and competencies in compliance with employers' requirements.

Scope of IT Higher Education in Ukraine

Currently in Ukraine there are 148 universities with study programs in Informatics and Computer Science. This includes national technical, classic, pedagogic, liberal and other Universities. Annually 15,5 IT specialists graduate from these Universities and join Ukrainian IT market.

There's a process of harmonization of IT industry and IT education. Professional IT standards are developed including requirements of IT industry to IT education, new technologies of upgrading competency level of IT graduates in accordance with IT industry requirements. Annual number of requests for IT graduates from IT companies and enterprises demanding IT systems support is 10-15% more than number of IT graduates. In order to improve quality of education Universities collaborate with IT companies, arrange Internship programs, lead scientific and business projects of IT companies. IT companies help universities to open centers of competencies, research and production laboratories, student engineering bureaus, etc. For further development of IT education it is important to preserve achievements.

Specializations and programs in world IT education

Based on analysis of Top 400 Universities for Engineering and Technology 2014 – 2015 [7], 15% of them provide program «Mathematics and Computer Science», 63% – «Computer Science», 35% - «Computer Engineering», 27% - «Software Engineering», 29% - «Information Technologies», 25% - «Information Systems». In universities outside of Ukraine there are no IT specialization with other names, such as "Development and management of data base and informational networks", etc.

In Ukraine elimination or reorganizing such specializations as Computer Science, Computer Engineering, and Software Engineering could jeopardize quality of IT education, increase deficit of IT specialists and decrease nomenclature of IT graduates specializations.

One of success criteria of IT education in Ukraine is integration of Ukrainian Universities with international education system and joining the market of international education services, e.g. double master's degree programs. IT curriculum in Ukraine is consistent with globally recognized practice of IT education. This makes possible introduction of globally recognized double master's degree programs in IT area. Today IT is a transnational field. That's why realization of Academic Mobility Program in accordance with Law "On Higher Education" is Ukraine's strategic course towards European Higher Education area. Academic Mobility means that students have possibility to study in foreign educational or scientific establishment with the same specialization one or more semester during the academic year. This period will be counted as a period of study in the student's home University. As current IT education in Ukraine is harmonized with Europe and USA (this includes Computer Science, Computer Engineering, Software Engineering), students, postgraduates and professors successfully participate in the process of integration. If the courses of study mentioned above are eliminated or renamed, the Academic mobility would be impossible, Ukraine would be excluded from global space of higher IT education.

IT jobs in Ukraine and in the world

Europe ICT framework competencies harmonizes IT programs and IT qualifications in the world. Competencies are defined for 23 professions. Qualification level is defined for each competency. Some of them are formed during the study courses based on globally recognized IT educational programs (Computer Science, Computer Engineering, Software Engineering, Information Technology, and Information Systems). Based on analysis of such international standards as Standard Classification of Occupations (ISCO), German classification of occupations (Klassifikation der Berufe – KldB), European Qualifications Framework (EQF), IT educational programs at Universities train specialists for the following jobs [4]:

- Computer network and systems technicians
- Computer network professionals
- Database and network professionals
- Database and network professionals not elsewhere classified
- Database designers and administrators
- Information and communications technology installers and servicers
- Information and communications technology professionals
- Information and communications technology sales professionals
- Information and communications technology service managers
- Information and communications technology user support technicians
- Information technology trainers
- Software and applications developers and analysts
- Software and applications developers and analysts not elsewhere classified
- Software developers
- Systems administrators
- Systems analysts
- Web and multimedia developers
- Web technicians

Based on analysis of vacancies in such leading Ukrainian IT companies as EPAM, LuxSoft, Global Logic, graduates of such study programs as Computer Science, Computer Engineering, and Software Engineering meet qualification requirements of IT companies.

Harmonization of IT bachelor's degree in Ukraine with world practice

International documents Computing Curricula 2001/2013 developed by international IT societies such as Association for Computing Machinery (ACM), IEEE – Computer Society, Association for Information Systems (AIS) are methodological basis of IT education in Ukraine [1]. Content and direction of IT education is defined in these documents: Computer Science, Computer Engineering, Software Engineering, Information Technology, and Information Systems. Such courses of study are recommended for IT higher education in Europe, USA, Canada, etc.

IT education in Ukraine in terms of bachelor's degree is being harmonized with global practice in such areas as Computer Science, Computer Engineering, and Software Engineering in accordance to Courses of Study List 2006 and respective specializations in terms of master's degree. During 2009 – 2011 Ministry of Education and Science ratified industry standards of higher education in these areas [3]. These courses of study in Ukraine's IT Higher Education make Ukrainian IT graduates competitive at international IT labor market.

According to Computer Science Curricula 2013 Computer Science area includes the following sections: Discrete Structures, Algorithms & Theories of Multiplicity, Data Structures & Programming Languages, Information Systems, Artificial Intelligence &

Intelligent Decision Support Systems, Computer Graphics, Human – Computer Interaction, Operational Systems, Computer Architecture & Computer Networks, etc. According to CE2004 Computer Engineering course includes the following sections: Computer Architecture, Digital Logics, Chains & Signals, Digital Signals Processing, Electronics, Embedded Systems, Human – Computer Interaction, Computer Networks, Operational Systems Database & Programming Basics, etc.

Areas of knowledge and IT specializations in the new structure of higher education in Ukraine

Including accepted in Europe IT courses of study to domestic higher education system insures its succession and harmonization with Europe IT educational programs which is transparent to IT industry, open to qualification and advanced training in terms of Double Degree and Academic Mobility of students, postgraduates and professors. According to international Computing Curricula recommendation IT education in Ukraine should include the following specializations (please see Table 1).

General Subject	Knowledge Area	Specialization
06 Information & Communication Technologies (ICT)	061 Information & Communication Technologies (ICT)	0611 Computer Science 0612 Computer Engineering 0613 Software Engineering 0614 Information Systems & Technologies 0615 System Analysis 0616 Information Security

Table 1: Project on IT Education structure

One of the projects (Project E, Table 5, Addendum A) meets requirements of International Standard Classification of Education (ISCE 2013) the most. It preserves current education system with new titles, keeps scientific schools which have been developed in higher education for many years.

Specialization 0616 Information and Communication Systems Security is added to the list because of the following reasons. The Law of Ukraine “On National Security” states that “Information Security” is a part of national state security as well as military and economic security. Information Security Doctrine of Ukraine states that in condition of global integration and tough international competition, information space become the main ring where national interests of different countries are crossed. Present information technologies let countries to pursue their interests without military forces, weaken or significantly affect security of competitive country which doesn’t have effective security system preventing negative informational impact. Fundamental difference between ICT and other general subjects is the list of specializations.

Computer Science Specialization: object domain analysis, automation models & processes mathematical models, applied software & databases, software development tools, web technologies, distributed system technology and parallel computing, artificial intelligence systems, data processing methods, computer infrastructure, including data storage.

Computer Engineering Specialization: computer systems and networks hardware & software, system software, information security devices, testing and diagnostic systems, distributed and clustered computer systems, local, global andcorporate computer networks.

Software Engineering Specialization: development and testing applied software for computer and information systems, web portals, and multimedia systems.

Information & Communication Systems Specialization: development, modification and support of Information Systems (IS) for automating organizational management and business processes in organizations with different forms of property intended to improve efficiency of the organizations using ISs , and software development and testing proccess management.

System Analysis Specialization: analysis, design, decision-making in complex systems of various nature based on system methodology.

Information Security Specialization: software and hardware tools that providie safe data transmission in computer networks including multichannel telecommunications.

Negative impact of elimination of current IT education nomenclature

Today Ukrainian universities qualify programmers, testers, software developers, system analysts, database analysts, quality managers, IT project managers, business analysts, and information technologies developer. These specialists successfully master different IT areas. As a result of elimination of current IT courses of study, Ukraine would receive only graduates with qualification of Information Technologies user. Such a poor quality of bachelor's qualification would make impossible to qualify research engineers and scientists required for effective execution of innovative tasks, producing new ideas, solving complex problems in professional and\or research and innovative activities. It would also make impossible gaining methodology of scientific and pedagogical practice and making independent research and studies with innovative, scientific, theoretical and practical results.

Poor quality of IT graduates qualification in specializations which are not recognized in the world would result in leaving Ukrainian market by IT outsourcing companies which partner with foreign customers, increase of brain drain among IT professionals, increase of demand for IT education abroad in specializations recognized in the world.

CONCLUSION

Computer Science, Programming & Computer Engineering Education process in Ukraine is consistent with international recommendations developed by Association for Computing Machinery (ACM), IEEE – Computer Society, Association for Information Systems (AIS).

Based on analysis of companies' demands for IT specialists, graduates in Computer Science, Computer Engineering, Software Engineering meet qualification requirements the most.

Elimination or transformation of the current IT courses of study could lead to significant deficit of specialists at IT labor market and eventually could make it impossible for IT industry to become the driving force behind the economic recovery of Ukraine.

The matter of harmonization of IT industry and IT education based on realization of effective mechanisms in social partnership is a main focus with a long term perspective. The Ministry of Education and Science of Ukraine pays special attention to this matter, e.g. there's a plan to develop Educational Engineering Centers in Universities and Companies, to align scope and content of educational programs on some professional disciplines with employers, etc.

REFERENCES

1. ACM/IEEE-CS Joint Task Force for Computer Curricula 2013. Computer Science Curricula 2013 Curriculum Guidelines for Undergraduate Degree Programs in Computer Science. (2013, December). http://dx.doi.org/10.1145/2534860

2. *Bulletin of the labor market. State Employment Service of Ukraine.*. Retrieved from *www.dcz.gov.ua/kie*
3. Industry Standard of Higher Education of Ukraine training direction 6.050101 "Computer Science". (2011). Kyiv, K: BHV.
4. ISCO-08 Group definitions - Final draft (Excel). Retrieved from http://www.ilo.org/public/english/bureau/stat/isco/isco08/
5. Law of Ukraine "On Higher Education" № 1556-VII (2014, July). Retrieved from http://zakon1.rada.gov.ua/laws/show/1556-18
6. The development of the software market. Retrieved from http://www.e.gov.ua/content/rozvytok-rynku-programnogo-zabezpechennya
7. World University Rankings 2014-2015. Retrieved from http://www.timeshighereducation.co.uk/world-university-rankings/2014-15/world-ranking

Tetiana Kovaliuk, Ph.D., associate professor of Department of Automated Systems of Information Processing and Management Department of National Technical University of Ukraine «Kiev Polytechnic Institute», Scientific secretary of NMC for the area of knowledges "Informatics and computer science", Director of the Ukrainian-Korean educational center of information technologies in NTUU „KPI”, Vice-Chairman of the Committee of education matters of the Ukrainian Federation of Informatics, Laureate of State Prize of Ukraine in Science and Technology, was awarded an "Excellence in Education of Ukraine”

Olena Chaikovska, Head of the Department of Computer Science of the Kyiv National University of Culture and Arts, Ph. D., Associate Professor, the Head of ‘IT in educational and cultural environment’ Work Group of the Committee of education matters of the Ukrainian Federation of Informatics, the member of organizing committee of the International IT Olympiad for students “IT Universe”, the jury member of the All-Ukrainian Contest-defend of scientific research works of the students-members of Minor Academy of Science of Ukraine (“Multimedia systems, educational and game programs” section), the member of Commonwealth of IT directors of Ukraine.

Addendum A.

General Subject	Area of Knowledge	Specialization
06 Information & Communication Technologies (ICT)	061 Information & Communication Technologies (ICT)	0612 Database & Information Networks Development and Management 0613 Software Development 0614 System Analysis and Information Technologies

Table 1. Project on IT Education structure (A)

General Subject	Area of Knowledge	Specialization
06 Information & Communication Technologies (ICT)	061 Information & Communication Technologies (ICT)	0613 System Science & Cybernetics 0614 Informatics, Computer Science & Automatics

Table 2. Project on IT Education structure (B)

General Subject	Area of Knowledge	Specialization
06 Information & Communication Technologies (ICT)	061 Information & Communication Technologies (ICT)	0612 Database and Computer Networks Design & Administration 0613 Software Development & Analysis 0614 Electronics & Automation 0615 System Analysis
	062 Informatics	0616 Informatics

Table 3. Project on IT Education structure (C)

General Subject	Area of Knowledge	Specialization
06 Information & Communication Technologies (ICT)	061 Information & Communication Technologies (ICT)	013.1 Informatics 013.2 Applied Mathematics 013.3 System Analysis
		014.1 Computer Science 014.2 Computer Engineering 014.3 Software Engineering
		015.1 Universal Information & Communication Systems 015.2 State Information & Communication Systems
		016.1 Universal Information & Communication Systems Security 016.2 State Information & Communication Systems Security

Table 4. Project on IT Education structure (D)

General Subject	Area of Knowledge	Specialization
06 Information & Communication Technologies (ICT)	061 Information & Communication Technologies (ICT)	0611 Computer Science 0612 Computer Engineering 0613 Software Engineering 0614 Information Systems & Technologies 0615 System Analysis 0616 Information Security

Table 5. Project on IT Education structure (E)

General Subject	Area of Knowledge	Specialization
Natural Science	Physics & Mathematics Science	Mathematics Physics Informatics Cybernetics & System Analysis Mechanics Astronomy
Applied Science	Technics & Technologies	General Engineering Science Electrical Engineering & Technologies Electrical Mechanics Radio Engineering & Telecommunications Computer Engineering Program Software Engineering System Engineering Automation & Computer Integrating Technologies Micro & Nano Electronics Electronics Heat & Power Engineering Nuclear Physicality Hydraulic Power Engineering Power Machine Building Mining And others

Table 6. Project on IT Education structure

What do we expect from graduates in CS? First results of a survey at university and company as part of a methodology for developing a competence model

Kathrin Müller[5], *kathyb@upb.de*
University of Paderborn, Germany

Abstract

This paper presents an empirically oriented approach to the development of a Computer Science Competence Model for the needs of curriculum development and evaluation in Higher Education. The main focus of this paper is a survey, which is part of the methodology. The survey is on expectations on fresh CS bachelor graduates and addresses university and company members. They should be involved in the education of CS graduates or who work with fresh CS bachelor graduates. The results of the survey will be used to build the competence model and an assessment.

Keywords

competence measurement, competence model, expectations on graduates, assessment

INTRODUCTION

The central demand of the Bologna reform is the outcome-oriented implementation of study courses, which include competence-oriented descriptions of the curricula and respective competence-oriented assessment procedures. These demands are often implemented only in a superficial way. To accomplish the demands of Bologna, it is not enough to adapt the curricula. Developing a competence-oriented course of studies is a process, which consists of several activities. According to Schaper(2012) these are e.g.: developing a competence-oriented curriculum, approaches to develop a competence-oriented organization of teaching and learning, methods for a competence-oriented assessment and also approaches for a course related advancement of competencies.

The base for most of these activities is a competence model. It is needed for the definition of learning outcomes on different levels of granularity of a study course like modules, lectures or seminars. Additionally for the needs of an assessment and in order to identify affirmative action a concept of competence measurement is necessary. During the development process of a competence model, it is important to ask the question what do we expect from our graduates, additionally what do the companies expect from the freshmen. The reason for the integration of the companies' opinion is that the task of universities is not only the scientific education. A vocational training is also expected because a bachelor or master degree is also an associate degree.

This paper describes a methodology for building a competence model by using a curricula analysis and a study with participants from university and company. The main focus is the study and the first results of it.

[5]née Bröker

BACKGROUND AND RELATED WORK

Many of the existing national and international projects, especially in computer science, focus on the competencies in schools, for future teachers, key competencies or on ICT key competencies in different subjects (UNESCO, 2012), (Denning, 2003). That implies that their main focus often is not on technical competence in CS. There are only a few projects with the aim to develop subject specific competence models with the focus on the technical competencies. The German projects MoKoM (Linck et. al., 2013) and KUI (Berges et. al., 2013) are two of these. The MoKoM project has its focus on developing a competence model for informatics modeling and system comprehension for students at schools. In contrast to that, the project KUI focuses on competencies for future teachers. The KUI competence model is divided into three parts: competencies on the subject matter knowledge (CK), competencies on pedagogical content knowledge (PCK) and non-cognitive competencies (NCC) (Hubwieser, Magenheim, Mühling, and Ruf, 2013), (Schaper et. al., 2013).In addition, there are some projects with research on students' competencies at universities. Nevertheless until now, no national or international project has developed a concrete competence model for subject-specific technical competencies in computer science for academics. The only existing models are, for example, the IEEE Software Engineering Body of Knowledge (SWEBOK)(Society, Bourque, and Dupuis, 2004), the ACM/IEEE Curriculum (The Joint Task Force on Computing Curricula Association for Computing Machinery IEEE-Computer Society 2012, 2013)or the IEEE Software Engineering Competency Model (SWECOM) (IEEE Computer Society, 2014). Nevertheless, the first two documents rather describe the knowledge, which should be part of a curriculum than real competencies. The SWECOM describes competencies but for software engineers who participate in the development of and modifications to software-intensive systems not for academics directly after their study. Unfortunately, these documents are additionally not empirically verified and often developed by people from the University. Consequently, we don't know if universities teach the topics mentioned in these curricula/competence models and if these are also the topics the companies expect from graduates. Only accreditation rules give us a first hint. Additionally, they are not as specific as the ACM/IEEE Curriculum. However, these documents are a good basis for the developing of a concrete competence model. In addition to the missing competence models, there is no assessment of the competencies by now.

DEVELOPING A COMPETENCE MODEL

We use an empirically oriented approach using a content analysis as a first step. The result should be a competence model for the area of Software Development, Software Engineering and Programming. The reason for this decision is that CS is a large field and for a Ph.D. project we had to focus. The applied research methodology (see Figure 1) consist of the following steps (Bröker, Kastens & Magenheim, 2014):

Content analysis of existing CS curricula of selected Universities on the basis of the topic categories provided by the ACM/IEEE recommendations (ACM/IEEE, 2013).

Conducting a survey in order to find out the expectations on graduates in the identified topic areas

Use the results of the survey with the results of the curricula analysis to gain a consolidated competence model in the topic area (Also influenced by the Anderson Krathwohl Taxonomy (AKT) (Krathwohl, 2002)).

Additionally the development of a test instrumentto measure students' competencies.

The competence definition used in this work is the definition of Weinert (2001) (original in German, translation by the author). Competencies are "the existence of learnable cognitive abilities and skills which are needed for problem solving as well as the associated motivational, volitional and social capabilities and skills which are needed for successful and responsible problem solving in variable situations."(Weinert, 2002, p.27f).

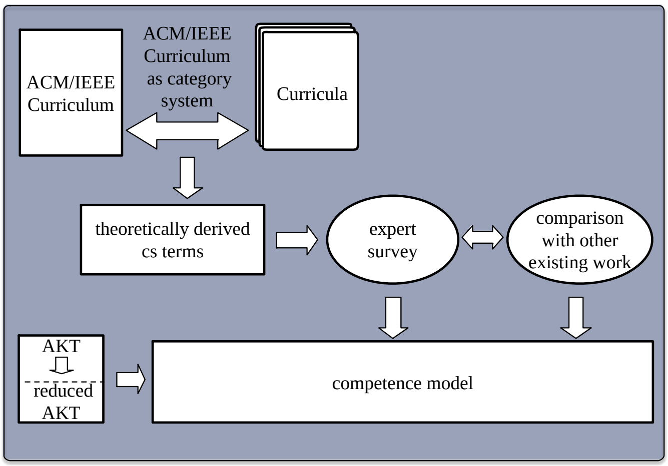


Figure 1: Developing the Competence Model (Bröker, Kastens & Magenheim, 2014)

Curricula Analysis

To develop a Competence Model by means of content analysis, we adapted the strategy used in previous projects like MoKoM and KUI. Figure 1 provides an overview of this process.

Firstly we analyzed several Computer Science Bachelor Curricula of universities from all over the world applying the method of deductive content analysis according to Mayring (2010). Here we start with the ACM/IEEE Curriculum (ACM/IEEE, 2013) in order to firstly derive classification categories. For practical coding, we used the software MaxQDA (www.maxqda.com). In the ACM/IEEE Curriculum, the basis of our analysis experts defined a catalog of 18 knowledge areas to cover computer science. Each knowledge area consists of several different knowledge units. For each of these knowledge units, a description of the content and the learning outcomes is shown. In addition, it is defined how many hours Universities should take for each knowledge unit during their computer science program. In short, the importance of each knowledge unit is shown.

We used these knowledge areas and knowledge units as our categories and subcategories for the content analysis. As the text corpus served different computer science bachelor curricula: The top ten of the QA Ranking[6] and, in addition, the top Universities of Switzerland, Singapore, India and eight of the Top Ten of German Universities in Computer Science[7]. We decided to use these additional Universities because we want to gain an overview of different countries. The Top Ten of the QA Ranking contains mostly Universities from the US. Because curricula often have obligatory and non-obligatory courses, we decided to analyze only the obligatory courses. Otherwise, there are too many variants. Additionally, our observations at our Learning Center and our Institute, indicate that students' problems often begin early

[6]http://www.topuniversities.com/university-rankings/university-subject-rankings/2013/computer-science-and-information-systems#sorting=rank+region=+country=+faculty=+stars=false+search= (last access: 05.02.1015)

[7]http://www.wiwo.de/ranking-die-besten-unis-und-fachhochschulen/8046582.html (last access: 05.02.2015)

in the basic courses. Thus, if we want to measure competencies according these courses, a look at obligatory subjects is sufficient. After building the category system and the text corpus, we defined the minimal (a single word/subject term) and maximal (paragraph) coding units. During the following analysis, the coder assigned the identified coding units to the appropriate categories of the ACM/IEEE curriculum. It was possible to map all code units to our category system derived from the ACM/IEEE curriculum, so there was no need to add other categories to the system during our analysis.

As a result of our analysis, we got the 21 categories are shown in Figure 2, which occur in minimum 50 % of the curricula (Bröker, Kastens & Magenheim, 2014). The coded aspects of the categories, which are belonging to the area of software development, are used for the next step, the empirical study at company and university.

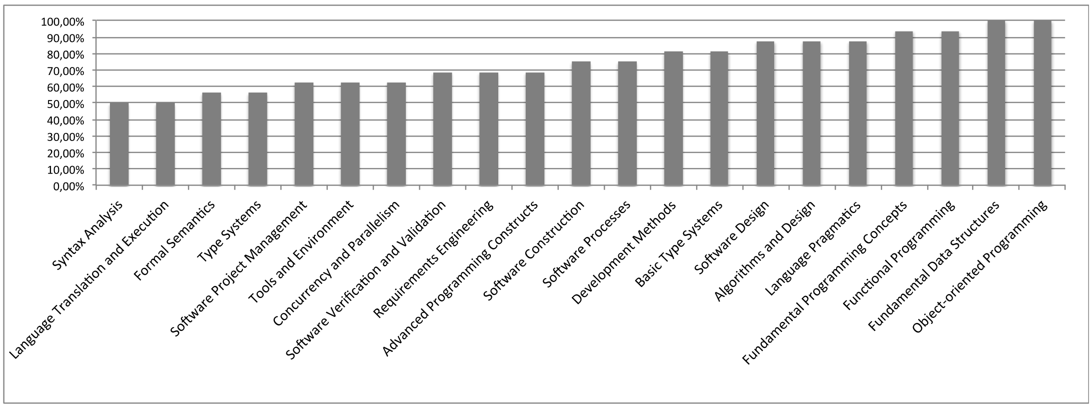


Figure 2: Results oft he Curricula Analysis: Categories that occur in minimum 50% of the curricula (Bröker, Kastens & Magenheim, 2014).

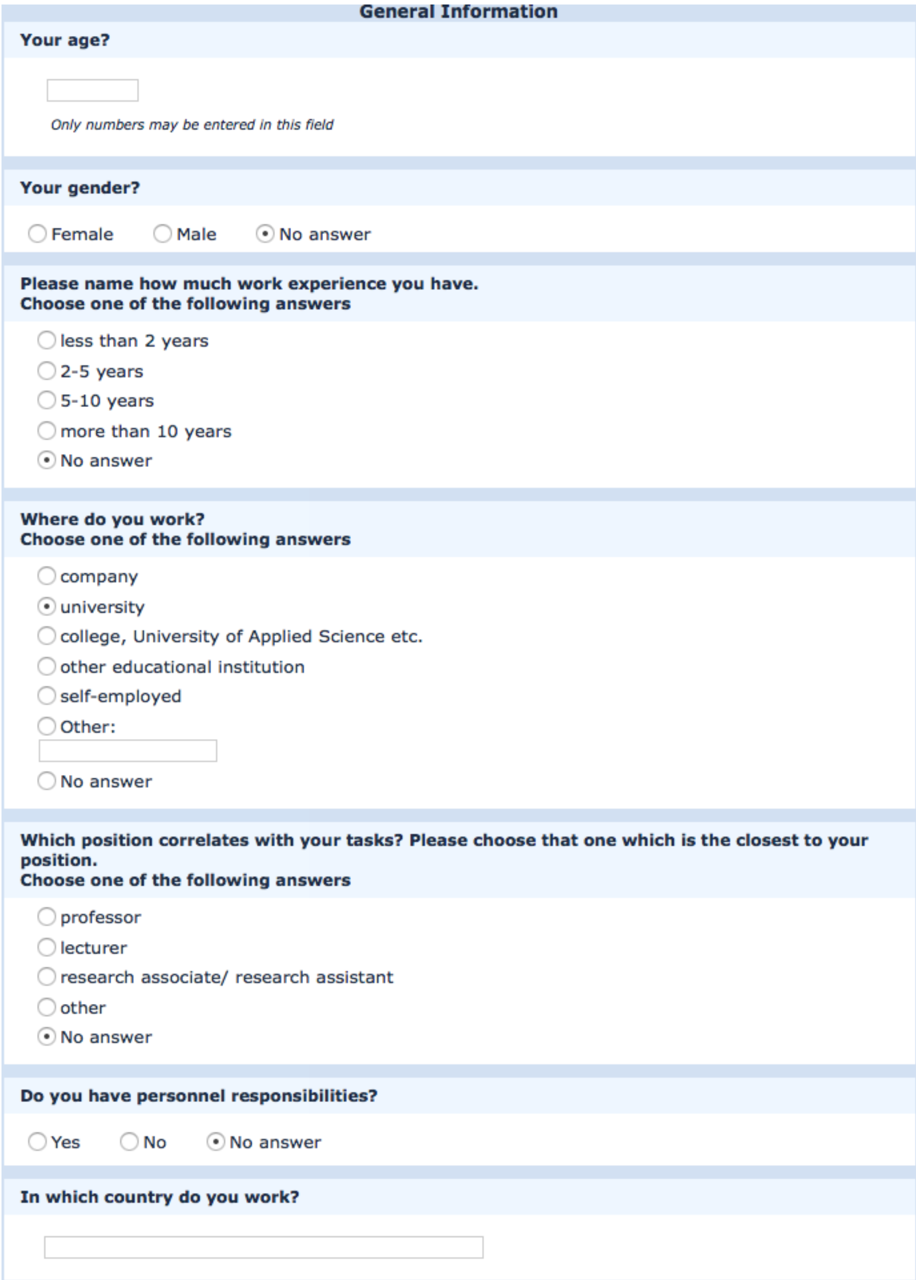
Study: What should graduates with a BSc. in CS know about software development?

On the basis of the curricula analysis, we developed a survey. This survey is the empirical component of our methodology for developing a competence model for CS bachelor students.Because of the wealth of data we decided to concentrate on those categories, which are belonging to the area of software development. Hence, the following categories are used for further steps:

- Fundamental Data Structures
- Fundamental Programming Concepts
- Algorithms and Design
- Software Design
- Development Methods
- Software Processes
- Software Construction
- Requirements Engineering
- Software Verification and Validation
- Tools and Environment
- Software Project Management

As a result of the curricula analysis for the named categories, we get round about 250 keywords and terms. These terms are clustered and summarized 53 points consisting of only one keyword or a connection of keywords are the result of this clustering and summarizing process.

Before we had a detailed look at the results of the curricula analysis, we thought we will find some competency definitions in our codings, but we only found keywords. However, we had to realize that only a very small number of curricula includecompetence definitions. Consequently, we had to design our survey in a way that we can build competence definitions from the result.For building competencies, we need knowledge about observable behavior of the graduates. We can describe this observable behavior with operators that are also used in taxonomies like the Anderson-Krathwohl Taxonomy (AKT) (Krathwohl, 2002). The operators describe to which cognitive dimension of the AKT model a named expectation belongs. Operators are words like: explain to describe that a person has knowledge about a concept or implement to describe that a person can use a concept. With these operators, we can describe if a student, for example, should only know a CS aspect like inheritance or should use it. Consequently, we decided to develop a survey in the following way.



General Information

Your age?

Only numbers may be entered in this field

Your gender?

Female Male No answer

Please name how much work experience you have.
Choose one of the following answers

- less than 2 years
- 2-5 years
- 5-10 years
- more than 10 years
- No answer

Where do you work?
Choose one of the following answers

- company
- university
- college, University of Applied Science etc.
- other educational institution
- self-employed
- Other:
- No answer

Which position correlates with your tasks? Please choose that one which is the closest to your position.
Choose one of the following answers

- professor
- lecturer
- research associate/ research assistant
- other
- No answer

Do you have personnel responsibilities?

Yes No No answer

In which country do you work?

Figure 3: General Questions in the case that someone works at University.

Study Design

Our study consists of a survey. It addresses university and company members because a university education is a scientific education and a vocational training. Consequently, both perspectives are important. Especially we ask people who are involved in the education of CS students or who work with graduates (freshmen) in a company.

Expectations on graduates with a BSc. in Computer Science

This is the main part of the survey. In the following you find a list of CS topics. Please think about your expectations of people with a bachelor's degree in CS, directly after their graduation. Please write down a sentence for each item, which describes your expectations (maybe with an application context) on the named CS topic.
Example: inheritance
possible expectations:
The graduate should understand the concept of inheritance and should also use it in programming.
The graduate should handle the concept of inheritance in everyday life of a programmer confidently.
The graduate should know and understand the concept.

Please write down a sentence for each item, which describes your expectations on the named cs topic as in the example. If you have any problems with a topic, you can leave it out. The items are random topics.

- syntax and semantics
- basic terminology of programs and their execution
- evaluation/analysis of algorithms (efficiency, time and cost models, time and space complexity)
- testing strategies and methods (Unit Tests, integration tests, system tests, automated certification, automated tests, critical system validation etc.)
- modern programming environments/ editors
- documentation during software lifecycle (e.g. functional specification, requirement specification document, notations for the analysis and development phase)
- requirements analysis (e.g. use case identification, application requirements, validating requirements etc.)
- project organisation, project management, principles of management, process management
- cost estimates
- simple I/O and file I/O
- graphs, trees
- time management
- give presentations

The topics above are only a small selection. Maybe you miss something you expect from graduates with a bachelor degree in CS. If so, please name these aspects.
Please consider that we focus on subject-specific competencies. Soft skills are not important here.

Figure 4: Example Questions: Expectations on graduates with a BSc. in Computer Science

The survey consists of two parts. In the first part, we ask some general questions about work experience or job position, for example. Figure 3 shows the questions for the case that someone works at University. For people in a company, we offered the positions: software developer; team leader, head of department, etc.; project management; product management and consulting.

The second part of the survey is on expectations on graduates with a BSc. in Computer Science and constructed in the following way: Each of the 53 CS topics or terms, the result of the curricula analysis described above, became one item in the survey. With these items, we built up 6 item sets consisting of 13 items and one additional open question. 30% of the items in each set also belong to another set.

Consequently, we have an overlapping. This is important, because every participant only gets one item set regarding the first letter of his/her last name.

We ask the participations to write a sentence with their expectations on graduates regarding each item. The survey shows an example for this at the beginning. The open question asks if there are any more expectations that are not named before. The reason is that every participant only gets one set, and maybe they have expectations that are not in the shown set, and that are not named in the curricula. Figure 4 shows an example set of items.

SURVEY RESULTS AND DISCUSSION

At the moment, we have 42 participants in our survey fromuniversity and company and different countries. In the following, we want to show our first results and discuss these and the survey itself.

Survey Results

In this section, we present our survey results. As said above, we have at the moment 42 participants from 8 different countries. Most of them are from the US and

N=42

Age:	
Average	39,64
Standard:derivation	11,37
minimum	22
maximum	70

Gender:		
Answer	Count	Percentage
female	11	26,14%
male	31	73,81%

personal.responsibilities		
Answer	Count	Percentage
yes	25	59,52
no	12	28,57
no:answer	5	11,9

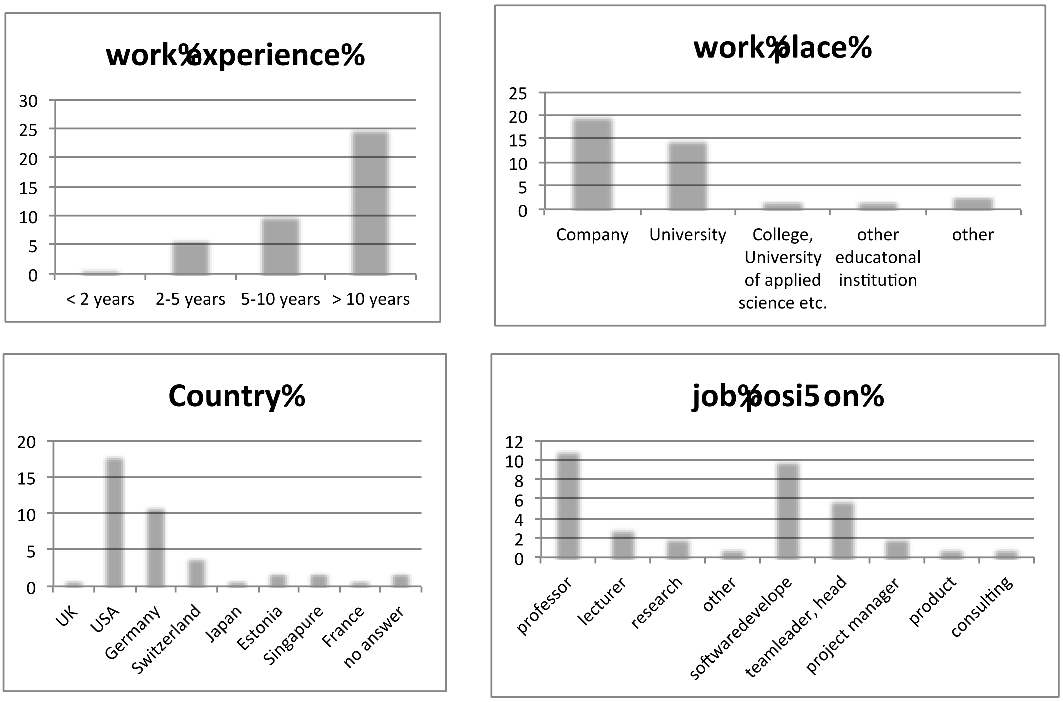


Figure 5: Results of the general questions

Germany. As often in CS the number of participants, which are male is larger than the number of female participants. Additionally the participants mostly have a long work experience from more than ten years. The mixture of university and company participants is very good. Figure 5 shows the complete results of the general questions.

Item	Summarized Answers	Relevance	Differences
small programs	able to read/ understand and develop small programms; additionally to debug them; knowing how to understand them	verry important	tendency to more practical implementation in the companies
software quality, quality assurance (product quality managment,metrics for process, project and product-quality, process improvement)	aware of it, understand the concepts implement standarts	medium	different opinions in the companies in case of relevance
software lifecycle management	aware of it; kow it; fundamental grasp	not so important	expectations in company are lower than at university
Evaluation/analysis of algorithm (efficiency, time and cost models, time and space complexity)	understand and use big O; estimate big O complexity by looking at an implementation or pseudocode; know runtime complexity of some classical algorithms	important	no differences
correctness, method of invariants, verification (e.g. deductive verification,verification according to floyd or hoare), termination etc.	University: know pre-post conditions; know the concept of invariants	not important	not relevant in company
Analysis modeling approaches, scenario based modeling, design based on requirements, models for analysis etc.	understand vs. create; University: understand requirement analysis; the development of scenarios and use cases and the role of theses approches in softwaredesign; Company: create abstract models of familiar constructs with at least 2 diffferent approaches eg. OOP vs functional	essential	University: grasp; Company : application
project planning, techniques for project planning	different classical models like: spiral;waterfall or agile should known	not important vs. medium understandig	different opinions in case of relevance at university and company
basic datastructures (f.e. arrays, lists and linked lists, records, strings	The graduate should know when and how to use basic data structures and have extensive experience through course work; should be able to manipulate these in their sleep	very important	company has a tendency through implementation
refactoring	know and can use the different methods; keyword: clean code	medium vs. important	Company: important, University: different opinions
mapping design to code	know and can challenge the relevant design concepts; implement them into code	important	Company: emphasize the implementation into code

Figure 6: Results of some example Items

To evaluate the second part of our survey we used the following methodology:

- For each item, we first start to summarize the common expectations. Additionally we note the difference: Is there a difference between company and university? And at least: Is the relevance of the topic mentioned?

- For the open question, we tried to map the answers to existing items. If this was not possible, we noted it as a new point (if it belongs to our relevant area of software development and engineering)

The result is a table with the summarized expectations, the relevance and the important differences between the answers. These results are surprising. We expected a difference between the expectations of people from university and company. Almost more practical aspects should named by company members. In contrast, the first results show us only small differences and only a weak tendency to more practical issues than expected. Additional aspects resulting from our study: The expectations from the companies are related to the domain in which the company works and additionally they are of course related to the job position. These results are not surprising. Often content related competencies are the most important. Figure 6 shows some example items and results mostly with some differences in the results.

If we discuss the survey itself, we have to mention that not all items are ideal. The comments of the participants show us that some items need a small context for a better understanding. Additionally some other items are too complex. Consequently, we add too many keywords to one item.

Some participants mentioned a numbered scale would be better than writing a sentence. But here we think with a numbered scale that only shows the importance of a topic we can't build competencies.

All in all the results will help us to develop the competence model. The number of participants in the survey is not as big as preferable. Furthermore, the answers of the different participants are often very similar. Additionally we now have more hints on expected cognitive processes. These are needed to build a competence model.

CONCLUSION AND FURTHER WORK

In this paper, we present an empirically oriented methodology for developing a computer science competence model. The main focus in this paper is a survey on expectations on CS bachelor graduates. This survey addresses university and company members and is part of our methodology. The items of the survey based on a curricula analysis. A curricula analysis is only a view on aspects universities want to teach. A bachelor degree is also an associate degree. The associate degree is the reason we integrated the expectations of company members to our work. The results of the survey are in some cases surprising. We expected that companies would expect much more practical experience than our survey shows. Additionally there was only one item where the company members said these topics are irrelevant.

Before we will build our competence model, we have to compare our results with other existing work like the model the German Computer Science Organization (GI) is developing (Bröker, Kastens, and Magenheim, 2014). After that, we will build our model and develop an assessment. This assessment provides the ability to find out the deficits of our students in a more concrete way than by means of observation only. These concrete results will give computer science departments the opportunity to develop competence-oriented curricula and specific interventions to help students in overcoming their deficits, and, in addition, to measure the effect of these interventions accordingly.

REFERENCES

Berges, M., Hubwieser, P., Magenheim, J., Bender, E., Bröker, K., Margaritis-Kopecki, M., Neugebauer, J., Schaper, N., Schubert, S. & L. Ohrndorf. (2013). Developing a competency model for teaching computer science in schools. *In*

Proceedings of the 18th ACM conference on Innovation and technology in computer science education (ITiCSE '13) (p. 327). ACM, New York, NY, USA.
Bourque, P. & Dupuis, R. (eds.) (2014). *Software Engineering Body of Knowledge (SWEBOK) ed. 3.0*. EUA: Angela Burgess
Bröker, K., Kastens, U., Magenheim, J., (2014). Competences of Undergraduate Computer Science Students. *In KEYCIT 2014- Key Competencies in Informatics and ICT.* Potsdam 2014.
Denning, P.J. (November 2003). Great principles of computing. In *Communications of the ACM*, 34(11), pp. 15-20
Hubwieser, P., Magenheim, J., Mühling, A. & Ruf, A. (2013), Towards a conceptualization of pedagogical content knowledge of computer science. In *Proceedings of the ninth annual international ACM conference on International computing education research*, pp.1-8.
IEEE Computer Society (2014). The software engineering competency model.
Krathwohl, D. R., (2002), A Revision of Bloom's Taxonomy: An Overview, *Theory Into Practice*, 41(4), pp.212-218.
Linck, B., Ohrndorf, L., Schubert, S., Stechert, P., Magenheim, J., Nelles, W. Neugebauer, J. & Schaper, N. (March 2013). Competence model for informatics modelling and system comprehension. In: *Global Engineering Education Conference (EDUCON'13),* pp. 85-93.
Mayring, P. (2010). *Qualitative Inhaltsanalyse: Grundlagen und Techniken.* Weinheim: Beltz Pädagogik, Beltz.
Schaper, N., Kompetenzorientierung in Studium und Lehre (2012), *Fachgutachten für die Hochschulrektorenkonferenz,* Bonn: HRK
Schaper, N., Magenheim, J., Schubert, S., Hubwieser, P. , Bender, E., Margaritis, M., Ohrndorf, L. & Berges, M. (2013). Competencies of Teaching Computer Science. In Blömeke, S., Zlatkin-Troitschanskaia, O. (Eds). *KoKoHs Working Papers no.3*, pp. 32-34.
The Joint Task Force on Computing Curricula Association for Computing Machinery IEEE-Computer Society 2012 (2013). *Computer Science Curricula 2013 - Ironman Draft.*
UNESCO (2012). *UNESCO ICT Competency Framework for Teachers*. UNESCO. Retrieved from: http://unesdoc.unesco.org/images/0021/002134/213475E.pdf, last access: 14.10.2013.
Weinert, F. E. (2001). Vergleichende Leistungsmessung in Schulen – eine umstrittende Selbstverständlichkeit. In Weinert, F. E. (Ed.). *Leistungsmessungen in Schulen* (pp. 17-31). Weinheim: Beltz

Kathrin Müller is research associate and PhD-Student at the Computer Science Education Group at the University of Paderborn, Germany. Her research areas are CSE at University and School. Her main research focus is on competence models, competence development and competence assessment.

A critical evaluation of information security education in nursing science: a case study

Irene Okere, *Irene.Okere2@nmmu.ac.za*
Nelson Mandela Metropolitan University, School of ICT, South Africa

Johan van Niekerk, *Johan.vanNiekerk@nmmu.ac.za*
Nelson Mandela Metropolitan University, School of ICT, South Africa

Kerry-Lynn Thomson, *Kerry-Lynn.Thomson@nmmu.ac.za*
Nelson Mandela Metropolitan University, School of ICT, South Africa

Abstract

The healthcare sector, like other sectors of society and most organisations of today, relies heavily on information for day-to-day operations. This information is increasingly used in a digital format. Due to this reliance on information, it is vital to ensure that healthcare workers have the necessary skills to use digital information resources whilst still ensuring the confidentiality, integrity and availability (CIA) of the information. Humans are often seen as integral to the protection of information. Humans, therefore, should have the desired knowledge and attitude with regards to information security. Information security is especially important in the healthcare sector where compromised information could lead to the loss of human life. Nurses, the largest group amongst the healthcare practitioners, have direct involvement with both patients and patient records which makes their role in healthcare critical. With such a critical role, it is vital that nurses are adequately educated regarding their roles and responsibilities with regards to information security. This paper evaluates the information security education component of the formal curriculum of the Nursing Science qualification at the Nelson Mandela Metropolitan University in order to determine the current information security components included in the curriculum, as well as the adequacy of this content. The paper highlights areas of concern regarding this education and presents recommendations for the improvement of such curricula, from an information security perspective.

Keywords

Digital literacy, Information security education, Healthcare, Nursing curriculum

INTRODUCTION

"Digital literacy involves more than the mere ability to use software or operate a digital device; it includes a large variety of complex cognitive, motor, sociological, and emotional skills, which users need in order to function effectively in digital environments" (Eshet&Eshie-t-alkalal2004).Further to this, information literacy is defined as "the ability to know when there is a need for information, to be able to identify, locate, evaluate and effectively use that information for the issue or problem at hand" (NFIL, 2015). It has become increasingly important in most fields of study for students of that field to be digitally, as well as information, literate.

The healthcare sector, like other sectors of society and most organisations of today, relies heavily on information for day-to-day operations. With such reliance on information, it is vital that information be protected to ensure its confidentiality, integrity and availability (CIA).

Humans, to a large extent as a result of their behaviour, are at the centre of protecting an organisation's information. Humans should therefore have the desired knowledge and attitude with regards to information security (Van Niekerk& Von Solms 2005).

Information security is particularly important in the healthcare sector due to the increased risks added by the fragmented nature of healthcare and also because the healthcare sector deals with patients' lives where incorrect or lost information could have serious consequences. The healthcare sector is fragmented as a result of diverse role players including doctors, nurses, paramedics, medical aid staff and administrative staff who utilise information differently and thus have different requirements from patient information.

Nurses, the largest group of healthcare practitioners, are directly involved with patients and patient records. Such involvement makes the role of nurses in healthcare critical (Benner et al. 2010). With such a critical role, it is vital that nurses are adequately educated in terms ofdigital literacy, information literacy **and** their roles and responsibilities regarding information security. This means that nursing educational programs should be developed in a manner that enables the Nursing graduate to effectively carry out his/her role with regards to information security in the workplace. The need for reform in nursing education has been debated and has become more important due to the deficiencies in the quality of patient care, as well as patient safety issues (Booth 2006; Forbes & Hickey 2009). With the emergence of electronic health (eHealth) technologies to support client care, undergraduate Nursing education calls for a curriculum that enables the Nursing graduate to function as an eHealth practitioner (Booth 2006). eHealth is "the use of information and communication technologies (ICT) for health including treating patients, conducting research, educating the health workforce, tracking diseases and monitoring public health" (World Health Organisation 2013).

This paper evaluates the information security education content of the formal curricula in the School of Nursing Science at the Nelson Mandela Metropolitan University (NMMU) in order to determine the adequacy of information security content in the Nursing Science curriculum. All nursing science students at the NMMU currently receive basic computer literacy education as one of their core second year modules. However, does the current nursing education provision adequately address the information security needs of nurses within the curriculum? This paper highlights areas of concern and presents recommendations for improvement of such curricula with regards to information security.

The remainder of this paper is structured as follows; the methodology used, the role of information in the fragmented healthcare sector, the role of humans in information security, information security education, formal information security education at the NMMU School of Nursing Science, discussion, and conclusion.

METHODOLOGY

This research performs a critical analysis of the existing state of information security education in the School of Nursing Science at the NMMU through a combination of literature reviews, qualitative content analysis and interviews.

Qualitative content analysis is a "research method for the subjective interpretation of content of text data through the systematic classification process of coding and identifying themes or patterns" (Hsieh & Shannon 2005). The analysis was conducted according to guidelines provided by Krippendorff (2004). A literature review was conducted by reviewing literature focusing on information security education in Nursing Science. Qualitative content analysis was carried out on the prospectus as well as specific subject study guides of the Nursing Science department of the NMMU to answer the thematic question: Does the Nursing Science prospectus or study guide include information security content in the Nursing Science curriculum? Further semi-structured interviews were conducted with some of the lecturers at the School of Nursing Science at the NMMU to elucidate their views with regards to information security content in the Nursing Science curriculum.The results from the critical analysis as well as the interviews are discussed.

INFORMATION IN THE HEALTHCARE SECTOR

Information is important for most organisations for their day-to-day operations. This is especially true for the healthcare sector where various role players with different information usage requirements are all heavily reliant on information. Without accurate, complete, and timely information many healthcare workers would be unable to administer proper treatment to patients.

This is especially true for nurses. A nurse is a healthcare professional who focuses on caring for individuals, families and communities, ensuring that they attain, maintain or recover optimal health and functioning. The nurse may practice in a variety of places such as hospitals, clinics, and physician offices (Medical_news_today 2009). The nurse has access to the patient's records in order to know how and what care to provide for a patient. As shown in Figure 1, many different types of healthcare professionals could play a role towards the administration and implementation of patient care. A nurse's role is just one of these functions, but, due to the nature of healthcare this role is a very central role.

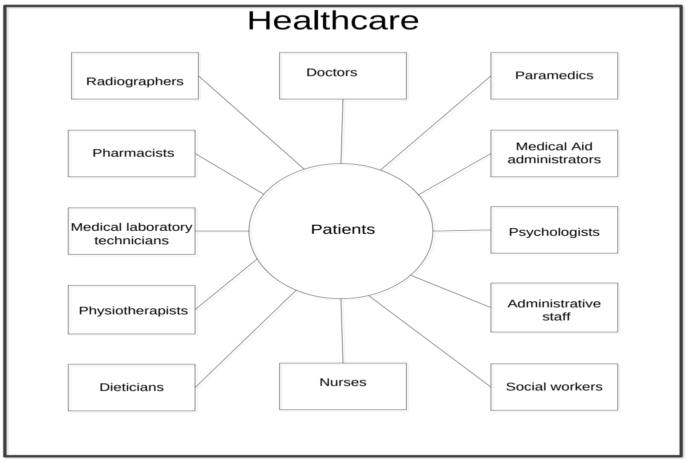


Figure 1: Different role players in the healthcare sector

Nurses do not necessarily diagnose patients, or decide on the specific treatments prescribed. They are, however, responsible for ensuring the treatment is administered correctly, and the patient is monitored for any signs of change in his/her condition. As such healthcare records play a vital role in enabling a nurse to perform his/her duties correctly.

Similar to records used in most other fields today, healthcare records have increasingly become digitized. Today electronic health records have become so ubiquitous that the "Institute of Medicine (IOM) and the Health Information Technology Act (2009) in America had recommended that electronic health records (EHRs) should be fully adopted by 2014"(Kowitlawakul, Chan, Pulcini, & Wang2014, p. 1). This recommendation has prompted educational institutions to assist in the preparation and education of healthcare workers to ensure that they are proficient in the use of EHRs (Kowitlawakul et al.2014). Similar trends towards the acceptance of electronic health records exist in many other countries and it can be argued that similar adoption of EHRs will eventually become the reality in most countries that have the necessary ICT infrastructure. Many factors could influence the successful use of EHRs. For student nurses, one such factor, which has been shown to have an effect, is the basic computer self-efficacy of the student nurses (Kowitlawakul, et al. 2014). Health records can also exist in a multitude of formats and EHRs could thus have elements ranging from text and numeric data, to various forms of laboratory test results including; graphs, photographs, or even video. It is thus vital for future nurses to be digitally literate.

Exactly what it means to be *digitally literate* has become increasingly difficult to define and the literature itself is inconsistent with this definition (Eshet&Eshie-t-alkalal 2004). However, for the term to have meaning in a nursing context one can assume that nurses should have a certain minimum set of skills relevant to the day-to-day tasks of a nurse. These tasks would include the creation, reading, and updating of EHRs and other relevant information assets. Due to the sensitive nature of health data it should also include the underlying sociological and emotional aspects that would enable them to work with data in cyberspace and avoid possible traps and pitfalls (Eshet&Eshie-t-alkalal 2004) posed by various forms of social engineering.

A nurse should thus, in addition to healthcare related tasks, be able to fulfil his/her role in ensuring the confidentiality, integrity and availability (CIA) of the information irrespective of the usage requirement.

All information should be protected to ensure its security. However, in some instances there could be a trade-off between confidentiality and availability depending on the context of the role player. For example, in life threatening situations, the availability of the medical information may supersede the confidentiality of the medical information and vice versa. It means that, to a large extent, the judgment call between confidentiality and availability of information depends on the judgment of the role player (human). Further, this ‘human’ judgement is not something that can be controlled or enforced by technology. In other words, information security needs to be enforced at the human level as well as the technology level in the healthcare environment and all nurses should thus be adequately trained with regard to their role(s) in the information security process.

THE ROLE OF HUMANS IN INFORMATION SECURITY

Humans play a vital role in the protection of information. However, humans are also often regarded as the weakest link in the security chain (Lineberry 2007) and thus constitute a challenge in the protection of information. According to the Ponemon Institute (2012), a number of security breaches have been reported in various healthcare institutions linked to the human factor. Such breaches could cause reputational damage, financial costs and the compromise of patient care in the form of misdiagnosis, delays in treatment, patient injury or death (Collmann& Grimes 2007). It, therefore, becomes important that the human factor is addressed. Humans need to have the desirable knowledge and attitude with regards to information security in order to assist in the protection of information assets (Van Niekerk& Von Solms 2005). To a certain extent the desirable knowledge and attitude can be instilled via education. Consequently, it is important that humans, in this case healthcare professionals, are educated regarding their role and responsibilities in information security so that they are able to make appropriate decisions regarding information and information security.

INFORMATION SECURITY EDUCATION

This paper focuses on the inclusion of information security education as part of the digital literacy of nurses. The focus was chosen due to the fact that nurses play a vital role within the healthcare sector and also due to convenience, since the School of Nursing Science at the NMMU is situated next to the School of ICT.

In order to examine the formal information security educational requirements for the nursing students within the School of Nursing Science at the NMMU, a comparison will be drawn based on the formal information security educational requirements for the students of the School of ICT at the NMMU. The School of ICT was chosen because ICT education should, presumably, have some information security education integrated into the course content.The paper thus wishes to propose that the formal structures to govern the Nursing Science curricula could be structured similarly to the way curricula are planned within the School of ICT; this structure is depicted in Figure 2.

The curriculum in the School of ICT, for example, isinformed by international guidelines found in the ACM/IEEE-CS curriculum.National formal guidelines stem from the international standards and guidelines, where each country, for example South Africa, adapts the international guidelines to suit South Africa's needs and can be used by universities to inform their curricula.

Following the national formal guidelines,a prospectus is produced which constitutes a binding contract between the university and the students. The prospectus level is followed by the lecturer level where the relevant lecturer, who should be knowledgeable about the content of the prospectus, creates a study guide, which indicates the detailed guidelines for the curriculum. Instructional activities are then planned according to the study guide.

Figure 2: Structure for formal information security education source

This proposed structure will be used to evaluate and/or identify any relevant curriculum guidelines that inform the existing state of information security education as part of the nursing student education at the School of Nursing at the NMMU.

ANALYSIS OF FORMAL INFORMATION SECURITY EDUCATION AT NMMU SCHOOL OF NURSING SCIENCE

This section analyses the existing state of formal information security education included by the School of Nursing Science at the NMMU. Firstly, a search was conducted to find internationally accepted curriculum guidelines that could serve a purpose similar to the ACM/IEEE-CS used for IT curricula.No similar international body was identified for Nursing Science. Nursing science, therefore, does not have any form of standardized internationally recognized curricula.

Secondly, at the National level, in the United States of America (USA), the American Association of Colleges of Nursing (AACN) was identified and provides well researched curriculum guidelines. The AACN provides a framework for baccalaureate and graduate nursing degree programs aimed at meeting the challenges of healthcare in the 21st century (American Association of Colleges of Nursing, 2008). The AACN framework comprises 9 delineated outcomes. An analysis of the AACN framework revealed that these curriculum guidelines do incorporate information security. However, in South Africa, the South African Nursing Council (SANC) and the South African Qualifications Authority (SAQA) provide guidelines on the content of the Nursing Science curriculum (South African Qualifications Authority (SAQA) 2005a; South African Qualifications Authority (SAQA) 2005b). An analysis of the documents from SANC and SAQA revealed no direct evidence of the inclusion of information security in the guidelines for exit level outcomes or critical cross-field outcomes. Governments play an important role in ensuring the effectiveness of various initiatives. For example, as stated by Klimburg (2012, p. 131) "Cyber security at the national level will fail when there is an inappropriate level of cyber security awareness and education. A nation requires its ministry of education and/or science to develop strategic/operational programmes for cyber security awareness and education". Similarly, information security education programs or initiatives should be established and encouraged by the government for it to be effective. In South Africa such guidance, specific to the information security needs of nursing science, are thus currently lacking.

Thirdly, at the prospectus level, an analysis of the NMMU School of Nursing Science prospectus for the year 2014 revealed no direct evidence of the inclusion of information security in the curriculum. However, it could be argued that the evidence could be present but was not found due to terminological differences.

At the lecturer level, interviews with a few of the Nursing Science lecturers revealed contradicting information, with some lecturers indicating the absence of information security content in the Nursing Science curriculum and one lecturer indicating the presence of information security relevant content in the Nursing Science curriculum in specific subject codes. However, an analysis of the study guides of the specific subject codes that were said to contain information security, revealed the absence of information security content.

DISCUSSION

The information obtained from the various levels with regards to information security content in the School of Nursing Science at the NMMU all converge. There is no evidence of formal consideration of the information security needs of nursing students at the NMMU at any level, ranging from international to individual lecturer. This lack of evidence triangulates as depicted in Figure 3 which is based on the convergence of evidence diagram according to Yin (2009).

According to Yin (2009), the evidence from multiple sources should converge in order to validate the result. In the study conducted in the School of Nursing Science at NMMU, the evidence (formal information security content in curricula) from multiple sources (International level, National level, Prospectus, Lecturer, Study guides) all converge to demonstrate a lack of formal consideration. The only evidence that was found in this case was at the lecturer level where a single lecturer indicated during interviews that information security concepts were taught. This statement was, however, refuted by subsequent examination of the underlying study guides.

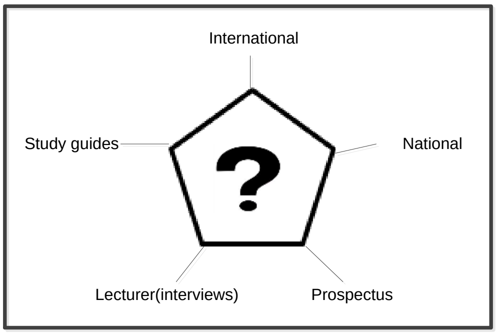


Figure 3: Convergence of evidence regarding information security content included in the curriculum of the School of Nursing Science at NMMU

Thus, at the international level, no international body/association was found that regulates the Nursing Science curriculum. At the international level, it would be beneficial to have an international body that prescribes an international curriculum standard that could be adapted by various countries. At the national level, the websites of SAQA and SANC revealed very little information. No direct evidence of the inclusion of formal information security content was found in the Nursing Science curriculum guidelines. At the national level, the South African government is ultimately responsible for information security and should ensure that information security is formally included in the Nursing Science curriculum at a national level and should cascade down to the prospectus to make it binding on the lecturers to teach information security.

An analysis of the Nursing Science prospectus did not reveal any direct evidence of formal information security content. Without information security being formally prescribed in the prospectus, there is no binding contract between the university and the student and thus the lecturer is not obliged to teach information security. Therefore, it will be beneficial for information security to be formally included in the prospectus to make it binding on the lecturers in the School of Nursing Science to teach information security.

Interviews with some of the Nursing Science lecturers revealed that information security might be taught, but **no documented evidence was found to support this assertion**. It could, thus, be interpreted that the teaching of information security, with regard to healthcare, could be lecturer dependent. This could prove detrimental considering the importance of information security in healthcare. If a specific lecturer were to leave the institution such lecturer dependent content would be lost.

With the inclusion of information security content, lecturers should become knowledgeable in the area of information security in order to teach the subject effectively. This could be achieved by lecturers receiving training to improve their knowledge of information security. Further, gathering information from lecturers using a questionnaire to ascertain their understanding of information security could be used to provide the appropriate training to the lecturers, based on their responses. Information security is important and individuals need to possess the desirable knowledge as well as attitude. Individuals with the right attitude to information security may still be unable to behave securely due to lack of knowledge. Thus it is important that lecturers are knowledgeable about information security and subsequently transfer such knowledge to students.

Finally, an analysis of the study guides for relevant subjects revealed no direct evidence of formal information security content in the subject exit level outcomes. Current ICT relevant education of nursing students consists primarily of basic “computer literacy” elements, including Word processing, Spreadsheets, Internet and email. In the authors' opinion, this level of "computer literacy" cannot be described as digital or information literacy. However, this analysis focussed specifically on the information security elements and this paper thus do not make any claims regarding the appropriateness of this education other than to comment on the lack of formal information security. If information security wasprescribed in the prospectus, it would have cascaded down to the lecturers and subsequently to the study guides ensuring that information security is formally taught, irrespective of the lecturer assigned to teach the subject.

CONCLUSION

With the advent of the use of ICT in formal healthcare, the importance of ICT education for nursing students has become paramount. Nursing students should become digitally literate. One element of such digital literacy is the ability to work with information in a way that will maintain the security of the information.Information security is important especially in the healthcare sector. This paper provided a critical evaluation of the information security content of the formal curriculum of the School of Nursing at the NMMU. From the discussion, it is evident that no information security education component currently exists in the formal curriculum of Nursing Science, as it is not directly included in the prospectus which is the binding contract between the university and the student. This paper suggests that information security education should be included in the prospectus so that it becomes binding on the lecturer(s) to teach. The lecturers should receive training so that they become knowledgeable about what to teach in terms of information security and should ultimately reflect in the study guides they prepare under the subject outcomes.

It should be noted that this paper researched the information security component of formal curricula of the School of Nursing Science only at the NMMU and no wider empirical studies were conducted. Additionally, nursing students were not interviewed to ascertain whether they received on-the-job training with regard to information security.

ICT education is becoming a must have in many professions today. For some of these professions it is critical that this education extends beyond mere "computer literacy" but also encompasses additional skills such as basic information security skills. The healthcare sector is one such sector where a compromise in the security of the information can have dire consequences. Nursing science education should thus include such education as a formal requirement. The authors believe that many similar limitations exist in other professions.

REFERENCES

American Association of Colleges of Nursing, 2008.The Essentials of Baccalaureate Education for Professional Nursing Practice.

Benner, P. et al., 2010. Educating nurses: A call for radical transformation 1st edition San Francisco: Jossey-Bass.

Booth, R., 2006. Educating the future eHealth professional nurse. International journal of nursing education scholoarship, 3(1).

Collmann, J. & Grimes, S., 2007. Managing information security in healthcare,

Eshet, Y., & Eshie-t-alkalal, Y. (2004). Digital Literacy: A Conceptual Framework for Survival Skills in the Digital era. Journal of Educational Multimedia and Hypermedia, 13, 93–106. Retrieved from http://www.editlib.org/p/4793

Forbes, M. & Hickey, M., 2009. Curriculum reform in Baccalaurete nursing education: Review of the literature. International journal of nursing education scholoarship, 6(1).

Hsieh, H. & Shannon, S., 2005. Qualitative Health Research.Health (San Francisco), 15(9), pp.1277–1288.

Klimburg, A., 2012. National cyber security framework manual A. Klimburg, ed., Tallinn, Estonia: NATO CCD COE publications.
Kowitlawakul, Y., Chan, S. W. C., Pulcini, J., & Wang, W. (2014). Factors influencing nursing students' acceptance of electronic health records for nursing education (EHRNE) software program. Nurse Education Today, 35(1), 189–194. doi:10.1016/j.nedt.2014.05.010
Krippendorff, K., 2004. Content analysis: An introduction to its methodology. 2nd editio., Thousand oaks. CA. Sage.
Lineberry, S., 2007. The Human Element: The Weakest Link in Information Security. Journal of accountancy. Available at: http://www.journalofaccountancy.com/NR/exeres/5128C268-5493-4E66-93F0-8FE6DCD7DF23.htm.
Medical_news_today, 2009. What is nursing? what does a nurse do? Available at: http://www.medicalnewstoday.com/articles/147142.php [Accessed April 11, 2013].
NFIL (National Forum on Information Literacy). What is Information Literacy? Available at http://infolit.org/about-the-nfil/what-is-the-nfil/ [Accessed February 5, 2014].
Van Niekerk, J. & Von Solms, R.., 2005. A holistic framework for the fostering of an information security sub-culture in organisations.In ISSA.
OCC info: Government of Alberta, Medical Laboratory Technologist. Available at: http://alis.alberta.ca/occinfo/content/requestaction.asp?aspaction=gethtmlprofile&format=html&occpro_id=71003140 [Accessed April 11, 2013].
Ponemon Institute, 2012. Third Annual Benchmark Study on Patient Privacy & Data Security Sponsored by ID Experts,
South African Qualifications Authority (SAQA), 2005a. Registered Qualification: Bachelor in Nursing,
South African Qualifications Authority (SAQA), 2005b. Registered qualification: Diploma in Nursing,
World Health Organisation, 2013.eHealth. Available at: http://www.who.int/topics/ehealth/en/ [Accessed February 20, 2013].
Yin, R., 2009. Case study research design and methods 4th editio., SAGE publications.

Model of Learning Computational Thinking

Tauno Palts, *tauno.palts@ut.ee*
University of Tartu, Institute of Computer Science, Estonia

Margus Pedaste, *margus.pedaste@ut.ee*
University of Tartu, Centre for Educational Technology of the Institute of Education, Estonia

Abstract

There is a high demand for qualified information and communication technology (ICT) practitioners in the European labour market, but the problem at many universities is a high dropout rate among ICT students, especially during the first study year. The solution might be to focus more on improving students' computational thinking (CT) before starting university studies. Therefore, research is needed to find the best methods for learning CT already at comprehensive school level to raise the interest in and awareness of studying computer science. Doing so requires a clear understanding of CT and a model to improve it at comprehensive schools. Through the analysis of the articles found in EBSCO Discovery Search tool, this study gives an overview of the definition of CT and presents three models of CT. The models are analysed to find out their similarities and differences in order to gather together the core elements of CT and form a revised model of learning CT in comprehensive school ICT lessons or integrating CT in other subjects.

Keywords

Computational thinking, models, comprehensive school, learning

INTRODUCTION

There is a high demand for qualified ICT practitioners in the European labour market. Recent reports warn of decreasing interest among young people in studying science, technology, engineering, and mathematics (STEM) in many countries (OECD, 2008). Therefore, educational research is needed to introduce the principles of computer science already at comprehensive school level in order to raise the interest in as well as the awareness of studying computer science.

The 21st century calls for an overall redefinition of the forms of knowledge, skills and competences that are necessary for the advancement of our societies. Computations thinking (CT) is a fundamental skill for everyone, not just for computer scientists. Some authors even argue that we should add CT as every child's analytical ability, just like reading, writing, and arithmetic (Wing, 2006).

CT has a long history within computer science. CT is viewed as the way computer scientists use decomposition, recursion, and algorithms to tackle difficult problems (Hoffmann, 2009). Known in the 1950s and 1960s as "algorithmic thinking", Denning defined CT through input and output – CT is a mental orientation to formulating problems as conversions of some inputs to outputs and looking for algorithms to perform the conversions (Denning, 2009). Aho defined CT as the thought processes involved in formulating problems so their solutions can be represented as computational steps and algorithms (Aho, 2012).

Wing started a new wave of the usage of the term CT, which has become the most commonly used definition in recent academic publications. Wing defined CT as the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form which can be effectively carried out by an information processing agent. The information processing agent can be a computer, a machine or a human being (Wing, 2006).

Glass opposes that the traditional and still meaningful name for the CT idea is problem solving. According to him, there are several reasons for objecting to the word “computational” (Glass, 2006):

- It has not, for at least several decades, described the work of computers. Computers rarely compute but do manipulate information.
- The implication is that CT is a course computer scientists should teach. Computer science concepts can certainly be part of such a course, but problem solving is a universal activity, and many disciplines are capable of teaching it.
- Problem solving is a centuries old discipline. While computers are a powerful new tool for doing it, the underlying discipline should be focused on problems and solutions, not just on solution approaches.

CT is a part of problem solving, but in this article the main focus is on CT as it is acknowledged as a specific term describing one kind of thought process.

As CT has a long history within computer science, it is important to understand where CT situates in the discipline of computer science. Denning concluded that there are four core practices in computer science:

- programming,
- engineering of systems,
- modelling, and
- applying.

CT can be seen either as a style of thought that runs through the practices or as a fifth practice. Denning (2009) sees it as the ability to interpret the world as algorithmically controlled conversions of inputs to outputs.

In order to find the best way of learning CT already at comprehensive school level a model of learning CT should be created. The problem with the models of CT published in the academic journals so far is the lack of systematic approach of various dimensions to learn CT. A way of organizing these models about CT is to sort them into three categories: concepts, practices and perspective (Lye & Koh, 2014).

This study gives an overview of these three approaches of the models of learning CT, which included principles that were mentioned more than once. All the models of CT were analysed to find out their similarities and differences in order to gather together the core elements of CT and create a revised model of learning CT.

METHODS

This paper presents the models of CT found in the academic journals from three different perspectives. Research was done through a systematic search with the EBSCO Discovery Search tool as it includes a considerable number of educational academic journals referring to the core articles of CT. Research was done using the following steps:

1. The search term "computational thinking" was used to make a search in the EBSCO Discovery Search, which resulted in 84 articles published about CT in academic journals.
2. Abstracts of the 84 articles were read. All the articles which were not directly about computational thinking (e.g., in the field of medicine and journalism) and articles in other languages than English were left aside along with articles which could not be accessed. This left 55 articles to be analysed: 24 viewpoints which were articles that included opinions on the topic of CT; and 31 studies which included reviews, experiments, and case studies.

3. 55 available articles were read to sort out articles which included any kind of models of CT. In this stage 31 articles were selected for the final analysis of the study.

4. For a systematic analysis of the 31 articles a table was created (Appendix 1). It was constructed so as to give an overview of the articles with the following given characteristics for each article (Lye & Koh, 2014):

- author(s) of the article;
- setting of the article (e.g., kindergarten, K-12, higher education);
- research type (survey, experimental, case study);
- intervention (e.g., essence, pair programming, game strategy creation, modelling of mathematical concepts, story-based e-learning, etc.); and
- articles sorted by computational thinking approach:
    i. concepts – concepts that programmers use, e.g., variables, loops;
    ii. practices – problem-solving practices that occur in the process of programming; and
    iii. perspective – students' understandings of themselves, their relationships to others, and the technological world around them, e.g., expressing and questioning about the world of technology.

5. Articles which described models of CT from other studies were used as a starting point of the article. If needed, references in these articles were used to get to the root of the models introduced.

RESULTS AND DISCUSSION

The analysis of the 31 articles found by systematic search showed that 13 different models of CT were introduced. 10 of the articles concentrated on different aspects of CT: the senses involved in teaching CT (Katai et al., 2014); creating profiles of the students (Shell & Soh, 2013); emotional self-awareness, empathy alongside CT (Daily & Eugene, 2013); methods towards analytical skill building (Tsalapatas et al., 2012); convergence of systems biology and CT (Navlakha & Bar-Joseph, 2011); effects of task goals on learning computing concepts (Miller & Settle, 2011); dependency cycle in CT skills (Wolz et al., 2011); personal development trajectory within a sociocultural context (Marina U., 2011); solving a problem by reducing it to another problem (Kilpelainen, 2010); and diagnostics and rubrics for assessing learning across the computational science curriculum (Manson & Olsen, 2010). However, those 10 models are not directly included in the further analysis because none of these approaches to CT has been used in more than one article found by the systematic search.

However, there are three approaches to form a model of CT which were found in several articles: i) Interaction between a Human and Computer (evidence found in 3 articles); ii) Conceptual Model (evidence found in 2 articles); and iii) Engineering Design (evidence found in 3 articles). The next three paragraphs give an overview of those three models. After that a revised model of CT based on the three most widely used models is created and presented.

Model 1: Interaction between a Human and a Computer

Interaction between a Human and a Computer approach for a model of CT is derived from Cooper et al. (2010) looking at computational learning as an iterative and interactive process between the human (the K12 student in that case) and the computer (or, in a more theoretical construct, a model of computation) (Figure 1). This model includes the human cognitive process (capacity for abstraction and for problem formulation) and two strengths of the computer (ability to present complex data sets, often visually, and capacity for storing factual and relational knowledge) (Cooper et al., 2010).

Wing supports the approach with the definition of CT which claims that formulated problems should be carried out by an information processing agent, which can be a computer, machine or human being (Wing, 2006). Denning includes in the definition of CT the role of an input and an output, which can be used by computers and human beings (Denning, 2009).

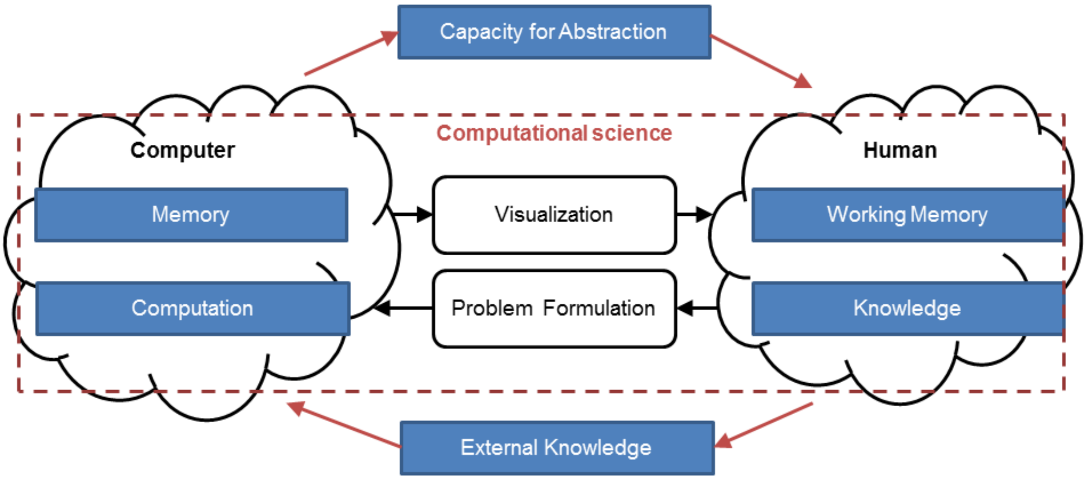


Figure 1: Model of Computational Learning (Cooper et al., 2010)

This model is rather similar to the definitions of CT presented in the introduction but does not include any information about the core concepts of CT (e.g., abstraction, automation) which are recognized as important parts of learning CT (Wing, 2006). Although the problem formulation is shortly mentioned in the model, not enough information is presented about the pedagogical side of learning CT. Interaction between a human and a computer should be presented in the revised model.

Model 2: Conceptual Model

In the Conceptual Model, CT is viewed as a link between discipline thinking and computing philosophy (Wenchong et al., 2014) including five core concepts of CT (Figure 2).

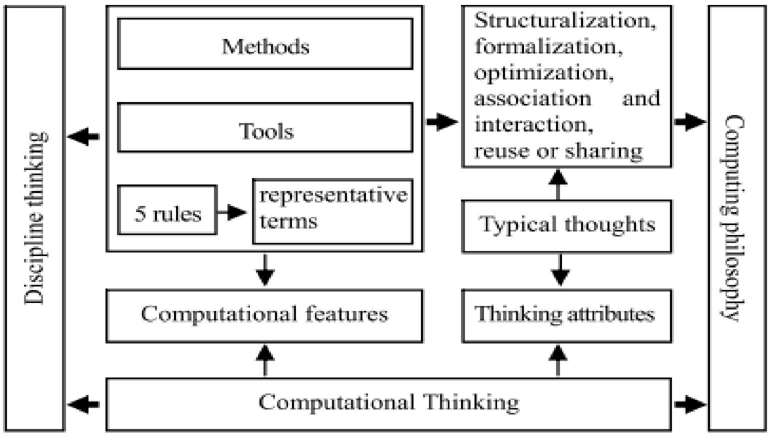


Figure 2: Conceptual Model of CT (Wenchong et al., 2014)

This model is based on the five typical thoughts of CT that form the core of CT:

1. Structuralization, which includes formatting and standardization. Examples of objects that are structural objects in the context of CT include data type, table, and file.

2. Formalization, which reflects ideas such as abstraction and visualization so as to be understood and universal. Examples of formalization include graph models, network protocols, and system formal specifications.

3. Optimization, which is reflected by concepts such as redundancy and complexity. Examples of optimization include lowering the time complexity by reducing the number of loops, from 1st normal form to 5th normal form of the database theory, choosing the data type appropriately.

4. Association and interaction including integrality, which is helpful for improving access speed, realizing automation and ensuring the validity of data. Examples of association and interaction: hyperlink in a web page embodies the association and interaction of files, media and location of information; object-oriented programming embodies the association and interaction of objects and data; transaction management of database theory embodies the association and interaction of events.

5. Reuse or sharing, which is one of the most important thoughts of computing science. It is beneficial to reduce the repetitive development of resources or to improve resource utilization. Examples of reuse or sharing: database technology, which makes data files separate themselves from the source programme and exist independently so that they can be applied to other programmes; resource sharing promoted by the wider coverage of Internet; the clipboard of Windows, the Cut, Copy and Paste operation of various software are all helpful for reusing the data or the files.

Compared to the previous model, this model includes some core elements of CT. Although there can be other ways of presenting the core concepts of CT, all other concepts (e.g., decomposition, pattern recognition (Grover & Pea, 2013)) can be recognized in various parts of the current model. All of the elements in the current model can be integrated into learning CT in various lessons, which means that instruments can be developed to analyse the students' understandings of the conceptual elements of CT.

This model lacks information about the steps of problem solving in learning CT: planning, prototyping, etc. All of these steps relate to CT in important ways that could be significant for education. The core concepts of the conceptual model of CT should be presented in the revised model.

Model 3: Engineering Design

The third way of approaching CT can be including the steps that are important when solving CT problems. Massachusetts Department of Education has created an Engineering Design process model which includes eight steps and is based on the cycle of problem solving (Figure 3) (Massachusetts Department of Education, 2006).

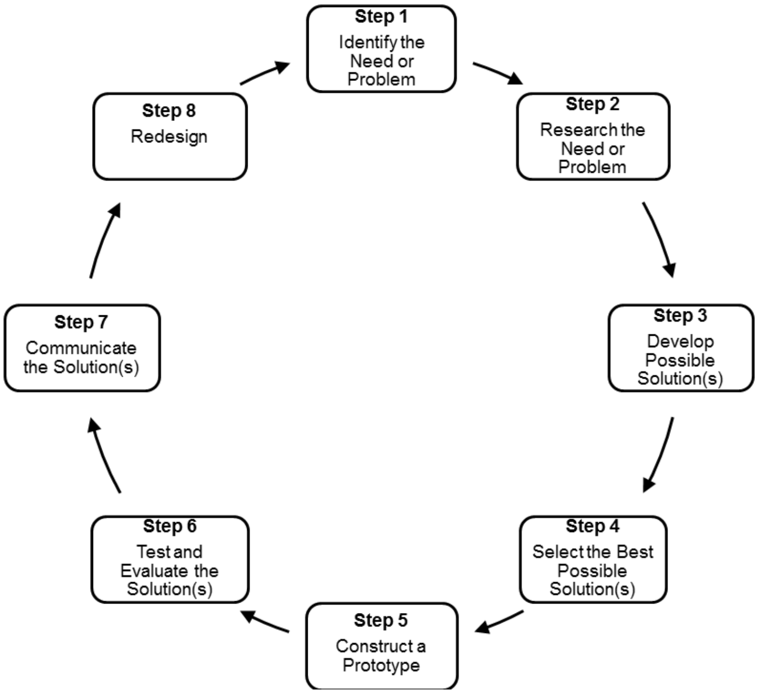


Figure 3: Steps of the Engineering Design Process (Massachusetts Department of Education, 2006)

This design process model consists of eight steps, which mostly follow each other in a linear order but can, in certain circumstances, be arranged in various types of order if needed. The steps of engineering design are as follows:

1. Identify the need or problem.
2. Research the need or problem: examine the current state of the issue and current solutions; explore other options via the Internet, library, interviews, etc.
3. Develop possible solution(s): brainstorm possible solution(s); draw on mathematics and science; articulate the possible solution(s) in two and three dimensions; refine the possible solution(s).
4. Select the best possible solution(s): determine which solution(s) best meet(s) the original need or solve(s) the original problem.
5. Construct a prototype: model the selected solution(s) in two and three dimensions.
6. Test and evaluate the solution(s) asking questions: Does it work? Does it meet the original design constraints?
7. Communicate the solution(s): make an engineering presentation that includes a discussion of how the solution(s) best meet(s) the initial need or the problem; discuss the societal impact and trade-offs of the solution(s).
8. Redesign: overhaul the solution(s) based on information gathered during the tests and presentation.

Steps 1–3 form the analytical, 4–5 the synthetical and 6–7 the evaluation part of the model. There are similar research oriented models created by Sengupta et al. (2013) and Lee et al. (2014), which include slightly different dimensions of CT, but those elements of CT are all included in the current model. Based on the current model, a simpler model has been created for youth (Lee et al., 2011). The current model has been effectively used for teaching CT in robotics (Bers et al., 2014).

The Engineering Design model has many specific steps for describing the process of learning CT, such as planning, prototyping, etc., which can be effectively carried out when learning CT in various comprehensive school lessons. The current model lacks information on the core concepts of CT, such as optimization, formalization, etc. The steps that this model represents can be used as essential parts of the revised model of learning CT.

Revised Model of Learning CT

All of the three applied CT approaches above (Interaction between a Human and a Computer, Conceptual Model, and Engineering Design) look at CT from different perspectives. When gathering together the core of all the models, a new model for learning CT can be created which includes the various dimensions of CT for learning CT at comprehensive school level (Figure 4).

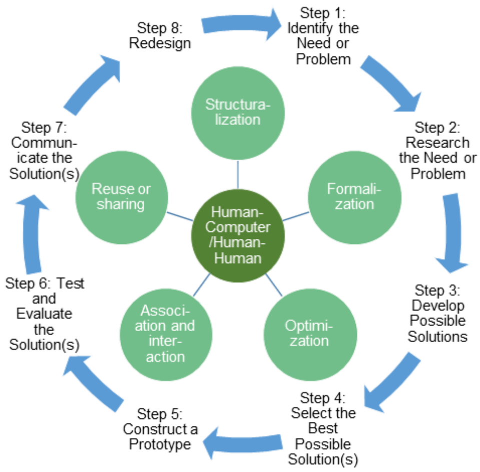


Figure 4. Revised Model of Learning CT

The revised model of learning CT includes three main components derived from previous models of CT:

1. Interaction between a Human and a Computer in the model of learning CT has a vital role and is presented in Figure 4 as the centre of the model.
2. Five core elements of CT (structuralization, formalization, optimization, association and interaction, reuse or sharing) are included in the model because those five components of CT can be taught in various lessons, in various key stages and involve the core elements of CT. In Figure 4 those five elements can be seen surrounding the centre of interaction of humans and computers. The five core elements can be rotated around the centre and used dynamically without any fixed order.
3. 8 steps are included in the model to go through all the steps that occur during the process of learning CT. The first step in the process of learning CT should be identifying the problem. The arrow pointing from step 1 to step 2 in Figure 4 indicates that after the problem is identified, research needs to be done, and after that, possible solutions are developed. The steps are following each other in a linear way and the various steps more or less include the core elements of CT. The circular arrows indicate that when one problem is solved (step 8), a new one can be started from the beginning (step 1).

The revised model of learning CT includes the definitions of CT, the core dimensions of CT, and the problem solving approach of learning CT. This model can be practically used to create scenarios for developing various CT skills by rotating the middle layer of CT concepts and then choosing one element from each layer. For example, in order to create a scenario for teaching structuralization in step 1 (Identify the need or problem), i) an interaction has to be chosen from the centre of the model (e.g., human-computer), and ii) the core concept (structuralization) has to be rotated towards step

1 (Identify the need or problem). As a result, for example, information about the need or the problem can be structured in a file with the valuable metadata and kept in a structured folder system. But in another scenario (e.g., step 5 – constructing a prototype) structuralization can be used to create a database to support the data collection in the solution.

CONCLUSION

In this study a model of learning CT was designed for creating scenarios learning CT in comprehensive school level ICT lessons or in other subjects.

Through the analysis of the articles found through EBSCO Discovery Search, this study gave an overview of the definition of CT and presented three models of CT. The core concepts from the models were integrated into a revised model for learning CT in comprehensive school ICT lessons or for integrating CT in other subjects:

- From the first model a principle of interaction between a human and a computer was added to the revised model.
- From the second model five core CT concepts (structuralization, formalization, optimization, association and interaction and reuse or sharing) were added to the revised model.
- From the third model engineering design steps (identify the need or problem; research the need or problem; develop possible solution(s); select the best possible solution(s); construct a prototype; test and evaluate the solution(s); communicate the solution(s); redesign) were added to the revised model.

The revised model also explains the relations between the various aspects mentioned above, which is an important part of the design of the learning process.

Further research is needed to investigate in more depth each of the five core CT concepts and to create instruments for evaluating the various concepts of CT. The relationships between the various elements and steps could be further researched. The model of learning CT is designed in such a way that it could be used in different subjects tackling computational problems. The activities for integrating the concepts of CT into the comprehensive school level classroom using various scenarios in lessons could be created and the effectiveness of the activities developing the awareness and knowledge of the CT principles could be researched. The revised model of learning CT is a theoretical one and we suggest creating scenarios of learning CT to test parts of it empirically in the future.

REFERENCES

Aho, A. V. (2012). Computation and computational thinking. Computer Journal, 55(7):832 – 835.

Brennan, K., & Resnick, M. (2012, April). New frameworks for studying and assessing the development of computational thinking. In Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada.

Cooper, S., Prez, L. C., and Rainey, D. (2010). Education k-12 computational learning. Communications of the ACM, 53(11):27.

Daily, S. B. and Eugene, W. (2013). Preparing the future stem workforce for diverse environments. Urban Education, 48(5):682 – 704.

Denning, P. J. (2009). The profession of it: Beyond computational thinking. Communications of the ACM, 52(6):28. Artificial Intelligence, Philosophy, and Cognitive Science, 21(2):301 – 322.

Glass, R. L. (2006). Call it problem solving, not computational thinking. *Communications of the ACM*, 49(9):13.

Grover, S. and Pea, R. (2013). Computational thinking in k12: A review of the state of the eld. *Educational Researcher*, 42(1):38 – 43.

Hoffmann, L. (2009). Qa: The upper limit. Communications of the ACM, 52(1):112.
Katai, Z., Toth, L., and Adorjani, A. K. (2014). Multisensory informatics education. Informatics in Education, 13(2):225.
Kilpelainen, P. (2010). Do all roads lead to rome? (or reductions for dummy travelers). Computer Science Education, 20(3):181 – 199.
Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., & . Werner, L. (2011). Computational thinking for youth in practice. ACM Inroads, 2(1), 32. doi:10.1145/1929887.1929902
Lee, T. Y., Mauriello, M. L., Ahn, J., and Bederson, B. B. (2014). Ctarcade: Computational thinking with games in school age children. International Journal of Child-Computer Interaction, 2:26 – 33.
Lye, S. Y., & Koh, J. L. (2014). Review: Review on teaching and learning of computational thinking through programming: What is next for K-12?. Computers In Human Behavior, 4151-61. doi:10.1016/j.chb.2014.09.012
Marina U., B. (2011). The tangiblek robotics program: Applied computational thinking for young children. Early Childhood Research and Practice, (2).
Manson, J. R. and Olsen, R. J. (2010). Diagnostics and rubrics for assessing learning across the computational science curriculum. Journal of Computational Science, 1(1):55.
Massachusetts Department of Education. (2006). Massachusetts science and technology/engineering curriculum framework. Retrieved from. Massachusetts Department of Education http://www.doe.mass.edu/frameworks/scitech/1006.pdf.
Miller, C. S. and Settle, A. (2011). When practice doesn't make perfect: E
ects of task goals on learning computing concepts. ACM Transactions on Computing Education, 11(4).
Navlakha, S. and Bar-Joseph, Z. (2011). Algorithms in nature: the convergence of systems biology and computational thinking. Molecular Systems Biology, 7:546.
OECD annual report. http://www.oecd.org/newsroom/40556222.pdf, 2008.
Repenning, A. (2012). Programming goes back to school. Communications of the ACM, 55(5):38.
Sengupta, P., Kinnebrew, J., Basu, S., Biswas, G., and Clark, D. (2013). Integrating computational thinking with k-12 science education using agent-based computation: A theoretical framework. Education and Information Technologies, 18(2):351.
Shell, D. F. and Soh, L.-K. (2013). Profiles of motivated self-regulation in college computer science courses: Differences in major versus required non-major courses. Journal of Science Education and Technology, 22(6):899 – 913.
Thiruvathukal, G. K. (2009). Computational thinking and doing. *Computing in Science and Engineering*, 11(6):4.
Tsalapatas, H., Heidmann, O., Alimisi, R., and Houstis, E. (2012). Game-based programming towards developing algorithmic thinking skills in primary education. Scientific Bulletin of the Petru Maior University of Targu Mures, 9(1):56 – 63.
Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3): 33 – 35.
Wenchong, S., Maohua, L., & Hendler, P. (2014). Computational Features of the Thinking and the Thinking Attributes of Computing: on Computational Thinking. Journal Of Software (1796217X), 9(10), 2507. doi:10.4304/jsw.9.10.2507-2513
Wolz, U., Stone, M., Pearson, K., Pulimood, S. M., and Switzer, M. (2011). Computational thinking and expository writing in the middle school. ACM Transactions on Computing Education, 11(2):1.

Appendix 1

No	Author	Setting	Research type	Intervention	Concept, practice, perspective		
					C	Pr	Pe
1	(Lye & Koh, 2014)	High s.	Survey	Conc. model of CT		X	
2	(Israel et al., 2014)	Teacher	Survey	Teaching CT		X	
3	(Berkaliev et al., 2014)	Uni.	Case study	CT in mathematics		X	
4	(Wenchong et al., 2014)	General	Survey	Conc. model of CT		X	
5	(Gynnild, 2014)	General	Survey	CT in journalism			X
6	(Katai et al., 2014)	General	Experiment	Multi-sensory approach to CT	X		
7	(Miller, 2014)	Uni.	Case study	Programming	X		
8	(Harlow & Leak, 2014)	Elem. s.	Case study	Programming	X	X	
9	(Bers et al., 2014)	Kinderg.	Case study	CT in robotics		X	
10	(Lee et al., 2014)	Prim. s.	Case study	Game design	X	X	
11	(Vakil, 2014)	Mid. s.	Experiment	Cognitive aspects in CT		X	
12	(Shell & Soh, 2013)	Uni.	Case study	Motivation of students		X	
13	(Libeskind-Hadas & Bush, 2013)	Uni.	Case study	CT in biology		X	
14	(Daily & Eugene, 2013)	Kinderg.	Experiment	Story based CT	X	X	
15	(Sengupta et al., 2013)	High s.	Experiment	Agent-based CT		X	
16	Kostadinov, 2013	Uni.	Survey	CT in math		X	
17	(Aiken et al., 2013)	Mid. s.	Case study	CT in physics		X	
18	(Rex & Ruben, 2013)	Uni.	Survey	Programming		X	
19	(Caballero et al., 2012)	Uni.	Survey	Programming	X		
20	(Tsalapatas et al., 2012)	Prim. s.	Experiment	CT in game creation	X		
21	(Navlakha & Bar-Joseph, 2011)	Uni.	Survey	CT in biology		X	
22	(Miller & Settle, 2011)	Uni.	Case study	Project-based learning		X	
23	(Wolz et al., 2011)	Mid. s.	Case study	Story based learning		X	
24	(Dodig-Crnkovic, 2011)	General	Survey	CT in computation			X
25	(Marina U., 2011)	Kinderg.	Survey	CT in robotics	X		
26	(Lee et al. 2011)	Elem. s.	Survey	Model of CT	X		
27	(Kilpelainen, 2010)	Uni.	Survey	Reduction in CT	X		
28	(Manson & Olsen, 2010)	Uni.	Survey	Rubrics in CT		X	
29	(Cooper et al, 2010)	Elem. s.	Survey	Model of CT	X		
30	(Molina & Mason, 2009)	Elem. s.	Experiment	CT in math	X		
31	(Ioannidou et al., 2009)	Mid. s.	Case study	Game creation in CT		X	

Table 1: Articles found from EBSCO search

Tauno Palts is an Assistant in Didactics of Informatics in the University of Tartu and has started doctoral studies in informatics in the Institute of Computer Science.

Acknowledgments

This research was supported by the European Union through the European Regional Development Fund. It is financed in the project 'Conceptual framework for increasing society's commitment in ICT: approaches in general and higher education for motivating ICT-related career choices and improving competences for applying and developing ICT'. In addition, this research was supported by European Social Fund's Doctoral Studies and Internationalisation Programme DoRa, which is carried out by Foundation Archimedes.

Classcraft: from gamification to ludicization

Eric Sanchez, eric.sanchez@ens-lyon.fr
EducTice, Institut français de l'éducation, ENS de Lyon

Shawn Young, shawn@classcraft.com
CEO Classcraft

Caroline Jouneau-Sion caroline.jouneau_sion@ens-lyon.fr
EducTice, Institut français de l'éducation, ENS de Lyon

Abstract

In this article, we discuss the concept of gamification, based on a literature review and preliminary feedback from teachers using Classcraft, a role-playing game supported by a digital platform and a mobile application that were developed to answer high school teachers' classroom management needs. We argue for the use of the term "ludicization" to emphasize that transforming a situation into a game doesn't consist of using elements that have a game-like aspect, but rather of a non-essentialistic vision of games, generating a metaphor around the situation to build a reflexive space within which the nature and meaning of interactions are modified.

KEYWORDS

Classcraft, gamification, ludicization, high school education

INTRODUCTION

Classcraft is a role-playing game that was developed for classroom management at the high school level. The game is now available as a mobile and web application in which a teacher can sign up his students. Teachers can make teams and assign an avatar to every student, as well as points and "powers" as rewards for proper classroom conduct. Thus, the objective is to transform the manner in which students experience the act of coming to class by adding a playful dimension.

The term *gamification* is generally used to describe the process in which one integrates aspects of play into a situation that is initially not playful. However, in this article, we argue for the use of the term *ludicization*, following from the idea that it's less about "making a game" (*gamify*) than it is about "making it possible for a situation to be seen as ludic" (*ludicise*). Thus, in our first section, a brief literature review pushes us to argue for the use of the term *ludicization* and to expose the key components of this concept. A second section presents the game *Classcraft* and the experiments we have conducted around the game. In the last section, we present an analysis of the feedback of the first usage experiments of *Classcraft* and the elements we have decided to use to define the concept of *ludicization*.

FROM GAMIFICATION TO LUDICIZATION

Gamification, origin of a neologism

According to Deterding, Khaled, Nacke, and Dixon (2011), the word *gamification* appeared in 2008 in the digital media economic sector. It was popularized during different conferences (Google Tech Talk) by Zimmermann in 2010 and Amy Jo Kim in 2011 (Kapp, 2012). Thereafter, the word spread across the fields of academic research, marketing, and game design (Bonenfant&Genvo, 2014). Since, different definitions have been suggested: "Gamification is the use of game design elements in non-game contexts" (Deterding, Khaled, et al., 2011) or "using game-based mechanics, aesthetics, and game thinking to engage people, motivate action, promote learning, and solve problems" (Kapp, 2012).

Gamification should be distinguished from *ludification*, which means the spreading of games in culture, a phenomenon described by Henriot (1969) before the development of digital technologies. *Gamification* is applied to various fields such as urban architecture or employees' relationships in companies, but the concept flourishes for the web interfaces design sector. Therefore, whether it is for catching the attention of consumers or building the loyalty of digital social network users, gamification is an economic approach of attention (Goldhaber, 1997). This approach aims at optimizing the mental engagement of an individual, ordinarily for economic purposes.

Etymologically, the word *gamification* is based on the Latin word *facere*, which reflects the idea that it is possible to "make the game." Therefore, *gamification* is considered to be an automatic and non-problematic transformation (Silva, 2013). In their article published in 2011, Deterding*et al*. improve their first definition by underlining that *gamification* is "the use (rather than the extension) of design (rather than game-based technology or other game-related practices) elements (rather than full-fledged games) characteristic for games (rather than play or playfulness) in non-game contexts (regardless of specific usage intentions, contexts, or media of implementation)" (Deterding, Dixon, Khaled, &Nacke, 2011). This definition suggests that some specific elements belong to games. However, these authors also consider the experience of the player so that *gamification* would consist in addressing playfulness (the experiential and behavioral dimensions), and in using these dimensions for the design of structures with ludic affordances (*Ibid.*).

Ludicization, play vs. Game

Bonenfant and Genvo emphasize that *gamification* "consists in adopting an essentialist approach of ludic phenomenon" (Bonenfant&Genvo, 2014). Therefore, with the support of the seminal work of Henriot (1969), Genvo (2013) proposes to adopt the word *ludicization* in order to focus our attention not on the artefact but on the situation that takes place when an individual accepts to play. We adopted a similar approach in a previous work dedicated to develop a theoretical model of play for educational purposes (Sanchez &Emin Martinez, 2014).

According to this model, there is no specific game element that can be used to make a game (*gamification*), but it is possible to subtly combine elements in order to design a learning context where play can take place. We consider that the term *ludicization* is more appropriate when it comes to design a learning situation that combines educational purposes and ludic characteristics. Indeed, *ludus*, the latin root of *ludicization*, means both game and school work. In addition, the suffix "icization" does not mean that it is possible to "make" the game as suggested by the suffix "fication" (facere) of *gamification* but mainly that it is possible to transform the situation (Sanchez, 2011). Indeed, play emerges from an intention, and it is not "in the materiality of objects, in the factuality of gestures, that we have some chance to find ludic elements" (Henriot, 1889). As a result, our approach leads us to inscribe the issue of game-based learning in the existentialist philosophy and to focus our attention on the behavior of the player within a frame that enables for its autonomy.

Aims and scope

This article proposes a discussion of the *gamification* concept based on an empirical study. We describe *Classcraft*, a platformthat facilitates the *ludicization* of classroom management. We analyse preliminary results of two experiments in France and Quebec in order to show how *ludicization* allows the teacher to manage classroom interactions. Our work also aims to identify the key elements that have been used for the *ludicization* process and their impact on student behavior.

CLASSCRAFT, A ROLE-PLAYING GAME FOR CLASSROOM MANAGEMENT

In this section, we describe the game and the context of our experiment in two schools in France and Quebec.

A multiplayer game

The objective of *Classcraft* is to transform the classroom into a role-playing game for the duration of the school year. For the teacher, the point is to foster desired behavior in students. Indeed, it is the positive behavior of students that allows them to progress in the game. For the student, the goal is to gain levels and thus acquire powers, to make their avatar progress and support their team.

Inspired by role-playing video games or RPGs (for example, *World of Warcraft*), the first version of *Classcraft* was conceived of by Shawn Young in January 2011. The first version of the digital platform, which was very basic, was built for personal use. Three years were then spent improving the rules. The first public version was made available in February 2014 as a beta version. The official global launch of the game was in August 2014.

Fig. 1. Screen capture of the game

Classcraft is not related to a specific school subject and the duration of the game depends on the teacher's expectations (from a few class hours to the entire year). The students play the game during schools hours and outside of class. In *Classcraft*, students are placed in teams of four to six members and play as mages, warriors, or healers. Based on their character class, they gain access to powers they can use as they see fit (as long as they have sufficient action points, or AP). These powers are either related to game mechanics (heal another player, protect another player, regenerate action points, etc.) or to privileges having an impact on players' real lives (being allowed to eat in class, listen to one's iPod in class, hand in an assignment a day later, etc.). These powers are either beneficial to the individual or to the individual's team. Thus, players want to acquire these powers to help themselves and their team.

In order to acquire powers, the player must demonstrate behavior that is expected of him by the school, such as participating in class, helping other students, etc. These actions are rewarded by experience points (XP), which are distributed by the teacher, who plays the role of game master. These points enable players to level up and acquire powers and gold pieces (GP) to customize the appearance of their avatar. However, if a player exhibits behavior that is inappropriate, such as arriving to class late or not doing one's classwork, the teacher can remove health points (HP). If a player loses all of his HP, the player acquires a sentence and all of his teammates also lose HP. The sentences are real-life punishments, such as detention, copying a text, and so on. When players use their powers to help teammates, they are automatically awarded XP. Thus, students are rewarded for helping teammates and penalized when their fellows behave inappropriately too often.

Every class, the system generates a random event, which has an impact on gameplay (for example, “Everyone loses 10 HP”) or classroom dynamics (for example, “Everyone must speak like a pirate for the day”). These events are random and affect the entire class. Like the powers, sentences, positive actions, and negative actions, these events can be completely customized by the teacher to adapt the game to their specific classroom setting.

Because these aspects have a direct impact on the real lives of the players, it is important for the teacher to customize them so that they are adapted to his students and classroom setting. For example, one of the default powers is to be able to listen to music during class work. However, in certain schools this isn’t possible (or permitted), and the teacher can then alter the power to change its effect.

Classcraft is first and foremost a web application (it operates in a browser connected to the Internet). To play, the teacher projects the application in front of the classroom and manages all aspects of school life. In a setting where students have access to electronic devices, they can connect to the platform and customize their avatar, activate powers, and access classroom content. One can also play *Classcraft* on smartphones and tablets, by using the Android and iOS apps. Thus, the game consists ofadding digital elements to the classroom, ludicizing real life interactions as they occur, without influencing the subject matter.

A large diffusion across the world

Since it launched in August 2014, *Classcraft* has gained rapid usage by many teachers. Indeed, as of February 1, 2015, more than 1,500 teachers were using *Classcraft* in more than 60 countries (eight languages). This represents over 60,000 students connecting regularly to the platform, with a total of more than 15,000 daily logins (students and teachers). This does not include classroom settings where students don’t actually connect to the platform. A class is considered active if more than 50 game events, concerning at least five students, were recorded in the previous month. If we take into account inactive accounts, more than 150,000 accounts have beencreated in the platform since its launch. Also, more than 1.1 million game events (using powers, losing HP, gaining XP, etc.) occur each month. The following graphs show constant growth, from September to December (the drop in the last week can be explained by the Thanksgiving holiday in the United States).

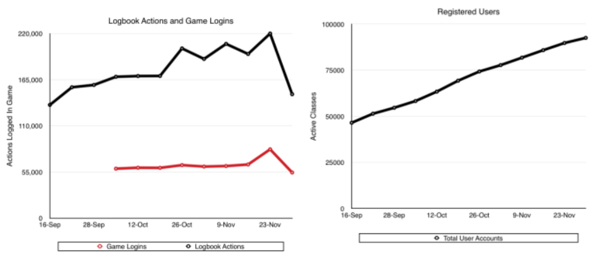


Fig. 2. Traction since September 2014

This data shows that *Classcraft* has gained approval from its market and meets the needs of numerous teachers.

Elements of play in Classcraft

The design of *Classcraft* rests on the combination of different game elements described by Caillois [12]. First of all, *Classcraft*, directly inspired from massively multiplayer online role-playing games (MMORPGs) such as *World of Warcraft*, is itself a role-playing game (*mimicry*). An avatar represents each player. *Classcraft* also leverages competition (*agon*). This competition exerts itself against the game itself, which, based on one's behavior, leads to gaining or losing points. It also exerts against the entire class because the points allow one to advance in relation to one's classmates.

Another gameplay element that is leveraged in *Classcraft* is that of randomness (*alea*). Indeed, every class starts with a random event that has an impact on the entire class. Randomness also manifests itself when, having lost all of their HP, the player must throw the "cursed die," which can have profound consequences like detention. To a degree, this die can lead to a feeling of vertigo (*ilinx*), another element identified by Caillois.

Thus, the design of *Classcraft* encompasses all of the gameplay elements formalized by Caillois, and this most likely explains some of its success. Indeed, the design of *Classcraft* doesn't limit itself to mobilizing *gamification* elements (avatars, points, etc.) but consists ofthe combination of multiple gameplay elements to create a situation in which the student will find a favorable context to develop a playful attitude.

Context and data collected

Other than the results pertaining to traction within a global market, the data we have pertains more specifically to two experiments that began in September 2014 and continue for the duration of the school year. The first one is in a history-geography class (32 students) in grade 10 at Germaine Tillionlycée in Sain Bel (Rhône, France), and the second one is in two physics classes (66 students) in grade 11 in Sherbrooke (Quebec, Canada).

In both cases, the schools have approximately 800 students in a well-off social context. In the case of the French experiment, the class of 35 students had a group of 10 undisciplined, talkative youths who recognized themselves as having inappropriate classroom behavior but stated that they were unable to controlthemselves. Many of these students said that they are badly oriented, and many are anxious aboutfuture academic challenges as they arrive ata new school where many of their classroom peers are strangers. In the case of the Canadian experiment, the students are generally academically successful and have access in class to a personal portable computer that is connected to the Internet.

The approach to this study is based on an ethnographic methodology (Whitehead, 2004). The participation of the two teachers (co-authors) involved in giving the course, allowed for a holistic approach to the case study design. The aim was to understand how the implementation wascarried out in the two contexts and which elements are involved in the the*ludicization* of classroom management.The data collected encompassed *in situ* observations and feedback from the teachers.

LESSONS LEARNED

In this section, we describe the way the game wasplayed in the two contexts, and we analyse the results of these experiments.

Description of two experimentations

To experiment with*Classcraft* in her classroom in Sain Bel (France), the teacher followed the tips given on the website: Once they are introduced to the game, the pupils can choose to play or not. All the pupils agreed to play and were asked to choose their five teammates following the teacher's instructions: Each team should contain different kinds of pupils, slow achievers, and good learners, with or without behavioral difficulties. It should be noticed, however, that this rule has been only partially followed and many of the groups were rather homogeneous.

The parents have been informed, and they have shown some interest in the game, if not being totally enthusiastic about the project. As advised by the headmaster, some rules of the game have been modified for compliance with the current code of conduct of Sain Bel. He particularly insisted on the fact that all pupils in Sain Bel must be treated equally. What is explicitly prohibited by the current code of conduct—such as using mp3 players, audiphones, and eating in class—cannot be allowed in the game. The modifications have been designed by the pupils themselves during dedicated class sessions. Some powers that violate the code of conduct (eating in class, playing music) have been replaced by equivalent and compliant ones. For instance, "eating in class" has been changed to "eating a sweet," since eating in class is prohibited on the ground that it may increase the need for cleaning, and "eating a sweet" provides an equivalent pleasure to"eating in class." "Listening to a music player" is a personal power that provides the pleasure through transgression of a rule, common to many classes. However, it is also prohibited to avoid trafficking and out of courtesy for other people. Pupils have proposed to replace this power with a very similar one, namely "listening to music," which makes no explicit mention to prohibited devices, such as "audio/video players." Then a "debating session," whose subject was "Classcraft, a perfect game for the lycée Germaine Tillion," took place among the pupils.

The game is now played each lesson, but the teacher faces some technical difficulties: the computer in the classroomcannot be used to show the game platform because it is filtered out by the firewall. For two months, the teacher used the mobile application, disturbing both the class (because she was forced to look at the phone instead of the pupils) and the game (because she was unable to show the website). She is currently using her own computer together with a mobile phone connection to show the website and manage the computationof the points.

As for the experimentation in Sherbrooke, the teams have been chosen by the teacher himself, based on the previous studentachievements, so as to obtain balanced teams, in terms of scholarly performance. The default rules have been used. Neither the parents nor the administrative staff have been explicitly informed about the game. They did not interfere during the experiment. The game has been played duringeach lesson, without technical problems, and the pupils accessed the platform using their personal computers.

Feedback on the Sain Bel experimentation

The teacher stressed that the game is an efficient way to enhance motivation about scholarly work: Work groups are often built as in the game, and the accomplishment of the scholarly productions is greater, especially for slow achievers. Oral participation also increased in particular because the teacher intensified the usage of Classcraft to reward actions. Pupils ask about their points, albeit not systematically, and they work hard to get them.

Nevertheless, behavioral problems have not disappeared completely. It highlights the fact that the teacher's role remains complex even in a playful situation. When the teacher acts as game master, their role is not deeply changed. The assignment of the points is not automatic: The teacher is still the one who evaluates and punishes. But the nature of the punishment changes, explicit rules constrain the assignment of the points, and positive actions induce explicit positive feedbacks.

An important goal of the game is to build upon collaboration between pupils to induce betterbehavior from those who show frequent misconduct.But collaboration between pairs, understood here as working towards a common achievement, remains problematic. It often takes the form of mutual assistance for scholarly work. It exists outside the strict game context, through direct as well as remote relationships (phone, Facebook). Involved pupils didnot think to claim their points for such mutual assistance. Also, misconduct in class did not always lead to collaboration. Indeed, watchful pupils sometimes failedto show their teammates the correct way to take part or speak in the class. Also, when it comes to choosing powers, individual powers were strongly preferred to collective ones. With the teacher being reluctant to withdraw health points, teams seldom stick together. Since then, the teacher has intensified usage of the game, including for punishments. Thishas induced the pupils to rely more on collective powers. The players are now eager to buy collective powers.

Feedback on the Sherbrooke experimentation

In Sherbrooke the game is played every lesson, so about 3 times a week per group. As in Sain Bel, the teacher reportedincreased motivation and deeper engagement in class work.Pupils tend to show more participative behavior, in every dimension of the lesson: to answer questions and to work in class. They want to claim points.

Pupils on the same team are more united than in the Sain Bel case. They help team members when they lose health points. Usage of power as a mutual assistance device is common, and they show the ability to self-govern. As the teacher often withdraws health points, pupils often feel unsafe, so they must develop survival strategies or modify behaviors not to lose health points (and gain experience points).

Also, computer access to the game allows for interactions between pupils without disturbing the course. They then have more opportunities to assist one another, to visit their status and train their partners. They then show more interest for the game since their interactions are more frequent, even on a voluntary basis.

CLASSCRAFT, A SPACE FOR REFLEXIVITY

In this section, we offer an analysis of the game and observations based on the concept of *ludicization* as defined in the first section.

From simulation to metaphor

Games are generally based on a model that allows for simulating a reference situation. The term “simulation” refers to the idea of an “experience of second kind” in contrast with a “first kind,” which is about the “immediate experience” (Varenne, 2006). Therefore, simulation is a field for experimentation that allows living a true empirical experience. This aspect of simulation is very often highlighted by authors who are interested in learning with digital games. It is also used to design games for educational use. However, in the case of *Classcraft*, the idea of simulation doesn’t account for the environment developed in the game. The game is also to be seen as a trope (Sutton-Smith, 1997). The expressed ideas are interpreted differently in order to build an imaginary world. There is an analogical relation between elements of the game and those of reference situation. For example, in *Classcraft*, mutual educational support is represented as powers that the healer can use to “heal” teammates or exclude them by pushing them to “fall in battle.” Simulation becomes a metaphor with a hidden meaning, that is of acceptable academic behavior, which is behind the imaginary world of *Classcraft*.

Furthermore, the distance between the metaphor and the reference situation it accounts for, that is the second degree in the game, gives it power and ontological significance because as in literature, a metaphor in a game captures the essence of a situation that it describes. Therefore, metaphors are a refined form of the reference situation, and the player is led to focus on the core of the situation. *Classcraft* can be considered as a way to metaphorize the functioning of a classroom as a battle combining collaboration and competition.

Game appropriation and engagement

Another important game dimension is the ability to encourage the involvement of students. In this case, this goal is achieved by transforming educational goals into play goals. Thus, decoding the teacher's expectations becomes easier. The goals for students are clear since it is not about behaving in class anymore but about interacting according to the game rules in order to earn points. The devolution of the teachers' goals is made easier because the game changes the meaning of their goals, and the game rules are a simple way to put those expectations into words.

Moreover, each player is represented by an avatar, which is a projective identity in two different ways. First, it allows the players to have a self-experience through introjection. For example, they are led to check the relevance of the decisions they make by earning or losing points. Second, the avatar, an emblematic figure of a warrior, a mage, or healer, becomes the projection of an identity that is being built and an experimentation field that allows it to be built. Roles that students play help them get involved in the situation.

Feedback and sense of competence

In *Classcraft*, feedback taking the shape of earned or lost points or powers is the response given by the game environment to the players' actions. They are not only reinforcement modalities used to design games based on a behavioristic approach, as described by Block & King (1987). They are information with potentially high semantic content that have to be analyzed and interpreted in order to rethink the implemented strategies if necessary. From a feedback perspective, what makes this game different from a regular class situation is that this feedback is continuously generated (Mayo, 2006). The game offers the students a space of liberty in making decisions. It also gives them information about the consequences of their choices, information that is necessary for the decision-making process. At any moment, players can judge the relevance of the decisions they are making. Giving feedback is made possible by the fact that the teacher is constantly collecting information about the players' actions and therefore about their ability to follow the classroom rules. So, the game provides the right environment for developing autonomy because it offers the players the liberty of choice and action as well as information, in the form of feedback, which allow them to practice their liberty of choice and action. The game is in this case considered as a space for reflexivity.

Moreover, the instant feedback increases the students' feeling of competency. Indeed, losing points or even "death" in the game is feedback that can be perceived as negative play-wise. However, because of the ludic context, negative consequences are less severe. It is always possible to go forward by carrying out actions to earn points and "resurrect" in the game. This negative feedback doesn't alter the feeling of being competent while positive feedback such as earning points or evolving in the charts increases the feeling of competency. This point is very important to note since the feeling of being competent is a key aspect of academic motivation (Ryan &Deci, 2000).

CONCLUSION

The observations that we were able to make show that, depending on the setting, the game is not experienced in the same way by students. These observations promote a model that consists ofconsidering the experience of the students, rather than the game itself. This experiment seems to depend on a multitude of factors, among which we've identified the institutional acceptability of the game, the equipment available in class, and the way the teacher presents and implements the game. Among these different factors, the role of the teacher and his own appropriation of the game are keys.

These elements considered, *ludicization* consists ofa reconfiguration of the class setting. This *ludicization* translates itself in the implementation of new interactions. In the game *Classcraft*, students are, for example, led to make decisions to “save” other students. Nevertheless, it is mainly the meaning of normal interactions within the classroom that is redefined. It isn’t about the student adopting behavior to conform to the class rules but rather about adopting behavior that, because it takes into account the rules of the game, leads to progressing within it. This progress materializes itself in points or other elements that then can be visualized in the platform. In *Classcraft*, the classroom rules are translated into arbitrary game rules. In this sense, we can say that *Classcraft* is a metaphor of class life. Indeed, it constitutes a refined version of the reference situation, and the player is incentivized to bring his attention to what is at the heart of the situation. This metaphor allows for the implementation of a space for reflexivity within which the player can test his ways of behaving because his decisions translate into immediate feedback. This space for reflexivity thus fosters autonomy.

Ludicizing doesn’t consist ofusing game elements in a mechanical way, but rather, with a non-essentialist vision of the game, in metaphorizing a situation to conceive of a reflexive space where the nature and the meaning of interactions are modified. Nevertheless, from this report stems the question of the emancipation of the player, who, in accepting to play the game, accepts to trade his freedom for a freedom constrained by the arbitrary rules that are the game rules (Duflo, 1997).

REFERENCES

Block, JH, & King, NR. (1987). *School play*. New York, NY: Garland.

Bonenfant, M, &Genvo, S. (2014). Une approche située et critique du concept de gamification. *Sciences du jeu, Questionner les mises en forme ludiques du web : gamification, ludification et ludicisation*(2).

Caillois, R. (1958). *Des jeux et des hommes. Le masque et le vertige.*(1967 ed.). Paris: Gallimard.

Deterding, S, Dixon, D, Khaled, R, &Nacke, L. (2011). *From Game Design Elements to Gamefulness: Defining “Gamification”*. Paper presented at the MindTrek’11, Tempere, Finland. http://www.sciencesdujeu.org/index.php?id=7278

Deterding, S, Khaled, R, Nacke, L, & Dixon, D. (2011). *Gamification: Toward a definition.*CHI 2011 Gamification Workshop Proceedings, Vancouver, BC.

Duflo, C. (1997). *Jouer et Philosopher*. Paris: Presses Universitaires de France.

Genvo, S. (2013). Penser les phénomènes de ludicisation à partir de Jacques Henriot. *Sciences du jeu* (1).

Goldhaber, M.H. (1997). The Attention Economy on the Net.*First Monday, 2*(4).

Henriot, J. (1969). *Le jeu*. Paris: Presses Universitaires de France.

Henriot, J. (1989). *Sous couleur de jouer - la métaphore ludique*. Paris: José Corti.

Kapp, KM. (2012). *The gamification of learning and instruction*. San Francisco, CA: Pfeiffer.

Mayo, MJ. (2009). Video Games: A Route to Large-Scale STEM Education? *Science, 323*(5910), 79-82.

Ryan, R M, &Deci, E L. (2000). Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being.*American Psychologist* (55), 68-78.

Sanchez, E. (2014). Learning, serious games, and gamification. *Inmedia*. http://inmedia.revues.org/814

Sanchez, E, & Emin Martinez, V. (2014). Ils jouent, mais quel jeu jouent-ils ? Du jeu sérieux au jeu épistémique *actes de la conférence TICE 2014* (pp. 25-36). Béziers.
Silva, H. (2013). La « gamification » de la vie : sous couleur de jouer ? *Sciences du jeu - 30 ans de Sciences du jeu à Villetaneuse. Hommage à Jacques Henriot, 1*(1).
Sutton-Smith, B. (1997). *The Ambiguity of Play*. Cambridge, M: Harvard University Press.
Varenne, F. (2006). *Les notions de métaphore et d'analogie dans les épistémologies des modèles et des simulations*. Paris: Petra.
Whitehead, T.L (2004). "What is Ethnography? Methodological, Ontological, and Epistemological Attributes". *EICCARS Working Paper*. University of Maryland, CuSAG.

Eric Sanchez is assistant professor, head of EducTice, a research team of the French Institute of Education (Ecole Normale Supérieure de Lyon). He is also adjunct Professor at the University of Sherbrooke (Canada). His research focuses on the use of digital technologies for educational purposes (elearning, simulation, serious games).

Shawn Young is the founder of Classcraft. With a unique background in physics, education, gamification and web development, he's interested in how we can create community in the classroom and how games can make learning fun and interactive. He holds a bachelor's degree in physics and master's in education from Université de Sherbrooke.

Caroline Jouneau-Sion is a secondary history and geography teacher. She is involved in research & development projects into the uses of digital technologies for secondary education at the EducTice, a research team of the French Institute of Education (Ecole Normale Supérieure de Lyon).

Evidence of assessing computational thinking

Cynthia Selby, *C.Selby@southampton.ac.uk*
University of Southampton, UK

Mark Dorling, *Mark.Dorling@computingatschool.org.uk*
Computing At School and BCS, Chartered Institute for IT, UK

John Woollard, *J.Woollard@southampton.ac.uk*
University of Southampton, UK

Abstract

Computational thinking is at the heart of the new English national curriculum for computing. There is a range of academic and pedagogic interpretations of the concept of computational thinking, a lack of understanding of the concepts and a close association of the subject with writing computer code using a programming language. Teachers might focus on a small aspect of the programme of study, thereby neglecting the breadth of content and the broader aims. In addition, the level descriptors associated with the curriculum have been removed creating a need for assessment guidance. In light of these changes, this paper explores the statutory requirements of the curriculum and the descriptions of computational thinking. It suggests a mechanism for assessment of achievement and progression for both computing and computational thinking.

Keywords

Assessment, computational thinking, computing, framework, theory

INTRODUCTION

From September 2014, pupils in state-maintained schools will be expected to follow the programmes of study set out in the national curriculum document (Department for Education (Department for Education, 2013b). This document addresses the subject of computing. In addition, the statutory assessment framework is being removed and the system of assessment levels is not to be replaced (Department for Education, 2013a).

Computational thinking sits at the heart of the national curriculum programme of study for computing. The opening sentence states “A high quality computing education equips pupils to use computational thinking and creativity to understand and change the world” (Department for Education, 2013b, p. 188). The scope of computational thinking is described in the first aim – “understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation” (Department for Education, 2013b, p. 188). There are many different interpretations of the concept of computational thinking. Jeanette Wing, when she first used the term, defined computational thinking as including “… a range of mental tools that reflect the breadth of the field of Computer Science” (Wing, 2006, p. 33).

However, there is a strong emphasis, being led by the media, implying that the new computing curriculum focuses on ‘coding’ (Crow, 2014; Nettleford, 2013). This misleading message, received by teachers and parents, could have a negative impact in the classroom. There is a danger of teachers focusing on a small aspect of the programme of study, thereby neglecting the breadth of the subject content and the broader aims.

Computational thinking is itself in danger of becoming a ‘buzz word’ in the teaching of computing. Teachers acknowledge the need to teach computational thinking but may struggle with the various and conflicting interpretations of its nature. This may be the result of debate by individuals and groups (Computer Science Teachers Association (CSTA), 2011; Henderson, et al., 2007; Lu & Fletcher, 2009; Naughton, 2012; Wing, 2006; Wing, 2008; Yadav, Zhou, Mayfield, Hambrusch & Korb,2011) concerning what is and what is not computational thinking. Some of these definitions are broad, overlapping other subjects (Bundy, 2007; Computer Science Teachers Association, 2011). In order to facilitate incorporation of computational thinking into classroom practices, a narrower definition is required. Once computational thinking is defined adequately, appropriate assessment instruments can be designed (National Research Council, 2010).

Recent developments in pedagogy have focussed upon thinking skills (Department for Education and Employment, 1999; Department for Education and Skills, 2002; Wickens, 2007) as underpinning areas of the curriculum. ‘Thinking Hats’, based on de Bono’s work (de Bono, 2000; de Bono, 2007), is a popular approach in which pupils are encouraged to think about the way they think. The computing curriculum is now challenging pupils to think using particular strategies for solving problems and understanding situations, referred to as computational thinking. There are a number of stages towards establishing a curriculum in which computational thinking can be taught and then assessed. These stages are:

- to establish an understanding of the current computing curriculum,
- to establish the meaning of computational thinking,
- to establish an assessment framework for the current computing curriculum, and
- to develop a method for evidencing the assessment of computational thinking.

CURRENT COMPUTING CURRICULUM

The programme of study has high-level aims in terms of the introduction of computer science (Department for Education, 2013b). The following extracts illustrate learner capabilities at different stages of primary and secondary education.

- At key stage 1 (ages 5-7), pupils should be able to “understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions” (p. 189).

- At key stage 2 (age 7-11), pupils should be able to (among other things): “solve problems by decomposing them into smaller parts” and also “use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs” (p. 189).
- At key stage 3 (ages 11-14), pupils should be able to: “design, use and evaluate computational abstractions …” and “use logical reasoning to compare the utility of alternative algorithms for the same problem” (p. 190).
- At key stage 4 (ages 14-16), pupils should be able to “develop and apply their analytic, problem-solving, design, and computational thinking skills” (p. 191).

These extracts demonstrate an emphasis on the progressive development of computational thinking skills. Teachers in England, engaging with the new programme of study, are now frequently hearing the term computational thinking and may question what it means to them as classroom practitioners.

Along with the move toward computational thinking, there is a withdrawal from the use of national standardised levels and level descriptors. Under the auspices of the Department for Education (2013a), schools are now free to design their own assessment models. There are many reasons for this move, including the suggestion that assessment leads the teaching (Barker, 2013; Passmore, 2007; Warner, 2008).

COMPUTATIONAL THINKING CONCEPTS

Jeanette Wing broadly defines computational thinking as "… the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent" (Cuny, Snyder & Wing, 2010, cited in Wing, 2011, p.20). Wing indicates that these solutions can be carried out by any processing agent, whether human, computer, or a combination of both (Wing, 2006). The emphasis in this statement is on thought processes, not the production of artefacts or evidences.

Given Wing's description of computational thinking, the next step is to decompose that definition into a set of concepts. This work has been undertaken by Selby and Woollard (2013). The result refines the definition of computational thinking to six concepts: a thought process, abstraction, decomposition, algorithmic design, evaluation, and generalisation. All of these concepts are employed in problem-solving processes. Again, the emphasis in this list of concepts is on thought processes, not the production of artefacts or evidence.

COMPUTING PROGRESSION PATHWAYS

Although there is some disagreement concerning at what level a computing assessment framework should be developed, from a classroom practitioner's perspective, there is definitely a need for one. This section introduces the Computing Progression Pathways (CPP) and describes how it can be used to acknowledge progression and reward performance in mastering both the computing programme of study content and computational thinking skills.

There is some debate about whether it is important that the arbitrary values of progression be standardised across schools. Naace (Harrison, 2014), in their guidance, indicate "…a school approach to assessment will need to be tailored to match their approach to the curriculum" (p. 1). Alternatively, the National Association of Head Teachers (NAHT) propose (2014) when translating the national curriculum into assessment criteria "… there is little room for meaningful variety, we suggest this job be shared between schools" (p. 10). Whether it is designed by a single school or a collection of interested parties, an assessment framework is required by classroom practitioners.

The Computing Progression Pathways (Dorling and Walker, 2014) is an example of a non-statutory assessment framework. It was produced by a small team of authors and reviewers, all teachers, based on their classroom experiences. It is an interpretation of the breadth and depth of the content in the 2014 national curriculum for computing programme of study. It includes the dependencies and interdependencies between concepts and principles. This may help non-specialist teachers and inexperienced teachers to understand what should be taught in the classroom. The Computing Progression Pathways is publically available at this link: http://community.computingatschool.org.uk/resources/2324 . It is also included in Appendix A.

The framework is grid-based. Five of the six strands, represented as columns, are aligned with the range and content categories from the Computing at School curriculum (Computing at School, 2012) and the requirements of applicants to initial teacher training courses (Department for Education, 2012). These include algorithms, programming and development, data and data representation, hardware and processing, communication and networks. The sixth strand incorporates the more traditional concept of information technology. This breadth affords an opportunity to view the subject of computing as a whole, rather than the separate subjects of Computer Science, Digital Literacy, and Information Technology. Each row represents a level of pupil progression. Annotation of the framework suggests that key stages 1-2 cover the first four colours (pink, yellow, orange, and blue), that key stages 3-4 cover the next four colours (purple, red, and black), and that GCSE covers the final colour (white). As an example, the purple cell under the "Hardware and Processing" strand states that a pupil "Recognises and understands the function of the main internal parts of basic computer architecture" (Dorling and Walker, 2014).

The colour-coded rows can aid teachers in assessing whether pupils are exhibiting competences at different levels and in recognising achievement and attainment. In addition, adherence to the colour-coded statements can provide standardisation across schools as identified by the National Association of Head Teachers (2014). Institutions planning to use this assessment framework with existing assessment or reporting systems may:

- assign values or levels to the coloured rows,
- agree the benchmark value, level, or entry point for a particular key stage,
- assign the benchmark value or level to the appropriate progression statements.

The Computing Progression Pathways also affords opportunities to celebrate achievement in computing. There is a growing interest in badges as an informal recognition of skill, knowledge, understanding, or attitude. They are made and awarded by commercial organisations, educational suppliers, websites, schools, teachers, and pupils (Hamilton and Henderson, 2013; Mozilla, 2014; Radiowaves Schools, 2014). Recognising and rewarding pupil achievement in each strand can be accomplished via coloured digital badges. Each strand can be assigned a separate digital badge. There may be two-tone badges for pupils working between coloured progression levels. Currently, there are no digital badge designs for the strands. Teachers and pupils who will be using the digital badge system are better placed to design and create them. The process of designing and creating the digital badges might promote learner ownership and student-centeredness (Reigeluth, 2013).

EVIDENCE OF ASSESSING COMPUTATIONAL THINKING

Given that computational thinking concepts have been defined (Selby and Woollard, 2013) and an assessment framework for the computing programme of study has been proposed (Dorling and Walker, 2014), a mapping can be developed to illustrate how computational thinking can be assessed over the full breadth and depth of the computing programme of study.

The key to developing this mapping lies in understanding that computational thinking concepts can be demonstrated in multiple ways. For example, decomposition is demonstrated by pupils breaking game logic down into levels (avoid traps, climb mountain, guess password). This can be mapped to the "Programming & Development" strand, blue row. However, it can also be demonstrated by pupils designing a library inventory (an inventory grid for DVDs, a different grid for books). This can be mapped to the "Data & Representation" strand, yellow row. These examples illustrate decomposition in terms of functionality and data structures, across strands (breadth) and across rows (depth).

Rather than provide specific examples, tied to activities, for each statement in the Computing Progression Pathways that illustrate one or more computational thinking concepts, it is best to consider the meaning of the computational thinking concept and how it might apply to the pathways' statement. This affords the opportunity for classroom practitioners to contextualise the pathways and computational thinking concepts in any way they see fit.

As an example of this approach, consider the purple cell of the "Hardware & Processing" strand of the Computing Progression Pathways. It requires that a pupil "Understands the concepts behind the fetch-execute cycle" (Dorling and Walker, 2014). The fetch-execute cycle can be viewed as an algorithm Understanding of this demonstrates the computational thinking concept of algorithmic thinking. Therefore, at a minimum, this pathways' statement maps to the computational thinking concept of algorithmic thinking. Once this mapping is complete, it is possible to identify, across the breadth and depth of the programme of study, all those activities with potential to enhance computational thinking skills.

The following tables are a reproduction of the blue row (key stage 2 and key stage 3 border) of the Computing Progression Pathways (Dorling and Walker, 2014). Where applicable, the computational thinking concepts associated with that statement have been indicated in the last column. The computational thinking concepts of abstraction, decomposition, algorithmic thinking, evaluation, and generalisation have been indicated.

CPP Strand: Algorithms	
CPP cell statement	**Computational thinking concept**
Shows an awareness of tasks best completed by humans or computers.	Evaluation
Designs solutions by decomposing a problem and creates a sub-solution for each of these parts.	Decomposition, Algorithmic thinking, Abstraction
Recognises that different solutions exist for the same problem.	Algorithmic thinking, Abstraction

Table 1: Computational thinking mapped onto CPP (Algorithms)

CPP Strand: Programming & Development	
CPP cell statement	**Computational thinking concept**
Understands the difference between, and appropriately uses if and if, then and else statements.	Algorithmic thinking, Decomposition
Uses a variable and relational operators within a loop to govern termination.	Algorithmic thinking, Abstraction
Designs, writes and debugs modular programs using procedures.	Algorithmic thinking, Decomposition, Abstraction, Generalisation
Knows that a procedure can be used to hide the detail with sub-solution.	Algorithmic thinking, Decomposition, Abstraction

Table 2: Computational thinking mapped onto CPP (Programming & Development)

CPP Strand: Data & Data Representation	
CPP cell statement	**Computational thinking concept**
Performs more complex searches for information e.g. using Boolean and relational operators.	Algorithmic thinking, Evaluation
Analyses and evaluates data and information, and recognises that poor quality data leads to unreliable results, and inaccurate conclusions.	Evaluation

Table 3: Computational thinking mapped onto CPP (Data & Data Representation)

CPP Strand: Hardware & Processing	
CPP cell statement	**Computational thinking concept**
Understands why and when computers are used.	Not specified
Understands the main functions of the operating system.	Decomposition, Abstraction
Knows the difference between physical, wireless and mobile networks.	Abstraction

Table 4: Computational thinking mapped onto CPP (Hardware & Processing)

CPP Strand: Communication & Networks	
CPP cell statement	**Computational thinking concept**
Understands how search results are selected, including that search engines use ‘web crawler programs’.	Abstraction, Evaluation
Selects, combines and uses internet services.	Algorithmic thinking, Evaluation
Demonstrates responsible use of technologies and online services, and knows a range of ways to report concerns.	Not specified

Table 5: Computational thinking mapped onto CPP (Communication & Networks)

CPP Strand: Information Technology	
CPP cell statement	**Computational thinking concept**
Makes judgements about digital content when evaluating and repurposing it for a given audience.	Evaluation
Recognises the audience when designing and creating digital content.	Evaluation
Understands the potential of information technology for collaboration when computers are networked.	Evaluation
Uses criteria to evaluate the quality of solutions, can identify improvements making some refinements to the solution, and future solutions.	Evaluation

Table 6: Computational thinking mapped onto CPP (Information Technology)

Care has been taken by 3 iterations of expert evaluation of the statements to avoid making assumptions about how the teaching might afford opportunities for computational thinking rather than strictly interpreting what is explicitly stated in the Computing Curriculum Pathways. For example, an exercise in a classroom might afford opportunities to identify suitability for purpose and efficiency of input and output devices. That would fall into the yellow cell of the “Hardware & Processing” strand, where a pupil “Recognises and can use a range of input and output devices” (Dorling and Walker, 2014). The teaching affords the opportunity for evaluation, although the statement from the pathways does not indicate that it would be an evaluation-based exercise. The teaching of the fetch-execute cycle, previously mapped to algorithmic thinking, usually incorporates the ideas of instructions and data, which correspond to the concept of abstraction. The teaching affords the opportunity for abstraction, although the pathways' statement does not explicitly anticipate this.

Using this strategy of identifying computational thinking concepts associated with the pathways′ statements enables computational thinking to be assessed using the same framework as the programme of study. From a practitioner′s perspective, there is no additional assessment or progression tracking required to fulfil the broad aim of the computing programme of study to incorporate computational thinking.

CONCLUSION

The computing programme of study (Department for Education, 2013b) includes the broad aim of incorporating computational thinking into the classroom. The subject content is detailed in the document, but the connection to computational thinking and its meaning is not. Removal of the statutory assessment frameworks, which did not assess computational thinking, leaves a void in assessing pupils' attainment. Both of these shortcomings have been addressed in this paper. An understanding of computational thinking, based on the work of Selby and Woollard (2013), has been established. An assessment framework, the Computing Progression Pathways, has been used to illustrate the dependencies and interdependencies between the concepts and principles of the programme of study (Dorling and Walker, 2014). This work has demonstrated how the Computing Progression Pathways can now be used to evidence attainment in computational thinking directly.

Computing Progressio

Pupil Progression	Algorithms	Programming & Development	Data & Data Representation
Pink	• Understands what an algorithm is and is able to express simple linear (non-branching) algorithms symbolically. (AL) • Understands that computers need precise instructions. (AL) • Demonstrates care and precision to avoid errors. (AL)	• Knows that users can develop their own programs, and can • demonstrate this by creating a simple program in an environment that does not rely on text e.g. programmable robots etc. (AL) • Executes, checks and changes programs. (AL) • Understands that programs execute by following precise instructions. (AL)	• Recognises that digital content can be represented in many forms. (AB) (GE) • Distinguishes between some of these forms and can explain the different ways that they communicate information. (AB)
Yellow	• Understands that algorithms are implemented on digital devices as programs.(AL) • Designs simple algorithms using loops, and selection i.e. if statements. (AL) • Uses logical reasoning to predict outcomes. (AL) • Detects and corrects errors i.e. debugging, in algorithms. (AL)	• Uses arithmetic operators, if statements, and loops, within programs. (AL) • Uses logical reasoning to predict the behaviour of programs. (AL) • Detects and corrects simple semantic errors i.e. debugging, in programs. (AL)	• Recognises different types of data: text, number. (AB) (GE) • Appreciates that programs can work with different types of data. (GE) • Recognises that data can be structured in tables to make it useful. (AB) (DE)
Orange	• Designs solutions (algorithms) that use repetition and two-way selection i.e. if, then and else. (AL) • Uses diagrams to express solutions. (AB) • Uses logical reasoning to predict outputs, showing an awareness of inputs. (AL)	• Creates programs that implement algorithms to achieve given goals. (AL) • Declares and assigns variables. (AB) • Uses post-tested loop e.g. 'until', and a sequence of selection statements in programs, including an if, then and else statement. (AL)	• Understands the difference between data and information. (AB) • Knows why sorting data in a flat file can improve searching for information. (EV) • Uses filters or can perform single criteria searches for information.(AL)
Blue	• Shows an awareness of tasks best completed by humans or computers. (EV) • Designs solutions by decomposing a problem and creates a sub-solution for each of these parts. (DE) (AL) (AB) • Recognises that different solutions exist for the same problem. (AL) (AB)	• Understands the difference between, and appropriately uses if and if, then and else statements. (AL) • Uses a variable and relational operators within a loop to govern termination. (AL) (GE) • Designs, writes and debugs modular programs using procedures. (AL) (DE) (AB) (GE) • Knows that a procedure can be used to hide the detail with sub-solution. (AL) (DE) (AB) (GE)	• Performs more complex searches for information e.g. using Boolean and relational operators. (AL) (GE) (EV) • Analyses and evaluates data and information, and recognises that poor quality data leads to unreliable results, and inaccurate conclusions. (AL) (EV)
Purple	• Understands that iteration is the repetition of a process such as a loop. (AL) • Recognises that different algorithms exist for the same problem. (AL) (GE) • Represents solutions using a structured notation. (AL) (AB) • Can identify similarities and differences in situations and can use these to solve problems (pattern recognition). (GE)	• Understands that programming bridges the gap between algorithmic solutions and computers. (AB) • Has practical experience of a high-level textual language, including using standard libraries when programming. (AB) (AL) • Uses a range of operators and expressions e.g. Boolean, and applies them in the context of program control. (AL) • Selects the appropriate data types. (AL) (AB)	• Knows that digital computers use binary to represent all data. (AB) • Understands how bit patterns represent numbers and images. (AB) • Knows that computers transfer data in binary. (AB) • Understands the relationship between binary and file size (uncompressed). (AB) • Defines data types: real numbers and Boolean. (AB) • Queries data on one table using a typical query language. (AB)
Red	• Understands a recursive solution to a problem repeatedly applies the same solution to smaller instances of the problem. (AL) (GE) • Recognises that some problems share the same characteristics and use the same algorithm to solve both. (AL) (GE) • Understands the notion of performance for algorithms and appreciates that some algorithms have different performance characteristics for the same task. (AL) (EV)	• Uses nested selection statements. (AL) • Appreciates the need for, and writes, custom functions including use of parameters. (AL) (AB) • Knows the difference between, and uses appropriately, procedures and functions. (AL) (AB) • Understands and uses negation with operators. (AL) • Uses and manipulates one dimensional data structures. (AB) • Detects and corrects syntactical errors. (AL)	• Understands how numbers, images, sounds and character sets use the same bit patterns. (AB) (GE) • Performs simple operations using bit patterns e.g. binary addition. (AB) (AL) • Understands the relationship between resolution and colour depth, including the effect on file size. (AB) • Distinguishes between data used in a simple program (a variable) and the storage structure for that data. (AB)
Black	• Recognises that the design of an algorithm is distinct from its expression in a programming language (which will depend on the programming constructs available). (AL) (AB) • Evaluates the effectiveness of algorithms and models for similar problems. (AL) (AB) (GE) • Recognises where information can be filtered out in generalizing problem solutions. (AL) (AB) (GE) • Uses logical reasoning to explain how an algorithm works. (AL) (AB) (DE) • Represents algorithms using structured language. (AL) (DE) (AB)	• Appreciates the effect of the scope of a variable e.g. a local variable can't be accessed from outside its function. (AB) (AL) • Understands and applies parameter passing. (AB) (GE) (DE) • Understands the difference between, and uses, both pre-tested e.g. 'while', and post-tested e.g. 'until' loops. (AL) • Applies a modular approach to error detection and correction. (AB) (DE) (GE)	• Knows the relationship between data representation and data quality. (AB) • Understands the relationship between binary and electrical circuits, including Boolean logic. (AB) • Understands how and why values are data typed in many different languages when manipulated within programs. (AB)
White	• Designs a solution to a problem that depends on solutions to smaller instances of the same problem (recursion). (AL) (DE) (AB) (GE) • Understands that some problems cannot be solved computationally. (AB) (GE)	• Designs and writes nested modular programs that enforce reusability utilising sub-routines wherever possible. (AL) (AB) (GE) (DE) • Understands the difference between 'While' loop and 'For' loop, which uses a loop counter. (AL) (AB) • Understands and uses two dimensional data structures. (AB) (DE)	• Performs operations using bit patterns e.g. conversion between binary and hexadecimal, binary subtraction etc. (AB) (AL) (GE) • Understands and can explain the need for data compression, and performs simple compression methods. (AL) (AB) • Knows what a relational database is, and understands the benefits of storing data in multiple tables. (AB) (GE) (DE)

Computational Thinking Concept: AB = Abstraction; DE = Decomposition; AL = Algorithmic Thinking; EV = Evaluation; GE = Generalisation

Note: Each of the Progression Pathway statements is underpinned by one-or-more learning outcomes (due for publication in 2014), providing greater detail of what should be taught to ad
and Matthew Walker. Reviewed by Simon Humphreys and Sue Sentance of Computing at School, CAS MasterTeachers, and by teachers and academics from the wider CAS community. C

n Pathways

Hardware & Processing	Communication & Networks	Information Technology
• Understands that computers have no intelligence and that computers can do nothing unless a program is executed. (AL) • Recognises that all software executed on digital devices is programmed. (AL) (AB) (GE)	• Obtains content from the world wide web using a web browser. (AL) • Understands the importance of communicating safely and respectfully online, and the need for keeping personal information private. (EV) • Knows what to do when concerned about content or being contacted. (AL)	• Uses software under the control of the teacher to create, store and edit digital content using appropriate file and folder names. (AB) (GE) (DE) • Understands that people interact with computers. • Shares their use of technology in school. • Knows common uses of information technology beyond the classroom. (GE) • Talks about their work and makes changes to improve it. (EV)
• Recognises that a range of digital devices can be considered a computer. (AB) (GE) • Recognises and can use a range of input and output devices. • Understands how programs specify the function of a general purpose computer. (AB)	• Navigates the web and can carry out simple web searches to collect digital content. (AL) (EV) • Demonstrates use of computers safely and responsibly, knowing a range of ways to report unacceptable content and contact when online.	• Uses technology with increasing independence to purposefully organise digital content. (AB) • Shows an awareness for the quality of digital content collected. (EV) • Uses a variety of software to manipulate and present digital content: data and information. (AL) • Shares their experiences of technology in school and beyond the classroom. (GE) (EV) • Talks about their work and makes improvements to solutions based on feedback received.(EV)
• Knows that computers collect data from various input devices, including sensors and application software. (AB) • Understands the difference between hardware and application software, and their roles within a computer system. (AB)	• Understands the difference between the internet and internet service e.g. world wide web. (AB) • Shows an awareness of, and can use a range of internet services e.g. VOIP. • Recognises what is acceptable and unacceptable behaviour when using technologies and online services.	• Collects, organises and presents data and information in digital content. (AB) • Creates digital content to achieve a given goal through combining software packages and internet services to communicate with a wider audience e.g. blogging. (AL) • Makes appropriate improvements to solutions based on feedback received, and can comment on the success of the solution. (EV)
• Understands why and when computers are used. (EV) • Understands the main functions of the operating system. (DE) (AB) • Knows the difference between physical, wireless and mobile networks. (AB)	• Understands how to effectively use search engines, and knows how search results are selected, including that search engines use 'web crawler programs'. (AB) (GE) (EV) • Selects, combines and uses internet services. (EV) • Demonstrates responsible use of technologies and online services, and knows a range of ways to report concerns.	• Makes judgements about digital content when evaluating and repurposing it for a given audience. (EV) (GE) • Recognises the audience when designing and creating digital content. (EV) • Understands the potential of information technology for collaboration when computers are networked. (GE) • Uses criteria to evaluate the quality of solutions, can identify improvements making some refinements to the solution, and future solutions. (EV)
• Recognises and understands the function of the main internal parts of basic computer architecture. (AB) • Understands the concepts behind the fetch-execute cycle. (AB) (AL) • Knows that there is a range of operating systems and application software for the same hardware. (AB)	• Understands how search engines rank search results. (AL) • Understands how to construct static web pages using HTML and CSS. (AL) (AB) • Understands data transmission between digital computers over networks, including the internet i.e. IP addresses and packet switching. (AL) (AB)	• Evaluates the appropriateness of digital devices, internet services and application software to achieve given goals. (EV) • Recognises ethical issues surrounding the application of information technology beyond school. • Designs criteria to critically evaluate the quality of solutions, uses the criteria to identify improvements and can make appropriate refinements to the solution. (EV)
• Understands the von Neumann architecture in relation to the fetch-execute cycle, including how data is stored in memory. (AB) (GE) • Understands the basic function and operation of location addressable memory.(AB)	• Knows the names of hardware e.g. hubs, routers, switches, and the names of protocols e.g. SMTP, iMAP, POP, FTP, TCP/IP, associated with networking computer systems. (AB) • Uses technologies and online services securely, and knows how to identify and report inappropriate conduct. (AL)	• Justifies the choice of and independently combines and uses multiple digital devices, internet services and application software to achieve given goals. (EV) • Evaluates the trustworthiness of digital content and considers the usability of visual design features when designing and creating digital artifacts for a known audience. (EV) • Identifies and explains how the use of technology can impact on society. • Designs criteria for users to evaluate the quality of solutions, uses the feedback from the users to identify improvements and can make appropriate refinements to the solution. (EV)
• Knows that processors have instruction sets and that these relate to low-level instructions carried out by a computer. (AB) (AL) (GE)	• Knows the purpose of the hardware and protocols associated with networking computer systems. (AB) (AL) • Understands the client-server model including how dynamic web pages use server-side scripting and that web servers process and store data entered by users. (AL) (AB) (DE) • Recognises that persistence of data on the internet requires careful protection of online identity and privacy.	• Undertakes creative projects that collect, analyse, and evaluate data to meet the needs of a known user group. (AL) (DE) (EV) • Effectively designs and creates digital artefacts for a wider or remote audience. (AL) (DE) • Considers the properties of media when importing them into digital artefacts. (AB) • Documents user feedback, the improvements identified and the refinements made to the solution. (AB) • Explains and justifies how the use of technology impacts on society, from the perspective of social, economical, political, legal, ethical and moral issues. (EV)
• Has practical experience of a small (hypothetical) low level programming language. (AB) (AL) (DE) (GE) • Understands and can explain Moore's Law. (GE) • Understands and can explain multitasking by computers. (AB) (AL) (DE)	• Understands the hardware associated with networking computer systems, including WANs and LANs, understands their purpose and how they work, including MAC addresses. (AB) (AL) (DE) (GE)	• Understands the ethical issues surrounding the application of information technology, and the existence of legal frameworks governing its use e.g. Data Protection Act, Computer Misuse Act, Copyright etc. (EV)

hieve each Progression Pathway statement and National Curriculum point of study. © 2014 Mark Dorling
Computational thinking mapping undertaken by Mark Dorling, Cynthia Selby and John Woollard.

COMPUTING AT SCHOOL
EDUCATE · ENGAGE · ENCOURAGE
In collaboration with BCS, The Chartered Institute for IT

REFERENCES

Barker, I. (2013). 'See-Sawing' Rankings Are Shaky Ground to Build On. *Times Educational Supplement* [Online]. Available: http://www.tes.co.uk/article.aspx?storycode=6317642 [Accessed May 9 2015].

Bundy, A. (2007). Computational Thinking Is Pervasive. *Journal of Scientific and Practical Computing,* 1**,** 67-69.

Computer Science Teachers Association (2011). *K–12 Computer Science Standards.* New York, NY, ACM.

Computing at School Working Group (2012). Computer Science: A Curriculum for Schools. London, UK, Computing At School.

Crow, D. (2014). Why Every Child Should Learn to Code. *The Guardian* [Online]. Available: http://www.theguardian.com/technology/2014/feb/07/year-of-code-dan-crow-songkick [Accessed May 9 2015].

de Bono, E. (2000). *6 Thinking Hats (Revised).* London, England, Penguin.

de Bono, E. (2007). *How to Have Creative Ideas.* London, England, Vermillion.

Department for Education. (2012). Subject Knowledge Requirements for Entry into Computer Science Teacher Training. Available: http://academy.bcs.org/sites/academy.bcs.org/files/subject%20knowledge%20requirements%20for%20entry%20into%20cs%20teacher%20training.pdf [Accessed May 9 2015].

Department for Education. (2013a). Assessing without Levels. Available: http://webarchive.nationalarchives.gov.uk/20130904084116/https://www.education.gov.uk/schools/teachingandlearning/curriculum/nationalcurriculum2014/a00225864/assessing-without-levels [Accessed May 9 2015].

Department for Education. (2013b). The National Curriculum in England, Framework Document. Available: http://www.education.gov.uk/nationalcurriculum [Accessed May 9 2015].

Department for Education and Employment. (1999). *Information and Communication Technology - the National Curriculum for England,* London, England.

Department for Education and Skills (2002). *Framework for Teaching ICT Capability: Years 7, 8 and 9,* London, UK.

Dorling, M. & Walker, M. (2014). Computing Progression Pathways. Available: https://www.hoddereducation.co.uk/Subjects/ICT/Series-pages/Compute-IT/Series-Box/Progression-Pathways/Progression-Pathways-Grid.aspx. [Accessed May 9 2015].

Hamilton, G. & Henderson, B. (2013). *So What Are Open Badges?* [Online]. Jisc. Available: http://www.jisc.ac.uk/blog/so-what-are-open-badges-28-aug-2013 [Accessed May 9 2015].

Harrison, J. (2014). Naace Assessment Panel Guidance Available: http://www.naace.co.uk/curriculum/assessment [Accessed May 9 2015].

Henderson, P. B., Cortina, T. J. & Wing, J. M. (2007). Computational Thinking. *Proceedings of the 38th SIGCSE technical symposium on Computer science education.* Covington, KY.ACM.

Lu, J. J. & Fletcher, G. H. L. (2009). Thinking About Computational Thinking. *In:* Proceedings of the 40th ACM Technical Symposium on Computer Science Education, 2009 Chattanooga, TN, USA. New York: ACM, 260-264.

Mozilla. (2014). *Open Badges* [Online]. Available: http://openbadges.org/about/ [Accessed May 9 2015].

National Association of Head Teachers. (2014). Report of the Naht Commission on Assessment. Available: http://www.naht.org.uk/welcome/news-and-media/key-topics/assessment/profession-takes-lead-on-assessment-after-the-end-of-levels/ [Accessed May 9 2015].

National Research Council (2010). Report of a Workshop on the Scope and Nature of Computational Thinking. The National Academies Press.
Naughton, J. (2012). Why All Our Kids Should Be Taught How to Code. *The Guardian* [Online]. Available: http://www.theguardian.com/education/2012/mar/31/why-kids-should-be-taught-code [Accessed May 9 2015].
Nettleford, W. (2013). Primary School Children Learn to Write Computer Code. Available: http://www.bbc.co.uk/news/uk-england-london-23261504 [Accessed 10-07-2013].
Passmore, B. (2007). Testing, Testing, Testing. *Times Educational Supplement* [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=trh&AN=24943568&site=eds-live [Accessed May 9 2015].
Radiowaves Schools Ltd. (2014). *Makewaves* [Online]. Leeds, England. Available: https://www.makewav.es/ [Accessed May 9 2015].
Reigeluth, C. M. (2013). *Instructional-Design Theories and Models: A New Paradigm of Instructional Theory.* London, England, Routledge.
Selby, C. & Woollard, J. (2013). Computational Thinking: The Developing Definition Available: http://eprints.soton.ac.uk/356481/ [Accessed May 9 2015].
Warner, L. (2008). The 'Freedom to Frame Questions ... Worth Asking', or Three Stories and Three (Other) Fragments of Research. *English in Education,* 42, 1**,** 88.
Wickens, C. (2007). Creativity. *In:* Kennewell, S., Connell, A., Edwards, A., Hammond, M. & Wickens, C. (eds.) *A Practical Guide to Teaching ICT in the Secondary School.* Oxford, UK: Routledge.
Wing, J. (2006). Computational Thinking. *Commun. ACM,* 49, 3**,** 33-35.
Wing, J. (2008). Computational Thinking and Thinking About Computing. *Philosophical Transactions of The Royal Society A,* 366**,** 3717-3725.
Wing, J. (2011). Research Notebook: Computational Thinking - What and Why? *The Link.* Pittsburgh, PA: Carneige Mellon.
Yadav, A., Zhou, N., Mayfield, C., Hambrusch, S. & Korb, J. T. (2011). Introducing Computational Thinking in Education Courses. *Proceedings of the 42nd ACM technical symposium on Computer science education.* Dallas, TX: ACM.

Teachers’ perspectives on successful strategies for teaching Computing in school

Sue Sentance, *sue.sentance@kcl.ac.uk*
King’s College London, UK

Andrew Csizmadia, *a.p.csizmadia@newman.ac.uk*
Newman University, UK

Abstract

With the introduction of Computing into the school curriculum in England, experienced teachers are having to teach it for the first time. This raises a number of questions, including whether teachers need to adapt their existing pedagogical strategies to deliver Computing in the curriculum. In this paper we address this particular question through an analysis of qualitative statements made about how to teach Computing by over 300 in-service teachers, who contributed as part of a larger survey. We identify a range of pedagogical strategies that teachers use in practice which can be categorised into the five areas of contextualised learning, computational thinking skills development, code manipulation, working collaboratively and learning away from the computer. We suggest that focusing on the use of a range of these strategies could help teachers to feel more confident in the Computing classroom.

Keywords

Computer science education, Pedagogy, Primary computing education, Secondary computing education, Teachers' perspectives, Computational thinking

INTRODUCTION

Computing is being introduced as a new subject in the school curriculum in many countries, and as an important part of informal learning opportunities in others. This brings with it both excitement and challenges, as for any new subject. For teachers facing curriculum change, how to teach it is very pertinent. Introducing new content does not merely mean that teachers have to equip themselves with new subject knowledge, which of course in many cases they do (Brown et al., 2013; Sentance, Dorling, & McNicol, 2013; Thompson & Bell, 2013). Teachers also need to learn appropriate pedagogies for delivering a new subject, particularly in those aspects of computer science that relate to algorithms, programming and the development of computational thinking skills.

In this paper, statements made by teachers who are currently teaching Computing in primary and secondary schools have been coded, categorised and analysed, describing both successful strategies for teaching and the difficulties they face. The teachers' perspective gives us some interesting evidence of what works for real teachers in their classrooms.

TEACHING COMPUTING IN SCHOOL

Recent literature relating to computer science education in school highlights a number of ways of making computer science concepts accessible, engaging and fun, and more importantly, giving learners a deep understanding of these concepts.

Constructivist theory, based on the work of Dewey (1938), Piaget (1950) and Bruner (1996) suggests that learning is a cumulative and active process during which the learner constructs knowledge and meaning for themselves as they learn, connecting with, and explaining new knowledge in terms of, what they already know.

Constructivist learning theories applied to computer science emphasize the active, subjective and constructive character of knowledge, placing students at the centre of the learning process (Ben-Ari, 1998). Specifically, constructivist learning, based on students' active participation in problem-solving and critical thinking, has profoundly influenced the teaching of programming (Ben-Ari, 1998).

Experiential learning that stems from constructivism describes the design of activities which engage learners in a very direct way. Working with tangible real world objects is a central tenet of Papert's constructionism (Papert, 1991) (which builds on constructivism). Thus, constructivist principles support the strategies of using more kinaesthetic and active approaches to teaching in the computer science classroom.

The “unplugged” style of activities which originated with the CS Unplugged project in New Zealand (Bell, Alexander, Freeman, & Grimley, 2009; Nishida et al., 2009) has resulted in many related, kinaesthetic activities which stimulate an understanding of a concept in a very concrete and practical way. CS4FN (Computer Science for Fun) (Curzon, McOwen, Cutts, & Bell, 2009) have generated many engaging activities and approaches by emphasising the importance of analogy as well as a kinaesthetic activity. Other research has highlighted the importance of providing a real world context for learning and relating it to students' interests and understanding and the value of a rich discourse regarding concepts (Grover & Pea, 2013).

Another key consideration in computer science pedagogy needs to be the development of computational thinking skills. Computational thinking was only recently popularised as a concept in 2006 by Wing (Wing, 2006) but teachers of computer science have been facilitating these skills in their students as long as this subject has been taught. For teachers in England, guidelines have been developed recently suggesting how computational thinking can be explicitly taught as part of the new curriculum (Curzon, Dorling, Ng, Selby, & Woollard, 2014).

Programming is the aspect of computer science in school which is perceived to be the most challenging. A range of activities can be used that allow students to collaborate and construct problem solutions. As an example, the following suggestions, drawing on a constructivist view of learning, are made by Van Gorp and Grissom:

- Code walkthroughs
- Writing algorithms in groups
- Insert comments in pairs into existing code
- Develop code from algorithm in pairs
- Find the bugs in code (Van Gorp & Grissom, 2001).

Reading and tracing code is also important in supporting the learning of programming has been demonstrated (Lopez, Whalley, Robbins, & Lister, 2008) and being able to do this is a pre-cursor to the problem-solving needed to write code (Lister et al., 2004). Lister later describes that novices need to be able to trace code with more than 50% accuracy before they can begin to confidently write programs of their own (Lister, 2011). In our study we were interested to see which types of strategies were being used in the classroom by the participants, and how they were supporting the development of strategies in reading, writing and tracing code.

This discussion about pedagogical approaches to teaching Computing can be related to the teachers’ pedagogical content knowledge (PCK), that is, the knowledge that a teachers has about how to teach their subject (Shulman, 1986).. But how does a teacher develop this PCK for teaching Computing? We hope initially through good initial teacher education, but also through professional development, sharing with other teachers, and learning from experience.

Research Focus

In this study we sought to ask a large number of active computing teachers how they recommend teaching the subject, in order to find out which strategies work well in practice. We balanced this with asking teachers about particular challenges they faced in teaching computing. Our research questions are quite simply the following:

- What pedagogical strategies do teachers report work well for teaching computer science in school?
- What challenges do teachers report that they face?

Black et al carried out a study in the UK where they asked teachers how they felt they could make the subject interesting (Black et al., 2013). The key aspects that they identified were the importance to teachers of making Computing fun and relevant. In carrying out our research we were interested to see whether the teachers' comments aligned with this study; in addition we asked more specifically for actual strategies that teachers use in their classroom that they feel to be effective.

This paper focuses purely on the teachers' perspective in addressing these questions. Diethelm et al emphasise the importance of the teachers' perspective to our understanding of computer science education as the teacher "*may work on many different abstraction levels or apply very different teaching methods for the same topic of the curriculum*" (Diethelm, Hubwieser, & Klaus, 2012). We wish to identify what these methods are, in particular identifying common themes that may help to provide guidance for teachers new to teaching the subject, as well as providing actual examples of teachers using effective strategies as we enter a phase of education when more and more students are studying computing in school.

In the next section the study carried out will be described. We will then report on the results of the content analysis that was used to analyse the responses of the teachers. Those aspects that require a whole new style of teaching for some teachers are identified. We then draw out how this can contribute to the general area of pedagogical content knowledge in our subject.

THE STUDY

The context: change in the curriculum

The UK has seen fast-paced change in the area of computer science education in the last few years (Brown et al., 2013; Brown, Sentance, Crick, & Humphreys, 2014). The state of computer science education is different in the four parts of the UK, with England having just implemented an ambitious new curriculum in Computing, to be taught from ages 5-16, and with a strong focus on computational thinking. This has been preceded by two years of preparation, as new qualifications were introduced and the draft curriculum proposed. Many schools and teachers in England had implemented elements of the Computing curriculum prior to the official starting date of the Computing Programme of Study of September 2014, as a void was left by the disapplication of ICT in January 2012 (Brown et al., 2014).

In the UK there is a strong subject association for computer science teachers, Computing At School (Brown et al., 2013). Through this grass-roots community of practice teachers are able to share resources, share experiences and attend local events. The participants of this study were to a very large extent members of this community. In the data collected in this study, they describe the experiences, successful strategies, and also the frustrations, of teachers who have begun to teach Computing in school over the last few years.

The Computing Programme of Study for the new English Curriculum (Department for Education, 2013) is based on computational thinking principles, and thus teachers of computer science welcome guidance on how to deliver computational thinking skills; which is beginning to emerge (Curzon et al., 2014).

Survey of teachers' perspectives

A wide-ranging survey was carried out of members of Computing At School. As one part of this, teachers were asked if they optionally wished to contribute free-text answers to the following four questions about their teaching.

1. What good techniques/strategies have you found for helping students to understand programming?
2. Please describe any good techniques/strategies you use for helping students to understand other aspects of Computing?
3. What difficulties, if any, have you experienced teaching programming?
4. What difficulties, if any, have you experienced teaching other aspects of Computing?

In the context of this survey, teachers in the England understand "other aspects of Computing" to be non-programming topics in the curriculum, which include learning about hardware, networking, data representation and logic (Department for Education, 2013) .

The survey was publicised via the Computing at School forum, as well as through social media channels. 1417 members completed the wider survey, with 357 teachers contributing at least one free text answer to the free text questions. In this paper we focus only on the 357 responses given by this self-selecting group of teachers but include reference to their other answers to survey questions where relevant.

The data was collected by an online questionnaire which was then input into qualitative data analysis software. The data consisted of the four free text questions described above, plus responses that these teachers gave to the other questions in the wider survey.

Study Participants

The 357 teachers responding to the questions were from different phases of education, including primary (ages 4-11), secondary (ages 11-18) and post-secondary education (ages 16 to 18). The majority (76%) of the teachers were from secondary education, with 20% from primary and middle schools and 4% from post-secondary education.

Teachers were asked to rate their confidence in being able to deliver the new Computing curriculum on a Likert scale from 0 to 10. This self-selecting group were largely confident in their Computing teaching, with 85% rating their confidence at 6 or more out of 10. Their confidence overall was greater than the confidence levels of the wider population completing the larger survey. The general confidence in the teaching of Computing will have contributed to their willingness to participate in a free text questionnaire on their practice and also will have a bearing on the content on their responses. This indicates that they may not be `typical' of the whole teacher population, but represent teachers who are more comfortable teaching Computing.

ANALYSIS

The data was initially coded in an inductive manner with respect to emerging themes, following the guidelines in (Mayring, 2000). The themes were then grouped to facilitate further analysis. The data was re-coded and verified by two researchers to ensure agreement on the interpretation of the teachers' statements. The coded answers were then examined in relation to the level of confidence that the participants had in the context of teaching computing.

Table 5 provides an overview of the particular strategies that teachers mentioned when describing their teaching.

Coded strategy (programming)	% mentions
Practice/little & often	14%
Unplugged/ teach away from computer	13%
Use of particular software	13%
Scaffolding/modifying code	12%
Vary activities	11%
Exercises around coding	10%
Use lots of examples	9%
Relate activities to the real world	8%
Demonstration & modelling	8%
Peer mentoring	8%
Learn through exploring	7%

Coded strategy (non-programming)	% mentions
Unplugged/ teach away from computer	19%
Hands-on activities	17%
Relate activities to real world	10%
Show videos	9%
Work in groups	8%
Use published resources	6%

Table 5: Strategies used by teachers teaching programming and non-programming aspects of the curriculum

This table shows that teachers emphasised unplugged, hands-on, contextualised activities and the importance of lots of practice. Approximately the same number of teachers mentioned working on tasks away from the computer as mentioned a particular software package that they used. In addition, a high percentage of teachers (13%) referred to particular software that helped them to teach programming and other concepts. The study looks entirely at free text comments with suggestion within the question; there are themes emerging quite clearly from this data around using activities away from the computer that promote understanding. These will be discussed in more depth in the next section.

Teachers reported a range of different challenges that they faced when teaching Computing. These are shown in Table 6. Some of the challenges mentioned relate to the teachers' own difficulties – for example, not being confident in the subject matter or not being able to differentiate sufficiently for a mixed-ability group, and other comments focus on the fact that the students have difficulty understanding the material and in problem solving. The data showed overall three areas of challenge for teachers: their confidence in teaching computing as a subject, the difficulties (or perceived difficulties) inherent in the subject matter and the issue of having sufficient resources, including technical support, in the classroom.

Challenges	% mentions (programming)	% mentions (non-programming)	Overall
Students not understanding / difficulty	20%	14%	17%
Teachers' own subject knowledge	20%	13%	16%
Challenge of differentiation	13%	3%	8%
Lack of resources	7%	9%	8%

Technical problems	9%	8%	8%
Students' abilities in problem-solving tasks	11%	2%	6%

Table 6: Challenges described by teachers teaching Computing

The remainder of the paper focuses particularly on the good strategies that teachers report that work well for them.

Key Themes Emerging

In this section we address specific themes that emerged from strategies used by teachers. Most of the individual strategies suggested by teachers could be grouped into a series of five themes, which are (in no particular order):

- Contextualisation of learning
- Collaborative working
- Computational thinking
- Code tracing and scaffolding
- Learning away from the computer

Typical quotes from teachers illustrate these themes.

Contextualisation of learning

Teachers talk about relating computing content to other aspects of the curriculum; they give examples of both relating what is being learned in computing to other subjects taught at school and also to concepts from home (so relating to real-life). The quote below is a typical example:

"*Scale it back to basics and use real-life examples for the activities e.g. making tea. Use lots of visual aids to help pupils and online resources to help scaffold activities*." (case 233, secondary teacher, confidence 7).

It is interesting to examine the range of ways in which teachers talk about the contextualisation of learning.

Collaborative Working

The analysis of teachers' qualitative responses highlights a variety of collaborative working strategies that they use within the classroom and would promote to other computer science teachers. These collaborative strategies included: team work, peer mentor, paired programming and collaboration. These strategies resonate with the concept of computational participation (Kafai & Burke, 2014) and strategies proposed to develop this within the classroom. In addition individual teachers commented on the positive motivational impact that collaborative working has on individuals, small groups and the class itself.

> "…*Developing digital leaders in students who can support others.* …" (case 345, primary teacher, confidence 9).
>
> "*Decomposing sample problems together as a class then team-coding …they can use peers for discussion of specific problems.* …" (case 12, secondary teacher, confidence 9).

Computational Thinking

Analysis of teachers' qualitative responses indicates a number of computational thinking concepts and processes that teachers want to promote and develop their students' competence in through using a variety of teaching and learning activities.

These concepts and processes include: logic (algorithmic) thinking, decomposition, problem solving and abstraction (Brennan & Resnick, 2012; Curzon et al., 2014).

> "*Breaking down the problem then breaking it down again then breaking it down again...* …" (case 109, secondary teacher, confidence 8).

> "*Organise the learning so that the pupils develop their programming skills using decomposition and abstraction. …."* (case 265, secondary teacher, confidence 10).

Code tracing and scaffolding

Closely related to the theme of computational thinking are the strategies that teachers use to help their students understand program code. One teacher describes a range of types of strategies used to support students learning programming, that involve:

> "… *giving code on paper not electronically, so they have to type it in, think about what they are typing and fix the errors that occur when trying to compile the program …* (case 113, secondary teacher, confidence 7).

> "*Discussion of what a specific algorithm does, then running trace tables on small programs …"* (case 310, secondary teacher, confidence 7).

Other strategies described included "scaffolding" as the student is given part of a program to extend, and programs to debug. Typing in code to give more chance that the program would work, but involving debugging errors caused by transcription errors is another supportive strategy for early programmers reported by teachers.

Unplugged-style or kinaesthetic activities

A significant proportion of teachers mentioned, unprompted, that they try to support students' understanding by using physical, or unplugged-style activities in the classroom. One teacher gives two examples of teaching different topics using physical visual-aids to support the learning:

> "*For example I use clear plastic drinking cups as memory locations and label them as variables or when demonstrating an algorithm like bubble sort add data (on pieces of paper)."* (case 229, secondary teacher, confidence 9)

Many of these activities are designed to promote both collaboration and computational thinking skills. In fact, whether the activity takes place on the computer or not may not be what is interesting. The key link between the statements made by teachers seemed to be their impression that actually physically being engaged in the activity was conducive to the students' learning. This is an area which needs further research to establish.

Putting it all together

As reported earlier, Diethelm (2012) discusses the way that teachers use a variety of strategies as part of their pedagogical content knowledge for the subject. This was most definitely the case with the data that we looked at. A number of teachers described strategies that they used for teaching which included a mixture of the types of approaches already described – these have been underlined in this comment:

> *"… Provide some examples which have <u>errors to be fixed</u> - or examples <u>that need re-writing</u> … more efficient perhaps and <u>get pupils to explain</u> their decisions) …Get them <u>working away from the computer</u> at times to ensure they*

consider the steps of the program they are undertaking rather than just hacking away on the computer . Ask questions and get them to explain program concepts i.e. to vocalise an understanding … Discussion is important at times - not just doing." (case 132, secondary teacher, confidence 10).

Here it can be seen that the teacher (who self-reports as having a high level of confidence in teaching Computing) is combining strategies around code exercises, using discussion (collaboration and computational thinking), and working away from the computer. The key for this teacher seems to be to utilise a variety of teaching strategies to support learning, rather than relying on one particular strategy. What is key in this description is the need for students to reflect on what they are learning in computing and be able to articulate it.

DISCUSSION

Examining the statements of teachers as they report what strategies work well for them in teaching Computing has enabled us to draw out particular themes. Ben-Ari (1998) advised teachers: "*Don't run to the computer*", and it seems that teachers are taking this advice in using a variety of other strategies to get concepts across. In addition, the use of collaborative work, peer mentoring, pair programming and other strategies is helping teachers to establish computational thinking skills in young students.

The teachers participating in the survey are, in the majority, members of Computing At School and as such have access to a lively and supportive grass-roots community of teachers with whom they can exchange ideas and classroom resources. It could be tentatively suggested that the presence and nature of this community of practice may also have an impact on the commonality between the approaches teachers are successfully using, but this cannot be verified from the data analysed.

It cannot be assumed that the teaching approaches described by teachers in this study are representative of all teachers teaching the English Computing curriculum. The participants are self-selecting and have mostly reported themselves as being confident in their delivery of Computing so the data gives us reports of good practice. We are also not able to provide evidence for which of these suggested approaches is more effective in helping students to learn without more empirical research; thus another useful angle on this question would be to examine students' own perspectives on how Computing is taught.

CONCLUSION

In this paper, we have identified a range of pedagogical strategies that teachers use in practice which can be categorised into the five areas of contextualised learning, computational thinking skills development, code manipulation, working collaboratively and learning away from the computer.

The study exemplifies a link between theory and practice as teachers report that they use strategies for teaching Computing that previous research has suggested to be of value. In addition, teachers who self-report as "confident" use a combination of these strategies to support their students' understanding.

We suggest that focusing on the use of a range of these strategies could help teachers to feel more confident in the computer science classroom. We believe the results of this study are significant in that they will give novice teachers (or experienced teachers new to computing) some ideas about pedagogical approaches that work. More research will be needed as to what extent students perceive these strategies to be effective also.

REFERENCES

Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2009). Computer science unplugged: School students doing real computing without computers. *New Zealand Journal of Applied Computing and Information Technology, 13*(1)

Ben-Ari, M. (1998). *Constructivism in computer science education*. Proceedings of the twenty-ninth SIGCSE technical symposium on Computer science education. Atlanta, Georgia, United States: ACM.

Black, J., Brodie, J., Curzon, P., Myketiak, C., McOwan, P. W., & Meagher, L. R. (2013). Making computing interesting to school students. *Proceedings of the 18th ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '13).* Canterbury, UK. 255. doi:10.1145/2462476.2466519

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. in . *Proceedings of the 2012 Annual Meeting of the American Educational Research Association, Vancouver, Canada.,* Vancouver, Canada.

Brown, N., Kölling, M., Crick, T., Jones, S. P., Humphreys, S., & Sentance, S. (2013). Bringing computer science back into schools: Lessons from the UK. *Proceedings of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE 2013),* 269-274.

Brown, N., Sentance, S., Crick, T., & Humphreys, S. (2014). Restart: The resurgence of computer science in UK schools. *ACM Transactions of Computing Education, 14*(2)

Bruner, J. (1996). *Towards a theory of instruction*. Cambridge, MA: Harvard University Press.

Curzon, P., Dorling, M., Ng, T., Selby, C., & Woollard, J. (2014). *Developing computational thinking in the classroom: A framework.* Available at: http://community.computingatschool.org.uk/files/3517/original.pdf).Unpublished report.

Curzon, P., McOwen, P., Cutts, Q., & Bell, T. (2009). Enthusing and inspiring with reusable kinaesthetic activities. *Proceedings of the ITICSE 2009,*

Department for Education. (2013). *National curriculum for England: Computing programme of study.*. London, England: Department for Education.

Dewey, J. (1938). *Experiential education*. New York: Collier Books.

Diethelm, I., Hubwieser, P., & Klaus, R. (2012). *Students, teachers and phenomena: Educational reconstruction for computer science education*. Koli, Finland: ACM.

Grover, S., & Pea, R. (2013). *Using a discourse-intensive pedagogy and Android's App Inventor for introducing computational concepts to middle school students*. Proceedings of the 44th SIGCSE technical symposium on Computer science education. ACM.

Kafai, Y. B., & Burke, Q. (2014). *Connected code why children need to learn programming.* Cambridge MA: MIT Press.

Lister, R. (2011). Concrete and other neo-piagetian forms of reasoning in the novice programmer. *Proceedings of the Thirteenth Australasian Computing Education Conference - Volume 114,* Perth, Australia. 9-18.

Lister, R., Adams, E. S., Fitzgerald, S., Fone, W., Hamer, J., Lindholm, M., . . . Thomas, L. (2004). A multi-national study of reading and tracing skills in novice programmers. *Working Group Reports from ITiCSE on Innovation and Technology in Computer Science Education,* Leeds, United Kingdom. 119-150.

Lopez, M., Whalley, J., Robbins, P., & Lister, R. (2008). Relationships between reading, tracing and writing skills in introductory programming. *Proceedings of the Fourth International Workshop on Computing Education Research,* Sydney, Australia. 101-112. doi:10.1145/1404520.1404531

Mayring, P. (2000). Qualitative content analysis. *Forum of Qualitative Social Research, 1*(2)

Nishida, T., Kanemune, S., Idosaka, Y., Namiki, M., Bell, T., & Kuno, Y. (2009). A CS unplugged design pattern. *Proceedings of the 40th ACM Technical Symposium on Computer Science Education (SIGCSE '09),* 231. doi:10.1145/1508865.1508951

Papert, S. (1991). *Constructionism* Ablex Publishing.

Piaget, J. (1950). *The psychology of intelligence*. Cambridge, MA: Harvard University Press.

Sentance, S., Dorling, M., & McNicol, A. (2013). Computer science in secondary schools in the UK: Ways to empower teachers. In I. Diethelm, & R. Mittermeir (Eds.), *Informatics in schools: Sustainable informatics education for pupils of all ages.Lecture notes in computer science* (pp. 15-30) Springer-Verlag.

Shulman, L. (1986). Those who understand: Knowledge growth in teaching. *American Educational Review, 15*(2)

Thompson, D., & Bell, T. (2013). *Adoption of new computer science high school standards by New Zealand teachers* ACM.

Van Gorp, M. J., & Grissom, S. (2001). An empirical evaluation of using constructive classroom activities to teach introductory programming. *Computer Science Education, 11*(3), 247-260.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33.

Sue Sentance is Senior Lecturer in Computer Science Education at King's College London and Research Coordinator for Computing At School. Sue teaches new and existing teachers of Computing, and has been heavily involved in the curriculum changes around primary and secondary-level Computing in England for several years. She runs a computer science teaching accreditation scheme for in-service teachers, as well as developing practitioner research in computer science education for Computing At School.

Andrew Csizmadia is Senior Lecturer in Computer Science Education at Newman University, Birmingham. Andrew has supported teachers in the West Midlands in the transition from ICT to computer science and is a regional coordinator for the Barefoot Computing Project which promotes computational thinking in schools.

OnCreate and the Virtual Teammate: An analysis of online creative processes and remote collaboration

Martyn Thayne, *mthayne@lincoln.ac.uk*
University of Lincoln, School of Film and Media, UK

Björn Stockleben, *bjoern.stockleben@hs-magdeburg.de*
University of Applied Sciences Magdeburg-Stendal, Germany

Seija Jäminki, seija.jaminki@lapinamk.fi
Lapland University of Applied Sciences, Finland

Ilkka Haukijärvi, ilkka.haukijarvi@tamk.fi
Tampere University of Applied Sciences, Finland

Nicholas Blessing Mavengere, nicholas.mavengere@uta.fi
University of Tampere, Finland

Muhammet Demirbilek, mdbilek@*sdu.edu.tr*
Süleyman Demirel University, Turkey

Mikko Ruohonen, *mikko.j.ruohonen@uta.fi*
University of Tampere, Finland

Abstract

The OnCreate project explores the specific challenges of implementing creative processes in online-based learning environments. The first research phase comprises a comparative qualitative analysis of collaboration practices in design-related study programmes at the ten participating universities. A key outcome of this research was identifying the shortcomings of the hierarchical models of established Learning Management Systems (such as Moodle or Blackboard) and the tendency towards evolving mash-up environments to support creative online collaboration.

Keywords

online collaboration, design processes, creativity, eLearning, eWork, communities

INTRODUCTION

The opportunity for users to collaborate and explore different perspectives is an important design element, especially for participants who are learning or working at a distance (Herrington & Herrington, 2006). Study programmes following blended-learning or even pure online approaches are becoming increasingly popular in higher education. Two main drivers for this development are increasing international cooperation among universities and with industry partners and the special requirements of students in continuing education programmes. Additionally, global and distributed organisations require team skills and online collaborative competences from employees and the best way to learn these competences is througheducation (Okuogume & Jäminki, 2011). In September 2014, 10 universities from across Europe started an EU Erasmus+ Strategic Partnership. This joint project is about the exchange, implementation and evaluation of processual and contextual knowledge of online collaborative courses, with focus on creation and innovation. Apart from questions of choice of tools and platforms, the project will especially investigate how to create the social and other “soft” contextual factors that foster creative collaboration in online learning environments. In physical rooms, we are used to providingspaces and processes to ideally support ideation and creation processes (Doorley, Witthoft, & others, 2011). The project seeks to create virtual spaces for creative learning in a similar manner, addressing frequent problems we encounter in online collaboration.

THE NEED FOR COLLABORATION IN ONLINE LEARNING

A range of factors, including technological innovation, a global economy, strong competition and information overload, characterizes the present environment. Learning is one of the ever-required virtues in this environment, and because of globalisation, virtual teamwork is of increasing importance. Thus, the development of more robust frameworks, pedagogies and tools for collaboration across the Higher Education sector couldhelp to produce more employable students who are better equipped to deal with the contemporary industrial setting. According to research by Woods Bagot (Holmes, 2012), (based on a survey of 500 business leaders), collaboration has become a key employment requirement that is currently being underserved in higher education. When asked, “what are the three most important skills or attributes new hires need in order to succeed at your organization” the top three were Problem Solving (49%), Collaboration/Ability to Work as a Team (43%) and Critical Thinking (36%).

A theoretical basis for collaborative learning was provided by Johnson & Johnson(1996) in terms of cognitive development theories and especially from a Vygotskian perspective; on the basis of social independence theory.According to Lev Vygotsky (Vygotsky, 1978), this type of social interaction involving cooperative or collaborative dialogue promotes cognitive development. Collaborative learning thus is a critical component that helps ensure quality in associated teaching and learning processes. The greater a student’s involvement in the design, assessment and evaluation of their learning, the greater their motivation and the stronger the sense of responsibility for their own learning process is(Holmes, 2012).

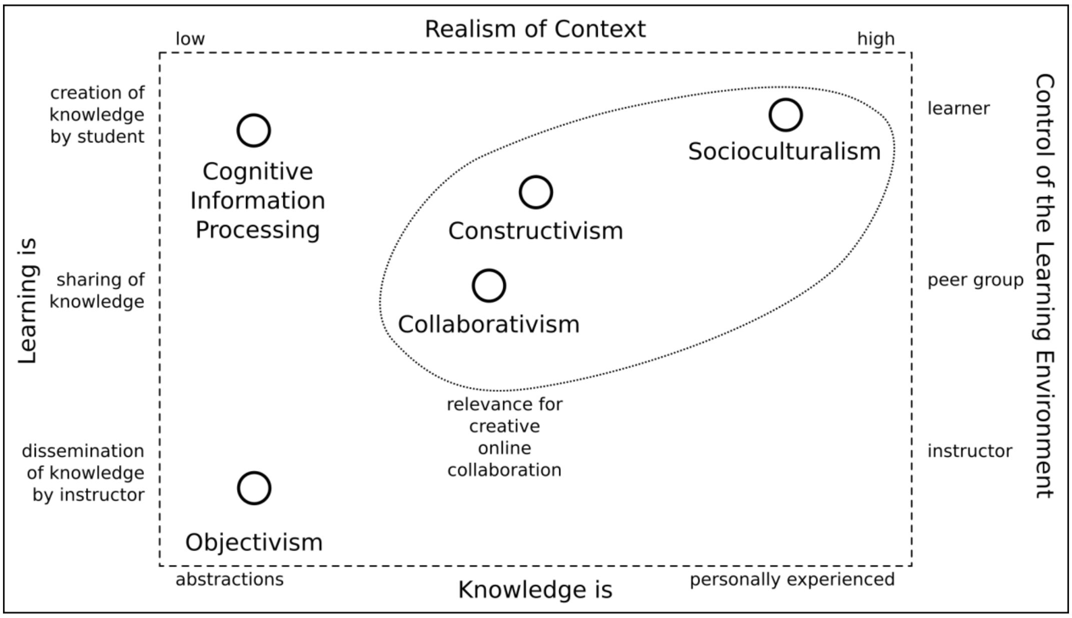


Figure 10 The Dimensions of the Learning Theories according to Leidner & Jarvenpaa(1995)

Such research demonstrates a necessity to understand more about the tools and frameworks needed to facilitate cooperative student partnerships and collaborative learning.Leidner & Jarvenpaa(1995) suggested that the effectiveness of information technology in contributing to learning will be a function of how well the technology supports a particular model of learning and the appropriateness of the model to a particular learning situation. Figure 10illustrates some of the common learning models and their tentative relevance for creative collaboration in online environments.

Creativity has a strong tie to constructivism, as the process of design can be seen as a process of recombination of formerly internalized experiences into something novel. Yet creative processes typically include externalization and discussion of ideas, as well as inspiration from everyday experiences, referring to Collaborativism and Socioculturalism respectively. As a conclusion, teaching creative online collaboration should refer to the intersection between Constructivism, Collaborativism and Socioculturalism.

THE CONCEPT OF 'CREATIVE PROCESSES' IN ONLINE COLLABORATION

Creativity is defined as an ability to produce a product or idea that is either original or possessing a new quality which people have not experienced before (Runco & Jaeger, 2012). Collaboration is defined as an action or interpersonal relationship between professionals to achieve a specific purpose (Schrage, 1995). It is a methodology through which work is produced by more than one person. In order to contribute effectively, collaborating partners should engage in a continuous reflective discourse on their work(Schoenfeld, 1999). Creative collaborators are persons who are interacting with others to discover genuinely new ways of thinking and doing something new together. Collaborators play principal roles on by conceiving and carrying the works out.

Creative collaboration is a process of exchanging of ideas among the participants helping to stimulate and enrich their own creativity to the extent that the solution they reach is novel and appropriate (Hong, 2013). The characteristics of creative collaboration are unique tasks, unpredictable results, shared understanding, communication, and joint decision-making (Kalay, 2006). Shared understanding of other collaborators' world-views and knowledge, communication, and joint decision-making, may help creativity. It requires higher cognitive taxonomy and critical reflections on other collaborative team members' social and educational background and allows each team member to criticize, incorporate the propositions into the emerging collective creation.The purpose of collaboration rests on the limits of individual abilities which prevent completing a given task and the expectation that collaboration can help people complete the task more quickly and more effectively as well as produce better quality than they could otherwise (Kalay, 2006).

The creative potential of online collaboration emerges in many forms, from epic cultural efforts such as Wikipedia, to open-source software, to smaller communities. Collaborative work can improve outcomes for both individual and community as a whole. According to Slavin (Slavin, 1980), working with others to achieve shared goals can promote social, motivational, and emotional benefits. Burke & Settles (2011) found that online creative collaboration processes improved the pursuit of the peers' own goalswhile at the same time made them behave in more community-favorable ways. Factors that affect the quality of collaboration are communication between collaborators, collaborations formed out of shared interests but different skills, andsmall status differences (Dow & Settles, 2013).

Within the project, online creative collaborative processes are defined as all such group activities, which aim to solve problems that do not have standard solutions, mediated through web-based tools. Typically, such problems require interdisciplinarity, lateral thinking, social empathy and extensive ideation with the aim of mutual inspiration. The processes applied are often nonlinear and rely on multimodal means of synchronous and asynchronous communication, with a special focus on visual tools. We deliberately chose to research the specific case of creative collaboration, as this poses particular challenges not sufficiently covered in our current educational practice. Table 7 summarizes the key challenges we have recognized in settings of virtual creative processes.

Characteristics of creative processes	Challenge in online environments
Mutually shared understanding of viewpoints (empathy) and knowledge	Online environments fall short of emotional cues and informal conversations. Substitutes for both are needed to build a successful team.
Unique tasks	Creative processes know a plethora of methods to tackle any creative challenge, but only a handful of them havebeen implemented online yet.
Unpredictable results	The flexibility of online-supported processes has to be increased to adapt processes quickly.
Joint decision-making	Creative works are judged by a lot of different criteria and decision-making is complex. Solutions to deeply discuss and decide about creative work online is needed.
Small status differences / flat hierarchies	Especially synchronous online methods are centred on a leader or moderator. Online creative processes should have the possibility to make anyone an actor driving the process at any time.

Table 7 Challenges of creative processes in online environments

The focus of the project is on online-based collaborative creative processes. Usually these processes are centred around very concrete problems, often implemented as project-based learning. Such kinds of courses have a high realism of context and the learning environment is in control of the group of learners. Knowledge is personally experienced and translated into competencies by the authentic learning context given(J. Herrington & Parker, 2013). More particularly, in project-based learning prior knowledge is shared among the peer group and integrated into competencies by students.

RESEARCH FRAMEWORK

As an Erasmus+ Strategic Partnership, OnCreate forms an umbrella under which the participating universities can coordinate their efforts in teaching and research. The consortium comprises a mix of universities and universities of applied sciences. The main strands of research are about different creative processes and their methods in online environments and about factors of successful online collaboration and their evaluation.

Partner	Short
University of Applied Sciences Magdeburg-Stendal, Germany	Magdeburg
University of Lincoln, UK	Lincoln
International YMCA University Kassel, Germany	IYU
Süleyman Demirel University, Isparta, Turkey	SDU
University of Applied Sciences Potsdam, Germany	Potsdam
University of Tampere, Finland	Tampere
University of Applied Sciences Tampere, Finland	TAMK
University of Aalborg, Denmark	Aalborg
University of Applied Sciences Lapland, Finland	Lapland
University of Ljubljana, Slovenia	Ljublana

Table 8 Project consortium

The first phase of OnCreate identifies the methods for collaborative teaching and learning that are currently employed by each institution (see Table 8). For this task, each university was paired with another institution to compare, discuss and evaluate good practices. In the context of this study, we consider methods and didactic concepts as “good practice” if they have been applied to the subjective satisfaction of staff and students in a university and are going to be applied in future courses. This was done in a formal descriptive way, circulating a form where partners described their uses of online collaboration in course contexts and as dialogue interviews in virtual meetings. The two partners then returned their mutual comparisons to the rest of the partnership.Issues of good practice were described in detail and suggestions were formulated in a way that could be easily translated to the design requirements of OnCreate. All results were compared side-by-side to highlight similarities and differences. Whilst the tools and frameworks currently being used to facilitate collaboration vary between each partner, the collection of good practices from across the partnership has identified a number of common structural issues, technical requirements and relevant pedagogies for facilitating collaboration. These issues are discussed below and will be evaluated and explored further throughout OnCreate.

RESULTS OF GOOD PRACTICE ANALYSIS

Evaluation of Tools for Collaboration

Virtual learning environments, social media and other digital technologies are embedded into the teaching undertaken at all OnCreate partner universities. Four partners currently run online courses, varying from large MOOC’s to more modest online university modules. These partners reported utilising a number of digital tools to enable remote collaboration. Six partners noted that they incorporated a range of methods for facilitating modes of blended learning in order to enable task-related group communication and to support the face-to-face teaching environment. Each partner institution used an affiliated learning management system to some extent, yet it was noted that these did not always meet the requirements of the department. As a result, other platforms were often utilised in conjunction with (or in place of) the core Learning Management System (LMS) to facilitate effective collaboration.

Moodle is the most popular LMS currently affiliated to the partner universities (8/10 use Moodle), although the use of this system varied widely between each institution. Moodle i is an open-source LMS that provides a central space on the web where students and staff can access a set of tools and resources. It is most commonly used to disseminate learning materials and departmental announcements, providing a place where staff can easily create web pages with course information and provide links to word documents, slides, and other resources that students can access. Blackboard, a proprietary LMS with similar functions, is used by Lincoln. Like Moodle, it is used as an archive and repository of information, lecture notes, course reading lists, relevant links, course information, module guides, etc. Moodle and Blackboard both provide opportunities to support classroom learning and structure the delivery of course content.

Whilst these two systems were found to be suitable for administrating university courses, it was suggested that they often provide a rather passive, one-way learning experience. Moodle and Blackboard do include interactive features that can be used to facilitate both synchronous and asynchronous collaboration (forum, wiki, discussion boards, ePortfolios, etc.), although it was reported that these platforms were not always sufficient for the learning culture at a number of universities. Despite providing opportunities for learners to work together as a whole class or in groups of various sizes, it was noted by IYU, Potsdam and Lincoln that these features were not particularly successful in generating meaningful collaboration, with students tending to create private Facebook groups for group interaction and discussion related to the course. As Curtis and Lawson (2001) highlight, effective remote collaboration is best achieved by using an appropriate platform where students are familiar with the interface, whilst communication with other group members and tutors is also made easily accessible. Lincoln noted that both students and staff were often ‘put off’ using Blackboard for communication and collaboration as the system is relatively slow and the interface is not intuitive, user-friendly or conducive for encouraging group participation and social interaction. Collaborating via social media, meanwhile, was found to be more in-line with students design expectations and web-habits (particularly features like closed groups, push notifications, file upload, and chat). Lincoln and Potsdam noted that it was hard to achieve parity or consistency across different module sites in both Blackboard and Moodle. This is largely due to the sheer amount of options available, and the fact that you have to enrol staff and students individually onto each module, meaning that students only have access to specific module content (preventing a shared, interdisciplinary learning culture).

All of the partner universities indicated they were using a range of other collaborative tools, either alongside their respective LMS or, in the case of Potsdam, in place of this altogether. Potsdam noted that whilst Moodle is the affiliated LMS for their institution, the Department of Design found that it was not suitable for their needs. As a result, the department developed a bespoke ‘learning-communication platform’ called INCOM, which they use in their teaching instead (it was also noted that Lincoln are in the process of designing their own bespoke LMS, based on the principles of collaborative learning). INCOM is used for inter-departmental communication, with users able to post news, appointments, offers for jobs/traineeships, etc. Despite similarities to features found in both Moodle and Blackboard, INCOM is non-hierarchical, meaning students have the same rights as teachers so they can create their own workspaces, invite people, upload material, post comments, etc. This method is particularly relevant to constructivist approaches to teaching as the learning environment is not dictated by the instructor/tutor. TAMK, Magdeburg and Lincoln also noted that collaborative design projects are best supported through this mode of student-lead, project-oriented teaching, where the instructor ‘coaches’ or ‘guides’ students to actively engage in the production of new knowledge. Whilst INCOM does overcome some of the drawbacks found in other LMS’s, it was noted that additional tools were still needed to support the learning environment. It was reported that INCOM is not suitable for presenting an overview of audio-visual material (Pinterest or Tumblr were used instead), and it does not allow for real-time chat (again, it was noted that students often utilised social media for this purpose).

SDU and Lapland both reported utilising wikis in their teaching practices. Features like tracking of page versions and delivery as a website position wikis as an alternative to other co-editing tools, especially when the results shall be available to a larger public. Students might use a wiki to collaborate on a group report, compile data or share the results of their research, while faculty might use the wiki to collaboratively author the structure and curriculum of a course. SDU stated that they use a wiki in their department for sharing course material and for peer-evaluation and peer-feedback activities.

Four partners were reported to use Google Drive, Docs and Hangouts alongside their respective LMS, especially when fostering group collaboration. It was noted that this suite of software “provides students with a range of interactive and accessible functions, enabling them to share knowledge with, and learn from, each other” (Muhammet Demirbilek, SDU). Respective partners adopted this approach due to the range of integrated software needs fulfilled by Google Drive and associated applications (word processing, spreadsheets, presentations, etc.). This enabled all participants to easily create and share documents from within the web browser, whilst the cloud system means that all users can access content across a range of networked devices (rather than being stored in one ‘physical’ location). A key advantage here is that when collaborative documents are prepared on Google Docs there is only one version, which is always up to date and documents all revisions. Subsequently, everybody has access to the same information, with each group member able to discuss the project in 'real-time' via the various comment and chat features.

The co_LAB research group at the University of Lincoln have incorporated Google Drive and associated apps across their teaching and research practices, encouraging students to use this platform as a shared space to discuss ideas, collaborate on shared artefacts and evaluate project research. Whilst this is a preferred method at Lincoln within the Digital Media department, it is not commonplace across the university and it was noted that there has been some resistance from both staff and students. It was suggested that some, especially those who are relatively new to working collaboratively online, may not feel comfortable using Google or Facebook in an educational context, as there are ethical concerns over intellectual property and data privacy (given that they are multi-billion dollar businesses that use personal data in their monetisation strategies). Nevertheless, the general feedback received from students has thus far been positive. This approach was adopted after being successfully implemented during another EU Erasmus project, 'Media Culture 2020', which brought together students and staff from 5 different universities from across Europe (including TAMK and Lincoln) to collaborate in a year long project that featured two physical workshops and a range of online activities. Google Drive and Docs were used alongside a closed Facebook group and dedicated blog to facilitate collaboration, share cultural knowledge, exchange and evaluate research, facilitate both synchronous and asynchronous communication, and to form social bonds within the mixed nationality teams (Thayne & Cooper, 2014).

A number of OnCreate partners (Lincoln, Aalborg, Magdeburg, Tampere) also recently took part in an online short course on Human-Centred Design, run by Acumen and IDEO. Participants were asked to form collaborative teams and respond to one of the available design challenges. This provided an opportunity to evaluate the delivery of the course and learning environment. The seven-week course was delivered entirely online via the NovoEd platform (some universities also ran physical workshops and formed local teams, while others worked remotely in international teams). NovoEd enables users to create shared documents with rest of the group, with Google Docs embedded into its collaborative 'workspace'. This demonstrates that the key functionalities of Google Drive and Docs can be successfully integrated into the delivery of online collaborative courses. Feedback from participants taking the course indicated that NovoEd was successful for fostering collaboration, for providing opportunities for teams to communicate and discuss assignments, and for delivering the course content and learning materials.

Suggestions and Requirements for Supporting Collaboration

As well as reviewing the technical solutions favoured by each university, the good practice interviews identified a number of structural and processual methods for facilitating collaborative learning. Suggestions were made by each partner for how OnCreate might respond to these concerns. A key requirement reported by all partners was the need to support group interaction and provide opportunities for both synchronous and asynchronous communications. This was found to be a central component for developing interpersonal, collaborative bonds during group work. As discussed above, there were differing approaches to how interpersonal connections were facilitated in an online environment across the partnership, although IYU did suggest that technology does not act as a simple substitute for face-to-face communication. All partners predominantly utilise a blend of remote and face-to-face modes of communication in their courses, although four institutions reported to run online courses where all communication and collaboration is done remotely. IYU reported to favour a ‘holistic’ educational approach, where learning takes place outside of the classroom in small personal groups and informal interaction with faculty on campus. They do run online programs that incorporate some elements of this approach, although there were concerns about the logistics of organising regular synchronous group meetings when working remotely (especially in an international context when dealing with students from different time-zones). IYU also noted that the internationalisation of online courses highlights a need to make sure the delivery of course materials are accessible to all, even students with limited internet access.

Aalborg and Ljubljana discussed how the ideation process in collaborative design projects might be improved. They found that in order to generate new ideas and non-formal discussion about a given project a change in environment is often beneficial. They suggested a number of ways a change of environment might be simulated when working in a shared office/classroom space. A simple solution would be a change of the screen and usage of background audio, a more advanced option would be usage of projectors to illuminate the office room with different patterns or objects and give it some kind of relaxed visual impression. The main obstacle is usage of additional equipment (projectors, etc.), which may not be available in all locations. It was also unclear how this might be translated to an online environment where teams are working remotely and in separate physical locations.

Another key suggestion that emerged from the good practice interviews was the importance of competence-based learning and self-assessment, where students are encouraged to reflect on the learning environment and evaluate creative processes (both individual contributions and the work of the team as a whole). In this approach, students are given opportunities to define learning objectives in the creative domain, which was noted as having a positive effect on student engagement and collaboration. Five partners reported to encourage students to chart their progress and reflect on their studies through what could be classified as an ePortfolio. JISC (Gray, 2008) define ePortfolios as “a learner-created collection of digital items: ideas, evidence, reflections, feedback, which presents a selected audience with evidence of a person's learning and/or ability” and ‘e-Portfolio-based learning’ as “…the process of planning, compiling, sharing, discussing, reflecting, giving and receiving feedback.”

CONCLUSIONS

All participating universities already embrace online collaboration to some extent. However, contrary to the promises of online Learning Management Systems, innovative online collaboration concepts were mainly found to be built on “mash-up platforms” comprising a mix of social media and current web services from outside of the domain of learning software solutions. The hierarchical and course-centred model of existing LMS allow for distribution of learning materials and online assignments, rather than for encouraging dynamic and effective peer collaboration. The challenge for future research is to develop a systematic approach of creating such mash-up environments for different challenges in creative collaboration, which is open to the fast-paced innovations in collaboration on the web.

Summary of key findings

1. Learning Management Systems are in use at all participating universities to distribute and structure course material.
2. The majority of the partners employ blended learning approaches.

3. Innovative and intense forms of collaboration are driven by web-based tools, notably facebook and google drive. Existing LMS are considered too slow and confuse users by a plethora of options.
4. Online collaboration efforts at the partner universities use no formal hierarchies and peer-based learning concepts.
5. However, the pace of adoption differs even inside the institutions. An important concern is the reservation of students and staff against using tools by large web corporations in an educational context, e.g. for privacy reasons.

Based on the findings in this paper, OnCreate will initiate further qualitative and quantitative research along with a specific schedule of experimental inter-university courses.

REFERENCES

Burke, M., & Settles, B. (2011). Plugged in to the community: Social motivators in online goal-setting groups. In *Proceedings of the 5th International Conference on Communities and Technologies* (pp. 1–10). ACM. Retrieved from http://dl.acm.org/citation.cfm?id=2103356

Curtis, D. D., & Lawson, M. J. (2001). Exploring collaborative online learning. *Journal of Asynchronous Learning Networks*, *5*(1), 21–34.

Doorley, S., Witthoft, S., & others. (2011). *Make space: How to set the stage for creative collaboration*. John Wiley & Sons.

Dow, S., & Settles, B. (2013). modeling online creative collaborations. *XRDS: Crossroads, The ACM Magazine for Students*, *19*(4), 21–25.

Gray, L. (2008). Effective Practice with e-Portfolios. Supporting 21st century learning. *JISC Innovation Group*. Retrieved from http://www.eife-l.org/publications/eportfolio/proceedings2/ep2008/presentations/Linda_GRAY.pdf

Herrington, A., & Herrington, J. (2006). *Authentic learning environments in higher education*. IGI Global.

Herrington, J., & Parker, J. (2013). Emerging technologies as cognitive tools for authentic learning. *British Journal of Educational Technology*, *44*(4), 607–615.

Holmes, J. (2012). *Education Futures: University Grads Don't Make the Grade*. Woods Bagot. Retrieved from http://ctibi2012.wikispaces.com/file/detail/Business_Schools_EducationFutures.pdf

Hong, S. W. (2013). *The Affordance of Online Multiuser Virtual Environments for Creative Collaboration*. University of California, Berkeley. Retrieved from http://escholarship.org/uc/item/91r0m1sx.pdf

Johnson, D. W., & Johnson, R. T. (1996). Cooperation and the use of technology. *Handbook of Research for Educational Communications and Technology: A Project of the Association for Educational Communications and Technology*, 1017–1044.

Kalay, Y. E. (2006). The impact of information technology on design methods, products and practices. *Design Studies*, *27*(3), 357–380.

Leidner, D. E., & Jarvenpaa, S. L. (1995). The use of information technology to enhance management school education: A theoretical view. *MIS Quarterly*, 265–291.

Okuogume, A., & Jäminki, S. (2011). Experiences of integrating R&D&I in Professional Master Degree Programme: The Case of Competence-Development-Based Education. Presented at the Combi Conference, Hämeenlinna.

Runco, M. A., & Jaeger, G. J. (2012). The standard definition of creativity. *Creativity Research Journal*, *24*(1), 92–96.

Schoenfeld, A. H. (1999). The core, the canon, and the development of research skills: Issues in the preparation of education researchers. *Issues in Education Research: Problems and Possibilities*, 166–202.
Schrage, M. (1995). *No more teams!: Mastering the Dynamics of Creative Collaboration*. New York, NY, USA: Currency Doubleday.
Slavin, R. E. (1980). Cooperative learning. *Review of Educational Research*, *50*(2), 315–342.
Thayne, M., & Cooper, G. (2014). Collaborative pedagogy and digital scholarship: a case study of'Media Culture 2020'. *Media Education Research Journal (MERJ)*, *5*(1), 46–59.
Vygotsky, L. (1978). Mind in society: The development of higher psychological processes. *Cole, M., et Al., LS Vygotsky. Harvard University Press, Cambridge*.

This work was co-fundedby the European Union through the ERASMUS+ programme by grant 2014-1-DE01-KA203-000706.

Martyn Thayne is a lecturer for the School of Film and Media at the University of Lincoln. He is a founding member of the co_LAB research network, which was set up to develop more open, interdisciplinary and collaborative pedagogical frameworks. Martyn is particularly interested in forms of blended learning, exploring the use of collaborative digital technologies to enhance the teaching and learning environment.

Björn Stockleben works as scientific coordinator of the Master programme Cross Media at University of Applied Sciences Magdeburg-Stendal. As well he is working for the innovation group of Rundfunk-Berlin Brandenburg as project manager and UX consultant in EC- and ESA-funded research projects. He is an expert in participative media, interactive service design and non-linear AV media, as well as didactics of blended learning and adult education.

Seija Jäminki works as an Online Tutor and researcher at Lapland University of Applied Sciences in Tornio, Finland. She has a PhD in eLearning and teaches online skills, team & project work and virtual practices. She is a member of the Lapland UAS Business team; Research team and Qualitative Team.Since 1988 she has been involved with several international projects focusing mainly on the integration of ICT in education.

Ilkka Haukijärvi is a development manager at Tampere University of Applied Sciences. Ilkka has been involved in higher education organisation´s development for the past seven years, currently the main area of responsibility being the development of e-learning in Tampere University of Applied Sciences, and the implementation of a five year strategy for the digitalization of learning and teaching.

Nicholas Blessing Mavengere is a doctor in Information and Systems and has worked as a researcher at the University of Tampere, Finland since 2009. His research interests include supply chain management, business strategy, strategic agility, ICT for development, the role of IT in business and the use of social media for learning purposes. His PhD completed in 2013, was selected best PhD at a top European conference and received an awarded from School of Information Sciences.

Muhammet Demirbilek is currently an Assistant Professor of Educational Technology in the School of Education at Suleyman Demirel University. He used to work as a graduate faculty and visiting research scholar in Digital Worlds Institute at University of Florida (UF). He earned his doctoral and master's degree in Educational Technology from the University of Florida worked as a Post-Doctoral Researcher at Games, Learning, and Society (GLS) at the University of Wisconsin-Madison.

Mikko J. Ruohonen is a professor at the University of Tampere, Finland. His interests are in information strategies, electronic business, knowledge management, inter-organizational learning, ICT for development and mass customization. He has published more than 130 articles and reports, four textbooks and a number of large industry reports. The International Federation of Information Processing (IFIP) granted him Silver Core Award 2007 due to his work for ICT and education.

Fine-Grained Recording of Student Programming Sessions to Improve Teaching and Time Estimations

Daniel Toll, *daniel.toll@lnu.se*

Tobias Ohlsson, *tobias.ohlsson@lnu.se*

Morgan Ericsson, *morgan.ericsson@lnu.se*

Anna Wingkvist, *anna.wingkvist@lnu.se*

Linnaeus University, Dept. of Computer Science, Sweden

Abstract

To have direct observation of students during an online programming course is impossible. This makes it harder for teachers to help struggling students. By using an online programming environment we have the opportunity to record what the students actually do to solve an assignment. We can analyse the recordings and provide teachers with valuable information. We developed and used an online programming toolwith fine-grained event logging to observe how our students solve problems. Our tool provides descriptive statistics and accurate replays of a student's programming sessions, including mouse movements. We used the tool in a course and collected 1028 detailed recordings. We compare fine-grained logging with existing coarse-grained logging solutions to estimate assignment-solving time. We find that time aggregations are improved by including time for active reading and navigation enabled by the increased granularity. We also divide the time users spent into editing (on average 14.8%), active use (on average 37.8%), passive use (on average 29.0%), and also estimate time used for breaks (on average 18.2%).Finally wesee a correlation between early student submission results and students that hand in later, but also see an example where the results differ significantly.

Keywords

Computer Science Education, Learning Analytics, Educational Data Mining

INTRODUCTION

In the past programming assignments often demanded access to hardware (computers) and software (compilers and tools) that was not easily available outside the campus. Students were scheduled to computer laboratory rooms, and participation was often mandatory. This allowed teachers to get a sense for what problems students experienced. Teaching was very agile; the class could be gathered during the laboratory session for impromptu lectures that provided more information on a certain topic or a quirk in the development environment. Computers and development tools are now widely available, and it is not uncommon for students to have a fully working laboratory environment at home. As courses move off-campus and online, this is almost mandatory. The scheduled laboratory work is more and more used for examinations or to assist with specific questions handled by teaching assistants. The opportunity for teachers to observe students while programming has almost disappeared. Based on experience from our own mixed mode programming courses, we feel that we do not understand the problems our students face. This feeling is confirmed by Fenwick et al. (2009) in their survey on student programming behaviour:

> "This disparity between student and faculty perceptions is an indication of the profound lack of understanding of student software development practices."(p. 296)

To understand students teachers of programming courses have recorded student programming sessions on different granularity levels, from the most coarse level of student assignment submissions to fine level logging of each keystroke (Jadud, 2006; Norris et al., 2008; Rodrigo & Baker, 2009; Helminen, Ihantola, & Karavirta, 2013; Matsuzawa, Okada, & Sakai, 2013; Vihavainen, Helminen, & Ihantola, 2014).

We make the following contributions:we implemented an even finer grained logging of student programming sessions with mouse movement, text selections and text cursor positions, as well as placing an upper limit on the amount of time students spend in the editor. We compare these finer grained logs with previous more coarse-grained in activity aggregations and find that the time for editing is only a fraction of the total amount of time students spend on an assignment. We divide the time spent into activities and use the fine-grained time measurements to determine how the sessions (e.g., problems experienced) of students that finish early compare to students that finishes later.

- **RQ1**. How much of total tool usage time can be estimated using coarse-grained models?
- **RQ2**. What is the fraction of programming time spent on editing, reading and browsing assignment code? And how much of the time do student spend outside of the tool?
- **RQ3**. Does early feedback predict what programming assignments the main group of students are going to struggle with?

BACKGROUND

In off-campus courses, for example MOOCs, an online programming environment is attractive; it provides the opportunity to record events when the students work on their assignments. Most current studies collect data at a coarse-grained level (e.g., submission times, code-changes, or compiler errors) to allow for more quantitative data to understand students. BlueJ[8]is a popular free Java development environment aimed at beginners. Jadud(2006)used BlueJ to record and investigate the Edit-Compile cycle of novice Java programmers by taking snapshots of the code on each compilation. His findings are in line with other research (Helminen, 2013; Fenwick, 2009), a majority of errors are cause by a small subset of compiler errors — most students make the same mistakes. Jadud(2006)also studied the compilation patterns of students and found that a large portion of students recompile without making any changes in the code. Using the collected compilation error data he defined the error quotient (EQ), a model to help identify students that struggle. Utting et al. (2012) plan to instrument BlueJ to allow for large-scale anonymous data collection and make the collected data available to other researchers. The goal is to allow for more power in quantitative analysis and also to study situations that do not occur that often which smaller samples may miss entirely. Helminen et al. (2013)include an interactive Python console in their online programming tool. They collect testing behaviour statistics on how many students write their own tests or uses the assignment's provided tests. Execution error statistics that indicate students' lack of API or language knowledge is also collected. Their granularity level is on recording code submissions, code edits, console interactions and also record when the student started their tool and closed it. Vihavainen et al. (2014) instrumented NetBeans to collect fine-grained recordings of keystrokes and events and compare different sampling granularities and report compilation success rates on beginner programmers. For further reading Helminen et al. (2013)provide a good overview of how recordings on various abstraction levels can be used to analyse student

[8]http://www.bluej.org/

behaviour. In a number of studies time is estimated using aggregation of coarse-grained events(Norris et al., 2008; Murphy et al., 2009;Matsuzawa et al., 2013). We have observed students on campus that spend a lot of their programming time thinking and reading before acting. Time measurements that do not include reading time would not do them justice.

Computer Science Quiz

We developed the Computer Science Quiz tool (CSQUIZ) to bridge the gap between the students and the teacher's perception of programming assignments. CSQUIZ is a complete programming and learning environment that presents instructions and theory relevant to an assignment, a multi-file code-editor that can execute code and provide feedback from the interpreter, and automated tests. Like many other tools, CSQUIZ is aweb-application that automatically assesses and records students when they work on their programming assignments. The overall goal is to support the teacher with visualizations and descriptive statistics to help find and address shortcomings in the course during its execution, and to perform a detailed analysis and suggest improvements for the next iteration of the course. CSQUIZ provides an exact replay of the students' interaction with the programming environment. The detailed recordings can be played back at different speeds and paused; mouse pointer, text selections, and file changes are all replayed accurately. This allows the teachers to build a deep and detailed knowledge about how students solved a problem.

CSQUIZ records student activities on a fine-grained level. A recorded programming session is a collection of time stamped events. Each session is stored on the server and is identified by a unique hash sum for the student and the name of the assignment. CSQUIZ records the following events: CSQUIZ becomes active/inactive, browser becomes in focus or out of focus, a source file is changed, the text cursor is moved, text is selected, a file is reset (all changes dropped), CSQUIZ is reloaded, the produced code is executed (output and errors are logged), switching file, and mouse movements. Mouse changes are recorded at 11 samples per second. The tool also records how long the editor is inactive. We built CSQUIZ for detailed time measurements, for example to record how long a student spend on reading instructions before starting to work on an assignment. Therefore the instructions and programming area fades if a student is inactive for more than 15 seconds. This effectively allows us to put an upper limit on the amount of time spent in the tool.

EMPIRICAL STUDY

To answer the research questions we use data collected by CSQIUZ. Other studies (Norris et al., 2008; Murphy et al., 2009; Matsuzawa et al, 2013) present time aggregated on a coarser granularity, e.g., using compilations or key events. Coarse granularity may overestimate the time it takes to complete an assignment by including time that students spend on other activities, or underestimate the time by considering breaks as inactivity and not pauses. Some of our assignments include several files and classes, and contains around 800 lines of text. Reading and browsing has been shown byKo et al.(2006)to be a large part of a programming effort. CQUIZ captures the time for browsing and reading to capture a larger proportion of the entire student effort. It is thus interesting to compare coarse measurements with fine-grained (RQ1) to determine if there is a difference. To find the fraction of programming time spent on different activities we do temporal aggregation from a sequence of discrete events. For example a stream of key press events closely related in time is aggregated into a single measurement ofediting time.

To be able to extract several different types of activities, we aggregate on four granularity levels similar to those found in other studies, from the coarsest(G1), reacting on application start up, saves, and compiles, to the finest(G4) based on text visibility.

- G1. Compilations: Tool is started, saving, running.
- G2. Text is changed: Key input, pasting, and file resets.
- G3. Active Use: Mouse movements, text selections, text cursor movements.
- G4. Time in tool. Text is visible

Each granularity level includes the events from coarser levels, e.g., G2 includes all events in G1. This enables us to derive the time for an activity by subtracting one granularity from another. An event threshold is used to connect events into time aggregations. An event that is followed by another event within the threshold is counted as the time between the events. For example if we use a threshold of 10 seconds, two text changes that happen with 9 seconds apart would count as one period of 9 seconds. With a threshold of three seconds these events would be considered separate. In our aggregation model a separate event that is not followed by another event within the threshold is counted as 0.5 seconds. We use CSQUIZ recordings from 66 second year university students completing 17 different programming assignments:resulting in a total of 1028 programming sessions.The assignments were divided into two blocks. A first block of 10 assignments was introduced on September 1,2014. During this block, the tool and assignments were updated on two occasions. A minor change in the output visualisation was introduced on September, 3 2014 to make it clearer for the students if application did not produce any output. The second update was the introduction of seven new assignments in block two on September 19,2014. Hence, the data was collectedbetween September1, 2014 andOctober 13,2014.

RQ1, Comparing granularities

Each increased granularity level may introduce additional issues. For example going from recording submissions to compilations requires some type of instrumentation of the programming tool. CSQUIZ records events with high frequencies (e.g. mouse and key events) resulting in a lot of network traffic and large log files. CSQUIZ also introduces inconveniences for the students by fading out text on inactivity. By increasing the event threshold, events that appear further apart are then regarded connected. Thus, it can be argued that the total time used in the tool could be estimated with a lower sampling granularity level but with a higher threshold.

Method

We measure on four granularity levels (G1-G4) and five different thresholds. We selected the thresholds: 3 seconds, 10 seconds, 1 minute, 5 minutes, and 15 minutes to be able to compare to related works and to reach overestimation for each granularity. Each combination of granularity and threshold forms an aggregation model and each of the 1028 programming sessions is aggregated using the 20different aggregation models. We compare the time aggregation values with CSQUIZ best estimation of Time in Tool (*TiT*) with 3-second threshold and compute the error (percentage). The value presented is calculated as *(TiT-GX(Y))/TiT*, where *X* is the granularity level and *Y* is the threshold. We present the error percentage and standard deviation to determine the precision of the coarser granularities. A negative percentage means an underestimation and a positive means an overestimation.

Results

In Table 1 we present error averages in percentfor total activity *TiT* depending on sampling granularities and thresholds. The values should be interpreted as follows: a measurement on ”Compilations” granularity(G1) and with 1 minute threshold results in an underestimation of the *TiT* and only captures on average 22%(-78%) of the Time in Tool. If we choose a coarser sampling granularity we get either an average underestimation or an overestimation, or get a large standard deviation. For example,measuring text changes (G2) with a threshold of 5 minutes results in quite accurate readings on average (-2%) but a high standard deviation. Using the ”Active Use” granularity (G3) improves the estimations, and we get quite accurate readings with a threshold of 1 minute. This is not surprising since mouse movements are used by the students to keep the text visible. If application start up is NOT included, the Compilation granularity (G1) constantly underestimates the time spent even with large thresholds. At a three hour threshold it creates acceptable average time but large individual errors.

Threshold:	3 seconds		10 seconds		1 minute		5 minutes		15 minutes	
	Error	stdev	Error	stdev	Error	stdev	Error	stdev	Error	std ev
G1. Compilations	- 99%	0.8%	-98%	3.4%	-78%	22.3 %	-27%	38.0 %	+6%	35. 7%
G2. Text change	- 83%	9.3%	-73%	13.6 %	-43%	23.0 %	-2%	24.5 %	+17%	31. 1%
G3. Active Use	- 37%	15.7 %	-21%	13.5 %	+2%	6.7%	+19%	27.5 %	+32%	59. 8%
G4. Time in Tool	0%	0%	+1%	1.9%	+7%	8.5%	+21%	30.3 %	+35%	72. 3%

Table 1: Comparing combination of thresholds and time aggregation granularities using 1028 recorded programming sessions to estimate how much of the ”Time in Tool” a combination captures on average. A negative value is an underestimation and a positive value an overestimation.

Discussion

The Active Use (G3) granularity estimations are probably higher than they would be if CSQUIZ did not fade text on inactivity. When looking at playback of the recordings we sometimes observe a passive period followed by a jerky mouse movement that is used to regain text visibility. The G1 granularity is perhaps a little unfairly represented here due to the way we treat separate events (as 0.5s), only compilations that are within threshold of each other are going to matter. Using larger thresholds and fine-granularities quickly result in overestimations and capture usage outside of the tool, which we use to answer the next question. It is clear from the values in Table 1 that in order to measure *TiT*, granularity and threshold matters. We consider capturing mouse worth the effort to get accurate readings. Fading text might be too intrusive.

RQ2, Time spent reading

To estimate the reading time without eye-tracking is difficult since reading does not necessarily involve any interaction. However, by fading text on inactivity CSQUIZ provides a definite time cap on tool usage, since students must stay active to be able to continue. CSQUIZ provides us with an opportunity to measure the time when students move the mouse over the application (active use) and the amount of time the text is visible (passive use).The true time for reading and navigation may be up to thesum of both but not higher.

Method

We use the time aggregation model described earlier and divide the time spent into activities. We start by computing a Total Time (*TT*) for all activities linked to an assignment. We measure on granularity level 4 (G4) but use a 15 minute threshold, so we will capture time spent outside of the tool, for example online searches or noise from other sources such as social media. We consider 15 minutes to be the longest break one would take and still be considered working in one session. We also measure the *TiT* by measuring on G4 but with threshold of 3 seconds. The Editing Time (*ET*) is measured using G2 with three second threshold and the Active Use Time (*AUT*) is measured on G3 with 3 seconds threshold.
From these values we compute the average fraction time for Editing *EF = ET / TT*, Active Use *AU = (AUT - ET)/ TT*, time Out of Tool *OT = (TT - TiT) / TT* and Passive Use *PU = 1 - (EF + AU + OT)*. These values are computed for each recorded programming session and statistics are computed and presented.

Results

We estimate the time a student spent on reading and navigating, and present the results in Table 2. Compared to Table 1 we show results as percentages of total assignment solving time, including time spent outside the tool. We find that on average, students are actively using mouse, doing text selections or moving the text cursor37.9% of the time. This time can be examined using CSQUIZ's replay functionality.The *PU* time represents time that we cannot account for; we know that the text has not yet faded, but we have no events from the user. When we watch the recordings, the mouse is sometimes moved outside the application, or just stops moving for a while. The amount of such breaks varies a lot between students, perhaps due to equipment. For example, a touchpad may have less frequent input than a mouse. We also find that on average 18.2% of the time is spent outside the tool. These breaks can be up to 15 minutes long (since we used a 15 minute threshold). Longer breaks are not included, since we find it unlikely that students would be actively searching for information on a task and not stay in the tool. We have not found any evidence of students trying to solve the tasks in other tools. For short assignments the *OT*time is smaller, and for the longer assignments students are taking more short breaks, perhaps to search for information or other things. The*OT* median was for all assignment 11.7%.

Activity	Average	Median	Min	Max	Standard deviation
Editing (*EF*)	14.8%	12.7%	0.2%	57.0%	9.8%
Active Use (*AU*)	37.9%	36.5%	2.8%	96.8%	16.2%
Passive Use (*PU*)	29.0%	28.2%	0.0%	75.0%	13.1%
Out of Tool (*OT*)	18.2%	11.7%	0.0%	93.2%	19.3%
Total Time (*TT*)	100%				

Table 2: Estimate time spent on activities based on 1028 recordings with 17 different programming assignments. Total time includes breaks up to 15 minutes.

Discussion

The actual time for reading and navigation is a combination of the *AU* and *PU* but not higher than their sum, on average 67%. It could be lower since the student could be moving the mouse over the application but not really reading. If more control could be added to the measurement situation, for example by eye-tracking or capturing the entire screen we could improve these values. Such recordings would explain more of what the students do during their short breaks, and provide clues on what happens during the passive time in our tool.

RQ3, Predictions

During the autumn 2014 we aimed to use CSQUIZ to focus our teaching resources on problematic assignments. Based on the statistics presented by CSQUIZ we tried to predict which assignments students got most frustrated with. After the course we compare feedbackthe teacher got from the students that completed the assignments during the first day, with the feedback from students that started later (RQ2). We expected three types of effects: motivated students have been reported to start early and this also correlates with better grades (Fenwick Jr., et al., 2009). We also know from experience that students help each other. Finally, lecture content and more time to study mayalso effects the results. A good prediction model would allow us to identify what assignmentswhere we should spend teaching resources. For example we could use lecture time to address issues in a problematic assignment.

Method

We compare the group of students who completed assignments before September 2, 8AM(morning of day two, in block one), with students that started and completed the assignments later. We remove recordings of students that started before but did not complete the assignment until after 8AM. We measure the time in the tool (G4), the number of forum questions, and the number of students that had completed the assignment on the first block of 10 assignments. The students had to complete an assignment before they were able to continue to the next one. Two assignments (6 and 9), has two versions, each part of an experiment, and the students were randomly divided to get one of them. We present correlation measures using Pearson's R and use Student's T to check for statistical difference between the two samples of students on each assignment.

Results

We compare the quick students with those who complete the assignments after the first day. The average time measurements are presented in Table 3 as "pre" and "post". There is a strong correlation (Pearson $R = 0.90$, $R^2 = 0.81$) between the pre and post time averages. We also compare the early and late group using a two tailed Students T-testto test for significant differences. We note that for the first three assignments solving time took longer for the students that started immediately. Assignment 3 (A3) takes significantly longer time at $P < 0.05$. For the assignments after A5 we see that the students that started early do these assignments on average faster. A10 shows significantly different at $P < 0.01$.

	Assignment											
	1	2	3	4	5	6.1	6.2	7	8	9.1	9.2	10
Pre (s)	239	245	1138	4078	4552	388	401	189	286	219	276	1775
Post (s)	190	153	737	5082	5188	530	926	520	685	648	699	5149
# Forum posts	0	0	0	4	4	1	0	0	0	0	1	2
Pre N	38	37	36	33	25	8	3	8	8	3	5	8
P	0.23	0.17	*0.016	0.31	0.47	0.45	0.39	0.25	0.11	0.18	*0.032	**0.008

Table 3: Table show average times to solve assignments between quick students (pre) and the students that completed the assignments after the first day(post). P is Student's-T p-values for Two-tailed tests between pre and post, * significant at P < 0.05. ** Significant at P < 0.01.

Discussion

From these measurements it seems reasonable that the 8 students that finished all ten assignments early are also more skilled or motivated. Excluding A10 from the prediction model gives a very good fit (Pearson R = 0.985, R^2 = 0.97), but the significant difference in A10 shows that judgements on how difficult an assignment is based on too few early measurements is unreliable.

The T-test results should be taken with a bit of scepticism since the later tasks are measured on a much smaller group of students.Both the normality and the equal variance requirements on T-tests are violated and the measurements are not completely independent. When we do multiple T-tests, we might find significant P-values by pure chance.

Two assignments (A4 and A5) stood out in terms of average time, number of compilations, and number of questions asked by the students. We based our conclusions on the faster students and missed the frustration of the struggling students in a third assignment (A10). During the course we misunderstood the problems students faced on A4 and based our added lecture content on what we thought was the challenging part of the assignment. Inspections of A4 recordings would have revealed that students spent much time trying to understand the assignment and CSQUIZ itself. Helminen et al. (2013)points out the problem of understanding a (new) tool. We used a group of six colleagues and two students to test CSQUIZ before the course started. While this group experienced some problems with interpreting CSQUIZ, most of the problems faced during the course were specific for the student group.

LIMITATIONS

The level of control of the programming situation is fairly low in the sense that students most probably helped each other.We have seen at least one example of an unlikely API-method choice in several student solutions. We have not seen any indications that students have directly copied a solution into our tool, but cannot rule out that a student copied by writing. CSQUIZ fades text on inactivity; this most probably affected the “Active Use”granularity (G3) since students must stay active to continue reading. All data presented in this work is the result of a single class of 66 undergraduate students. In the group of students there is a subgroup of skilled programmers with years of experience, while others have little experience. This may have an effect on our prediction where a group of fast students completed the assignments early. We also present a limited sample of 17 short PHP assignments, many of which included files ranging from a couple of lines to around 800 lines. The assignments is supposed to be short (minutes to an hour) and our result may thus not be valid for longer sessions.

RELATED WORK

There are a number of tools that compute and use development time to support teaching and learning. ClockIt calculates and presents time to both students and teachers(Norris et al., 2008).ClockIt capture start, exit, compile, save, error, and file change events using instrumentation in BlueJ. This is similar to our granularity level G2. Extrapolating from our results, assuming similar exercises and students, having a threshold of 5 minutes could give them acceptable time estimations (see Table 1). Rodrigo and Baker (2009)also instruments on compilation level, and estimates student frustration. They intend to try more fine-grained approach with a full replay of keystrokes, mouse movements and other events. We think that having access to fine-grained recordings like ours would benefit their work.

Retinais another tool that collects compile and runtime errors(Murphy et al., 2009). Retina approximates time spent from the number of programming sessions (e.g., compilations that are more than 30 minutes apart). These approximations are used to present time predictions to students. This is an interesting approach and we intend to use our data to see if student solving time can be accurately predicted. Matsuzawa et al. (2013)used detailed logging of key strokes in a tool to enable students to observe, analyse and improve their own programming process. They aggregate data on 5 minute thresholds and would thus (using our result) get a quite good estimate on average but get a high standard deviation. Ko et al. (2006)let a group of 10 developers solve five maintenance tasks during 70 minutes and captured their work using full screen recordings. They divided their activities to be even finer grained than ours. Using their results and aggregating similar activities show a combined reading and navigation time around 45% compared our "Active Use" of 37.9%. Their editing time was around 20% compared to our 14.8%. The main difference is that we compute these values automatically.

Marmoset[9]and Web-Cat[10] are popular assignment submission systems. Data from these systems (and modifications of them) have been used to find patterns in basic student behaviour and high-level strategies used to solve the assignments. The major pattern found is that students that start early on an assignment typically perform better and students that submit late tend to continue to do so. This behaviour also correlates positively with lower grades (Fenwick Jr., et al., 2009). More advanced analyses and model building have also been performed to find patterns in how students solve their assignments. For example, Hidden Markov Models was used to find problem-solving strategies used by students, where some strategies in the model correlate with higher student performance (Piech, 2012; Kiesmueller, 2010). This is in line with our results that early students may be faster on assignments (A4-A10). But we also see that for (A1 to A3) this group is actually slower in completing the assignments than the latter group.

CONCLUSIONS AND FUTURE WORK

We present an online tool called CSQUIZ to help teachers understand how students solve programming assignments. A detailed recording of what each student do during their problem solving can provide interesting statistics and visualizations. A teacher can watch a full replay of a student's problem solving effort and gain in-depth knowledge about it. We improved the granularity level of the recordings to include mouse movements, text-selections and text cursor positions. We compare different granularity levels and find that active use reduces much of the error when estimating time spent in tool (RQ1). We are able to divide the time spent into editing, active use, passive use, and time out of tool. The amount of time for active use such as reading and navigating the tool is estimated to 37.9% on average (RQ2). We calculate the maximum amount of time spent in the tool, but not actively interacting with it(on average 29.0%), thus providing a measurement of the time that cannot be accounted for but could be used to read code or instructions. Descriptive statistics derived from fine-grained recordings further helped us identify which assignments were the most problematic for the students. We see a correlation between early student submission results and students that hand in later but also see an example where the results differ significantly, seeA10 in Table 3 (RQ3).

In the future, we plan to use the fine-grained logging to examine how different assignments affect student behaviour. We are aware that fine-grained logging (timed

[9]http://marmoset.cs.umd.edu

[10]http://web-cat.org

key strokes and mouse movements, etc.) is used in a different but related context; to understand the cognitive writing process (Leijten & Van Waes, 2013), e.g., differences between novice and expert writers, and burst analysis (how much is written before pausing). In this context, logs have been complemented by eye-tracking and thinking-aloud protocols to help determine what the writer was looking at and thinking. Analysis using network and data mining techniques has also been used to provide information about writing strategies using multiple electronic sources. It seems highly relevant to explore how students use online sources as part of their assignment solving process. We plan to explore these techniques in the future.

REFERENCES

Fenwick Jr., J. B., Norris, C., Barry, F., Rountree, J., Spicer, C., & Cheek, S. (2009). Another Look at the Behaviors of Novice Programmers.*Proc. of the 40th ACM Technical Symposium on Computer Science Education* (pp. 296-300). Chattanooga, TN, USA: ACM.

Helminen, J., Ihantola, P., & Karavirta, V. (2013). Recording and Analyzing In-browser Programming Sessions. *Proc. of the 13th Koli Calling Int. Conf. on Computing Education Research* (pp. 13-22). Koli, Finland: ACM.

Jadud, M. (2006). Methods and Tools for Exploring Novice Compilation Behaviour. *Proc. of the 2nd Int. Workshop on Computing Education Research* (pp. 73-84). Canterbury, United Kingdom: ACM.

Kiesmueller, U., Sossalla, S., Brinda, T., & Riedhammer, K. (2010). Online Identification of Learner Problem Solving Strategies Using Pattern Recognition Methods. *Proc. of the 15th Conf. on Innovation and Technology in Computer Science Education* (pp. 274-278). Bilkent, Ankara, Turkey: ACM.

Ko, A., Myers, B., Coblenz, M., & Aung, H. (2006). An Exploratory Study of How Developers Seek, Relate, and Collect Relevant Information during Software Maintenance Tasks. *Software Engineering, IEEE Transactions on Software Engineering, 30(12)*, 971-987.

Leijten, M., & Van Waes, L. (2013). Keystroke Logging in Writing Research. *Written Communication, 30(3)*, 358-392.

Matsuzawa, Y., Okada, K., & Sakai, S. (2013). Programming Process Visualizer: A Proposal of the Tool for Students to Observe Their Programming Process. *Proc. of the 18th ACM Conf. on Innovation and Technology in Computer Science Education* (pp. 46-51). Canterbury, England, UK: ACM.

Murphy, C., Kaiser, G., Loveland, K., & Hasan, S. (2009). Helping Students and Instructors Based on Observed Programming Activities. *Proceedings of the 40th ACM Technical Symposium on Computer Science Education* (pp. 178-182). Chattanooga, TN, USA: ACM.

Norris, C., Barry, F., Fenwick Jr., J., Reid, K., & Rountree, J. (2008). ClockIt: Collecting Quantitative Data on How Beginning Software Developers Really Work. *Proceedings of the 13th Annual Conference on Innovation and Technology in Computer Science Education* (pp. 37-41). Madrid, Spain: ACM.

Piech, C., Sahami, M., Koller, D., Cooper, S., & Blikstein, P. (2012). Modeling How Students Learn to Program. *Proc. of the 43rd ACM Technical Symposium on Computer Science Education* (pp. 153-160). Raleigh, North Carolina, USA: ACM.

Rodrigo, M. M., & Baker, R. (2009). Coarse-grained Detection of Student Frustration in an Introductory Programming Course. *Proc. of the 5th Int. Workshop on Computing Education Research* (pp. 75-80). Berkeley, CA, USA: ACM.

Utting, I., Brown, N., Kölling, M., McCall, D., & Stevens, P. (2012). Web-scale Data Gathering with BlueJ. *Proc. of the 9th Int. Conf. on Computing Education Research* (pp. 1-4). Auckland, New Zealand: ACM.

Vihavainen, A., Helminen, J., & Ihantola, P. (2014). How Novices Tackle Their First Lines of Code in an IDE: Analysis of Programming Session Traces. *Proceedings* *of the 14th Koli Calling International Conference on Computing Education Research* (pp. 109-116). Koli, Finland: ACM.

Creating Digital Devices in Science Classes

Michael Weigend, *mw@creative-informatics.de*
University of Münster, Institut für Didaktik der Mathematik und der Informatik, Germany

Abstract

Engineering digital devices can make science education more attractive. This contribution presents easy-to-make projects from photometry. The students construct spectrophotometers and sophisticated digital devices that measure the light absorbance of dye solutions. All they need is a computer, a PicoBoard with light sensor, and materials and tools that can be found at home and in a regular chemistry lab at school. The programming is done in Scratch.

Keywords

Science, Programming, Photometry, Chemistry, Scratch

INTRODUCTION

Hands on activities in a chemistry lab at a high school include planning and conducting experiments, observing chemical reactions and explaining the observations, developing research questions and hypotheses. Students take the role of a scientist, trying to understand and explore fundamental laws of nature. This paper presents chemistry projects that challenge the students in a different way. They take the role of an engineer instead of a scientist, using chemistry knowledge to design useful digital devices applying the visual programming language Scratch (Resnick 2009). In these projects science knowledge is elaborated as well, just the perspective is different. The goal is not to explore the world as it is but to change it and to make it a better place.

Distinguishing between science and engineering seems to be quite common.

Petroski (2010) points out that engineering tasks imply greater responsibility than scientific tasks. For example, the oil spill in the Gulf of Mexico 2010 was a consequence of poor engineering decisions, which allowed gas to escape and cause an explosion. Nobody blamed science for this disaster.

According to Kolb's Experiential Learning Theory (ELT), scientific and engineering activities correspond to different learning styles. ELT defines a learning cycle consisting of four elements (learning orientations): concrete experience (CE), reflective observation (RE), abstract conceptualization (AC) and active experimenting (AE). Individuals have different preferences (learning styles). "Convergers" prefer AC and AE. They tend to major in mathematics and engineering. Those with "Assimilating" styles prefer AC and RO and often have a background in mathematics, chemistry and physical sciences (Kolb 2005). Both styles include abstract conceptualization, but "Assimilators" prefer reflective observations whereas "Convergers" prefer active experimenting.

Engineering assignments foster creativity to a greater extent than scientific discovery tasks can do. Traditional discovery tasks in Chemistry - like investigating properties of a material - tend to lead to the same activities: Try to burn the material, to solve it in water, measure the melting point and so on. And when students have unconventional ideas, it might get dangerous. Chemical experiments must always be approved by the teacher for safety reasons.

On the other hand, engineering projects are much more complex and imply a wider range of thinking and doing. A student has to consider not just relevant scientific facts and laws but also the practical purpose, the person who is going to use the device, resources that are available, expected benefit for the individual users and the community and so on. There exist some popular non-digital engineering tasks for chemistry classes:

Separate plastic, glass and metals from garbage.

Find a way to separate the ink from old newspapers to make it recyclable.

Elaborated competitive engineering activities with limited resources are called “egg races”. The concept was popularized by the BBC show "The Great Egg Race" (broadcasted from 1979 to 1986). Participating teams were challenged by tasks like this: Construct a machine to transport a fresh egg the greatest possible distance using only the energy which can be stored in a small rubber band.

However, laboratory equipment is limited, chemicals may be dangerous and hands-on experimenting in a chemistry lab (including cleaning the workplace) takes much time. Thus in a high school chemistry lab, the possibilities of being creative and making something unique and unconventional are rather limited. The design and implementation of *digital* artifacts is much quicker, safe and cheap. Scratch, the visual programming language, has been especially designed to keep the distance between an idea and its implementation as short as possible. The environment offers a small set of blocks representing program statements. The blocks are all visible and in this way invite the developer to use them. They can only be put together in a way that leads to a syntactically correct program. Scratch encourages tinkering: “They’ll snap together a few bricks, and the emerging structure will give them new ideas.” (Resnick et. al. 2009). Resnick (1998) summarizes reasons for digital design projects, including these: Design activities challenge students as active learners, encourage “pluralistic thinking” by suggesting that multiple solutions are possible, offer an opportunity to explicate internal mental models and encourage students to put themselves into the mind of others since they need to use their (digital) constructions.

Since digital design tasks are complex and cover a broad spectrum of activities and real-life aspects, they have the potential to be interesting for a bigger audience than “pure scientific”, discovery-oriented tasks.

CLASSIC SCIENTIFIC CLASSROOM ACTIVITIES IN PHOTOMETRY

In this section I discuss two classic experiments from the chemistry of colors. They imply discovering a scientific law and exploring properties of a material.

The first activity is the discovery of the Beer-Lambert law. The law relates the attenuation of light to the properties of the material through which the light is traveling. In a photometer monochromatic light passes through a glass cell containing a solution of a colored substance. It then reaches a light sensor that measures the intensity of the light (I). Part of the light is absorbed by the solution and the glass.

The absorbance (A) of is defined as

$$A = log_{10}\left(\frac{I_0}{I}\right)$$

I_0 is the intensity of light after passing through an identical cell with the same solvent but without the colored substance. Each colored substance absorbs different amounts of light of different wavelengths. The capacity of a substance to absorb light of a given wavelength is an important property. According to the Beer-Lambert law, the concentration c is linear to the absorbance.

$$A = \varepsilon \cdot l \cdot c$$

If the attenuation coefficient ε and the light path length l are known, the concentration c of the substance can be calculated.

So how to “discover” the Beer-Lambert law? A student, say Anna, creates solutions with different concentrations of a dye – for example Methylene Blue. She selects the wavelength that is absorbed most by the substance (the complement). First she measures I_0, the intensity of light passing through pure solvant (water) without the colored substance. Then she measures the prepared solutions and notes the values of concentration and absorbance in a table using spreadsheet calculation. Finally she creates a diagram, which suggests a linear relation. The procedure is rather straight forward. The challenge – for a beginner like Anna – is to follow the instructions and to understand, what she is doing. There is not much room for creativity and imagination. The „discovery“ in this case is just to see that a cloud of dots in a diagram represents linearity. Scientific research turns out to be much routine work.

The second activity is to take absorption sprectra of dye solutions with a spectrophotometer. The absorption spectrum is the fraction of light absorbed by the material over a range of wavelengths (see figure 1). You can identify an unknown sample by comparing a measured spectrum with a library of reference spectra. For example, in a lab activity about dyes, the students synthesize the blue jeans dye Indigo, take the absorption spectrum and compare it with spectra of different blue dyes like Methylene Blue or Patent Blue V (This is part of a programme for high school chemistry classes at the Afried-Krupp-Schülerlabor at the University of Bochum, Germany, http://www.aks.rub.de)

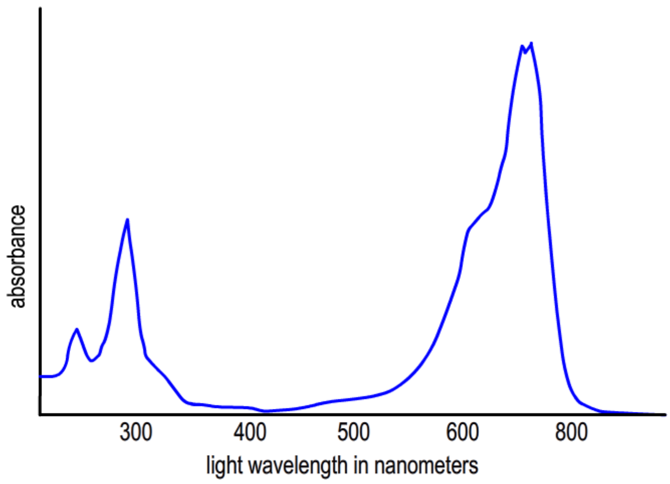


Figure 1: Absorption spectrum of methylene blue (redrawn from wikipedia)

Carefully supervised by the instructor or tutor, Anna puts solutions into cuvettes, puts them into the spectrophotometer and creates spectra by pressing the correct buttons. All this takes something like five minutes. And the spectrophotometer remains a black box.

SCRATCH PROJECTS IN PHOTOMETRY

A professional photometer is designed for efficient measuring and it hides the internal structure. Elaborating and understanding photometry can be stimulated by opening the black box. Within minutes students can create their own simple spectrophotometer using a computer, a PicoBoard, Scratch, regular chemistry lab equipment (like test tubes, cylinders, stand, clamp etc.), a color changing LED spot, and some additional parts made of paper, aluminum foil and adhesive tape. Color changing bulbs contain three LEDs that emit monochromatic red, blue and green light.

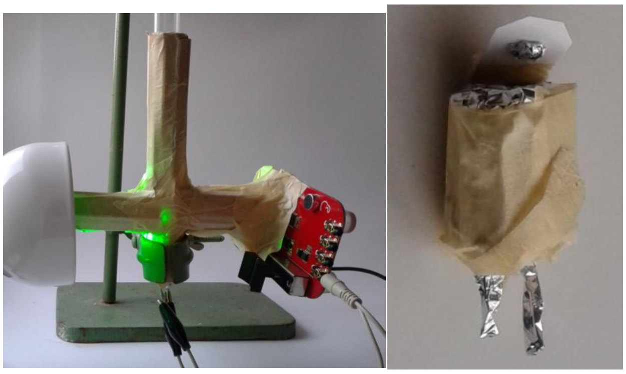


Figure 2: Homemade photometer with an LED light-changing lamp and a switch) that is closed by an inserted test tube.

Figure 2 depicts a self-made photometer. Instead of cuvettes, regular test tubes are used. They are put into a vertical pipe made of cardboard. Monochromatic light from the LED spot travels through a horizontal pipe through the test tube and the solution to the light sensor of the PicoBoard. For more advanced projects one can add a switch (made of paper and aluminum foil) to the bottom of the vertical tube. The switch is connected to a resistance sensor via alligator clips. In this way a Scratch project could check whether a test tube has been put into the photometer or not.

Project 1: Taking A Discrete Absorbance Spectrum

The homemade photometer can be used to take simple discrete absorbance spectra, which show the absorbance of a dye at three different wavelengths. This is – of cause – not precise enough to determine the absorbance maximum of a colored solution but it can be used to distinguish between different dyes. The procedure has two steps: calibration and measuring.

For calibration the user inserts a test tube with water and makes three blank tests with the colours blue, green and red (increasing wavelengths). The blank test is made by hitting the keys b, g and r. The three squares representing the absorbance jump to the base line.

Then a test tube with a dye solution is inserted. The user measures the absorbencies of blue, green and red light by clicking on the squares, starting with the square most left (blue). The square jumps up to a certain vertical position representing the absorbance. In that way a simple absorbance spectrum with three values is created.

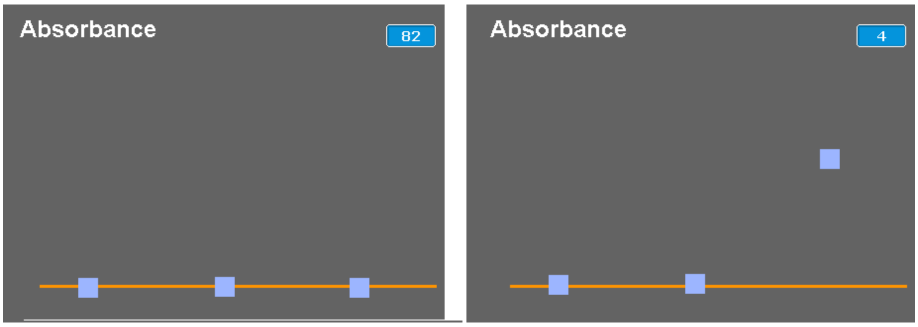


Figure 3: Simple discrete absorbance spectrum of blue food dye.

The three squares are represented by three sprites, named Blue, Green and Red like in figure 4. The scripts of the other sprites are almost identical.

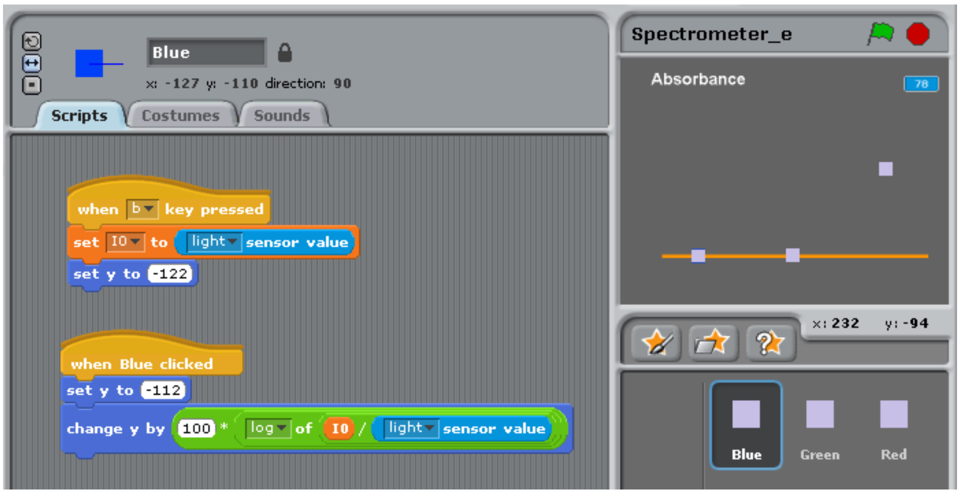


Figure 4: Scripts for sprite Blue indicating the absorbance of blue light.

Project 2: Measuring the Absorbance

An easy programming project is to create a photometer that calculates and displays the concentration of a dye solution. The concentration is relative to a reference concentration. The device is used in three steps:

- First the user inserts a test tube with water into the photometer, select an appropriate light source and clicks on the button *Blank test*. The button is implemented by a sprite with one single script. It stores the value of the light sensor in the Variable Io.
- The user inserts a test tube with a reference solution into the photometer and clicks on *Reference*. The script of the corresponding sprite calculates the absorbance coefficient (based on the Beer Lambert law) and stores it in a variable. Usually the concentration of the reference solutution is 100% (1). But the user can change this concentration with a slider.
- Now the photometer is calibrated and can be used for measuring samples with unknown concentrations.

Figure 5 shows a simple implementation. The Scratch project contains three sprites (each with a *when … clicked*-script) representing the buttons. This algorithm assumes that the light sensor values are linear to the light intensity. Unfortunately this is not the case.

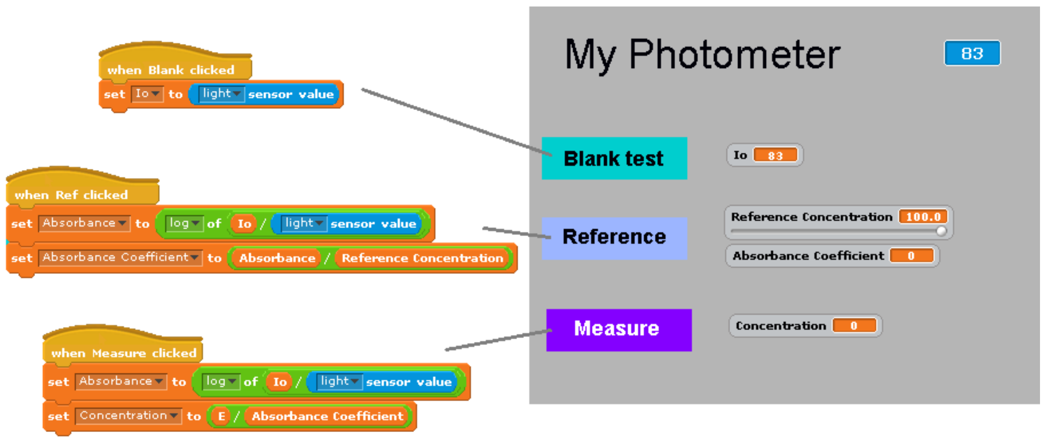


Figure 5: A Scratch Photometer with three buttons (sprites).

Project 3: Measuring Concentrations Using a Calibration Curve

A standard experiment in chemistry is to measure the concentration of a dye solution in a naïve way by using a calibration curve. The user – say Anna - creates a aqueous solution of a dye – say Brillant Blue – with a certain concentration – for example 0.1 grams per litre. This is called 100%. By diluting this solution, Anna creates further solutions of 20%, 40%, 60% and 80%. She inserts a test tube with water into the photometer and measures the intensity of the light that travels through the water. Then she measures the light intensities with the dye solutions and plots the values (concentration, light intensity) in a diagram. The result is called calibration curve. This diagram can be used for detecting the unknown concentration of a sample. This procedure is called naïve, because the Beer-Lambert law is not used. Using the calibration curve does not require any mathematical calculation. Anna just draws a horizontal line through the value of the light sensor on the vertical axis, detects the intersection with the calibration curve and finally finds the corresponding concentration of the horizontal axis.

Figure 6 shows screenshots from a Scratch project that supports this procedure. The user interface contains three buttons:

- Clear: Wipe out the calibration curve.
- Calibration: Measure the light transmission of a solution with a known concentration.
- Sample: Measure the dye concentration in a sample using the calibration curve.

Anna first creates a calibration curve by inserting test tubes with dye solutions into the photometer and clicking the button Calibration - starting with 0% and ending with 100%.

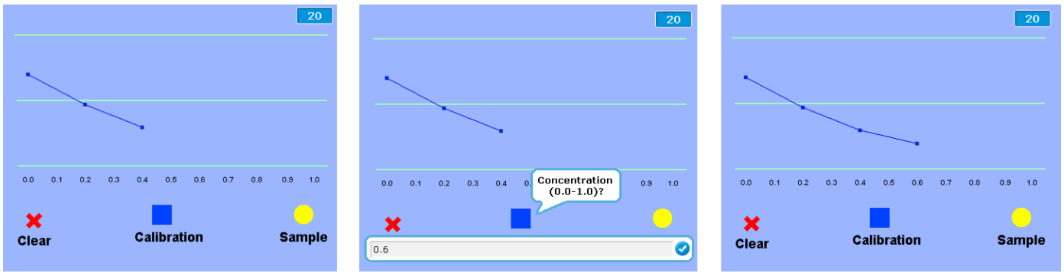


Figure 6: Creating a calibration curve.

When the calibration curve is finished, Anna inserts a test tube with the unknown solution and clicks on *Sample*. The computer measures the light intensity and finds the corresponding concentration in the same way as Anna would do it with pencil and paper. On the screen a small square moves from the left to right like a flying airplane. Its height (y-position) represents the value of the light sensor. When it hits the calibration curve – this can be sensed by a sprite using touching color ? –, the red square changes to a vertical line indicating the concentration of the sample.

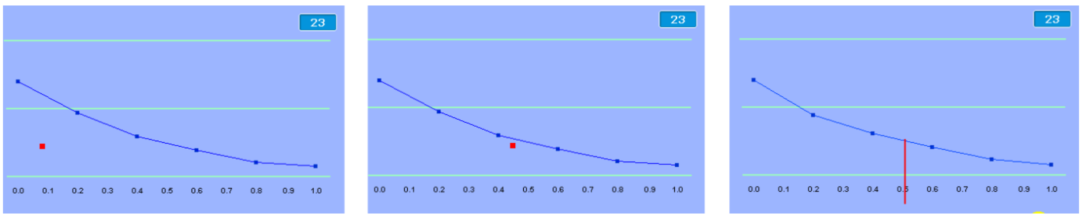


Figure 7: Measuring the concentration of a dye solution in the style of a shooter computer game.

This is just one way to find the concentration adopting techniques from video game design. It is completely free from math and it is a nice example of modeling a real life procedure by a computer program.

Advanced Engineering Projects Related to Photometry

Scratch supports the design of multimedia user interfaces. Animations, sound and interactivity are easy to program. This makes it possible to cover a huge variety of projects that are not related to scientific research but to practical everyday purposes. These projects illustrate the impact of science and informatics to social life and its potential to make the world better. Let me shortly discuss two examples: 1) Design a photometer that can be used by people. In this case the major engineering decisions would concern input and output. External switches that can be found easily or voice control might be useful. Output would be acoustic. For example, a concentration or light intensity could be represented by a sound of a certain pitch.

2) Design a “smart” photometer that can be used be everybody without specific knowledge. A possible design decision is to add a “mentor” that explains what to do.

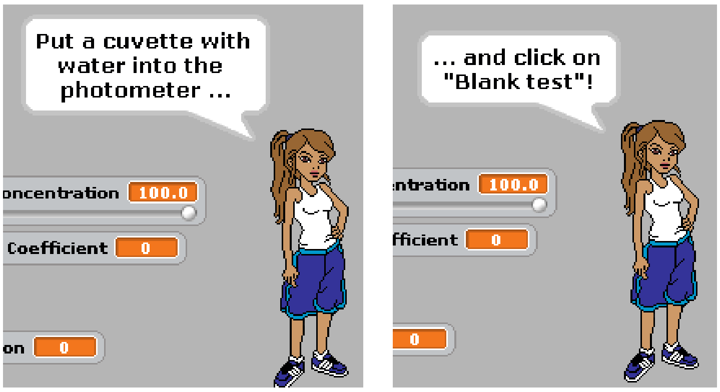


Figure 8: Just-in-time eplanations of a “smart” photometer.

During the project, the students not only learn how to program but also develop science-related competences like explaining the measurement process (using correct technical terms like “blank test”, “cuvette”, “absorbance”) and using the correct mathematical equations for calculating absorbance and concentration. The Scratch project could replace a regular report, students usually have to write when they perform chemical experiments.

Relevance of Science Projects

Are digital projects relevant for high school students? In winter 2014/2015 German high 37 school students (15 female, 22 male, age between 17 and 20 years) from chemistry classes at two different schools were asked to answer a questionnaire about their motivation to perform projects in photometry. They got a list of concise project descriptions and had to pick at most five projects they would like to perform. Table 1 displays the results. The most attractive projects were those with a clear practical purpose like a machine that can recognize a dye and distinguish it from other dyes or a spectrophotometer. Projects related to scientific research methods were less attractive.

Project	Chosen by
1. Develop a device that automatically recognizes a dye. It can distinguish Brilliant Blue from other blue dyes.	73%
2. Develop a machine that automatically controls the quality of a product (e. g. lemonade) by measuring the light absorption.	54%
3. Develop a talking spectrophotometer which tells the user what to do. Anybody should be able to operate it without special training.	51%
4. Develop a device that represents the colors of solutions by sounds. It should be usable for blind people.	49%
5. Design a machine that automatically determines the concentration of dye solutions and displays the result. The machine should be easy to use.	46%
6. Build a spectrophotometer following step-by-step instructions and try it out.	43%
7. There are chemical reactions in which the color gradually changes. Examine such reaction using a photometer.	32%
8. Perform a series of experiments in which you measure the absorption of dye solutions at different wavelengths (absorption spectrum).	24%
9. Plan a series of experiments, in which the absorption of dye solutions is measured. You perform the experiments and evaluate them.	16%

Table 1: Attractiveness of project ideas to high school students (n=37, age 17-20)

Additionally the participants were asked to rate the attractiveness of activities that are typically part of a science project like those which were presented before. The results reveal a quite remarkable inconsistency: Constructing an elaborated device (that for example represents color by sound), implies engineering activities like improving, error searching and especially programming. In fact, the process of designing a digital device is mostly programming. However, many students stated that they found programming to be rather uninteresting.

Activity	Avg. attractiveness
Work with laboratory equipment: Mix liquids, heat a substance, etc.	3.35
Observe a chemical reaction.	2.95
Assemble a device from individual parts (cardboard, aluminum foil, light sensor, etc.).	2.73
Improve and develop a device.	2.68
Develop a computer program.	2.56
Create a research question that can be used for designing experiments.	2.16
Search for an error, if something does not work	2.11
Discuss observations from experiments.	1.95

Table 2: Attractiveness of project activities to high school students, 1 = “very uninteresting”, 4 = “very interesting” (n=37, age 17-20)

Scaffolds for Diving into Computing

The projects discussed in this paper are for students who do not necessarily intend to learn programming but who want to create an artifact that can be used in a chemistry lab. They dive into computing because they need this knowledge for implementing design idea.

Which kinds of scaffolds are useful to this audience? In this section I discuss starter projects, step-by-step-instructions and domain-specific hints.

(1) Starter projects are predesigned applications that are simple and quick to make. Students begin the development by making (or copying) a starter project. Since students may be not primarily interested in programming as such, the starter project has to be within the same domain as the intended artifact. A starter project in photometry is a minimal spectrophotometer like the one described before. In fact this is a rather demanding project which requires specific scientific prior knowledge. The developer should be familiar with the idea of calibration, light absorbance and absorption diagrams.

According to the expectance-value-theory (Widfield & Eccles, 2000, 2002) the motivation for choosing a task depends not only on the attractiveness of the project goal and the activities during the development (incentive and attainment values, utility values, cost) but also on the expectation of success. The latter implies that it is not sufficient that the starter project *is* (objectively) easy. It also must look easy and Anna must be convinced than she can do it – or do it even better than expected.

Starter projects may have obvious weaknesses that students are able to recognize and to improve with the programming skills they already have acquired. The starter project of the spectrophotometer is difficult to use and it displays absorbance values in a suboptimal way. But it is easy to make improvements: Add hints what to do next. Add colors and wavelengths to the diagram so that the user can see which wavelength (color) is absorbed to what extent. Make the diagram scalable using a variable, which can be changed with a slider.

(2) Written step-by-step instructions or videos explain how to make a starter project. The advantage to just importing the project from the website and then changing it is that the students learn how to use Scratch by constructing a concrete artifact.

A feature of the step-by-step principle is that all steps have a similar level of difficulty. This implies that the very first step is an indicator of the difficulty of the whole process. If Anna is able to do the first step she might take this as evidence that she can do the whole thing. If there is a difficult step in the procedure, it should be told to the reader in order to maintain optimistic self-schemata.

Instructions do not kill a playful attitude. In the contrary, to understand a verbal instruction or explanation completely it is often necessary to experiment, play and tinker. Instruction may include suggestions to try out things. In (Weigend, 2014) it was observed that children who followed step-by-step instructions were more satisfied with their Scratch project than those who ignored instructions.

(3) Sometimes students come to the teacher and ask “I have this idea. How can I do it?” Then it is nice if the teacher could name a section in a reference book or give a card with specific explanations (like “Scratch cards” on the Scratch website). The Scratch context help system offers explanations for each block. This gives an idea how to use this block and is very helpful for those who are exploring Scratch. But someone who is “diving into Scratch” and who is not willing to learn more programming than necessary wants information that is directly related to his or her design idea. Examples of design ideas from the photometer context are:

- When the user clicks on a certain button the figure on the screen tells the light intensity. Solution: Use message passing. When the button-sprite is clicked, it broadcasts a message. This is received by the sprite representing the figure.
- The value light intensity (I) should be displayed rounded to two decimal places. Solution: Use this term $\frac{round(100 \cdot I)}{100}$.

According to expectancy-value-theory the motivation to elaborate and use a hint results from the value of the design idea and the simplicity and understandability of the hint.

CONCLUSION

Programming seems to be not a very attractive activity for many high schools students. Science-related projects implemented with Scratch give the opportunity to gain positive experience and to overcome a negative attitude. Scratch makes it easy to dive into computing. That means creating highly valued products and acquiring necessary skills en route. The first step to Computer science is to get fascinated by the possibilities of digital technology. Toy projects that are quickly to make with Scratch stimulate imagination. They might not just motivate to go deeper into computing but they seem to make science more interesting as well.

The projects mentioned in this paper are part of training program for chemistry teachers in the German federal state North Rhine Westphalia. If one wants to make programming more popular at schools it might be wise to address not only students but non-informatics teachers as well.

REFERENCES

BBC: The Great Eggrace. http://www.bbc.co.uk/archive/great_egg_race/

Kolb, A. Y. and Kolb, D. A. (2005). Learning styles and learning spaces: Enhancing experiential learning in higher education. Academy of management learning & education 4.2, 193-212.

Kolb, David A. (1981). "Learning styles and disciplinary differences." The modern American college 232-255.

Felder, Richard M., and Linda K. Silverman. "Learning and teaching styles in engineering education." Engineering education 78.7 (1988): 674-681.

Wiley, H. S. (2010). Scientists vs. Engineers. The Scientist, July 1st 2010, online available http://www.the-scientist.com/?articles.view/articleNo/29115/title/Scientists-vs--Engineers/

Henry Petroski: Engineering Is Not Science. And confusing the two keeps us from solving the problems of the world. IEEE Spectrum 2010, 23. November 2010. http://spectrum.ieee.org/at-work/tech-careers/engineering-is-not-science#

Resnick, M. (1998). Technologies for lifelong kindergarten. Educational technology research and development, 46(4), 43-55.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., ... & Kafai, Y. (2009). Scratch: programming for all. Communications of the ACM, 52(11), 60-67.

Weigend, M. (2014). The Digital Woodlouse–Scaffolding in Science-Related Scratch Projects. Informatics in Education-An International Journal, (Vol13_2), 293-305.

Wigfield, A., & Eccles, J. S. (2000). Expectancy–value theory of achievement motivation. Contemporary educational psychology, 25(1), 68-81.

Wigfield, A., & Eccles, J. S. (2002). The development of competence beliefs, expectancies for success, and achievement values from childhood through adolescence. Development of achievement motivation, 91-120.

Michael Weigend studied Computer Science, Chemistry and Education at the University of Bochum and the University of Hagen and received a PhD in Computer Science from the University of Potsdam. He is a teacher at a secondary school in Witten, Germany and he has taught Didactics of CS at the University of Hagen for almost 20 years. He has published several books on computer programming, web development and visual modelling.

Analyses of difficulty and complexity of items in international on-line competition “Beaver”

Ekaterina Yagunova, *katrin.home@mail.ru*
Saint-Petersburg State Electro technical Institution LETI after V.I.Ulianov (Lenin), Russia
St Petersburg Academic University, “Physical-Technical School” Lyceum, Russia

Sergey Pozdnyakov, *pozdnkovl@gmail.com*
Saint-Petersburg State Electro technical Institution LETI after V.I.Ulianov (Lenin), Russia

Nina Ryzhova, *ryzhova.nina@gmail.com*
State Corporation ‘Institution of Training – ARB Pro’, Russia

Evgenia Razumovskaia, *evgeniar@yahoo.com*
The University of Edinburgh, Old College, United Kingdom E

Nikolay Korovkin, *nikoay.korovkinl@gmail.com*
St. Petersburg State Polytechnic University, Russia

Abstract

Several approaches to evaluation of difficulty and complexity of tasks were considered using the analysis of protocols of the International online informatics competition “Beaver-2012”. It was shown that a priori evaluation of complexity by the organizers does not correspond to the difficulty of the task for the participants. The pupils, especially primary school kids, underestimate the complexity of the tasks frequently. A method of division of tasks into subgroups based on the analysis of difficulty and complexity was offered. Subsequently, it can be used in other disciplines.

A cluster of tasks the difficulty of which was significantly underestimated was identified. In addition to that, it was demonstrated that the complexity of tasks can be related to the interface of the competition in some cases. The shortcomings of the organization of online protocols were elicited. The elimination of the defects will allow the results of the competition to be more valid.

Keywords

complexity of tasks, difficulty of tasks, on-line competitions, educational tests, typology of tasks, competition "Beaver".

INTRODUCTION

Competition as a measuring procedure

Any competition in a school subject is in fact a ***test***, i.e. it represents a standard evaluation procedure. The competence and skills of school children in informatics have been assessed during “The Beaver” on-line competition. Mandatory test attributes are: standard set of tasks, standard representation of tasks, formal description of answers and processing procedure, adequate test key. A test key is an algorithm of projecting the protocol of answers to some point on a scoring scale. The main requirement for the key is its correspondence to the measured feature (Ball, 1990). Only upon availability of the attributes, it is possible to make an objective evaluation of an individual using a quantitative scale indicating the evidence of measured feature (Shmelev, 2013).

The simplest test key suggests the addition of all the scores of fulfilled tasks. The scores can be weighted or unweighted.

There is a huge number of ways to determine task weights (Socan, 2009, Chang, 2009). There are two different approaches to assigning an item score: a priori

determination of task weights by organizers (experts) or an a posteriori weighting while taking into account the results of test performance by participants. In case of an a posteriori "weighting" number of participants who gave the correct answer shall determine the weight of task. The greater value shall be attributed to tasks which have been performed by minority of participants. When a priori weighting is used, more complicated tasks are also supposed to have greater weights.
The simplest scoring scheme (a participant receives 1 point for any right answer and null for a wrong answer) assumes that the tasks are presented to participants in the increasing order of complexity (Lord, 1952). Therefore, the level of complexity has to be evaluated a priori.
When weighting a priori, the consistency of competition results shall be defined by experts' proficiency. Shmelev (2013) underlines that "it is not worth being under an illusion that experts can truly assess the complexity of tasks". According to him, the best complexity measure is the statistics of real answers given by real participants.

Methods for estimating task complexity and difficulty

A task which is easy for one participant may be difficult for another one. The task difficulty reflects the relationships between a task and an individual who performs it. To underline this feature many authors separate the notions of "complexity" and "difficulty" (Ball, 1990; Golikov & Kostin, 1996; Navon & Gopher, 1979; Sammer, 1997). Complexity means a certain objective feature of a task while the difficulty is understood as a subjective feature, i.e. how a participant interprets a task. While speaking about the difficulty the authors focus on the individual's activity to perform a task – to analise and to process the information, to design and to make decisions, to forecast consequences of their own decisions and to build operation images and frameworks (Leontiev, 1975; Ponomarenko, 1982).

The task complexity may be measured upon competition results by counting a share of participants who got right answers. A measurement or at least evaluation of task difficulty requires serious efforts.

The difficulty of a task for a subject is formed of their mental workload (cognitive, informational, emotional, attentional loads) and expenditures for their own state control (Sammer, 1997).

The most accurate methods of workload estimation suggest measuring of various human factors. A diagnostic procedure may be accomplished only during "live" competitions with limited number of participants. The procedure itself may be an additional stress for participants. This may lead to the increase of effort made to control their state (Kantovits & Sorokin, 1991). Meanwhile, it is only possible to assess the state of participants of remote competitions using self-reflection tests during task performance. The results of such questionnaires shall be adequate only for senior school students because it will be difficult for primary school children to make an objective evaluation of their state and abilities (Towler & Broadfoot, 2006). According to Piaget (Piaget, 1951), a primary school child is at the stage of concrete operational intellectual development. Typically for this age, thinking restrictions affect not only the cognition of outside world but also the manner of children to perceive themselves. It is fair to start talking about conceptual thinking only by the age of 11-12 years.

The level of the development of thought process is associated with the age as well as the appropriateness of self-assessment. That is why primary school children may not cope with solution of tasks that require the operation with abstract notions (and this is natural!) (Piaget, 1951). During the period from 8 to 10 years the capacity of memory is rapidly increasing, attention can be switched much better. So, even minor age difference in this period may cause significant differences in results when solving same tasks.

According to Narvon and Gopher, the results of “human system” activity exhibit the relationship between the quality of operation information (quality and quantity of stimuli, coding, distribution etc.) and the capacity of resources available (Navon & Gopher, 1979). In subject competitions the attention load, the processes of short-term and operative memory may be assessed with the difficulty of the text of problem statement. Among numerous ways of assessment of the difficulty of text, the most straight forward is the length of the statement, i. e. number of stimuli to be processed for solution (Benjamin, 2011).

The workload may be a function of the level of difficulty and the number of tasks to be performed within a unit of time (Riley et al., 1994). Using the protocols of on-line competitions one can evaluate the workload of participants taking into account the time spent by them to solve tasks (Gibson & Clark-Midura, 2013).

The rules of “The Beaver” competition presume the possibility by participants to complete only a part of tasks. In this case the participant’s refusal to solve the task is to be considered as their assessment of task difficulty upon binary rating scale (“difficult” – “not difficult”). Solved tasks are assessed by a participant as “not difficult”, while those that are not presented – as “difficult”. A share of participants who found the task as “difficult” shall determine the difficulty of a task for the whole body of competition participants.

A prior estimation of tasks by experts (weighting) shall be correct once both the task complexity and difficulty are taken into account for participants of each age group. Only analysis of the results of competition can establish whether the difficulty has been really considered at weighting, whether a prior estimate is in agreement with an objective complexity.

It should be noted that some authors use the notions of “complexity” and “difficulty” as synonyms because not the content of mental processes is fundamental for them, but the execution – whether a schoolchild can or cannot solve the task (Krotov, 1999). Further on we shall differ the terms “complexity” and “difficulty” by high lightening them in bold. If the matter is a complex evaluation with account made for both parameters, we shall use the term “complexity” without high lightening it.

Research objectives

1. To evaluate the validity of measurement procedure to be performed during processing of the results of international competition on informatics “The Beaver”. To assess the quality of a set of competition tasks and scoring method.
2. To estimate the adequacy of expert estimate of task complexity. To compare various estimates of task complexity and difficulty.
3. To classify competition tasks upon their complexity and difficulty.
4. To evaluate age differences of task perceiving by schoolchildren.

MATERIAL AND METHODS

“The Beaver” competition: organization, tasks selection

“The Beaver” international on-line competition on informatics started in 2003 (Dagienė, 2006). The task pool is prepared by representatives of participating countries. The statements of the problem should meet certain requirements (Opmanis et al., 2006, Dagienė & Futschek, 2008). The statements are discussed and formulated in English. Each country chooses the problems for the competition out of the task pool. Some tasks are “compulsory”, i.e. they should be included by all participating countries. The organizing committee of each country is responsible for translations of the statements to the national language of each country.

Russia took part in this competition for the first time in 2012. The competition in Russia is organized for six age groups. The participants are proposed to solve the tasks of three levels of complexity: for schoolchildren of 1 and 2 grades – 4 tasks for each group (weighted of 3, 6 and scores respectively), for schoolchildren of 3-10 grades – 5 tasks for each group (weighted of 6, 9 and 12 scores). 8 simple and 7 complicated tasks (weighted of 9 and 12 scores respectively) are proposed to senior schoolchildren. The tasks of 2012 can be found http://ipo.spb.ru/bebras-files/beaver-2012-rus.zip (in Russian) or http://ipo.spb.ru/bebras-files/beaver-2012-eng.zip (in English). 40 minutes are given for task completion. Every wrong answer is fined, the penalty rate makes one-third of task value.

Competition tasks are numbered, simple tasks are of smaller order and complicated tasks have larger numbers. The tasks of the same complexity are randomly numbered (the order is different for each participant). Participants may solve the tasks arbitrarily, they may return to solved tasks.

Participants are to choose from multiple answers, three of them are wrong and one is correct. A participant can choose the option “no answer”. In this case they will receive neither score, nor penalty. Certain tasks prepared for students of 1-2 grades are dynamic – certain actions with the mouse are required to be made. These tasks will be scored in the same way as others. A sum of scores and penalties gained shall make the result of competition for each participant. Participants gained the highest results shall become winners in each age group.

Scales and analysis

The protocol of competition where all participants’ actions are recorded shall be taken as a basic data set (Yagunova & Ryzhova, 2013). The time when a participant started his/her work and the time spent for selection of every answer is known. If a participant introduced successively several answers the last one is to be taken into consideration when counting the results. The total number of competition participants in 2012 was 6602 schoolchildren. The distribution of the participants by age is given in table 1.

level of contest	grade	number of pupils	
1	1	297	733
	2	436	
2	3	661	1483
	4	822	
3	5	793	1603
	6	810	
4	7	672	1329
	8	657	
5	9	615	1148
	10	533	
6	11	306	6602

Table 1: number of students in each age group, who took part in the competition in 2012

We used several scales to evaluate the complexity and difficulty of competition tasks.

Scale 1 – scale of expert estimation. It is performed upon three-point scale (1 – simple, 3 - complicated) during the meeting of international steering committee. In order to consider the estimate as correct both the objective complexity and difficulty of tasks should be taken into account.

Scale 2 – share of participants who chose the “no answer” option for particular task. When the answer is arbitrarily chosen out of 4 proposed options, the probability (p) to choose a correct answer makes ¼. In this case a mathematical expectation of scores gained is $p*x+ (1-p)*(-x/3) =x (4p-1)/3=0$ (where x – task value), that coincides with total scores received when choosing “no answer” option. If one of the answers proposed is rejected as certainly wrong, the probability of arbitrary choice of a correct answer out of remained options is greater than ¼ and a mathematical expectation of score gained for solving a task becomes positive. Therefore, a participant has no reasonable motive to choose “no answer” button. The use of this button may only be due to psychological reasons – for example, a fear of failure (and penalty for it) or an extreme lack of self-confidence, a fear of task statement. As statistical expectation of score may be hardly calculated by schoolchildren, the choice of “no answer” option reflects their feelings about the relation between winning and failing probabilities. In all cases the choice of “no answer” option is the result of an interaction between the schoolchild and the task, i.e. it features the difficulty of a task for a participant.

Scale 3 – share of participants who gave a correct answer among those who decided to solve the problem. It shall be determined upon completion of a competition and shall contribute to proper evaluation of the task complexity. We must underline that this is just a task complexity because it is calculated only taking into consideration those who decided to solve it, i.e. among those who understood it as “not difficult” upon scale 2.

Scale 4 – number of symbols in problem statement. This is an indirect assessment of task difficulty because it relates to memory and attention loads.

Because scales 1-3 are ordinal, the comparison of scale 1 with scales 2 and 3 has been made using Spearman's correlation coefficient.

Clustering of tasks has been made by Ward’s method while using Euclidean metric.

In order to assess the adequacy of scoring method chosen for this competition and to evaluate the quality of a set of competition tasks by using Lilliefors test, the control of competition results normality has been done. The following hypothetical procedures have been implemented as an alternative to used scoring method (Weighted tasks, scores established a priori with penalties for the wrong answer):

- Weighted tasks, scores established a priori without penalty scores for the wrong answer;
- Unweighted tasks with penalty scores for the wrong answer;
- Unweighted tasks without penalty scores for the wrong answer.

RESULTS

Competition results

The normal-theory test of results of schoolchildren (composite score you got) is rejected at a level of significance $p<0.05$ for all school grades except for 6th, 7th and 8th. The values of Kolmogorov-Smirnov test are given on Figure 1 (row 1). For junior (1-6) and higher (9-11) grades the distribution of results has a positive skew, i.e. most of the school children have low results.

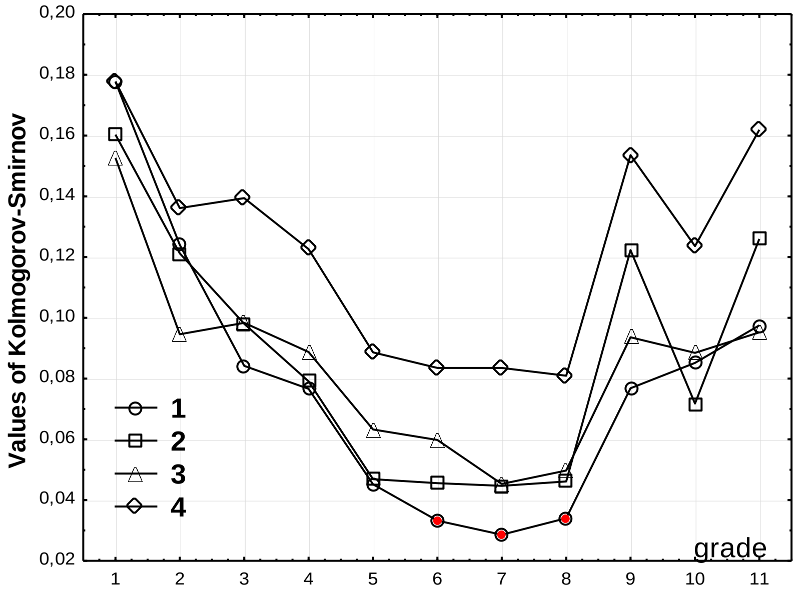


Figure 1: Normality test of competition results when real and hypothetical scoring methods are used. Values of Kolmogorov-Smirnov test are given for: 1 – scores weighted a priori with penalties; 2 – scores weighted a priori without penalties; 3 – unweighted scores with penalties; 4 – unweighted scores without penalties for the wrong answer. Filled circles are the points at which the deviation from normal distribution can be counted as insignificant at $p<0.05$.

An analysis of hypothetical versions of competition results estimation has shown that for all grades from the 3rd the scoring method used in the contest is the best out of those with a priori estimation of tasks (Fig. 1). Abolishment of penalties, when tasks have different weights, and equalized tasks, when penalties are available, make a real distribution of results insignificantly worse. The most unsuccessful is the version of tasks having the same weights without penalties. For all tested hypothetical scoring methods, even the distribution of results of 6-8 grades becomes far from normal. As far as it concerns schoolchildren of junior grades (1-2) the situation is quite different: a real scoring method proved to be the least successful while penalty exception or equalizing of task weights improves the distribution. It should be noted once again that only in three of all considered cases deviations from normal distribution may be found incidental.

Task complexity and difficulty

Task assignments upon difficulty (scale 2) and complexity (scale 3) are far from normal (Figures 2). The skew of task complexity distribution is positive while that of task difficulty distribution is negative, i.e. tasks of low difficulty prevail in competition but the most part of tasks are of high complexity. There are test versions for all grades where difficulty makes less than 10 %. The task of the lowest difficulty was found in a test for the 8th grade (1, 1%) and of the highest difficulty - in a test for the 10th grade (47, 6%). The task complexity within test versions for each grade varies from 10-20 to 80-90 %. The lowest complexity task was found in the 2nd grade version (7, 7%) and the one of the highest complexity – in the 7th grade task set (89, 8%).

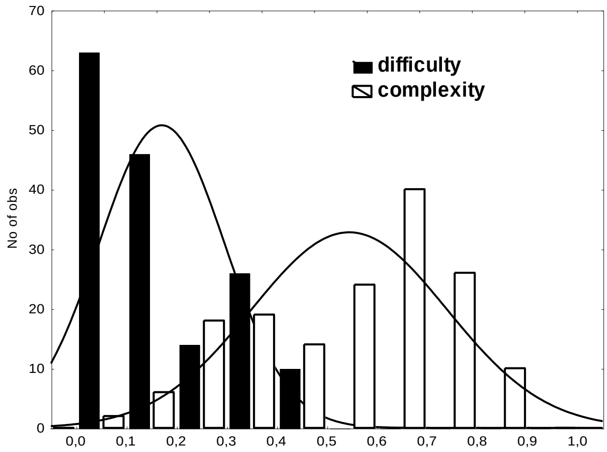


Figure 2: Distribution of competition tasks upon their **difficulty** and **complexity**

Adequacy of expert evaluation of task complexity

Spearman's rank correlation coefficients of expert tasks evaluation (scale 1) with their complexity (scale 3) and difficulty (scale 2) for participants calculated for all competition tasks are significantly positive ($p<0.01$) and equal to 0.56 and 0.60 respectively(table 1). There was no correlation of expert evaluation of complexity with task difficulty revealed for junior grades (1, 2, 3). There is a significantly positive correlation for grades staring from the 4th. The best concurrence of expert evaluation with task complexity was found for 1-2 grades. As far as it concerns the versions for 7-8 and 11 grades the evaluation of task complexity by organizers did not correspond to real complexity of tasks for school students.

Grade	1	2	3	4	5	6	7	8	9	10	11
Tasks **difficulty**	0,41	0,46	0,51	**0,67**	**0,53**	**0,76**	**0,76**	**0,76**	**0,63**	**0,79**	**0,84**
Tasks **complexity**	**0,74**	**0,8**	**0,62**	**0,64**	**0,59**	**0,57**	0,28	0,25	**0,62**	**0,66**	0,49

Table 2: Spearman's rank correlation coefficients of expert evaluation of tasks complexity for each grade with their **complexity** and **difficulty.** Correlation coefficients significantly at $p<0.05$ are highlighted in bold.

Tasks classification

Four clusters have been marked (Fig. 3) using comparison of the values of task complexity and difficulty with the use of cluster analysis. Two clusters contain highly complicated tasks while remaining two clusters are of low and medium complexity. There was a cluster of tasks of high difficulty and the one of low difficulty tasks separated among highly complicated tasks. All tasks of low and medium-case complexity are of low difficulty.

Link between task statement length and its complexity and difficulty

Table 2 shows correlation coefficients between the number of characters in task statement (scale 4) and its complexity and difficulty (scales 2 and 3). In junior grades (from the 1st to the 4th) a significantly positive link ($p<0.01$) has been found between task statement length and its difficulty (number of "no answer" responses). Moreover, the link between the task statement length and its objective complexity was revealed only for tasks of the 3rd - 4th grades.

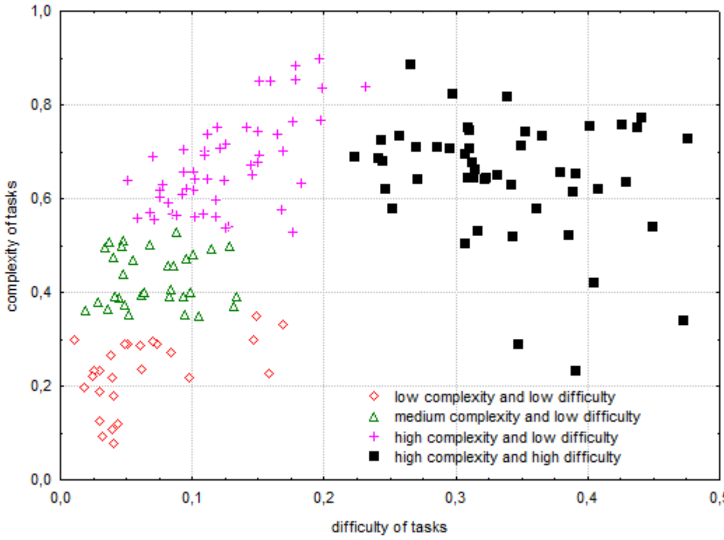


Figure 3: Task clusters marked on the basis of comparison of their **difficulty** and **complexity**

Age differences in task perception and competition results

Schoolchildren of two grades took part in each of the first five competition levels. The results of comparison of junior and senior schoolchildren in each pair of grades are shown in Figure 4. At each level (i.e. among schoolchildren who were solving the same tasks) the results of younger participants were lower (Fig. 4A). Junior schoolchildren of levels 1 & 2 assessed task difficulty as higher (i.e. they chose “no answer” response, Fig. 4B) while those who were in the 4^{th} and 5^{th} age group assessed it as lower. Tasks were complicated (number of wrong answers, Fig. 4C) for junior children at all competition levels except the first one. We have to note that it is not correct to compare the results of children of different levels because the number and sets of tasks differed.

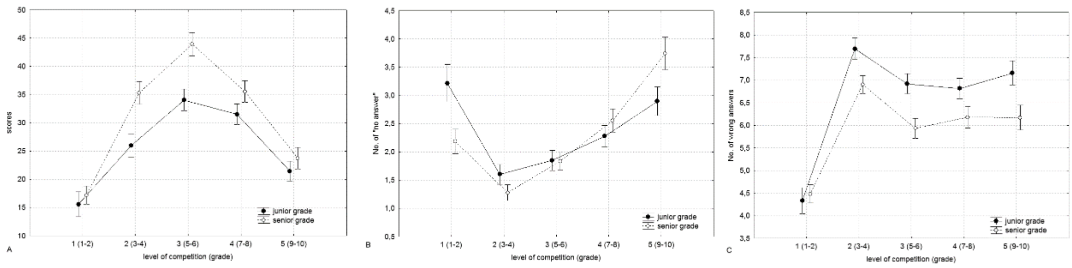


Figure 4: Differences of results of junior and senior schoolchildren at each competition level. A. Differences of average score B. Differences in assessment of task **difficulty**. C. Differences in task **complexity**. 95%-confidence interval of averages is given.

Table 3 indicates the task clustering by grades. When comparing junior and senior grades within competition levels we note that all tasks for senior schoolchildren with a single exception are of the same or lower complexity and difficulty. Only ninth task of fifth competition level is of low difficulty for juniors (9^{th} grade) and of high difficulty for seniors (10^{th} grade). The task complexity for all participants is the same, and it is high.

Grade	1	2	3	4	5	6	7	8	9	10	11
Task difficulty	**0,90**	**0,83**	**0,77**	**0,78**	0,17	0,13	-0,04	-0,06	0,10	0,04	-0,14
Task complexity	0,37	0,36	**0,77**	**0,78**	0,29	0,32	-0,18	-0,20	-0,13	-0,01	-0,23

Table 3: Spierman's rank correlation coefficients of task statement length and its **difficulty** and **complexity.** Correlation coefficients significant at p<0.05 are bolded.

DISCUSSION

Validity of the measuring procedure

When considering both the distribution of competition results and the difficulty and complexity of tasks, it is possible to conclude that tasks complex for school children prevail in the competition. The most appropriate tasks for participants were those for the 7th – 8^{th} grades. In that set, the tasks of high complexity and difficulty do not make more than a half of all the tasks. As a consequence, only the distribution of the competition results for schoolchildren of the 7th – 8^{th} grades is close to normal while in the test versions designed for the remaining groups the task complexity is beyond the abilities of participants.

Among different options of scoring supposing an a priori weighting of tasks the option chosen by organizers (tasks of different values, penalty scores for wrong answer) is effective for a given set of tasks for all grades starting with 3rd. For juniors the introduction of penalty scores and task clustering upon complexity is an unnecessary sophistication which impairs the distribution of results which is bad enough even without that.

Due to a significant right skewness of distribution of total scores, the selection of winners and ranking of the strongest participants run at high point. Ranking of the main body of participants is rough. A set of tasks used for this competition could be more appropriate if its goal was to select the best students. Taking into account that the competition is aimed at the general public in order to heighten the interest in the subject and tailored for students of general education schools, the set of tasks must be admitted to be far from successful.

Expert estimation of tasks

As it was mentioned above, most of the tasks proved to be complicated for the majority of participants. There is a possibility to make some assumptions about the reasons for that. The assumptions were made based on the results of comparison of expert estimation of tasks and their complexity and difficulty. As far as it concerns the tasks proposed to junior pupils, the correlation between expert estimation and task complexity is high while the one related to the difficulty is insignificant. The elder the pupils were, the closer was the task assessment by organizers to its difficulty set up in the protocol of competition. However, the correlation between expert opinion and task complexity was not always established in tasks designed for senior students. That is to say, the experts do not evaluate accurately the difficulty of tasks for juniors and their complexity for seniors.

Our results confirm to some extent the opinion of Navon and Gopher (Navon & Gopher, 1979) that one of the components contributing to the task difficulty is its statement length. In junior classes the tasks became significantly more difficult with the increase in statement length. Perhaps, this factor was not taken into account by experts when estimating tasks for junior grades which resulted in making the task excessively complicated. We suppose that the task length was the factor which determined a large number of refusals in junior classes, even though the tasks were not complex, in fact. This must have skewed the results of measurement of knowledge and skills of junior pupils. Because of lower level of development of mental processes junior pupils misunderstand long texts. It was proved that complexity of text provokes loss of interest to its content (Fulmera & Tulis, 2013), therefore, tasks with too intricate statements should be avoided.

Another complicating factor is interface peculiarities of a competition. Tasks containing long texts may have not enough space to be presented on one screen. In this case to read it from the screen some skills related to computer-literacy will be needed (to know what “vertical scroller” means and how to use it), as well as fine motor skills shaped in a certain manner (to know to use a mouse).

For senior students the length of text is not an extra complicating factor. Moreover, during educative process a personal experience to assess the difficulty of a task by eye is being gained as well as schemata to differ “difficult” and “simple” tasks. That is why the tasks estimated by experts as difficult prove to be such for senior participants.

Having made mistake with assessment of task difficulty for junior schoolchildren, the experts evaluated best of all the complexity of tasks for them. As far as it concerns senior pupils the objective complexity of tasks did not coincide with an a priori opinion of organizers. It probably means that an objective complexity of a task is due to students’ knowledge to a large extent. The knowledge of senior students participating in the competition on informatics was overestimated by organizers. The impression is that the experts were oriented to select and estimate competition tasks for a standard student of secondary school.

When considering the causes of incorrect a priori evaluation of task complexity, we should bear in mind that the competition is international. It is well-known fact that same tasks of TIMMS and PISA can have different weights for children from different countries. This is due to varied linguistic and cultural backgrounds (Grisay et al., 2007, Arffman, 2012). Appropriate complexity evaluation by the experts is essential for adaptation of the tasks of international competition for the specialties of each country. It would be interesting to compare the results gained by us with the data from other coutries.

Tasks categorization

Tasks of low difficulty prevail in the competition. Pupils are more disposed to solve tasks than to choose “no answer” version. As shown above, the selection of “no answer” version gives no advantage in scores over a simple guessing. The tasks where a large part of pupils refused to find a solution are of higher interest – these are tasks of second cluster. Most of them were found in tests designed for junior and senior students. We suppose that this cluster contains nonstandard tasks that frighten pupils by their presentation. The probability of giving a right answer is instinctively assessed as extremely low (this is not consistent with the reality – this probability is not less than ¼). Three other clusters contain more intelligible and/or ordinary tasks. The simplest (tasks of low difficulty and complexity) are those called in competition slang“consolation tasks”. Their solution is possible for practically each participant. The existence of such tasks in competition gives a positive emotion even to those who solved few tasks. The number of such tasks proposed for competition was negligible.

Age differences with regard to task perception

The above results show that a mere one year gap makes significant differences with regard to the perception of the same tasks. Junior participants of elementary schools are prone to assess tasks as more difficult and they are ready to choose “no answer” version more often than senior pupils. Elder pupils of elementary schools are prone in the contrary to assess tasks as less difficult (Tab. 3). The complexity of tasks for younger pupils of elementary school is in fact higher which becomes apparent in larger number of wrong answers and it is not surprising that it leads to lower results.

		Low difficulty			High difficulty
Level	Grade	Low complexity	Average complexity	High complexity	High complexity
1	1	3-4-7	6	1	2-5-8-11-9-10-12
	2	3-4-7	6-1	9-10-12	2-5-8-11
2	3	2	4-3-5-10	1-6-8-9-11-12-13-14-15-7	
	4	2-3-5-10	4-7	1-6-8-9-11-12-13-14-15	
3	5	10	2-7-15-3-4-8	1-6-9-14-5	11-12-13
	6	10-3-4-8	2-7-15-5	1-6-9-14	11-12-13
4	7	10-2-12	1-3-6-4	5-7-8-9	11-13-14-15
	8	10-2-12-4	1-3-6	5-7-8-9	11-13-14-15
5	9	1	5-7	2-6-8-3-9	4-10-11-12-13-14-5
	10	1	5-7-3	2-6-8	4-10-11-12-13-14-15-9

Table 3: Task clustering for each grade. Numbers of tasks are given in cells. Arrows show changes of complexity or difficulty.

CONCLUSION

1. The tasks proposed for “The Beaver-2012” competition allow selecting the best students but do not allow grading the main body of participants. Different weights of tasks and penalty scores for wrong answers enhance the quality of measuring procedure for the 3rd – 11th grades but do not make it accurate enough.
2. When preparing tasks for competition the organizers did not take into account the age difference of junior pupils, especially their low capacity to understand long texts. In addition, the standard of knowledge of senior pupils has been overestimated.
3. The proposed procedure of task clustering allows identifying nonstandard tasks.
4. One year difference makes significant difference with regard to perception of task difficulty and complexity by pupils and, therefore, their results.

RECOMMENDATIONS TO COMPETITION ORGANIZERS

- In general, the competition needs to be simplified: to increase the number of “consolation” tasks for all grades. Nonstandard tasks for junior and middle grades are to be simplified and more carefully formulated. The tasks with long text statements should be bypassed in junior classes.
- To think over the utility of “no answer” version. Perhaps, it is worth changing the relation between added scores and penalties or the number of proposed answers, so as to add a more pragmatic sense to “no answer” version.
- To think over the possibility to determine a posteriori task weights while keeping the penalty scheme for wrong answers. Such an approach shall give a positive educative impact – the tasks with bigger weights shall not frighten at a first glance the participants who are not self-confident. At the same time, a posteriori determination of task values shall compensate experts’ mistakes at their a priori assignment.
- If school students of different grades solve the same set of tasks, the announcement of the results of competition and the selection of winners must be done separately for each grade.

- To record into protocols of competition not only the time of answer introduction but the time spent by a participant to solve the task. By doing so it will make alternative estimation of the tasks difficulty possible.

REFERENCES

Arffman, I. (2012). Translating International Achievement Tests: Translators' View. Finnish Institute for Educational Research. Reports 44. 92P.
Ball, G.A. (1990). Theory of training tasks: Psychological and educational aspect. Moscow, Russia: Pedagogy.
Benjamin, R. G. (2012). Reconstructing Readability: Recent Developments and Recommendations in the Analysis of Text Difficulty: Educational Psychology. Vol. 24, pp. 63 – 88.
Chang, S.-W. (2009). Choice of Weighting Scheme in Forming the Composite: Bulletin of Educational Psychology. Vol. 40, N 3, pp. 489–510
Dagienė, V. (2006). Information Technology Contests--Introduction to Computer Science in an Attractive Way: Informatics in Education-An International Journal. Vol. 5, N1, pp. 37-46.
Dagienė, V., Futschek G. (2008). Bebras international contest on informatics and computer literacy: Criteria for good tasks. s. In R.T. Mittermeir and M.M. Sysło (Eds.), Informatics Education – Supporting Computational Thinking, Vol. 5090 of Lecture Notes in Comput. Sci.. Springer, Berlin, 19–30.
Delandshere, G. & Petrovsky, A. R. (1998). Assessment of Complex Performances: Limitations of Key Measurement Assumptions: Educational Researcher. Vol. 23(2), pp. 14 – 24.
Fulmera, S. M. & Tulis, M. (2013). Changes in interest and affect during a difficult reading task: Relationships with perceived difficulty and reading fluency: Learning and Istruction. Vol. 5, pp. 11 – 20.
Gibson, D. & Clarke-Midura, J. (2013). Some Psychometric and Design Implications of Game-Based Learning Analytics. Perth, Australia: Curtin University.
Golikov, Yu. Ya. & Kostin, A. N. (1996). Psychology of facilities management automation. Moscow, Russia: RAS Institute of Psychology.
Grisay, A., de Jong, J., Gebhardt, E., Berezner, A., & Halleux-Monseur, B. (2007). Translation equivalence across PISA countries. Journal of Applied Measurement, 8 (3), 249–266.
Kantovits, B. & Sotokin, R. (1991). Human factor: Mir. Vol. 4, pp. 85 – 113.
Kiriukhin, V. M. (2012). Methodological recommendations on elaboration of tasks for school and municipal stages of All-Russian competition on informatics for students in 2012/2013 academic year. Moscow
Krotov, V. M. (1999). On complexity of problems on physics: Physics: teaching problems. Vol. 3, pp. 69 – 74.
Leontiev A. N. (1975). Activity. Comsciousness. Personality. Moscow, Russia: Politizdat.
Lord, F. M. (1952). Theory of Test Scores: Psychometric monograph. Vol. 7.
Navon, D. & Gopher, D. (1979). On the economy of human information processing systems: Psychology Review. Vol. 86, pp. 214 – 255.
Opmanis, M., Dagienė, V., Truu, A. (2006). Task Types at Beaver Contests Standards. In: Dagienė, V., Mittermeir, R. (eds.) Proc. of the 2nd Int. Conference Informatics in Secondary Schools: Evolution and Perspectives, Vilnius, pp. 509–519.
Piaget, J. (1951). Psychology of intelligence. London, Great Britain: Routledge and Kegan Paul.
Ponomarenko, V.A. Cherniakov, G. M. & Kostritsa, V.G. (2013). Operator's mental states as the object of engineering and psychological researches: Cybernetic issues.

Riley, V. Lyall, E. & Wiener, E. (1994). Analytic workload model for flight deck design and evaluation: Proceedings of the Human Factors and Ergonomic Society. Vol. 38, pp. 81 – 84.

Sammer, G. (1997). Concepts of mental workload in psychological research: Proceedings of the 13th Triennial Congress of the International Ergonomic Association. Vol. 5, pp. 368 – 370.

Shmelev, A. G. (2013). Practical testology. Moscow, Russia: Maska.

Sočan, G. (2009). Scoring of multiple choice items by means of internal linear weighting: Review of Psychology. Vol. 16, N2, pp. 77-85.

Towler, L. & Broadfoot, P. (1992). Self-Assessment in the primary school: Educational Review. Vol. 4(2), pp. 137 – 155.

Yagunova, E., Ryzhova, N. (2013). Use of protocols of online competitions for an assessment of complexity of tasks and increase of a validity of measuring procedure. Komp'ûternye instrumenty v obrazovanii. Vol. 6, pp.33-44.

SHORT PAPERS

Songs are the tapping of keys Coding events for Kids

Nigel Beacham, *n.beacham@abdn.ac.uk*
University of Aberdeen, UK

Janet Carter, *J.E.Carter@kent.ac.uk*
University of Kent, School of Computing, UK

Bruce Scharlau, *b.scharlau@abdn.ac.uk*
University of Aberdeen, UK

Abstract

In this pilot study we look at the UK based weeklong Festival of Code, which is organized by Young Rewired State, and has for the last five years, and explore what impact this has had on the 7-18 year old attendees and their subsequent development. This team based coding exercise for children has grown from 50 to 1000 participants and will have important lessons about how we address computational thinking and teaching computing science to the next generation, as well as what lessons can be learned from these informal events.

Keywords

Computer Science Education, Coding for children, programming for children, YRS, Young Rewired State.

INTRODUCTION

Each year Young Rewired State (YRS) organize the weeklong Festival of Code for children aged 7-18 at multiple sites across the UK, which culminates in a weekend event at one location for all participants. The weeklong event has been happening for five years now. We wanted to ask participants what they and their parents have experienced at the events to see if it offered insights into what will follow as more children start to code.

BACKGROUND

Coding for Children Assumptions

For a number of years groups in both the US and the UK have been arguing that children should learn to code, and to understand computational thinking (Strauss, 2014). This is seen in both the Computing at Schools Working Group in the UK (Peyton Jones, 2013) and the Common Core State Standards initiative in the US (Stephenson, 2013), and has led to the 'hour of code' approach (http://hourofcode.com) as a means to introduce children to computational thinking (Williams, 2007).

A number of groups such as Coder Dojo and Code Club sprang up to meet this need. Coder Dojo is for primary and secondary aged children interested in coding and technical skills founded in 2011 (https://coderdojo.com/). Code Club started in 2012 for primary school aged children and follows a specific lesson plans at each session (Smith, Sutcliffe, Sandvik, 2014).

Before either of these groups started, YRS was quietly evolving and having an impact on children and parents. It started as a weekend hack event in London in 2009 (Cellan-Jones, 2009, Mulqueeny, 2014b), and has always helped the push for STEM teaching and the need to address the number of women participants, which has grown to about 30% of YRS attendees. However, this paper only addresses the perceptions of the YRS participants and their parents.

YRS Background

Rewired State launched in 2008 in order to show the power of open data based applications to organisations by framing projects around small experiments with limited resources. The first Rewired State event had few young people, so YRS was launched in 2009 as a means to encourage young people into the joys of community hacking events based around open data, and saw 50 participants at the first event. The 2010 event ran for a week, and had twice the number of participants (http://www.rewiredstate.org/legend, Mulqueeny, 2014b). In 2014 the YRS week saw 1000 children in 61 centres taking part (Mulqueeny, 2014b). This is a lot of participants, which they repeatedly attend if possible, as shown later in Table 3, it makes sense to ask them about their experience, and to see if there are patterns in attentees' characteristics, beliefs and attitudes, and development.

The Festival of Code week runs as follows: Children are hosted at centres around the UK Monday to Thursday. Each centre provides space, wifi and mentors, plus possibly lunch and snacks depending upon their sponsorship arrangements. Parents drop children off at the centre in the morning and collect them in the late afternoon; all centre staff are vetted for working with children. At the centres teams are formed on the Monday around software application ideas generated by the children with help from mentors, who also guide them during the week. All children, and parents of those under 14 travels to the weekend centre on Friday. Teams present their applications to judging panels on the Saturday, and the final judging panel taking place on the Sunday. While the weekend event was in London for several years, after outgrowing the venue there YRS moved to Birmingham for 2012 and 2013 before taking place in Plymouth in 2014, and returning to Birmingham for 2015.

In each of the YRS events the goal is for the children to develop applications using open data (http://theodi.org/guides/, Mulqueeny, 2014b). This can be a game, a tool, or anything, which lets the users play, or explore the data, or to better understand and use the data in the app. The apps can range from websites pulling disparate data sets together, to mobile apps and physical hacks based around open data, built around Raspberry Pi and Arduino boards.

The YRS events provide an opportunity for the children to learn from each other while building their apps. The YRS philosophy fits in with the gradual shift since the 1990s away from instructionalism towards creating constructivist learning environments (Ben-Ari, 2001) and promoting active learning (Beck, Chizhik, 2008; Biggers, Yilmaz, Sweat, 2009; Boyer, Dwight, Miller, Raubenheimer, Stallmann, Vouk, 2007; Chinn, Martin, Spencer, 2007; Hendrix, Myneni, Narayanan, McIntyre, 2004; Horowitz, Rodger, Biggers, Binkley, Frantz, Gundermann, Hambrusch, Huss-Lederman, Munson, Ryder, and Sweat, 2009; Hundhausen, Agrawal, Fairbrother, Trevisian, 2010; Huss-Lederman, Chinn, Skrentny, 2008; Lewandowski, Johnson, Goldweber, 2005; Shackelford, McGettrick, Sloan, Topi, Davies, Kamali, Cross, Impagliazzo, LeBlanc, and Lunt, 2006; Williams, 2007).

Apps developed by teams of young coders in the past (http://hacks.rewiredstate.org and http://hacks.youngrewiredstate.org) include Postcode Wars as a way to compare the population, crime and other statistics for UK postcodes; PiCyle to provide navigational information to cyclists using handlebar-mounted LEDs and a RaspberryPi; Tourify to provide custom tours based on your available time; and Miles Per Pound, which tells you how the cost of your driving is broken down by insurance, depreciation, repair, and millage so you know how many miles you can travel on one UK pound sterling.

The Survey Approach

Over the years little research has been conducted observing, monitoring and theorizing the impact such events are having on children's development. We want to start to change this. We felt that a survey was a good approach to use to explore the children and their YRS experience. We could not reach the children at the event itself, and we could not travel to a number of locations to conduct the surveys in person. This meant internet delivery using Typeform (http://www.typeform.com) to conduct the survey and exported the results to Excel for analysis.

We prepared a number of anonymous surveys. One was for the participant children under 18, and a second for those participants now 18 and over. A third survey was for the parents of the participating children, because without the parents the children cannot get to the events, and parents have to be there for the weekend events too if their child is under 14.

We proposed almost the same questions for an online survey of parents and children so that we would have comparable results. The questions fell into a number of distinct categories: location, events attended, length of time coding, and questions about the good/bad aspects of the events. Given changes in research policy, we needed to obtain ethical board approval before running the survey, and while approval for those 18 and over was forthcoming, approval for children under18s proved impossible in the timeframe available. Therefore, this is a pilot study and not the full study, which we will run during Young Rewired State 2015.

In order to have a significant uptake of the survey we need to run this in person with children, and not rely upon social media to spread the word about the survey. Our online pilot survey illustrated that our questions worked well. However, we need to couple this with staff asking participants to fill in the survey.

Children

We wanted to find out what drives the children to participate and what they did afterwards. We asked them the following:

(1) What year were you born?
(2) What is your gender?
(3) Where do you live? (Scotland, Wales, East of England, etc)
(4) What kind of coding or digital design do you do? (HTML/CSS, Ruby, Java, Blender, Unity, etc)
(5) How long before attending a YRS event did you either start to code, or do digital design work?
(6) Did you have any coding/digital design friends before attending a YRS event?
(7) Which YRS events did you go to?
(8) If you went to any weeklong Festival of Code events, did you also go to the weekend part?
(9) How far did you travel (one way) each day to get to the YRS centre host site?
(10) Did you tell others to sign up for the Festival of Code?
(11) What did you enjoy most about the YRS Festival of Code?
(12) What did you enjoy least about the YRS Festival of Code?
(13) If you could change one thing about the YRS Festival of Code, what would it be?
(14) What did you feel you learnt by attending the Festival of Code?
(15) What have you done that's related to coding or digital design since going to YRS Festival of Code?

(16) Have you used anything you learned from attending Festival of Code after the event?
(17) Please complete the following sentence. Attending the Festival of Code was like….
The questions were aimed at gathering information about the children in general, and also to determine whether the events were helping to grow a community in which participants can establish, share and deepen technical as well as problem solving skills in an inclusive community, which are some of the YRS programme goals. (Mulqueeny, 2014b; Slay, 2011; http://www.yrs.io)

Parents

Parents are key partners in this venture along with inclusive practices (Gamarnikow, Green, 1999), as without them the children do not attend events. Therefore they too should be questioned as many of them stay for the weekend event to help and support their younger children. We asked them the following:
(1) What year was your child born?
(2) What is the gender of your child?
(3) Where do you live? (Scotland, Wales, East of England, etc)
(4) How long before attending a YRS event did your child start to code, or do digital design on their own?
(6) Did your child have any coding/digital design friends before attending a YRS event?
(7) Which YRS events did you go to?
(8) If you went to any weeklong Festival of Code events, did you also go to the weekend part?
(9) How far did you travel (one way) each day to get to the YRS centre host site?
(10) Did you tell others to sign up for the Festival of Code?
(11) As a parent, what was the best thing about the YRS Festival of Code?
(12) As a parent, what was the worst thing about the YRS Festival of Code?
(13) If you could change one thing about the YRS Festival of Code, what would it be?
(14) What have you done, or have you noticed you child doing since going to YRS Festival of Code?
The questions, as with the children' ones, should point to changes, which help confirm behavior consistent with the goals of the YRS programme (http://www.yrs.io).

Results

We had fewer results than expected from our survey so this is a more qualitative than quantitative analysis. We had eleven parents respond (2 male, 8 female, one other), and ten children (4 male and 6 female) respond to our survey over a three-week period in November and December 2014. We ran the survey later than intended with less social media, and other help as our wait for ethical approval meant we missed key events where we could have had more opportunies to gain survey participants.

The children were born between 1994 and 1997, and the children of the parents were born between 1997 and 2000. This makes almost all of the participant children part of the 97er generation (Mulqueeny, 2014a), who are digital natives and who are aware of social media from their youngest age. These are digital natives, who grew up after Windows 95 was released and after the Internet and the Web became mainstream.

Less than half of the participants had coding/design friends before taking part in an YRS event. 50% of the children, and 6 of 11 parents' children had no coding/design friends before attending.

Most people told others about the YRS events after they found out about them. 8 of the 10 children, and 10 of the 11 parents did this.

Almost all people went to the weekend event after attending the weeklong event. This ends up being due to prior family arrangements, and the location of the weekend event.

Survey participants came from across the UK.

Home	**Parent's**	**Children**
London	4	1
South East England	1	2
South West England	0	4
East of England	1	2
Northwest England	1	2
West Midlands	2	0
Yorkshire and the Humber	1	0

Table 1. Participant's home locations

Table 1 shows that survey responses were from across a range of locations.

The importance of parental support for out-of-school group coding activity as provided by the YRS Festival of Code week is highlighted by the distances travelled by participants as shown in Table 2. More than half of them travel at least 10-20 miles to arrive at their weekly hosting centre.

The children and parents make an effort to attend the event because of a long-standing interest in the area for most children, which suggests that they hoped to find it a special event. This is also why they almost all told someone about it too.

Distance	**Parents'**	**Children**
< 5 miles	1	2
5-10 miles	3	2
10-20 miles	1	4
20-40 miles	2	1
> 40	4	1

Table 2. Distance to the YRS host centre

Participants like to repeatedly attend YRS events as shown in Table 3. None of our participants were born in 2001, which leaves that year empty. Table 3 combines all of the participants across both the children and parents surveys.

		YRS events			
Participants	**Born**	**2011**	**2012**	**2013**	**2014**
2	1994		2		
3	1995		1	2	1
4	1996	1			3
3	1997	1	2	1	2
3	1998	1	1	3	3
2	1999			1	2
2	2000			1	1
0	2001				
1	2002		1	1	1
1	2003			1	

Table 3. Survey participant attendance at YRS events

The children also showed a wide range of skills across both coding and design as seen in Table 4. ‘Other’ here includes Haskell, Perl and bash scripting as well as Visual Basic.

Language/Skills	Number
HTML/CSS	9
Javascipt/Jquery	6
Ruby	2
Python	4
Java	3
C#	2
Photoshop	3
Other	3

Table 4. Language and Skills Abilities

Given the small numbers it is hard to draw any conclusions about the coding and design skills here other than to point out the preponderance of web skills, which also shows up in the types of hacks developed by the teams over the years (http://hacks.rewiredstate.org and http://hacks.youngrewiredstate.org). Beyond this we can speculate that web skills are an easy entry point for those wanting to learn to code, and that compiled languages are probably learned in order to do something specific like a Minecraft mod using Java, or to pursue a game idea using the Unity game engine in C#.

These coding and design skills have developed over time as seen in Table 5.

Time	Parent’s	Children
1-6 months	2	2
6-12 months	2	2
Over a year	1	2
Over 2 years	4	0
Longer	2	2

Table 5. Time coding before attending YRS

More than half of the children had been coding for more than a year before attending YRS and some for over two years. This implies that the participating children find coding something more than a passing interest.

Correlating age of the child with how long they were programming before starting with YRS shows that 8 of 19 children, who coded for two years or longer were born between 1996 and 2000. This means there are a growing number of children, who are starting to code when they are aged around ten, which is usually well before they start any formal computing science in school. This is something, which needs to be explored further with a larger sample.

The open-ended questions provided more insight into what the attendees gained from the event.

The children thought the best thing about YRS was: Collaborating on a project for a week, where you get to meet like-minded people and make new friends and develop their interest.

The parents saw similar benefits: This was a free chance for them to help support their child's interest with knowledgeable mentors in a way that would boost their social circle, and also build their self-confidence.

The problems the children saw included the judging be only partially based on the technical ability; the groups becoming too big so each person contributed less, that the weekend might be too far away, when you did arrive, the vegetarian pizza might be all gone, and depending upon the venue, the wifi and sleeping arrangements for the weekend might be a problem.

The parents saw similar problems too. For parents the issues revolve around how they can support their children. The first issue might be what to do while their child is at YRS during the week and it is too far to head home. Then comes the consideration of whether the weekend part is too far away. The parents also find the judging unclear, and mostly wish they had heard about it two years sooner.

The YRS Festival of Code is not perfect, and everyone would change something if they could. The children would change things this way: change the weekend location and increase publicity so more children attend; make it more frequent than once a year; add workshops to the weekend event; enforce the competition rules more strictly; also allow those over 18 to attend.

The parents would like to see a central weekend location and clearer competition and judging criteria while adding events and activities for those not presenting, or perhaps remove the competition aspects as not all children care about that aspect. Other parents would like to enable more robotic kits and other hardware for the children to use, and to have people register for food to reduce the wastage at the event.

So far we have explored basic expectations of children and parents about events. However, we also wanted to know about what happened next. "What did you feel you learnt by attending the Festival of Code?" The children felt they learned a variety of direct computing things as well as transferable skills. They gained new technical skills and gained some insight into career options, and found ways to get involved with various companies. Some express regret at not finding out about YRS sooner, because they found there are a lot of young coders out there. Some pushed their skills further by teaching and mentoring others in programming

Under transferable skills the children gained presentation skills, and teamwork and time management. Others found that they are now better at working with others and have used this to help them with their school studies, and used this experience in their personal statements for university admissions.

And what happened next? Where did YRS participants take this experience? The children said that they are doing many things. Some give back to the community by taking part in more hackathons, helping local code clubs, and contributing to open source projects. Others continue learning by studying computing at higher levels in schools, or computing science/electrical engineering at university. Others have gone on to entrepreneurship and released an app with a friend, another is a freelance software engineer, and some work for startups. Others moved into the corporate world and participated in a company internship, or, work for an ISP, and some work other organisations.

The parents said that their children have changed. Their children have grown in confidence and started moving into hacking things with coding skills, and are keen to learn more. Some children have continued their YRS project further, and now think in terms of 'projects', and continue going to other coding events. One parent has started an YRS hyperlocal centre so that children in their area can attend YRS events more frequently.

The final question was aimed at finding out what else there is of the YRS experience that we might have overlooked. We asked the children to complete the sentence "Attending the Festival of Code was like…" Here are the answers of those who completed the question:

"Attending the Festival of Code was like… an escape."

"Attending the Festival of Code was like… An insight into the world of coding and hackathons which made me want to learn more."

"Attending the Festival of Code was like… a chance to be myself and find people who understand my interests."

"Attending the Festival of Code was like… becoming part of a mad hyperactive family of awesome people."

"Attending the Festival of Code was like… being in a room with a bunch of teachers, willing to share their passion."

"Attending the Festival of Code was like… a music festival but where there's as much pizza as mud and the songs are the tapping of keys..."

CONCLUSIONS

Coding events for children provide all of the expected 'upfront' benefits that we would expect. The children do gain in self-confidence; they do help each other learn new skills, and do learn how to work together as a team to achieve a specific goal. This was revealed in the answers to our open ended questions. Event activities such as these align particularly well with situated learning theory (Lave, Wenger, 1991), problem-based learning and experiential learning (Kolb, 1984).

We also found a wider range of outcomes too. The children gain a better understanding of career options, and gain opportunities to practice their craft through networking with other children and adults at the events too. They find that they are not alone. This is possibly the most important outcome. Whilst not part of formal mainstream education, the learning environments created by the events provide evidence of good inclusive practices. They promote key pedagogical principles, such as co-agency, working with others and trust; core ideas underpinning transformability and the capacity to learn [9].

The children want to know that they are not the only ones who are interested in the coding and developing apps. They too want to know other children, who share their same interests (Mulqueeny, 2014b).

The experiences of the survey participants seem to confirm studies of high achievers in computing too. Competitions, such as those run by the BCS, ACM, Microsoft and IBM, can motivate some students – winning looks good on the CV – but the challenge may not fit well with student knowledge and experience and may, ultimately, prove to be a discouragement; children may not be in a position to be able to enter, or parents may lack enough knowledge to be able to support them. Activities designed in the specific context of students' life experiences, such as YRS seeks to provide, can therefore have greater educational strengths. Furthermore, students can be encouraged to learn new skills and extend those that we consider important for their future (Carter, White, Fraser, Kurkowsky, McCreesh, Wieck, 2010).

It is also accepted that in order to motivate our more gifted students we need to engage them in a different way from the rest of the cohort. It seems the success of these approaches is entirely dependent upon the skill and enthusiasm of the instructor leading the initiative. YRS works with this by using skilled mentors and providing a peer-led educational experience where the students study at their own pace, which improves the learning experience (Carter, White, Fraser, Kurkowsky, McCreesh, Wieck, 2010).

For some children it appears that these events show that not all paths have to lead through the doors of a university, but that a coding career is feasible straight out the door of the school. However, this is the experience of our small survey sample, and a wider sample size might show this as a small segment. We have also skewed our results insofar that we are only gathering results from those now aged 18 and over. Younger children might have different outcomes, or might not yet have reached their peak.

The Future

We found that this is an area worth exploring further. We have shown that there is useful data waiting to be collected from the YRS participants and their parents. We now need to plan how to achieve this before the next event so that the ethical committee is satisfied that we will meet their requirements for the under 18 children.

We intend to do this by working with YRS to achieve more responses during and after the next Festival of Code in 2015. We aim to bring you more quantitative results next year.

Acknowledgements

Our thanks to Emma Mulqueeny, and Ruth Nichols as well as the other Young Rewired State staff, who helped make the survey possible.

REFERENCES

Beck L.L., and Chizhik AW. (2008) An experimental study of cooperative learning in cs1, *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education*, Portland, OR.

Ben-Ari M. (2001) Constructivism in Computer Science Education, *Journal of Computers in Mathematics and Science Teaching* 20(1), 2001

Biggers M, Yilmaz T., and Sweat M. (2009) Using collaborative, modified peer led team learning to improve student success and retention in intro cs, *Proceedings of the 40th ACM Technical Symposium on Computer Science Education*, Chattanooga, TN.

Boyer K.E., Dwight R.S., Miller C.S., Raubenheimer C.D., Stallmann M.F., and Vouk M.A. (2007) A case for smaller class size with integrated lab for introductory computer science, *Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education*, Covington.

Carter, Janet, White, Su, Fraser, Karen, Kurkovsky, Stanislav, McCreesh, Colette, and Wieck, Malcolm. (2010) ITiCSE 2010 working group report motivating our top students. In *Proceedings of the 2010 ITiCSE working group reports* (ITiCSE-

WGR '10), Alison Clear and Lori Russell Dag (Eds.). ACM, New York, NY, USA, 29-47. http://doi.acm.org/10.1145/1971681.1971685

Cellan-Jones, Rory. (2009) “Teenage hackers: Making a better world” BBC News http://www.bbc.co.uk/blogs/legacy/technology/2009/08/teenage_hackers_making_the_wor.html visited 28 December 2014.

Chinn D., Martin K., and Spencer C. (2007) Treisman workshops and student performance in CS, *Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education*, Covington.

Gamarnikow, Eva, Green, Anthony G. (1999) The third way and social capital: Education action zones and a new agenda for education, parents and community?, *International Studies in Sociology of Education*, 9:1, 3-22, DOI: 10.1080/09620219900200032

Hart, S., Dixon, A., Drummond, M.J. and McIntyre, D. (2004) *Learning without limits*, New York: Open University Press.

Hendrix D, Myneni L., Narayanan H., and Ross M. (2010) Implementing studio-based learning in CS2, *Proceedings of the 41st ACM Technical Symposium on Computer Science Education*, Milwaukee, WI.

Horwitz S, Rodger S.H., Biggers M., Binkley D., Frantz C.K., Gundermann D., Hambrusch S., Huss-Lederman S., Munson E., Ryder B., and Sweat M. (2009) Using peer-led team learning to increase participation and success of under-represented groups in introductory computer science, *Proceedings of the 40th ACM Technical Symposium on Computer Science Education*, Chattanooga, TN.

Hundhausen C., Agrawal A., Fairbrother D. and Trevisan M. (2010) Does studio-based instruction work in CS 1?: an empirical comparison with a traditional approach, *Proceedings of the 41st ACM Technical Symposium on Computer Science Education*, Milwaukee, WI.

Huss-Lederman S., Chinn D., and Skrentny J. (2008) Serious fun: peer-led team learning in CS, *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education*, Portland, OR.

Kolb, D. A. (1984) *Experiential learning: Experience as the source of learning and development* (Vol. 1). Englewood Cliffs, NJ: Prentice-Hall.

Lave, Jean, Wenger, Etienne. (1991) *Situated Learning. Legitimate peripheral participation*, Cambridge: University of Cambridge Press.

Lewandowski G, Johnson E., and Goldweber M. (2005) Fostering a creative interest in computer science, *Proceedings of the 36th SIGCSE Technical Symposium on Computer Science Education*, St. Louis, MO.

Mulqueeny, Emma. (2014a) *Introducing the 1997 Digital Natives, 97ers, and their networked communities of learning* http://mulqueeny.wordpress.com/2014/02/02/97ers/ visited 28 December 2014.

Mulqueeny, Emma. (2014b) *One thousand children, one room and a lot of code - welcome to the future.* http://www.theguardian.com/technology/2014/jul/31/children-code-young-rewired-state-festival-of-code-plymouth visited 28 December.

Peyton Jones, Simon. (2013) “The computing at school working group” In *Proceedings of the 18th ACM conference on Innovation and technology in computer science education* (ITiCSE '13). ACM, New York, NY, USA, 1-1. http://doi.acm.org/10.1145/2462476.2483794

Shackelford R, McGettrick A, Sloan R, Topi H, Davies G, Kamali R, Cross J, Impagliazzo J, LeBlanc R and Lunt B, *Computing Curricula 2005: The Overview Report*, ACM SIGCSE Bulletin 38(1), 2006

Slay, Julie. (2011) *Creating Stronger and More Inclusive Communities* http://www.neweconomics.org/publications/entry/creating-stronger-and-more-inclusive-communities visited 10 January 2015.

Smith, Neil, Sutcliffe, Clare, Sandvik, Linda. (2014) Code Club: Bringing Programming to UK Primary Schools through Scratch. 2014. *SIGCSE* ’14, March 05 - 08 2014, Atlanta, GA, USA. http://dx.doi.org/10.1145/2538862.2538919

Stephenson, Chris, Gal-Ezer, Judith, Humphreys, Simon, and Julie King. (2013) Transforming computer science in schools. In *Proceedings of the 18th ACM conference on Innovation and technology in computer science education* (ITiCSE '13). ACM, New York, NY, USA, 309-310. http://doi.acm.org/10.1145/2462476.2462510

Strauss, Valerie. (2014) *All students should learn to code. Right? Not so fast.* http://www.washingtonpost.com/blogs/answer-sheet/wp/2014/05/29/all-students-should-learn-to-code-right-not-so-fast/ visited 28 December 2014.R

Williams L, *Lessons learned from seven years of pair programming at North Carolina State University*, SIGCSE Bulletin 39(4), December 2007

Wing, Jeannette M. “Computational Thinking”. *Communications of the ACM* 49,3 (March 2006), 33-35. DOI=http://dx.doi.org/10.1145/1118178.1118215.

Nigel Beacham is a teaching fellow at the University of Aberdeen. He is director of Postgraduate Teaching within Computing Science. He has over 20 years’ experience working in Higher Education carrying out research into the computation, design, implementation and evaluation of e-learning systems and user interfaces of educational technologies.

Janet Carter is a senior lecturer at the University of Kent. She is director of undergraduate studies and senior tutor for the school of computing as well as the science faculty director for student experience. Her research interests lie in the realm of student experience: helping the strugglers; enthusing the most talented; gender issues in computing.

Bruce Scharlau is a Senior Teaching Fellow at the University of Aberdeen. He is the Director of Undergraduate Teaching in Computing Science. He is interested in using games for teaching, and in engineering more opportunities for students to work with live clients, developers and designers. He is a Lego Serious Play Facilitator.

Development of Interactive Teaching and Learning Materials for Bilingual Mathematics Education

Christine Bescherer,
University of Education Ludwigsburg

Abstract

A learning unit in middle school mathematics on the interactive, communication oriented, intercultural, and bilingual construction of triangles will be evaluated during a pilot study in early 2015. This paper gives some of the theoretical background on which the design of the unit is based as well as some examples of typical tasks from the lessons.

Keywords

Interactive devices for learning, geometry teaching, communication in mathematics education, content and language integrated learning/CLIL

Introduction

The INTACT-project (http://www.intact-comenius.eu/) is an EU-funded project with six project partners from Germany, Hungary, Ireland, Portugal, Romania and Spain. It aims at the development of interactive teaching and learning resources for content integrated language learning (CLIL) in the areas of school subjects and second language learning. The INTACT-project fosters interactivity in a double sense – interactivity with digital learning resources and international interactivity connecting learning all-over Europe using a common INTACT-platform.

This case study describes the development and early implementation of a learning unit about communication in mathematics education. Learning units from other school subjects like biology, history, geography, which are part of the project, will not discussed at this time.

THEORETICAL BACKGROUND

Designing a math lesson that uses interactive media like whiteboards or tablets in intercultural and bilingual settings draws on several theoretical backgrounds that are contributing different aspects to the pedagogical decisions.

Communication in Mathematics Education

Communication in mathematics education is an important part of the curriculum in itself. The National Council of Teachers of Mathematics (NCTM) identifies communication as one of the five processes all students should be able to master by the end of their school years.

> "Instructional programs from prekindergarten through grade 12 should enable all students to

- organize and consolidate their mathematical thinking though communication;
- communicate their mathematical thinking coherently and clearly to peers, teachers, and others;
- analyse and evaluate the mathematical thinking and strategies of others;
- use the language of mathematics to express mathematical ideas precisely." (NCTM, 2000, p. 60)

However, communication is also a very important instrument for teachers to support students' learning or to diagnose their learning i.e. by identifying misconceptions (see i.e.Huang& Normandia, 2009; Brendefur & Frykholm, 2000). So learning scenarios that allow students (and teachers) to communicate substantially in mathematics classroomsneed special planning.Typical topics where communication is an issue even in traditional mathematics classrooms are the descriptions of constructing congruent triangles in middle school (see older textbooks or i.e. Perks & Prestage, 2006). Toexemplify each step in the construction process students really need to understand how triangle constructions work. Bycomparing the drawing with the description teachers can easily identify mistakes and even students' misconceptions regarding geometrical objects.

Content Integrated Language Learning (CLIL)

The CLIL methodology comes originally from foreign language learning and is related to the concepts of content-based instruction, bilingual instruction or immersion (Doyle, 2007). However, CLIL focusses more on content learning than other approaches. In school contexts 'content' means usually the traditional school subjects most often natural or social sciences.

> "In CLIL, with regard to cognitive processes, the foreign language becomes an instrument for processing and storing of information. The use of a foreign language requires a different, deeper way of information processing and leads to enhanced acquisition of both the language and the non-linguistic content matter. Due to the differences of "mental horizons" reflecting the work in a foreign language, CLIL also influences the formation of notions, and thus literally shapes the way we think."
>
> (Hofmannová; Novotná& Pípalová, 2008, p. 21)

CLIL can be used in different structural frames i.e. a parallel stetting where some subjects are taught in the foreign language for one or more school years or a modular setting where only selected topics are taught in a foreign language over a limited period during a school year (Breidbach & Viebrock, 2012). The CLIL philosophy integrates pedagogical concepts like 'authenticity' (material, topics, language, etc.), 'negotiation of meaning', 'students as active learners', 'multi-channel learning' (visualizing, hands-on activities, etc.), 'scaffolding of the learning processes', 'step-by-step approaches', and 'demonstrative representations'.

Many of these concepts are realized in the lessons about constructions of triangles and descriptions of the construction processes.

Intercultural Education

'Intercultural competences' is seen as the "ability to effectively and appropriately interact in an intercultural situation or context" (Perry & Southwell, 2011, p. 453) and is part of the educational standards in a wide range of countries.While getting in contact with students from other nationalities is an important part of intercultural education in itself (Allemann-Ghionda, 2008) there are domain specific aspects as well: In mathematics education, 'intercultural learning' is often used to further the understanding i.e. of algorithms by comparing long multiplication done in Germany or in Greece or the US. However to talk about the names of i.e. geometrical shapes or forms in different languages like "triangle" in English vs. "Dreieck" (three-vertex) in German or "quadrilateral" vs. "Viereck" (four-vertex) can also lead to a deeper understanding of the geometrical properties of these objects.

Use of Interactive Devices for Learning

There are numerous approaches to using interactive learning devices like interactive whiteboards, tablets, smartphones, for learning in various age groups and learning settings. A lot of the research on the use of these devices in classrooms describes an increase of interaction between teachers and students or among students (see Manny-Ikan, Dagan,Tikochinski &Zorman,2011;Shi, Y.; Yang, Z.; Hao Yang, H. & Liu, S., 2012, Higgins, S. E., 2010) or allows an authentic learning environment in i.e. language and literacy classrooms (Reid & Ostashewski, 2011).

The main advantages relevant in the INTACT-project are the interactivity of the resources, the integration of multimedia in particular the possibility of audio in foreign language learning and the possibilities to connect with mobile devices such as smart phones. Especially the use of audio in the communicative scenarios of language learning is considered very helpful for students (Yang & Wang, 2012, Son & Park, 2012).

The combination of interactive resources with bilingual education scenarios and collaborative learning across different nations and languages makes the INTACT-project very special. Because of the different technological solutions and incompatibilities of software, theINTACT-project focuses on theimplementation of html5 (http://www.w3.org/TR/html5/)as a technical solution independent from any particular technology.

INTERACTIVE AND BILINGUAL CONSTRUCTION OF TRIANGLES

The learning unit is part of geometry class in the 7th or 8th grade. The class should be used to dynamic geometry systems (DGS, i.e. Geometer's Sketchpad or Cinderella) and of course have had several years of foreign language learning.

The aims[11] of the unit are:

- Students draw and construct geometrical shapes using appropriate tools like compass, straightedge, protractor or dynamic geometry systems.
- Through constructing triangles students deepen their understanding of the properties of parallel or perpendicular lines, perpendicular bisectors, angle bisectors, heights, median lines in triangles, the Thales theorem, …
- Students apply their knowledge of the properties of parallel or perpendicular lines, perpendicular bisectors, angle bisectors, heights, median lines in triangles, the Thales theorem, … for solving geometrical and real world problems.
- Students communicate mathematically – and in foreign languages - using the description of the construction processes.

The unit consists of five lessons. The shown tasks are demonstrations of key problems of the lessons.

Lesson 1 - Construction of triangles using a given description of a construction process:

[11] The aims regarding the mathematical aspect are commen obejctives of German mathematics classrooms in 8th grades.

Task1: Work individually on the construction problem and compare afterwards your triangles.

Description of constructing process

1. Draw the line segment AB = c = 6 cm.

2. Draw a circle whose midpoint is the midpoint of c with radius c/2 = 3 cm.

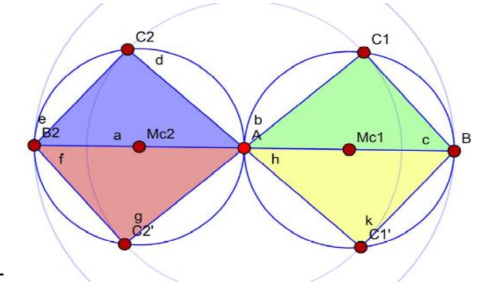
3. Draw a circle around A with radius 4,5 cm. The point C is at the section of the two circles.

4. Draw the line segment CB called a.

How many congruent triangles do you get?

Checking: Compare the lengths of the sides and the widths of the angles of your triangles with the ones in the sample construction.

Task 2: What are the connections of this construction process with the construction you worked on?



A	Punkt(3.76|-6.28)	(3.76|-6.28)
a	GeradeDurch(A;179,17°)	y = -0.0146x - 6.2252
C0	Kreis(A;6)	$(x - 3.76)^2 + (y + 6.28)^2 = 6^2$
B	Schnittpunkt(C0;a)	(9.7594|-6.3674)
B2	Schnittpunkt(C0;a)	(-2.2394|-6.1926)
Mc1	Mitte(A;B)	(6.7597|-6.3237)
Mc2	Mitte(A;B2)	(0.7603|-6.2363)
E	Kreis(Mc1;B)	$(x - 6.7597)^2 + (y + 6.3237)^2 = 3^2$
H	Kreis(Mc2;A)	$(x - 0.7603)^2 + (y + 6.2363)^2 = 3^2$
C3	Kreis(A;4.5)	$(x - 3.76)^2 + (y + 6.28)^2 = 4.5^2$
C	Schnittpunkt(C0;C3)	(1:-0 - i*1:0)
D	Schnittpunkt(C0;C3)	(1:-0 + i*1:0)
C1	Schnittpunkt(E;C3)	(7.178|-3.353)
C1'	Schnittpunkt(E;C3)	(7.0913|-9.3053)
C2'	Schnittpunkt(H;C3)	(0.342|-9.207)
C2	Schnittpunkt(H;C3)	(0.4287|-3.2547)
b	Strecke(C1;A)	_
c	Strecke(C1;B)	_
d	Strecke(C2;A)	_
e	Strecke(C2;B2)	_
f	Strecke(B2;C2')	_
g	Strecke(C2';A)	_
h	Strecke(A;C1')	_
k	Strecke(C1';B)	_
Poly0	Polygon(A;B;C1)	_
Poly1	Polygon(A;C1';B)	_
Poly2	Polygon(B2;A;C2)	_
Poly3	Polygon(B2;C2';A)	_

- Lesson 2 - Describing construction processes esp. while using DGS for construction:
- Lesson 3 & 4 - Posing and solving construction of triangles problems using the description of the construction processes:
- Lesson 5 - Reflection on the shared experience and reviewing the vocabulary/mathematical terms

Lessons 1, 2 and 5 take place in the normal classroom settings but during the lessons 3 and 4 the two classes meet online on the INTACT-project platform, exchange their problems, and discuss the solutions of the others. The platform offers among other features tools for chats, video conferencing, audio as well as iframes, which allow embedding of different documents or objects. The platform is scalable to interactive whiteboards, tablets, monitors or smartphones.

OUTLOOK

The INTACT-project platform was finished early 2015 and the piloting of the learning unit will take place during the first halfof 2015. The plan is for two classes from

Tasks - *work together in groups of three or four:*

- *Pose at least three different construction of triangles problems or choose out of the given problems and work the construction by using paper and pencil as well as DGS.*
- *Describe the construction processes in German and in English (or mother tongue and foreign language).*
- *Exchange the texts with your partner group in the other countries.*
- *Do the construction of your new problems and compare the results with your partner group.*
- *Talk with your partner groups about the constructions especially the mistakes and improve the descriptions.*

Germany and Spain who are used to CLIL – with English as the foreign language – and know already about the properties of congruent triangles to work together. The case study will evaluate the feasibility of the scenario as well as the effects on the students and the teachers, mainly using qualitative methods like interviews. Vocabulary tests and a content analysis of the artefacts produced by the students during the unit will complement the interviews.

REFERENCES

ALLEMANN-GHIONDA, C. (2008).*Intercultural Education in Schools***.** European Parliament Committee Culture and Education, Policy Department B: Structural and Cohesion Policies, IP/B/FWC/2006-169-Lot2-C01-SC02, online at http://www.pedz.uni-mannheim.de/daten/edz-ma/ep/08/EST20951.pdf, last retrieved, 02/23/2015.

BREIDBACH, S. & VIEBROCK, B. (2012). CLIL in Germany: Results from recent research in a contested field of education. *International CLIL Research Journal*, 1(4), 5 – 15.

BRENDEFUR, J., & FRYKHOLM, J. (2000). Promoting mathematical communication in the classroom: Two preservice teachers' conceptions and practices. Journal of Mathematics Teacher Education, 3(2), 125-153. doi:10.1023/A:1009947032694

COYLE, D. (2007). Content and Language Integrated Learning: Towards a Connected Research Agenda for CLIL Pedagogies, *International Journal of Bilingual Education and Bilingualism*, 10:5, 543-562. doi:10.2167/beb459.0

HOFMANNOVÁ, M., NOVOTNÁ, J., & PÍPALOVÁ, R. (2008). Assessment approaches to teaching mathematics in English as a foreign language. *International CLIL Research Journal*, 1(1), 20-35.

HUANG, J., & NORMANDIA, B. (2009). Students' Perceptions on Communicating Mathematically: A Case Study of a Secondary Mathematics Classroom. *International Journal Of Learning*, *16*(5), 1-21.

MANNY-IKAN, E., DAGAN, O.; TIKOCHINSKI, T.; ZORMAN, R. (2011). Using the Interactive White Board in Teaching and Learning – An Evaluation of the SMART CLASSROOM Pilot Project. *Interdisciplinary Journal of E-Learning and Learning Objects,* 7, 249–273.

NCTM - National Council of Teachers of Mathematics (2000). *Principles and Standards for School Mathematics*. Reston, Virginia, USA.

PERKS, P., & PRESTAGE, S. (2006). The Ubiquitous Isosceles Triangle: Part 1—Constructions. *Mathematics In School*, *35*(1), 2-3.

PERRY, L. & SOUTHWELL, L., (2011). Developing intercultural understanding and skills: models and approaches. *Intercultural Education* 22(6). doi: 10.1080/14675986. 2011.644948

REID, D. & OSTASHEWSKI, N. (2011). iPads in the Classroom – New Technologies, Old Issues: Are they worth the effort? In T. Bastiaens & M. Ebner (Eds.), Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2011 Chesapeake, VA: AACE 1689-1694. Online at http://www.editlib.org/p/38089 last retrieved 02/23/2015.

SHI, Y.; YANG, Z.; HAO YANG, H. & LIU, S. (2012). The impact of interactive whiteboards on education. In *Proceedings of the 4th International Conference on Internet Multimedia Computing and Service* (ICIMCS '12). ACM, New York, NY, USA, 213-218. doi:10.1145/2382336.2382397

Son, J. B., & Park, J. Y. (2012). Intercultural usability of language learning websites. *International Journal of Pedagogies and Learning*, *7*(2), 135-141.
doi:10.5172/ijpl.2012.7.2.135

Higgins, S. E. (2010). 'The impact of interactive whiteboards on classroom interaction and learning in primary schools in the UK.', *Interactive whiteboards for education: theory, research and practice*. Hershey, Pa: IGI Global, 86-101. doi:10.4018/978-1-61520-715-2.

TUPAS, R. (2014) Intercultural education in everyday practice. *InterculturalEducation*, 25:4, 243-254, doi:10.1080/14675986.2014.883166
Yang, K. T., & Wang, T. H. (2012). Interactive WhiteBoard: Effective Interactive Teaching Strategy Designs for Biology Teaching. tech, E-Learning-Engineering, On-Job Training and Interactive Teaching, 139-154. Online http://cdn.intechopen.com/pdfs-wm/32046.pdf last retrieved 05/14/2015

Needs analysis for an online learning service

Armelle Brun, *armelle.brun@loria.fr*
Université de Lorraine, France

Monique Grandbastien, *monique.grandbastien@loria.fr*
Université de Lorraine, France

Julie Henry, *julie.henry@ulg.ac.be*
Université de Liège, Belgium

Etienne Vandeput, *evandeput@ulg.ac.be*
Université de Liège, Belgium

Abstract

The development of fully or partially online training, as well as the provision of digital learning materials for students, requires the development of systems that provide students with innovative services tailored to their needs. We are interested in students who undertake all or part of their course in a foreign language and who face difficulties due to limited language skills. Our goal is to provide them, on demand, online and in their native language, with a document similar to the one that they find difficult to understand. In this paper we present the needs analysis phase, related to the engineering process of the online service.

Keywords

Digital educational resources, resource provision in another language, online service, needs analysis

INTRODUCTION AND CONTEXT

International student mobility is a big challenge. While moving abroad, many "mobile students" (ERASMUS, 2013; Endrizzi, 2010) face major difficulties such as a different style of education (more abstract, more intensive, etc.), a lack of prior knowledge and required skills, new requirements, etc. For a non-native student, such challenges are enhanced by language difficulties (Coulon & Paivandi, 2003; Erlenawati, 2005).

The issue is complex and can be addressed from several viewpoints: institutional, organizational, technological, educational and methodological, among others. From the technological side, it is clear that many digital teaching materials that are made available to students have few functional features, little interactivity and no smart support for the learning process. Thus, despite many studies and prototypes (Devedzic, 2006; Melis et al., 2009), existing e-learning systems, platforms and services rarely take advantage of modern semantic and adaptive technologies. That is the reason why it is crucial to develop services that provide non-native students with new and innovative services tailored to solve their problems.

Our goal is to experiment with an on-demand online service, focusing on scientific, technological and management disciplines. It has been developed in the framework of the Interlingua project[12] whose partners are research and academic institutions from Belgium, France, Germany and Luxemburg. That service aims to help non-native students who face difficulties understanding a learning resource. It recommends a similar resource in their native language. Notice that "similarity" does not mean

[12] http://www.interlingua-project.eu/

“translation” and therefore has to be defined. Moreover it also provides automatically generated MCQ based on keywords to facilitate self-assessment.

So, the Interlingua service is an innovative online learning and training service that implements an integrated approach. This approach mixes technological and pedagogical expertise and combines semantic linking technologies for multilingual educational materials and an automatic generation of self-assessment items

The goal is to help the learner to become familiar with the language of the course by understanding terms and expressions used in the context of the discipline. By providing a student with a resource in his own language we expect he will be able to establish interesting connections between the current learning resource and the language support resource

The system manages a multilingual resource database. Once a resource, written in the language of the course, is proposed to the learner, the Interlingua service offers the opportunity to identify the main concepts (words highlighted by the service), to select a paragraph or a slide (if the resource is a presentation). Then, the learner may ask the service to provide him with a similar resource in his mother tongue.

The original resource can include a description of an exercise, a recommended reading, course notes, etc. The recommended resource (in the native language of the student) allows him to better understand the concepts as linguistic problems are removed. The new resource is not a translation of the original one but an additional resource which presents the same concept or the same issue. This new resource may contain other examples or illustrations, and should improve comprehension, addressing linguistic problems and problems related to other aspects like education style, prerequisite, culture, etc.

This paper presents the phase related to the needs collection and analysis for engineering this online service. This step is the very beginning of the service design process and does not include the human machine interface design. However the results gained through this step will inform the next design phases.

METHODOLOGY AND DATA COLLECTION

Supporting students who are in mobility is the main objective of the Interlingua project. So, it is crucial to understand in detail the nature of the problems experienced by students in the specific context of a course, when it is given in a language different to their mother tongue. In this project we specifically focus on problems faced by students in the “Great Region”[13] who may study in German, French and/or English. Obviously, the problem is more general and is not limited to the Great Region case. As a first experience, the Interlingua prototype will consider courses related to basic statistics as such a course is included in many curricula, so we can collect the testimonies of many students.

An important issue regarding the needs analysis was to know who had to be questioned. Firstly we identified the international students within the project’s participating institutions and also identified students who take courses abroad. These students were asked about the difficulties they faced. Secondly, we identified statistics teachers who regularly meet such students to collect their perceptions of the difficulties these students face. We also wanted to get perceptions of teachers from other disciplines who are in the same situation, in order to identify possible common characteristics.

[13]Saarland - Lorraine - Luxemburg - Rheinland - Palatinate - Wallonia (including the German-speaking Community of Belgium) situated between Rhein, Moselle, Saar and Meuse

We designed a questionnaire for students in mobility. The development of this questionnaire was done in two stages. We met individually 10 students who were or had been studying abroad and asked them, through informal interviews, the main difficulties they encountered. The questionnaire was then structured according to their answers. The resulting questionnaire is comprised of 24 questions split into four sections. Some questions may contain several sub-questions. The first section collects data about student mobility: original language, semester, language used in courses, institutional framework of this mobility (i.e. ERASMUS, joint degree programs, etc.) and number of courses. The second section is used to identify the problems faced by these students in relation to linguistic or cultural aspects, working habits, prerequisites, etc. The third part is related to solutions to overcome these problems, whether these solutions are offered by the hosting institution, implemented by students or simply considered useful. The last section deals with personal information: age, sex, birth country, etc. The questionnaire was distributed widely within the project's partners, *i.e.* universities from Lorraine (France), from Liège (Belgium) and from Saarbrück and HTW (Germany).

To supplement the collected data, it also seemed appropriate to interview some students. We decided to conduct semi-structured interviews in which students were asked to explain about problems they had faced and how they overcame them. An interview guide was designed. This interview guide has been used to verify that the students provided the expected information in the survey and not as a formal interview guideline. The techniques used were therefore essentially reformulation of questions, demand for details and more rarely questions to stimulate conversation. With students' consent, the interviews were recorded, transcribed and qualitatively analysed. The results of the analysis have been compared with the results of the online survey.

At the same time, we conducted semi-structured interviews with teachers who regularly interact with students in a situation of mobility or teachers who were familiar with the difficulties of language proficiency. The aim was to collect information about the relevance of the Interlingua project and to get recorded with the agreement of teachers and transcribed before analysis.

MAIN RESULTS

Quantitative analysis

The questionnaire was made available online[14]. It was available in three equivalent versions in German, English and French.

During September and October 2014, we received 460 responses but only 210 were complete. The 250 incomplete responses mainly corresponded to students who just started filling in the questionnaire and then exited. The following quantitative analysis is related to the 210 complete responses provided.

Education profile of those who filled in the questionnaire

The three versions (one per language) have been answered in the same proportions. Furthermore, 30% of students are native German, followed by French and Italian natives (15% each). Students' countries of origin are mainly Germany, Italy, France, Belgium and Luxemburg. Percentages about the countries of study are nearly the same. The courses taken by students were mostly taught in French and German.

[14] http://enquete.univ-lorraine.fr/collegium/index.php?sid=75276&lang=en

Their mobility experience was rather recent. For over 88% of them, it was less than one year.

Difficulties encountered

In relation with the amount of additional work requested:

- 44% of students did not notice a significant increase,
- 30% did.
- 26% have no opinion

When questioning students about possible reasons for the difficulties faced, we noticed that the answers were equally distributed between the required autonomy, the difficulty in understanding course material and the number and difficulty of exams. Compared to what they face in the university from where they come:

- 35% did not notice any difference,
- 32% noticed an increased difficulty,
- 33% noticed a decreased difficulty.

In contrast, the main difficulties encountered by more than 50% of students were taking notes, understanding the explanations of teachers, reading comprehension of course materials, wording of tests and examinations.

Additional difficulties were expressed as free text in the answers:

- lack of additional course materials,
- feeling of isolation due to language.

Some students found it stressful that teachers and other students had discussions in their native language and not in the official language of the course.

Furthermore, the cultural references are noticed as problematic by some students (around 33% of the 36% who bring answers to this open question):

- different ways of teaching,
- implicit references or attitude.

Problems with the vocabulary (understanding, oral or written production) were often mentioned through open questions in similar proportions.

Solutions offered by institutions or adopted by students.

Among the solutions offered by the universities to limit these problems, we found:

- language courses,
- special arrangements for examinations,
- additional exercises on platforms including online services.

For their part, students also adopted solutions:

- taking language courses,
- using specific support for the technical vocabulary (e.g. online dictionary),
- asking for help from their peers,
- using online resources.

When asked about online resources (the focus of the Interlingua service) such as online books, online course materials, online courses in their native language or Internet research, 80% of students thought it would be a good help.

Qualitative analysis

A qualitative analysis of the joint interviews (students and teachers) led to identify five dimensions:

- the variety of contexts faced by the target audience,
- the nature of the language problem and consciousness of teachers,
- the reactions of teachers and students when they are confronted with innovative strategies and resources,
- the current strategies of students in response to the problem,
- the interest of teachers and students in access to online resources.

Interviews with students suggested they developed different strategies, whether individual or collective ones, but sometimes more substantial than those that might be provided by the Interlingua service. These strategies change according to the discipline, for instance in Statistics, a student noticed that:

> "This is an area where the statements and questions can lead to many interpretations, the nuances of language can be more difficult to take into account than in Mathematics..."
>
> “During exams, it is sometimes difficult, especially for multiple choice questions (MCQ), because if I do not fully understand a word I hesitate because, in my opinion, I understand the course, but I do not understand the phrase which expresses the question”.

A German-speaking student taking courses in French explained that, like her classmates, when she does not understand something, she searches for other books in French but also in German:

> "Looking on the Web in German helps me, that's true”.

Furthermore, different cultural aspects were highlighted. In the context of offering assistance, the data support the provision of an equivalent resource written in the native language, and not a translated resource.

CONCLUSION

The students who answered the survey and/or were interviewed confirmed that there are linguistic and cultural issues linked to understanding learning resources amongst other difficulties. These data support the aim to propose and test a helpdesk to make the understanding of these documents easier.

Needs analysis shows that, in order to face linguistic problems, the students implement a wide diversity of strategies to help them get a better understanding of the language of the course. This includes taking language courses to improve their fluency in the target language, using technical vocabularies, using textbooks in their native language, and asking their peers to explain.

Taking into account these results, the Interlingua project decided to develop an open solution that gives prominence to students’ autonomy and vocabulary learning. We also decided to support students’ learning and self-assessment by highlighting of concepts, provision of glossaries and MCQ, targeting vocabulary learning and text understanding. The service also aims at providing feedback that incites them to discover more (another resource, for example) rather than providing them with very formal feedback (like the “correct answer”), using the study language to formulate assessment questions, etc.

The Interlingua service will provide a range of functions and enable students to choose the best way to use them. The Interlingua consortium is developing an automatic analysis device for documents and similarity calculations (Aletras, Stevenson & Clough, 2014) from semantic models of statistics area. This system will develop the process described in (Sosnovsky, Hsiao & Brusilovsky, 2012). It remains to be seen what use is made of the service by students, including students of cross-countries existing courses, in the Great Region.

Acknowledgments:

The Interlingua project is partly funded by the INTERREG IV A program under the reference FEDER 138 GR DeLux 2 3 274 “Interlingua”

REFERENCES

Aletras N., Stevenson M., Clough P., (2014) Computing similarity between items in a digital library of cultural heritage, JOCCH, 5 (4), 16 p.

Coulon A., Paivandi S., (2003) Les étudiants étrangers en France : l’état des savoirs, Rapport pour l’Observatoire de la vie étudiante, Université Paris 8, Centre de Recherches sur l’Enseignement Supérieur.

Devedzic V., (2006) "*Semantic Web and Education*", Monograph, Springer, Berlin Heidelberg New York.

ERASMUS, *Rapport d’activité 2013*, Agence Europe Education Formation France, retrieved from: http://www.agence-erasmus.fr/docs/2160_ra-2013.pdf

Endrizzi L., (2010). « La mobilité étudiante, entre mythe et réalité ». *Dossier d’actualité de la VST*, n° 51, février. Retrieved from : http://www.inrp.fr/vst/LettreVST/51-fevrier-2010.php

Erlenawati S., (2005) Language difficulties of international students in Australia: The effects of prior learning experience, *International Education Journal*, 6(5), p.567-580

Melis E., Goguadze G., Libbrecht P., Ullrich C., (2009), ActiveMath – a Learning Platform With Semantic Web Features in *Semantic Web Technologies for e-Learning*, Dicheva D., Mizoguchi R., Greer J. (eds) IOS Press

Sosnovsky S., Hsiao, I-H. Brusilovsky P., (2012) Adaptation “in the wild”: Ontology-based personalization of open-corpus learning material. In A. Ravenscroft, S. Lindstaedt, C. Delgado Kloos, & D. Hernández-Leo (Eds.), *Proceedings of EC-TEL'2012: 7th European Conference on Technology Enhanced Learning*, p. 425-431,. Berlin/Heidelberg, Germany: Springer

Armelle Brun is an assistant professor in Computer Science at Université de Lorraine (France). Her research focuses on artificial intelligence, user modelling, data mining, applied to e-education. She is the author of more than 50 papers in international conferences and journals.

Monique Grandbastien is an emeritus professor in Computer Science at Université de Lorraine (France). Her research interests include e-learning, online resources and Artificial Intelligence applied to Education, as well as Informatics education. She is on the editorial boards of several journals in the field and on many conference committees

Julie Henry is a chemist and a PhD student in Education Sciences. Her research topics are ICT didactics, didactics of informatics and ICT in education.

She is lecturer at Université de Liège (Belgium).

Étienne Vandeput is a mathematician and a computer scientist. Lecturer at Université de Liège (Belgium), he is interested in ICT in education. He is also an associate professor at Université de Genève (Suisse) where he is responsible for a research seminar in Didactics of Informatics. He is responsible for the organisation of the 6th Didapro Conference in Namur in January 2016.

Compile Error Collection Viewer: Visualization of Learning Curve for Compile Error Correction

Motoki Hirao, *hirao@sakailab.info*
Shizuoka University, Graduate School of Informatics, Japan

Yoshiaki Matsuzawa, *matsuzawa@inf.shizuoka.ac.jp*
Shizuoka University, Graduate School of Informatics, Japan

Sanshiro Sakai, *sakai@inf.shizuoka.ac.jp*
Shizuoka University, Graduate School of Informatics, Japan

Abstract

We have developed CocoViewer (Compile error Collection Viewer) for learners in programming to enable them to conduct analysis for compiling error records. CocoViewer generates charts that show a trajectory of reducing the correction time of the compile error that is calculated by logs recorded in students' computers during a programming course. Students can see lists of charts for many kinds of compile errors, as well as a particular detailed circumstance of error that is selected by a student. We hypothesized that the system promotes clear understanding regarding their compile error learning, and the following three effects were expected: (1) to encourage more experiences of compilation error correction, (2) improvement of the procedure of correcting errors, and (3) reducing unarticulated anxiety for the compile error. The system was tried in an undergraduate introductory programming course for approximately 100 non-CS, liberal arts students. Both qualitative research regarding students' use of the system and quantitative data in the questionnaire showed that the students appreciated the system and enjoyed using it. The system promoted dialogue among students regarding compile errors, and we succeeded in reducing unarticulated anxiety for students.

Keywords

Programming Education, Compile error correction, Reflected Learning

INTRODUCTION

In introductory programming environments using common text-based languages such as Java, C, or Python, it is inevitable that the learners have to spend some time correcting compile errors. Learning to correct compile errors has been the first obstacle for novices. Seeing numerous compile errors has caused "nightmares" for many beginning programmers. In the classroom, teaching assistants are always busy supporting and correcting errors. This results in both students and assistants having an inability to focus on the most important practice in the introductory level: thinking about workable algorithms. This had been manageable for CS (Computer Science) students, however, recent demands to develop higher level problem-solving skills using the computer, (Computational Thinking) (Wing, 2006), has broadened the target to non-CS students.

In reality, non-CS students may feel more negative about compile errors than teachers think. Our survey for students in our introductory programming course for non-CS students showed that 62% of students answered that they were feeling literal "fear" regarding compile errors. They also answered the question "What was the percentage of time you spent on compile error correction per total programming process?". The result of the students' perception was 28.4% where as our actual calculation was 15%.

In the literature, researchers have discussed two approaches to solve the problem. One is to avoid compile error occurrences by using a visual programming language such as Squeak (Ingalls et al., 1997) or Scratch (scratch.mit.edu). The visual programming approach enables beginning learners to build their program by block building as in a jigsaw puzzle style (Maloney, 2004), which can completely eliminate any compile errors, because it is not possible to connect blocks unless those blocks can connect grammatically.

Another approach is to develop an environment which provides some scaffolding where the students can easily learn how to correct compile errors. In reality, the number of occurrences for compile errors is not a concern for professionals. They can correct errors quickly because compile error messages give them clear clues for correcting errors. An advantage of this approach is that correcting error experiences promotes understanding of grammatical knowledge. This is a reasonable approach for redesigning compile error messages for novices, because the standard error messages that were assumed for professional use in design include many terminologies, therefore, it is hard to understand for beginners who have little grammatical knowledge.

We can take the first approach described above (visual language) in the early stage of programming education. However, it may be disappointing for highly motivated students, because many popular languages which provide attractive functions to solve problems by a computational approach, such as JavaScript in the browser, are still text-based. At least in a migration phase from visual to text-based language, they cannot avoid learning some grammatical knowledge. On the other hand, the second approach (error correcting scaffolding) never succeeded in completely wipe out beginners' fears. Once a student faces stacking in a number of compile errors, his/her motivation for learning programming is going to decrease and is unlikely to return due to the negative experience.

In this research, we tried to improve the situation in the second approach (error correcting scaffolding), by proposing a tool to encourage formative assessment regarding compile error correction learning by students themselves. Using the latest information technology, we can easily collect information concerning the learning procedure, and the tool providing visual information for students. Therefore, we took an approach using the computer as a learning-reflection tool (Collins & Brown, 1988). Our hypothesis behind this is that the students' "fear" about compile errors came from their lack of information and knowledge regarding actual compile error learning status. Thus, from this point on in this paper we refer to "fear" as "unarticulated anxiety".

We hypothesized that the system promotes clear understanding regarding their compilation error learning, and the following three effects were expected:

1. to encourage more experiences of compilation error correcting
2. improvement of the procedure of correcting errors
3. reducing unarticulated anxiety concerning compilation errors.

The remainder of this paper is organized as follows: We will discuss works related to this research in Section 2. We will introduce our course design, analysis tool in Section 3. Evaluation method is in Section 4. The results of the qualitative analyses are given in Section 5. Section 6 concludes with a discussion of the results and limitations of the study.

RELATED WORK

This study was built on research regarding individual skill growth in software engineering. Specifically, we focus on the chronological changes of the compile error correction time. PSP: Personal Software Process (Humphrey, 2001) is the classic theory and the only one which provides results of the chronological analysis of compile error correction time. Although the data was gathered in the professional development situation, PSP research results show the compile error correction time gradually decreases by experience. We can say it is an improvement of their skills and that seeing the phenomena visually encourages students toward further learning of programming. However, the original PSP has to be done by manual logging, which was unreasonable to conduct in the introductory programming course.

Recent information technology allows us to easily collect logs which record all operations in the students' computers. In this decade, many recording environments in the educational field have been proposed, resulting in a significantly low cost for collecting logs even in real-time (e.g. Alammaly et al., 2012). However, there are a few articles that try either analysis or visualization of the records, as we review in the following.

There are studies that attempted to discover indicators which show learning status by analyzing compile error correction logs. A number of reports statistically analyzed an occurrence of a compile error number in their programming education (Jadud, 2005) (Jackson et al., 2005) (Mow, 2012). They examined the difficulty of each compile error kind by counting the occurrences. Jadud (Jadud, 2005) examined calculating average correction time for each kind of detection of difficulty, and Thompson concurred (Thompson, 2006). Although the research concluded that the average time was useful to detect difficulties of the compile error kind, their research omitted the aspects of individual differences and temporal aspects of learning. Jadud, (Jadud, 2006) also tested a similar problem to detect the difficulty of each compile error kind. They proposed EQ (Error Quotient), which is calculated by how many times the students tried to compile in order to correct an error. However, the research omitted the aspects of correction time.

Recently, an approach in which manual qualitative analysis was included in the quantitative collected data was done in several studies. Marceau et al., 2011(Marceau et al., 2011) attempted to detect compile error difficulty for each kind by learners' activities. They defined activities which can be identified by learners' error correction processes, and their data in the actual education process was manually analyzed by the defined activities. Bringula et al., 2012(Bringula et al., 2012) proposed taxonomy of compile errors from a human cognitive aspect. The category was composed of six types (Thought error, Sensorimotor error, Omission Error, Memory error, Knowledge error, Habit error), and their regression analysis revealed the relationship between those types.

There is another approach for improving the problem by proposing software tools which directly support learners in solving their compile errors. In this category, the most frequent approach is to redesign compile error messages for novices (Hristova et al., 2003) (Nienaltowski et al., 2008) (Marceau et al., 2011). Specifically, two reports (Nienaltowski et al., 2008) (Marceau et al., 2011) show clear results were achieved through certain levels of improvement in the actual environment. Some research proposed tools that provide hints for learners to resolve their errors by using a database where the cases accumulated (Watson et al., 2012) (Hartmann et al., 2010). The approach is similar to the expert system. A situation of programming language might also be affected by a language kind. In this view, some researchers claimed advantages of the particular language (e.g. Scheme (Crestani et al., 2010)).

Our research does not take an approach which directly facilitates learners solving the problems they face, but takes the route which encourages the learners' learning process in the meta-cognitive layer. In this view, some studies tried a similar approach. For example, Chiken et al., 2005 proposed a tool which supports the learners' reflection process using failure knowledge. In another example, Kay et al., 2007 tried to let learners have self-assessment experiences for the activity of evaluation of several other’s programs. Belski, 2010 also claimed the importance of the metacognitive process in programming education, although their research continues to conduct self-assessment by questionnaire.

COCOVIEWER

System Design

We developed a system by which learners can analyze compile errors themselves. We call the System “CocoViewer (Compile error Collection Viewer)” from the system function characteristics, collection compile error correction data. Figure 11 shows the outline of our system flow.

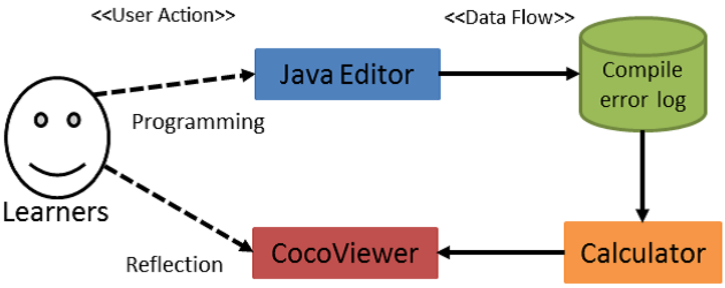


Figure 11: The outline of our system flow

Correction Time Chart

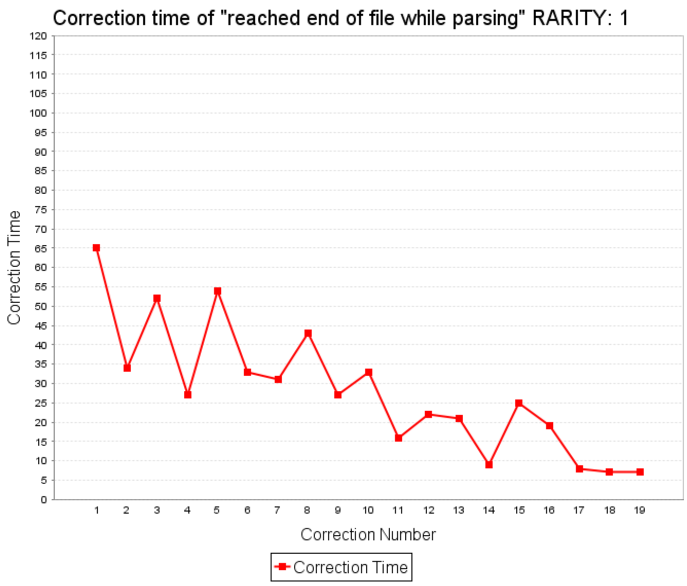


Figure 12: Correction time chart example

Figure 12 shows an example of a correction time chart. The chart targets personal compile error correction data. The chart's X-axis is compile error correction number (unit: counts) and the Y-axis is compile error time (unit: seconds). One chart is created per one compile error kind. For example, “`;' expected” is one compile error kind.

To illustrate, we can read in Figure 12 that the learner spent a compile error correction time of 65 seconds the first time and 34 seconds the second time.

CocoViewer Main Window

When CocoViewer is launched, the CocoViewer main window appears Figure 13 shows CocoViewer's main window. CocoViewer's main window shows a correction time chart table and whole compile error correction data. Compile error correction is different between learners. All charts are different for each learner.

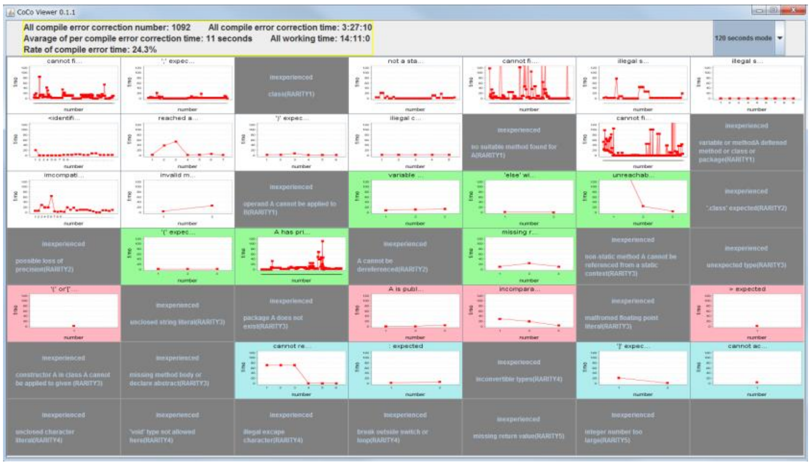


Figure 13: CocoViewer main window

The character of the CocoViewer's main window shows correction time charts. Learners can verify at a glance how many kinds of compile errors they correct and what type is most numerous by character.

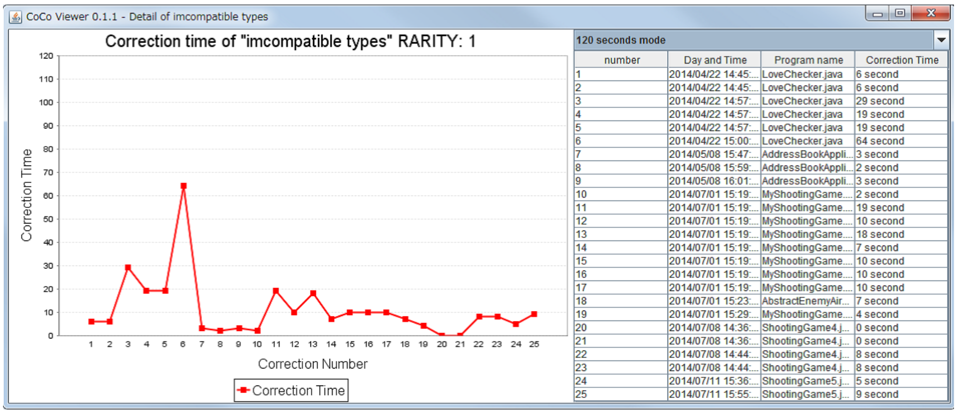
Moreover, the design we adopted assists learner is becoming highly motivated for compile error correction. The design is the following:

1. Set RARITY of compile error kinds
2. Change compile error correction chart background by RARITY
3. For each compile error message that is not corrected, a hint is given to assist with correction

Learners can see CocoViewer's design like a picture book. The design aim is that learners positively analyze their compile error correction.

RARITY means frequency of compile error kinds correction count. The frequency data used for this study involved gathering statistics of compile errors for the entry level Java programming class for liberal arts freshman during the Shizuoka University 2010 Autumn term.

Correction detail Window



number	Day and Time	Program name	Correction Time
1	2014/04/22 14:45:...	LoveChecker.java	6 second
2	2014/04/22 14:45:...	LoveChecker.java	6 second
3	2014/04/22 14:57:...	LoveChecker.java	29 second
4	2014/04/22 14:57:...	LoveChecker.java	19 second
5	2014/04/22 14:57:...	LoveChecker.java	19 second
6	2014/04/22 15:00:...	LoveChecker.java	64 second
7	2014/05/08 15:47:...	AddressBookAppli...	3 second
8	2014/05/08 15:59:...	AddressBookAppli...	2 second
9	2014/05/08 16:01:...	AddressBookAppli...	3 second
10	2014/07/01 15:19:...	MyShootingGame....	2 second
11	2014/07/01 15:19:...	MyShootingGame....	19 second
12	2014/07/01 15:19:...	MyShootingGame....	10 second
13	2014/07/01 15:19:...	MyShootingGame....	18 second
14	2014/07/01 15:19:...	MyShootingGame....	7 second
15	2014/07/01 15:19:...	MyShootingGame....	10 second
16	2014/07/01 15:19:...	MyShootingGame....	10 second
17	2014/07/01 15:19:...	MyShootingGame....	10 second
18	2014/07/01 15:23:...	AbstractEnemyAir...	7 second
19	2014/07/01 15:29:...	MyShootingGame....	4 second
20	2014/07/08 14:36:...	ShootingGame4.j...	0 second
21	2014/07/08 14:36:...	ShootingGame4.j...	0 second
22	2014/07/08 14:44:...	ShootingGame4.j...	8 second
23	2014/07/08 14:44:...	ShootingGame4.j...	8 second
24	2014/07/11 15:36:...	ShootingGame5.j...	5 second
25	2014/07/11 15:55:...	ShootingGame5.j...	9 second

Figure 14: Correction detail window

The correction detail window opens when the chart showing on the CocoViewer main window is clicked. Figure 14 shows a correction detail window example. This example is about “incompatible types”. The correction detail window consists of an expended correction time chart and correction detail table which includes correction count, correction date, target program name and correction time.

SCENARIO OF READING CHART

Learners can analyze their compile error correction by the correction time chart's curve. The chart's curve type is classified by 3 kinds of patterns: Decrease type chart, Jagged type chart, Increase type chart.

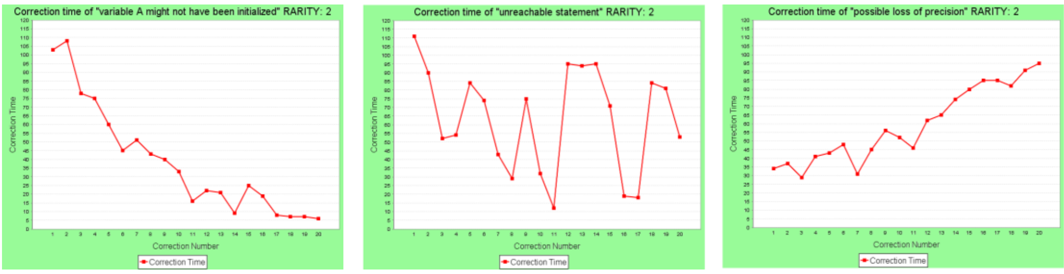


Figure 15: 3 chart’s curve type

The left side of Figure 15 shows decrease type correction time chart (decrease type chart) example. By reading their chart, learners can overcome and understand this compile error kind because decrease type chart shows compile error correction time decreasing as a result of compile correction experience. We expect a lot of decrease type charts because ideal learners decrease compile error correction time by compile error correction experience. Decrease type chart is an ideal type.

The middle of Figure 15 shows jagged type correction time chart (jagged type chart) example. By reading their chart, learners partially understand this compile error kind because jagged type charts show compile error correction time decreasing without compile error correction times. Learners change jagged type chart to decrease type chart to understand the cause of jagged type compile error occurrence and the method of compile error correction.

The right side of Figure 15 shows increase type correction time chart (increase type chart) example. By reading their chart, learners recognize that they do not understand this compile error kind because increase type chart shows compile error correction time increasing with compile error correction times. Increase type charts occur because the learner cannot understand the grammar of the program process as it is difficult for them and they have difficulty correcting compile errors between classes.

Scenario of Reducing “unarticulated anxiety”

CocoViewer users reduce “unarticulated anxiety” by the following processes:

1. View correction time chart, analyze chart curve
2. Decrease the gap between their expectation for compile error correction and actual compile error correction
3. Recognize extent of learning own compile error correction

EXPERIMENTAL PLAN

Hypothesis

Learners will reduce “unarticulated anxiety” by improving their incorrect compile error image by using CocoViewer.

Environment

The research targeted learners taking our introductory programming course for non-CS freshmen in 2013 autumn. There were approximately 100 freshmen. They had not experienced programming before this class.

Summary

We performed our experiment in the twelfth class. Learners received additional points for participating in the experiment. If learners did not want to use CocoViewer, they did not have to join the experiment. The experimental procedure was as follows:

1. We distributed questionnaires to learners
2. We explained the experiment summary and correction time chart reading when the class started (10 minutes)
3. Learners completed a questionnaire about using CocoViewer (10-15 minutes)
4. We gathered the questionnaires after the class

Targeted learners were able to complete the questionnaire after our explanation.

We responded to questions about the questionnaire and CocoViewer during the class. We observed and recorded learners’ experience using CocoViewer during the experiment.

Analytical Method

We removed the learners who did not complete the essay question because we determined that they did not complete the experiment correctly.

We analyzed the questionnaire and determined that 61 out of 71 users completed correctly.

RESULT

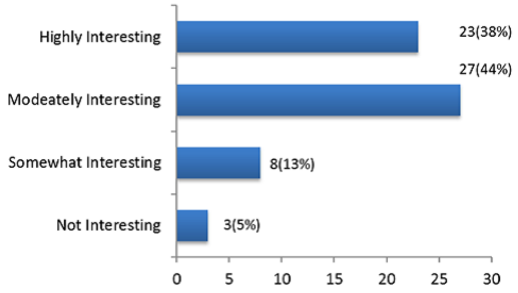


Figure 6: The result of “Did you find the compile error collection table interesting?”

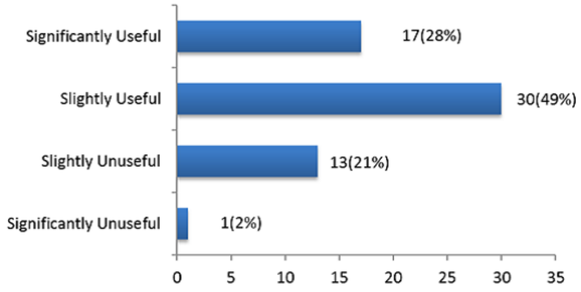


Figure 7: The result of “Did you feel the compile error correction time table was useful for learning?”

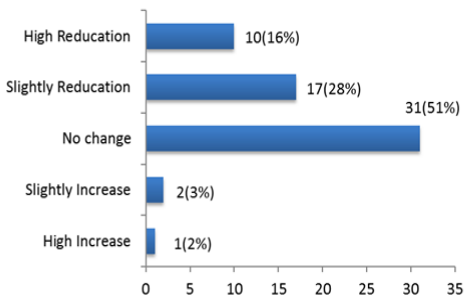


Figure 8: The result of “Was your “unarticulated anxiety” reduced through this experiment?”

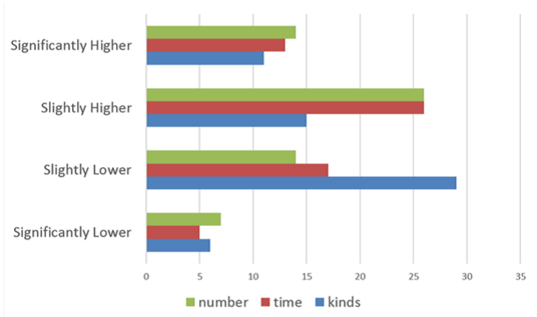


Figure 9: The result of the difference between expected and actual.

Figure **6** shows responses about interest in the compile error correction record. 82% indicated “highly interesting” or “moderately interesting”. The high interest in compile error correction records was due to the fact that learners had not had the opportunity to view compile error correction records before.

Figure **7** shows responses about the usefulness of this system for learners. 77% indicated it was “highly useful” or “slightly useful”. Respondents felt it was useful for analyzing their compile error correction record.

Figure **8** shows responses about reducing “unarticulated anxiety”. 44% indicated “high reduction” and “slight reduction”. The reducing of “unarticulated anxiety” was at least partially due to using the system.

Green line in Figure **9** shows responses about the difference between expectations of compile error correction number and the actual number. 66% indicated “significantly higher” or “slightly higher” than expected. The responders recognized a much higher number of compile error correction than they had expected.

Red line in Figure 9 shows responses about the difference between expectations of compile error correction time and the actual correction time. 64% indicated “significantly longer” and “slightly longer”. Responders recognized that compile error correction took longer than expected.

Blue line in Figure 9 shows responses about the difference between expectation of compile error correction kinds and the actual results. 58% indicated “significantly lower” and “slightly lower”. Responders indicated that they had fewer compile error correction methods because they corrected fewer kinds of compile errors.

DISCUSSION

Evaluation of interest in Compile Error

The result of Figure 6 shows that the majority of learners are interested in analyzing compile error correction.

Those opinions were the result of learners experiencing compile error correction through CocoViewer by which they were able to view compile error correction records quantitatively. Learners gain interest in compile error correction through experience with compile errors.

The result of Figure 7 shows that learners felt that CocoViewer was useful for analyzing compile error correction. Learners want to use CocoViewer to understand the cause of compile error occurrence and the method of correction.

Accordingly, CocoViewer supports learners compile error correction study interest.

Evaluation of Reducing “unarticulated anxiety”

The primary method for reducing “unarticulated anxiety” is that by using CocoViewer the learner can decrease the gap between a vague image of compile error correction and the actual method of compile error correction.

We think the responses revealed the following from the results of Figure 9.

1. Compile error correction experience is sufficient because it is more than the actual compile error correction number expected and it is longer than the actual compile error correction time expected.
2. Learners experienced few compile error correction methods because they corrected few kinds of compile errors.

It's an especially important point that learners experienced few compile error kinds; the occurrence of compile error kinds was minimal. Learners primarily corrected similar compile errors because they corrected few compile error kinds. Learners felt they mastered compile error correction methods by correcting compile errors which occurred many times.

But there were some unexpected learners who did not experience a decrease in compile error correction time. The “unarticulated anxiety” did not decrease in those learners who did not learn good compile error correction methods.

Accordingly, the hypothesis is partially supported.

CONCLUSION

We developed a system that allows learners to analyze their compile error correction records. Our experiment took place in a novice programming class with approximately 100 students. Using the system increased learner interest in compile error correction study and at the same time learners indicated an interest in using the system. The system partially helps reducing “unarticulated anxiety” (unarticulated anxiety) for compile error by learners experiencing compile error correction many more times than they expected. The system closed the gap between the leaner's compile error expectation and actual correction experience. The system partially reduced the “unarticulated anxiety” for compile errors by learners experiencing compile error correction much more than expected.

REFERENCES

Alammary, A., Carbone, A., & Sheard, J. (2012). Implementation of a Smart Lab for Teachers of Novice Programmers. Proceedings of the Fourteenth Australasian Computing Education Conference, Vol. 123, 121–130

Bringula, R., Manabat, G.M., Tolentino, M.A., & Torres, E. (2012). Predictors of Errors of Novice Java Programmers. World Journal of Education, 2(1), 3.

Belski, I. (2010), The Impact of Self-assessment and Reflection on Student Learning Outcomes. Proceedings of the 21st Annual Conference for the Australasian Association for Engineering Education, 216-221

Chiken, K., Hazeyama, A., & Miyadera, Y. (2005). A Programming Learning Environment Focusing on Failure Knowledge, J. IEICE in Japan, Vol.J88-D1, No.1, 66-75

Collins, A. & Brown, J.S. (1988). The computer as a tool for learning through reflection. in Mandl, H. and Lesgold, A. (Eds.): Learning for Intelligent Tutoring Systems, 1–18, Springer-Verlag, New York.

Crestani, M. & Sperber, M. (2010). Experience report: growing programming languages for beginning students. In: Proc. 15th ICFP, Vol. 45, No. 9, 229-234

Hartmann, B., MacDougall, D., Brandt, J., & Klemmer, S.R. (2010), What Would Other Programmers Do: Suggesting Solutions to Error Messages. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1019-1028.

Hristova, M., Misra, A., Rutter, M., & Mercuri, R. (2003). Identifying and correcting java programming errors for introductory computer science students. ACM SIGCSE Bulletin, Vol.35, No.1, 153-156.

Humphrey, S.W. (1997). Introduction to the Personal Software Process. SEI Series in Software Engineering. Addison-Wesley

Ingalls, D., Keahler, T., Maloney, J., Wallace, S., & Key, A. (1997). Back to the Future: The Story of Squeak. A Practical Smalltalk Writtern in Itself, Proc. of ACM OOPSLA' 97, Vol. 32, No.10, 318-326

Jackson, J., Cobb, M., & Carver, C. (2005). Identifying Top Java Errors for Novice Programmers. Frontiers in Education, 2005. FIE '05. Proceedings 35th Annual Conference, T4C24-T4C27

Jadud, M.C. (2005). A first look at novice compilation behaviour using BlueJ, Computer Science Education. Vol.15, 25-40.

Jadud, M.C. (2006). Methods and tools for exploring novice compilation behaviour. In Proceedings of the second international workshop on Computing education research, 73-84

Kay, J., Li, L., & Fekete, A. (2007). Leaner Reflection in Student Self-assessment. ACE '07 Proceedings of the ninth Australasian conference on Computing education, Vol. 66, 89-95.

Maloney, J., Burd, L., Kafai, Y., Rusk, N., Silverman, B., & Resnick, M. (2004). Scratch: a sneak preview. Proceedings Second International Conference on Creating Connecting and Collaborating through Computing 2004, 104-109

Mow, C.T.I. (2012). Analyses of Student Programming Errors In Java Programming Courses. Journal of Emerging Trends in Computing and Information Sciences, Vol. 3, No.5, 739-749

Marceau, G., Fisler, K., & Krishnamurthi, S. (2011)., Mind your language: on novices' interactions with error messages, Proceedings of the 10th SIGPLAN Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software, 3-18

Nienaltowski, M.H., Pedroni, M., & Meyer, B. (2008). Compiler Error Messages: What Can Help Novices?. Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education, Vol.40, No.1, 168-172

Scratch Team Lifelong Kindergarten Group MIT Media Lab, (n.d.). Scratch - Imagine, Program, Share. Retrieved February 21, from http://scratch.mit.edu/

Thompson, S.M. (2006). An Exploratory Study of Novice Programming Experiences and Errors. Master’s thesis, University of Victoria

Watson, C., Li, F.W., & Godwin, J.L. (2012). Bluefix: Using crowd-sourced feedback to support programming students in error diagnosis and repair. Advances in Web-Based Learning - ICWL 2012, 228-239

Wing, J.M. (2006). Computational Thinking. Commun. ACM, Vol.49, No.3, 33-35

Motoki Hirao received the bachelor degree in Informatics from University of Shizuoka (Japan) in 2014. He is currently a graduate student of Informatics. His research interests include include educational software development, and compile error learning.

Yoshiaki Matsuzawa received his PhD in Media and Governance in 2008 from Keio University, Japan. He is an assistant professor in the Faculty of Informatics at Shizuoka University. His research interests include software-engineering education, educational software development, modeling and simulation of complex systems, and learning sciences.

Sanshiro Sakai received his PhD in engineering in 1984 from Shizuoka University, Japan. He is a professor in the Faculty of Informatics at Shizuoka University. His research interests include computer supported collaborative learning, software development environment and programming learning support system.

Finding Threshold Concepts in Computer Science Contest

Eglė Jasutė, egle.jasute@mii.vu.lt

Gabrielė Stupurienė, gabriele.stupuriene@mii.vu.lt

Vilnius University, Institute of Mathematics and Informatics, Lithuania

Abstract

The main goal of this paper is to find the relation between threshold concepts in computer science and the tasks of the *Bebras* contest. The *Bebras* contest is a worldwide initiative whose goal is to promote Informatics (or Computer Science, or Computing) and Computational Thinking. Threshold concepts plays a very important role for students' understanding and progression in education. In this paper, we discuss about an opportunity that the *Bebras* tasks could be the best way for students to learn and understand threshold concepts.

Keywords

Threshold Concepts, The *Bebras* contest, Informatics education, Computer Science education.

INTRODUCTION

Competition makes teaching of informatics (computer science, computing) more attractive for children. During contests students have the possibility to test their skills among peers from different schools or even countries and to make friends in a field that they are interested. The contest on informatics and computer fluency named *Bebras* (it is a Lithuanian word for "beaver") may be the key to the potential of informatics science knowledge and an attractive way to bind up technology and education (Dagienė, Stupurienė, 2014).

Algorithms and programming are included in "big ideas" of computing (Grover, Pea, 2013). Algorithms are tools for developing and expressing solutions to computational problems and programming is a creative process that produces computational artifacts. A little bigger attention is being paid to developing algorithms and programming and these are included in optional IT modules in Lithuanian informatics curricula. Therefore the *Bebras* contest gains a higher value in motivating pupils to understand computer and master it creatively.

The international *Bebras* community relates over than 40 countries in the whole world (bebras.org). In 2014, there were over 890 000 participants. Every year the International *Bebras* workshop participants develop a set of new tasks (some months before workshop each country provides 5-10 of task proposals). The group of international experts, teachers and students in pedagogy and computer science (in total more than one hundred people), follows a process that allows creativity in finding new tasks and ensures a high quality of the output. It is not easy to create tasks that satisfy all the criteria (the task can be solved within 3 minutes; the problem statement is easy understandable; the task is presentable at a single screen page; the task is independent from specific systems). Often it is a process of several versions from an imperfect task formulation to an acceptable formulation (Dagienė, Futschek, 2008). The national organizers make up their national task set from this pool.

The contests consists of a set of short questions or tests usually called *Bebras* tasks. These tasks can be answered without prior knowledge about informatics, but are clearly related to informatics concepts. Each *Bebras* task can both demonstrate an aspect of informatics and test the talent of the participant, regarding understanding of informatics (Dagienė, Stupurienė, 2014). Proposal for topics of the *Bebras* contests on informatics and computer literacy (Dagienė, Futschek, 2008):

Information (conception of information, its representation (symbolic, numerical, graphical), encoding, encrypting); **Algorithms** (action formalization, action description according to certain rules); **Computer systems and their application** (interaction of computer components, development, common principles of program functionality, search engines, etc.); **Structures and patterns** (components of discrete mathematics, elements of combinatorics and actions with them); **Social effect of technologies** (cognitive, legal, ethical, cultural, integral aspects of information and communication technologies); **Informatics and information technology puzzles** (logical games, mind maps, used to develop technology-based skills).

It is important to highlight that for the authors of tasks there are no requirements for the topic of tasks (not take into account the specific informatics concepts) which are proposed for workshop. But the task must be suitable for the topics mentioned above. The *Bebras* contest essentially focuses on informatics concepts. Understanding and handling the basics and foundations of informatics are more important than knowing technical details (Dagienė, Stupurienė, 2014).

Concepts (a concept is a unit of knowledge created by a unique combination of characteristics [ISO 1087-1: 2000]) of computer science (informatics, computing) always is classified using the categories such as core, fundamental, central, key, important and similar. These classification descriptors are focused on the conceptual structure of the subject and are indifferent to the learner's experiences and the difficulties encountered with the acquisition of conceptual understanding (Shinners-Kennedy, Fincher, 2013). Threshold concepts are different to core or central concepts. For example, core concepts are building blocks that must be understood; central concepts are compiled by using Schwill's four criteria (horizontal criterion, vertical criterion, criterion of time, criterion of sence) for fundamental ideas of computer science (Zendler, Spannagel, 2008). Threshold concepts are a subset of the core concepts in a discipline and they represents a transformed way of understanding, or interpreting, or viewing something without which the learner cannot progress (Meyer, Land, 2003).

In this paper we focus on a specific group of students (age 15-19) and consider what the threshold concepts exist in *Bebras* contest.

THRESHOLD CONCEPTS IN COMPUTER SCIENCE

A threshold concept suggests knowledge related to ideas, thoughts, and perceptions. If core concepts are the building blocks of knowledge, then threshold concepts can operate to transform and integrate that knowledge into a new way of thinking (Barradell, Peseta, 2014).

Threshold Concepts are defined by Meyer and Land (Boustedt, et al., 2007):

- Transformative: they change the way a student looks at things in the discipline.
- Integrative: they tie together concepts in ways that were previously unknown to the student.
- Irreversible: they are difficult for the student to unlearn.
- Potentially troublesome for students: they are conceptually difficult, alien, and/or counter-intuitive.
- Often boundary markers: they indicate the limits of a conceptual area or the discipline itself.

Threshold Concepts can be as a possible way to organize and focus learning in computer science (Eckerdal, et al. 2006). Identifying threshold concepts can help lecturers consider what they teach, why they teach particular ideas over others, how they teach, and when to teach those ideas. Embedding threshold concepts successfully in a curriculum carries the potential to positively shape what, and how, the student learns (Barradell, Peseta, 2014). A popular approach to the accumulation of evidence supporting a particular classification of concepts has been to ask experts.

In 2005 a group of computer science education researchers have decided to identify threshold concepts in computer science. The group's efforts represent the most persistent and varied attempt at investigating threshold concepts in the discipline of computing. List of investigated threshold concepts: functions, parameters, scope, iteration, recursion, encapsulation, graphs, objects, class, polymorphism, programming language grammars, addressing memory (pointers, references), control structures (e.g. if, loops), balanced trees, computer architecture, aggregation, algorithm, modeling, state, cohesion, abstraction, complexity, templates, threads, lists (arrays), trees, abstract data type (Shinners-Kennedy, Fincher, 2013).

METHODOLOGY AND ANALYSIS

In this research, we used set of 27 threshold concepts (investigated in the research in 2005) to find out whether they exists in *Bebras* tasks. This study is based on the analysis of the sets of *Bebras* contest held in Lithuania in 2012, 2013, 2014. We have analyzed sets of *Bebras* tasks for aged 15-19 pupils (the contest has five age groups: Little Beavers (age 8-10), Benjamin (age 11-12), Cadet (age 13-14), Junior (age 15-16) and Senior (age 17-19)). In total 88 tasks: 23 tasks in 2014, 31 tasks in 2013, 34 tasks in 2012. Every task have at least 3 keywords (what student can learn from the task) so it was the main data used in this analysis.

12 out of 27 threshold concepts exist in *Bebras* contest tasks: 7 tasks of the contest in 2012, 10 tasks of the contest in 2013 and 5 tasks of the contest in 2014. As we can see in the diagram (Fig. 1) algorithm and graphs are the most popular. Other 10 threshold concepts play not very important role. The tasks for sets in each participating country are chosen by their experts and they do not depend on the national informatics curriculum or concrete list of certain informatics concepts.

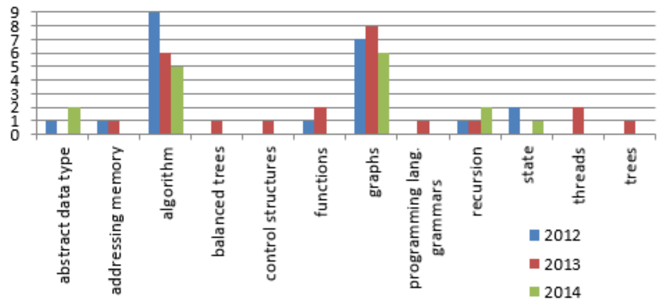


Figure 1: Frequency of threshold concepts in the *Bebras* tasks

As the diagrams illustrate, the threshold concepts algorithm and graphs are very important for the sets of contest. The reason could be based on the criteria for the *Bebras* tasks. As mentioned before, the tasks should be a short text, representing a quickly understandable problem. In 2013, Tomcsányiová and Kabátová discussed the opportunity that it is often possible to significantly shorten the verbal part of the task by using pictures. Pictures, illustrations, visualization and various multimedia in education materials are considered effective at enhancing the learner's ability to understand the instruction and thus help him to learn. Examples of the *Bebras* tasks (Fig. 2) show that the easiest way to create a task related with concepts graph and algorithm are by using pictures.

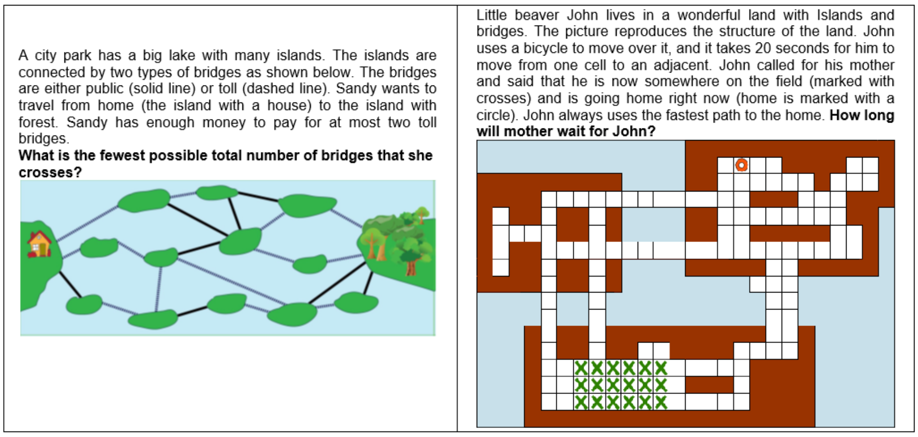


Figure 2: Task related with the graph concept from *Bebras* contest in 2014 (on the left) and task related with the algorithm concept from contest in 2013 (on the right)

12 *Bebras* tasks from 2012, 7 tasks from 2013, 6 tasks from 2014 are related with other than threshold concepts. For example, combinatorics, coding, data base. The question is whether the list of threshold concepts (investigated in 2005) is the best? Maybe we need to refresh it by doing new research and by asking other experts? It is possible to think that e.g. combinatorics, could be included to the list of threshold concepts and so on?

At the end of analysis of this *Bebras* tasks it is very important to remember that the purpose of *Bebras* contest is to introduce advanced informatics concepts to young school students, but not to test students' knowledge from the curriculum. To solve the *Bebras* tasks, students need to think about various aspect of computer science (information, discrete structures, computation, data visualization, data processing), but they also must use algorithmic as well as programming concepts.

CONCLUSION AND DISCUSSIONS

In conclusion, we have presented an exploratory analysis of the tasks of the *Bebras* contest regarding the threshold concepts in computer science. In the progress of the investigation we have found that about 23 percent tasks of every set for each age group are related with concept algorithm. It can be explainable by viewpoint that concept algorithm is very wide and can to cover critical thinking. In the other hand, about 26 percent tasks of every set for each age group are related with concept graph. We consider that it is related with criteria for creating *Bebras* tasks. The tasks related with the graph theory are more attractive and almost always are presented with pictures.

This research was the first attempt to view to the Bebras contest as the opportunity to use it like a tool for learning threshold concepts. In the future works, we would like to explore and offer the modified list of threshold concepts for each grade of students.

REFERENCES

Barradell, S., Peseta, T. (2014). Promise and challenge of identifying threshold concepts: a cautionary account of using transactional curriculum inquiry. *Journal of Further and Higher Education*. DOI:10.1080/0309877X.2014.971105.

Bebras. International Contest on Informatics and Computer Literacy. http://bebras.org/?q=countries, accessed February 23, 2015.

Boustedt, J. Eckerdal, A., McCartney, R., Moström, J.E., Ratcliffe, M., Sanders, K., Zander, C. (2007). Threshold concepts in computer science: do they exist and are they useful? ACM SIGCSE Bulletin, 39(1), 504-508.

Dagienė V. (2008). The BEBRAS Contest on Informatics and Computer Literacy – Students' Drive to Science Education. Joint Open and Working IFIP Conference. ICT and Learning for the Next Generation, Kuala Lumpur, Malaysia, 214–223.

Dagienė, V., Futschek, G. (2008). Bebras international contest on informatics and computer literacy: Criteria for good tasks. *LNCS*, Vol. 5090, 19-30.

Dagienė, V., Stupurienė, G. (2014). Informatics Education based on Solving Attractive Tasks through a Contest. Proceedings of IFIP Conference, 51-62.

Eckerdal, A., McCartney, R., Moström, J.E., Ratcliffe, M., Sanders, K., Zander, C. (2006). Putting Threshold Concepts into Context in Computer Science Education. ACM SIGCSE Bulletin, 38(3), 103-107.

Grover, S., Pea, R., (2013) Computational Thinking in K–12: A Review of the State of the Field*. Educational Researcher*, 42(1), 38–43.

ISO 1087-1.2000. Terminology Work—Vocabulary—Part 1: Theory and Application. International Organization for Standardization, Geneva, Switzerland.

Meyer, J. H. F., Land, R. (2003). Threshold Concepts and Troublesome Knowledge: Linkages to Ways of Thinking and Practising within the Disciplines. OCSLD, 412-424.

Shinners-Kennedy. D.,Fincher, S. (2013). Identifying Threshold Concepts: From Dead End to a New Direction. ICER '13, 9-18.

Tomcsányiová, M., Kabátová, M. (2013) Categorization of pictures in tasks of the bebras Contest. Berlin: Springer, 184–195.

Zendler, A., Spannagel, C. (2008). Empirical Foundation of Central Concepts for Computer Science Education. *ACM Journal on Educational Resources in Computing*, 8(2).

Eglė Jasutė is a scientist in the Vilnius University Institute of Mathematics and Informatics at the Department of Informatics Methodology. She is a teacher of informatics and mathematics in Vilnius Jesuit High School too. She is interested in innovation technologies for teaching mathematics and informatics in lower and upper secondary school.

Gabrielė Stupurienė is a PhD student in the Vilnius University Institute of Mathematics and Informatics at the Department of Informatics Methodology. Since 2010 she has been working on Informatics contest *Bebras* tasks. She has developed and defended her Master thesis "Conceptualisation of Informatics Fundamentals through Tasks" in 2011.

Motivation and disengagement of online students in remote laboratories

Zdena Lustigova, zdena.liotta@gmail.com
Charles University in Prague, Czech Republic

Pavel Brom
Charles University in Prague, Czech Republic

Abstract:

To achieve effective learning, motivational aspects like engagement play a very important role. Within online learning applications disengagement detection and prediction based on real data (not always in real time) is becoming more and more popular among educational specialists. Many E-learning systems, and virtual or remote learning environments, could be improved by tracking students' disengagement that, in turn, would allow personalized interventions at appropriate times in order to re-engage students.

The present article describes the results of a medium-scale (number of students N = 56) study of log files from the Open Remote Laboratory at Charles University in Prague, Faculty of Mathematics and Physics, to observe students' behaviour during their work in a virtual environment (spring 2011). Log files analysis and simple data mining and text mining techniques were used to reveal individual user's behavioural patterns and to detect disengagement and its reasons.

The results were used mainly to improve the systems' adaptability to students' requirements and to prevent their disengagement.

Keywords:

Log files analysis, disengagement detection, remote laboratories, educational data mining, behavioural patterns

1. INTRODUCTION

At the end of last century, E-learning and online learning (also referred to as web-based education) started to generate large amounts of information describing teaching-learning interactions. This information is endlessly generated, easily and ubiquitously available but rarely processed. At the beginning the variety and amount of data from learning and teaching process was often seen as a blessing: plenty of information readily available just a click away. Equally it could be seen as an exponentially growing nightmare, in which unstructured information chokes the educational system without providing any articulate knowledge to its actors.(Castro 2007)

Educational Data Mining was born to deal with problems like this. As a field of research, it is almost contemporary to e-learning. It is, though, rather difficult to define. Not because of its intrinsic complexity, but because data mining has most of its roots in the ever-shifting world of business. At its most detailed, it can be understood not just as a collection of data analysis methods, but as a data analysis process that encompasses anything from data understanding, pre-processing and modelling to process evaluation and implementati. (Chapman 2000).

1.1 Motivation and engagement

Engagement and disengagement are usually presented as a part of motivation. In effective learning, motivational aspects play a very important role. In relation to motivation, engagement is just one aspect indicating that, for one reason or another, the person is motivated to do the activity he/she is engaged in, or the other way, if the person is disengaged, he/she may not motivated to do the activity; in other words, engagement is an indicator of motivation. Many motivation theories focus on the amount of motivation, with higher levels said to result in improved outcomes. However, as educators we should not focus on generating increased motivation from our learners but instead focus on creating conditions that facilitate the internalization of motivation from within our learners (Koedinger 2008).

Several concepts are used in motivational research, besides motivation itself: engagement, interest, effort, focus of attention, self-efficacy, confidence etc. These concepts are related though not identical to engagement (Pintrich and Schunk, 2002).

1.2 Engagement and disengagement in online learning

Several efforts to detect motivational aspects from online learners' actions are reported in the literature. However, almost all these efforts are concentrated on Intelligent Tutoring Systems (Walonoski 2006), problem solving environments (Beck 2005, Johns 2006 and others) or online content-delivery systems (Cocea, Weiblezahl, 2007, 2011).

1.3 Disengagement detection in remote laboratories

Our research was focused mainly on disengagement detection and prediction within remote laboratory activities. Remote laboratories represent one of the three commonly used laboratory landscapes, together with so called virtual labs (also known under the name simulated labs) and computer-mediated, hands-on labs.

Remote labs enable experimenting and lab work in virtual conditions and with the use of remote access. Although this work is often done in environments and conditions for recent generations of students unimaginable, the main goals of laboratory work are still the same. Nowadays students have also to master their basic science concepts, to understand the role of direct observation, to distinguish between inferences based on theory and the outcomes of experiments, to cooperate and to develop collaborative learning skills. But they have to do all this being exposed to uncertain and not exactly defined situations, since the whole virtual and remotely controlled working environment is more complicated and thus more unpredictable (Lustig, Lustigova et al. 2009. This brings also more and more unpredictability for the teacher (or online supervisor) and also places greater demands on the analyst and remote lab developers, who themselves have often grown up and learned in different conditions.

Educational research within remote labs conditions has to deal with higher fuzziness and unpredictability. While in e-learning or online learning environment researchers have at their disposal plenty of structured and unstructured textual information, including discussion threads, all kinds of communication between teacher and student, student-student, student-team of students, student – learning material (in form of personalized comments, reviews, etc.), in remote labs the situation is different. The remote lab communication tools are very limited and the whole work is usually task oriented: to setup the experimental environment, to gather data and to process them. If there is a team work and the negotiation connected, it is observable directly, at place (see Lustig, Lustigova 2011).

Remote laboratory environments offer communication tools like chats, discussion clubs or cafés, whether synchronous or asynchronous, very rarely. This means, that there is virtually no textual information available and the researchers often have to work just with log files.

Within the latest “state of art” literature review focused on remote laboratories, we did not find any study based on log files analysis. It follows that log file data from remote laboratories is more often collected than analysed. Most of research papers in the field are focused on remote experiments development, online access improvement and other technical and engineering aspects of the problem. Studies of users' behaviour and learning process are quite rare and often based on direct (at place) observation, results and reports discussion, or survey data (Lustigova et al, 2009.

Since virtual and remote laboratories are increasingly used in both formal and informal education, there is a need to extend this research to encompass these types of systems as well. The interaction in these systems is less constrained and structured compared with problem-solving environments or intelligent tutoring systems. This poses several difficulties to an automatic analysis of learners' activities.

To address this challenge, we restricted our research to one motivational aspect, disengagement, and

1/ looked at identifying the relevant learners' behavioural patterns to be used for its prediction and

2/ looked for system setup and conditions (system behavioural pattern) when it occurs to adapt it to reduce attrition

3/ looked for mixed conditions and patterns to target disengaged learners with some hints or system help.

Our main goal during processing log file data from students' activity was not only to reveal disengagement but to prevent such a situation by improving the online learning and measuring environment. We researched mainly to avoid objective causes of disengagement, such as unnecessarily long wait for the event or feedback, confusing information and instructions or other problems, that cannot be easily identified with the use of traditional techniques.

We also wanted to discover behavioural and problem solving patterns.

2. METHODS AND SAMPLE

2.1 Task definition and sampling

Within our research we processed data from log files, collected in the spring and summer 2012 at a remote laboratory belonging to Charles University in Prague, Faculty of Mathematics and Physics.

The remote laboratory at Charles University in Prague belongs to so called “open remote laboratories”, which means that the local laboratory through a remote control option is available to any visitor, who is interested. In the spring and early summer 2011 the most engaged were students of 5 secondary schools (N=56, age 17), who were asked by their teachers, involved in the experiment, to measure and process data from a photo effect experiment and report the results as a part of their school laboratory work.

2.2. Specifics of our online laboratory

Unlike many remote laboratories, the laboratory at Charles University offers quite favourable conditions for high school students. The impression of the real presence is emphasized by installed web cameras that provide real time image transmission of the most interesting parts of selected experimental setup or its results. Simultaneously, different variables are measured and visualized in a form of graphs (see figure 3). Also the buttons for remote setup and control are user friendly and do not require knowledge of technical terms.

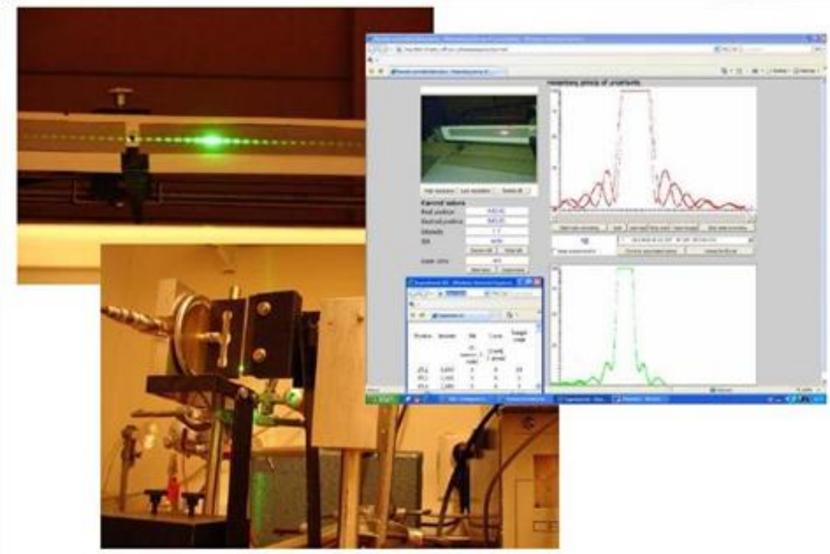


Figure 1: Example of a remote experiment: Fraunhoffer's diffraction in remote laboratory at Charles University in Prague: image transfer (results and experimental setup), gathered data (light intensity) visualisation and buttons for remote setup and control (Lustigova, Lustig 2009)

2.3 Data processing methods

As a first step the huge log files data were pre-processed to create small files (e.g. 24 MB to 38 kB (csv) or 100 kB (xls).

IP	Date	Time	Method:Circumstance A(0)=Alone/W(n)=Waiting for n users	Total	ID
88.102.115.32	11.4.2011	14:04:51	0:W(1)*D	122	ID(85)
88.102.115.32	11.4.2011	14:06:53	0:W(2)*D	4	ID(88)
88.102.115.32	11.4.2011	14:13:33	0:A(0):Q*D	1659	ID(89)
88.102.115.32	11.4.2011	14:15:29	0:W(1)*D	1543	ID(90)
88.102.115.32	11.4.2011	14:30:32	0:W(2){20}*Pd(1){11}*Pd(1){22}*Pd(1){17}*Pd(1)*D	84	ID(91)
88.102.115.32	11.4.2011	14:32:01	0:W(2){21}*Pd(1){8}*Pd(1)*D	852	ID(92)
88.102.115.32	11.4.2011	14:33:53	0:W(3)*D	8	ID(93)
88.102.115.32	11.4.2011	14:35:25	0:W(3)*D	12	ID(94)
88.102.115.32	11.4.2011	14:38:20	0:W(3)*D	11	ID(95)
88.102.115.32	11.4.2011	20:32:14	0:W(1)*D	2	ID(115)
88.102.115.32	11.4.2011	20:35:01	0:W(2){100}*Pd(1)*D	1628	ID(117)
88.102.115.32	11.4.2011	21:40:18	0:A(0)*D	57	ID(124)
88.102.115.32	11.4.2011	21:41:22	0:A(0):Q*D	667	ID(125)
88.102.115.32	12.4.2011	6:35:54	0:A(0)*D	213	ID(129)
88.102.115.32	18.4.2011	20:52:54	0:W(1){14}*Pd(1)*D	382	ID(20)
...					
81.25.16.87	11.4.2011	20:16:39	0:A(0):Q*D	458	ID(112)
81.25.16.87	11.4.2011	20:20:56	0:W(1):Q:Q:Q*D	971	ID(113)
81.25.16.87	11.4.2011	20:37:07	1:W(2)*D	68	ID(119)
81.25.16.87	11.4.2011	20:38:15	0:W(2){1749}*Sl1{2}*Sl0{4}*Sl1{3}*Sr(100){9}*Mv(-450.16){0}*Mv	2447	ID(120)
81.25.16.87	11.4.2011	20:47:39	1:W(3)*D	3	ID(121)
81.25.16.87	11.4.2011	21:01:01	0:W(3)*D	3	ID(122)
81.25.16.87	11.4.2011	21:19:03	1:W(1)*D	45	ID(123)
81.25.16.87	17.4.2011	18:37:29	0:W(1){88}*Sv1{23}*Sv1{10}*Sr(100){71}*Sl1{1}*Sl0{4}*Sl1{7}*Mv	1035	ID(4)
81.25.16.87	17.4.2011	18:37:34	0:W(2)*D	113	ID(5)

Figure 2: A part of the log file data sorted with respect to IP address, date and time of users' sessions, concerning the experiment topic "Photoelectric effect".

Each particular record contains a string, describing individual user activity, without losing any information (see an example of an individual user activity recorded in a form of a string below).

```
81.25.16.87     17.4.201118:37:29  1035 s     ID(4)

0:W(1){88}*Sv1{23}*Sv1{10}*Sr(100){71}*Sl1{1}*Sl0{4}*Sl1{7}*Mv(-12.16){0}*Mv(-445.85){0}
*Mv(-477.93){0}*Mv(-1000.00){1}*Mv(-1000.00){4}*Ma0{160}*Sf(0){1}*Sf(1){3}*Sf(0){10}*Ma1{46}
*Pr(1){9}*Ma1{43}*Sf(1){3}*Ma1{43}*Sf(2){3}*Ma1{44}*Sf(3){3}*Ma1{42}*Sf(4){3}*Ma1{43}*Sf(5){8}
*Sl0{5}*Ma1{44}*Ps(1){0}*Pd(1){12}*Pd(1)*D
```

Figure 3: The example of an individual user's activity string, derived from the log file

While the first line in the figure above identifies the user's computer IP address, the date and time he started to measure, the whole time in seconds his activities (connection) lasted, and the original ID in log file under which we can find original data, the second long line contains the full description of user activities. The legend follows:

Method or the experiment variant selected by the user (at the beginning of the session record):	
0:	Volt-ampere characteristics of a vacuum phototube.
1:	Charging a capacitor up to the stopping voltage.
Circumstances (introduced by ":" delimiter character)	
:A(0)	User was alone. He/she was not limited by time, just by the 3-minutes timeout.
:W(n)	User was not alone, his serial number was n+1. He had to wait when n users before him have finished their sessions, as a maximum n x 12 minutes. The parameter n is specified in each session record.
:Q	During users work another user asked for an access to this experiment. Thus the queue was noticed and remaining time of the current user was limited to 12 minutes.
End of the users work (at the end of the session record)	
*X	Server broke down.
*D	The user closed the browser window or used the standard way to disconnect.
*T	The control of the experimental setup was interrupted by the server due to the user's inactivity. The timeout is 3 minutes.
Codes describing the main user activity with the classification (introduced by the delimiter character "*") The classification is specified by the first capital character.	
*S…	The user **setup up** the experimental settings or changed existing settings – **depends on the topic**, e.g.:
*Sv1	The user switched the mercury lamp on.
*Sv0	The user switched it off.
*Sl1	The user switched the other source of light on.
*Sl0	The user switched it off.
*Sf(m)	The user selected the interference filter specified by the parameter m (available filter wavelengths: 0=365nm, 1=405nm, 2=435nm, 3=546nm, 4=568nm, 5=shielded phototube).
*Sr(x)	The user set the amplification resistor specified by the parameter x
*M…	The user is performing the measurement –e.g.:
*Ma1	The user started the automatized measurement.
*Ma0	The user stopped the automatized measurement prematurely by the stop button before it had been finished in standard way.
*Mv(u)	The user changed the voltage manually, the new value is specified by the parameter u.
*Mc	The user started the capacitor charging (the other available variant of the experiment topic).
*Md	The user discharged the capacitor.

Figure 4: Example of a legend for all recordable items of user's activity and their classification

3. RESULTS

3.1 Descriptive statistics

From the collection of 613 sessions within the first half of 2011, just 155 belonged to the our group (April 2011) and from that number just 15 sessions (students) completedthe whole task (e.g. Measurement or data downloading).

The length of the connections (sessions) changes from very short to very long (up to one hour). But surprisingly the length of the connection says nothing about the meaningfulness of the activities. Some short connections finished with data downloading, while some very long connections contain absolutely no activity (see histogram of connection length, notice that time axis is nonlinear). The average length of any connection was 354.7 seconds, while the average length of meaningful connection (connection finished with data download or measurement) was 756.2 seconds.

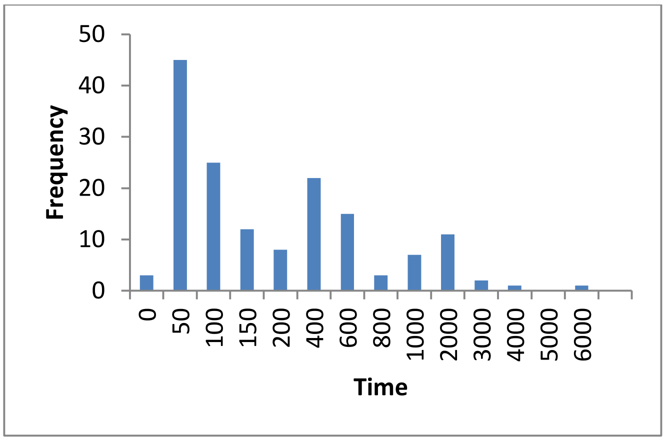


Figure 5: Time duration of an individual user connection (absolute frequencies histogram)

Our experimental group users connected from 43 different IP addresses. The users preferred to work in late afternoons and evenings. We noticed that some of these secondary school students worked after midnight as well.

If we define a session as a chronological series of a connections from a defined IP address within the same day and setup the interconnection "no activity interval" up to 15 minutes (900 s), the number of sessions decreases to 56. Since the number of participants in our experimental group was slightly higher, it gives us evidence that some of them were not able or did not want to work within the remote lab environment.

3.2 Disengagement detection

The number of connections, where the user was alone, apparatus ready to measure, but he/she disconnected after a while for unknown reason, is surprisingly high (108). It is even higher than the number of connections finished because of the necessity to wait (28). All connections that finished with meaningful activities were the "wait" connections: waiting for one user (3 minutes) 9 connections, waiting for two users (6 minutes) 6 connections. Our hypothesis that users are disengaged because of waiting in queue was not verified.

3.3 Behavioural patterns

Individual user observation/examples of different behavioural patterns:

1/ User A (IP: 88.102...) connected to the remote experiment repeatedly and had to wait in a queue (W). Finally he/she downloaded someone else's (User B's) data (Pd).

2/ User B (IP: 81.25...) on 11/4/2011 (see figure 3) first explored the volt-ampere characteristics of a vacuum phototube. On 17/4/2011 user B had to wait in a queue, but after 88 s of waiting user B took control of the remote experimental setup, explored the interface and after a short time of playing at the beginning he started with systematic measurement activities afterwards. User B spent 1035 s (i.e. 17 min) performing the remote measurement with data acquisition.

3/ The sessions recorded under IP address 81.25.16.87 (Fig 5) from April 11 2011 inform us about different behavioural pattern. This whole session lasted approximately 63 minutes. The user spent 2447 seconds (i.e. approximately 40 minutes) with playing all buttons and measuring. He/she started at about 8 p.m. and luckily was alone. But he did not use the occasion. After while (waiting for 2) he/she took the control and started to work. The activity record, presented in the log file (see figure 6), belongs to the longest ones, but surprisingly has no real output.

```
1:W(2){1749}-Sl1{2}-Sl0{4}-Sl1{3}-Sr(100){9}-Mv(-450.160156){0}-Mv(-650.160156){1}-Mv(-759.062500){0}-Mv(-
775.125000){0}-Mv(-1000.000000){1}-Mv(-1000.000000){2}-Ma1{1}-Sf(0){26}-Sf(1){8}-Sf(2){4}-Sf(2){4}-Sf(1){1}-
Ma1{3}-Ma0{7}-Sf(0){11}-Sl1{9}-Sf(0){4}-Ma1{9}-Mv(-566.316406){0}-Mv(-542.222656){0}-Mv(-293.253906){0}-Mv(-
285.222656){1}-Mv(260.894531){0}-Mv(260.894531){1}-Mv(268.925781){0}-Mv(276.957031){0}-Mv(903.390625){0}-
Mv(919.453125){2}-Mv(967.640625){0}-Mv(983.703125){0}-Mv(999.765625){0}-Mv(999.765625){6}-Mv(-
237.039063){0}-Mv(501.832031){2}-Ma0{0}-Mv(999.765625){0}-Mv(999.765625){3}-Mv(991.734375){0}-
Mv(991.734375){1}-Mv(999.765625){0}-Mv(999.765625){2}-Ma0{2}-Ma0{1}-Mv(-700.000000){1}-Mv(-
200.000000){0}-Mv(200.000000){1}-Mv(349.238281){0}-Mv(397.425781){0}-Mv(758.828125){0}-Mv(782.921875){1}-
Mv(999.765625){6}-Mv(999.765625){6}-Sf(1){4}-Sf(2){1}-Pr(28){2}-Sf(0){2}-Sf(5){3}-Sl0{3}-Sl1{36}-Sf(1){84}-
Sf(1){1}-Sf(0){3}-Sf(1){23}-Sf(0){49}-Sf(0){1}-Sf(0){2}-Sv0{4}-Sl0{5}-Sv1{1}-Sf(1){31}-Sr(10){2}-Sr(100){9}-Sf(0){1}-
Sf(3){4}-Sv0{0}-Sv1{81}-Pd(22){1}-Pd(26){2}-Pd(28){4}-Pd(21){6}-Pd(23){7}-Pd(13){3}-Pd(14){5}-Pd(14){28}-
Pd(14){37}-Pd(22){4}-Pd(21){5}-Pd(19){3}-Pd(17){6}-Pd(18){19}-Sf(1){11}-Sl1{3}-Sl0{4}-Sl1{2}-Sl0{2}-Sl1{2}-Sl0{2}-
Sl1{15}-Sl1{1}-Sl0{2}-Sl1{2}-Sl0{2}-Sl1{2}-Sl0{2}-Sl1{2}-Sl0{1}-Sl1{2}-Sl0{2}-Sl1{2}-Sl0{2}-Sl1{2}-Sl0{2}-Sl1{1}-
Sl0{5}-Sl1{2}-Sl0{2}-Sl1{7}-Sl0{2}-Sl1{3}-Sl0{2}-Sl1{1}-Ps(28){0}-Pd(28){4}-Sv1{9}-Sf(1){6}-Sl0{1}-Sl1{2}-Sl0{2}-
Sl1{2}-Sl0{2}-Sl1{2}-Sl0{2}-Sl1{1}-Sl0{2}-Sl1{2}-Sl0{2}-Sl1{2}-Sl0{2}-Sl1{3}-Sl1{1}-Sl0{2}-Sl1{89}-Sl0{1}-Sl1{2}-
Sl0{3}-Sl1{10}-Sl0{1}-Sl1{2}-Sl0{2}-Sl1{4}-Sl0{1}-Sl1{2}-Sl0{1}-Sl1{31}-Sl0{1}-Sl1{2}-Sl0{2}-Sl1{3}-Sl0{3}-Sl1{16}-
Sf(1){1}-Sf(4){1}-Sf(5){5}-Sf(4){19}-Pd(13){10}-Sf(4)-D
```

Figure 6: Example of an activity record when the user might have been confused by the user interface or unsure with the assignment itself. He/she just played with all control elements

3.4 Users preferences

Since our remote laboratory interface offers both, 1/ to run experiments in real time and 2/ to provide asynchronous interaction with premeasured data, we decided to check users' preferences. Contrary to Corter, Nickerson, at al.'s study (2007), where they refused real-time interaction and forced fixed scheduling on the students, we decided to offer both, real time measurement and pre-measured data. Premeasured data available within our laboratory remote experiments do not bring any additional information (no information about the author is available, except id, time and IP address). A preview in both numerical and graphical ways is available.

The results were as follows: "Early birds" students, who followed the recommended time schedule, preferred real time measurement (app ¼ within each group), while those "last minute" students, queueing to operate remote lab devices, frequently used pre-measured data, often without checking their quality and reliability. They used them even if their own data were in fact of a higher quality.

Although the remote lab offers up to 200 stored data sets, the users in the experimental group usually selected from the last 3 offers.

4. CONCLUSIONS

Although the students from the experimental group presented nicely processed reports, the reality hidden in log files was different. Drawing on educational data mining techniques, we revealed, that:

1. Although our remote laboratory is open to individual secondary school students, the overwhelming majority of them are not able to practice in the laboratory without meaningful training. If they are forced to do so, they leave the environment without any meaningful activity or they play for a while, but then also prefer data withdrawal to the real measurement.

2. The “play phase” seems to be very important. Just those, who played for a while, were able to setup the apparatus, to start the measurement, to finish it correctly and to save the measured data. But finally, even these students mostly preferred data download.

3. The credibility of pre-measured data (it doesn’t matter what they look like and who their author is) is very high.

4. Students do not trust to their own results. It might be associated with the learning and teaching paradigm change in general (teamwork x individual work), lack of supervision; they are not used to, and/or increased uncertainty in the virtual environment.

5. REFERENCES

Beck, J. (2005). Engagement Tracing: Using Response Times to Model Student Disengagement. In C. Looi et al., (Eds.), *Artificial Intelligence in Education: Supporting Learning through Intelligent and Socially Informed Technology*, 88-95, IOS Press.

Castro, F., Vellido, A., Nebot, A., & Mugica, F. (2007). Applying Data Mining Techniques to e-Learning Problems. Springerhttp://sci2s.ugr.es/keel/pdf/specific/capitulo/ApplyingDataMiningTechniques.pdf

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). CRIPS-DM 1.0 Step by Step Data Mining Guide. CRISP-DM Consortium.

Cocea, M., & Woibelzahl, S. (2007). Log files analysis for disengagement detection in e/learning environments. Available from: http://link.springer.com/article/10.1007%2Fs11257-009-9065-5#page-1

Cocea, M., & Weibelzah S. (2011). Disengagement Detection in Online Learning: Validation Studies and Perspectives*. IEE transaction on leasing technologies. 4*(2).

Corter, J. E., Nickerson, J. V., Esche, S. K. Chassapis, C. Im, S., & Ma, J. (2007). "Constructing Reality: A study of remote, hands-on and simulated laboratories", *ACM Transactions on Computer Human Interaction 14*(2).

Deci, E.L., & Ryan, R.M. (2000). The 'What' and 'Why' of Goal Pursuits: Human Needs and the Self-Determination of Behaviour. *Psychological Inquiry. 11*(4).

Johns, J., & Woolf, B. (2006). A Dynamic Mixture Model to Detect Student Motivation and Proficiency. *Proceedings of 21st National Conf. Artificial Intelligence* (AAAI-06).

Koedinger, K., Cunningham, K., Skogsholm, A., & Leber, B. (2008). An open repository and analysis tools for fine-grained, longitudinal learner data. *Proc. 1st Int. Conf. Educ. Data Mining*, Montreal, QC, Canada, 157–166.

Lustigova, Z. & Lustig, F. (2008). New e-learning environments for teaching and learning science. *Learning to live in the knowledge society*. Milan, Italy: Springer.

Lustigova, Z., Lustig, F., Mechlova, E., & Malcik, M. (2009). A New E-learning Strategy for Cognition of the Real World. *The New Educational Review*, *17*(1).
Lustigova, Z., & Lustig, F. (2009). Remote and Open Laboratory in Science Education - Technological, Educational and Psychological Issues. *Proceedings of the ITI 2009 31st International Conference on Information Technology Interfaces*. Cavtat. Croatia.
Lustigova, Z., Dufresne, A., & Courtemanche, F. (2010). Acquiring Physiological Data for Automated Educational Feedback in Virtual Learning Environments. *New Educational Review*, *21*(2).
Ma, J., & Nickerson, J. V. (2006). Hands-on, simulated, and remote laboratories. *ACM Computing Surveys, 38*(3), 7-Es.
Pintrich, P., & Schunk, D. (2002). *Motivation in education: Theory, research, and applications*. Englewood Cliffs, N.J.: Merrill.
Walonoski, J., & Heffernan, N.T. (2006). Detection and Analysis of Off-Task Gaming Behaviour in Intelligent Tutoring Systems. In M. Ikeda, K. Ashley & T.W. Chan, (Eds.), Proc. Eighth Int'l Conf. *Intelligent Tutoring Systems*, 382-391.

Zdena Lustigova is the head of the Laboratory of Distance Education, Faculty of Mathematics and Physics, Charles University in Prague, Czech Republic. Her educational work and especially her research and development effort have been concentrated on neurophysiologic aspects of e-learning and computer aided teaching and learning.

Pavel Brom is a PhD student lecturer at Faculty of Mathematics and Physics, Charles University In Prague Czech Republic. His educational work and especially his research and development effort are focused on remote laboratories and the use of ICT in science education.

Experiences of primary school in-service teachers undertaking on-line professional learning sessions in Kenyan coastal county

Gioko Maina *gioko@agakhanacademies.org*
Aga Khan Academy, Mombasa

Rosemary Waga B Ed. *rwaga@akesk.org*
Aga Khan Primary, Mombasa

Abstract

Online professional Learning (OPL) has become an option to access profession learning (PL) due to limited opportunities in the developing world. With the Ministry of Education Science and Technology and the in-service teachers service commission demanding for PL for learner to maintain their teaching licenses in Kenya the demand for PL is high against the limited seat and offering as well as lack of time or study leave for the PL. The paper explores the experience of the in-service teachers undertaking OPL to understand their experiences so as to offer information to what the learner goes through. The research was undertaken to learner who had been given opportunities for OPL in a professional development centre. Data collection was through mobile based interviews through the WhatsApp platform. 80 in-service teachers were involved with responses received from 55 in-service teachers. The platform allowed follow-up question through voice, text and video response. The responses were analysed, coded and emerging themes identified and discussed against literature. The finding bought out the following as challenges attitude, competency, connectivity, electrical power, resources, support and time. The learner comes out with innovative ways of addressing the challenges so as to access OPL. The finding confirms the challenges of OPL in developing world according to literature. The innovative approaches to manage the challenges provide an insight to OPL providers, educators and designers on the experiences of the in-service teachers. The learner will derive maximum benefits of they are supported to overcome the challenges.

Keywords

Professional Learning, Online, Training

INTRODUCTION

The education landscape is changing in Kenya with Registration of Examination candidates, checking of National Examination results and access of pays slips and salary statement for in-service teachers is all done online. This exemplifies the embracing of technology by the education sector. On the other hand the Ministry of education science and technology (MoEST) and Teacher Service commission (TSC) has offered a directive for in-service teachers to undertake professional learning (PL). This directive has taken a serious angle with a threat to deregister in-service teachers or to demote headteachers who have not attended PL in two years. This has made in-service teachers and school leaders to seek and undertake PL. Over the last few decades there has been increased use of computing devices in educational institutions in developing countries (Trucano, Hawkins & Iglesias, 2012). The use of the web, computer, and mobile-based technologies has drawn a lot of interest among in-service teachers, who use them for educational purposes as well as for social networking. This at least implies a degree of familiarity with these technologies and the skills for using those (Trucano et al., 2012).

One challenge has been the availability of PL sessions as the few that are available can only accommodate a small percentage of the in-service teachers. Out of a population of 40 Million people Kenya has 171301 in-service teachers at primary schools and 48.087 at secondary schools. OPL also has the potential to offer access to professional learning to a large number of in-service teachers (Karim & Hashim, 2004). Economic challenges have pronounced the difficulty of paying for PL sessions. OPL has started to emerge in many developing countries where it has the potential to help meet an increasing demand for education and address the growing decline of trained in-service teachers (UNESCO, 2006). This has led to in-service teachers who have access to internet to explore online professional learning (OPL) opportunities.

The exploration of OPL is an opportunity to offer PL to in-service teachers who are able to access. OPL offers great flexibility in learning (Kocur & Kosc, 2009). This flexibility provided by the various forms in which the learning material can be presented, allows the in-service teachers a variety of options to learn from at their own pace and time. However this opportunity comes with its challenges. Knowing the challenges the in-service teachers are undergoing will offer insights to professional leaning providers, trainers and other stakeholders on how best to support and facilitate the in-service teachers in OPL. This has led to the question “What are the experiences of educators undertaking online professional learning session in a Kenyan county? The focus was in-service teachers undertaking OPL facilitated by professional development centre.

SIGNIFICANCE

Positive learning climate and performance expectations affect learner satisfaction, and performance expectations provide the greatest contribution (total effect) to learning satisfaction. Educators will hold positive attitudes towards OPL if they recognize that it would help them improve their learning and teaching effectiveness and efficiency (Rahamat, Shah, Din, Puteh, Aziz, Norman, 2012). Chen and Huang stated that understanding learner attitudes can help expand OPL system functions and meet learner needs, which should further increase the impact of learning and enhance satisfaction with the learning process. Aixia and Wang (2011) found that the vast majority of in-service teachers who were satisfied with an OPL environment held positive believes and attitudes towards it; perceived satisfaction was identified as one of four factors that helped explain 83.8% of the variance of learner attitude. According to Rhema and Miliszewska (2014) OPL and its potential benefits for developing countries have been discussed in the literature, research on user perspectives of OPL in those countries is limited. Given this background to the current situation in Kenya’s educational system, it’s imperative to explore the challenges posed to a successful implementation.

LITERATURE REVIEW

As indicated by the UNESCO report, the gains from the adoption and implementation of OPL are many and varied among which are the flexibility in learning (Kocur & Kosc, 2009), lower cost compared to on-campus presence, ability to absorb the increasing number of learner, availability of re-usable content, more avenues for human development, increased educational opportunities, among others. Moreover the application of OPL in developing countries has gradually advanced in recent years with an improved availability of Internet connections, local area networks, and IT support (Omidinia, Masrom, & Selamat, 2011). These advantages are however faced with some challenges that are fundamental and critical to a successful implementation in a developing country. It must be noted however that these challenges are not alien to developed economies (Andersson & Grönlund, 2009). Notable among these challenges are infrastructural, technological, financial and institutional support.

Bhuasiri, Xaymoungkhoun, Zo, Rho, and Ciganek (2012) found that in developing countries the most significant factors were related to increasing technology awareness and improving attitude toward OPL, enhancing basic technology knowledge and skills, improving learning content, requiring computer training, motivating users to utilize OPL systems, and requiring a high level of support. Moreover in-service teacher's characteristics are regarded as a critical success factor in OPL in developing countries (Bhuasiri et al., 2012). These characteristics include computer self-efficacy, Internet self-efficacy, computer experience, Internet experience, computer anxiety, and attitudes toward OPL (Chu & Chu, 2010).

In-service teacher's attitudes towards OPL have been identified as critical to the success of OPL (Zhang & Bhattacharyya, 2008). Attitudes are influenced by the quality and perceived ease of use of OPL courses, functionality of OPL platforms, and the level of computer skills (Aixia & Wang, 2011). Omidinia, Masrom, & Selamat (2011) identified learner attitudes as a factor that determined how OPL was adopted. In-service teacher's computer experience including perceived self-efficacy, enjoyment, and usefulness of using OPL also played a role (Liaw & Huang, 2011). According to Selim (2007) the development of positive attitude is realise by the use of the devices by people who are familiar with web technologies and the skills needed to use computer and mobile devices. In turn, positive in-service teacher's attitudes and behaviours towards OPL are critical to their OPL readiness and acceptance (Lim, Hong, & Tan, 2008). In-service teachers' attitudes and beliefs towards OPL, as well as their satisfaction with technology and past OPL experiences are regarded as success determinants of future OPL initiatives.

In-service teacher levels of access to technologies represent an initial factor that would shape their attitudes towards OPL, and their willingness to use it; the availability of reliable ICTs and the convenience of accessing these technologies reflect learner attitudes toward OPL (Rhema & Miliszewska, 2014). Having access to technology in OPL is evidently an enabling or disabling factor. Access here implies the physical access to a computer, an internet connection, the reliability of the connection and bandwidth, as may be needed to access the full range of the content needed (Burn & Thongprasert, 2005). Sweeney and Geer (2010) found that limited access to ICT constrains learner capabilities, attitudes and experiences. According to Hussain (2007), In-service teachers selected for a study on OPL indicated that they faced many difficulties in accessing ICT facilities and this limited their ability to use technologies. Accessibility of technology tends to affect learner and instructor attitudes and competencies and correlates positively with the level of technology use (Agyei & Voogt, 2011). The in-service teachers' ability to use ICTs was significantly hindered by the low level of technology access (Hussain 2007). Many research studies identified correlations between positive computer experience and positive attitudes, competence and comfort with computers (Papaioannou & Charalambous, 2011, Paris, 2004) and an inverse relationship between computer experience and computer anxiety (Olatoye, 2011). Other studies disagreed with these findings and claimed that computer experience did not play a significant role in reducing computer anxiety or developing positive computer attitudes (Felton, 2006).

The amount of time in-service teachers have to and want to commit plays an important role in the success of OPL implementation (Agyei & Voogt, 2011). When several activities compete for the attention of the learner, without prioritization and discipline, very little can be realized from an OPL programme. Furthermore, Agyei and Voogt conclude that without making time for studies reading materials, answering test questions, engaging in group assignments among others due to other equally important activities like job and home requirements, the expected impact will not be felt. This would create dissatisfaction, high dropout rates, and a rippling effect of discouraging potential in-service teachers.

The academic confidence of a learner can predict the success or failure of a learner in an OPL (Simpson, 2004). Andersson (2008) describe that previous academic experience and qualifications can best describe in-service teachers' performance. Where a learner's self-efficacy, which is the learner's confidence in his or her ability to successfully complete a course, is high, the potential for impacting positively on the success of an OPL implementation can be positive. Where the self-efficacy is low, it can result in difficulties in the implementation.

In-service teachers also need to have the necessary computer skills and feel confident in the use of computers (Agyei & Voogt, 2011). The lack of these skills can be a hindrance to learning, especially for in-service teachers who are entirely new to computers as computer confidence accounts for much of the predictive power of good achievements. Many in-service teachers in Kenya have either not been introduced to computers or have a difficult time grasping the concepts and skills due to many factors like no access to computers, little time spent using computers due to the number of in-service teachers wanting to access them. Where this is the case, implementing OPL systems become a challenging task both for the implementers and the users.

A stable and supportive study environment affect OPL to a very large extent and some research suggest that this is perhaps the most important factor influencing drop out and retention (Andersson & Grönlund, 2009). Social support can be about the time, and help in-service teachers get from family and friends, and employers for those working. The very nature of OPL that suggests self-study, either in part or whole shows that a conducive environment devoid of distractions and full of support from the home and work place is required. Where there is little understanding of OPL and its contributions to personal, organizational and national development, the denial of this support can negatively affect its implementation. There are claims that contact or intervention from the institution and support from the tutor or other staff improve learning and pass rates (Andersson, 2008). A low level support provision would discourage many people from using an OPL system.

METHODOLOGY

An analysis of relationships between learner attitudes towards OPL and their demographic characteristics, access to technology, use of technology for learning, skill in technology, and satisfaction with technology were included. The target group was in-service teachers who were facilitated to access resources and undertake OPL. OPL is a blended model of offering Professional Learning Session where concepts are taught face to face and complimented with online learning, For example teaching comprehension to early years was complimented by an online course from NovoEd which was offered after the session was taught. The whole certificate course was 378 hours over the year and 60 hours out of that was allocated for online learning. The approach was 20% offered by the facilitators to supplement, 30% offered by the facilitators as self-study and 50 % options selected by the course participants. Mobile qualitative research used to collect data. This was easy and comfortable for the participants as it involved video, audio and text responses. This technique was used to reach in-service teachers who were in a nationwide strike and it had the possibility of follower up.

The study aimed to explore the experiences of learner on online professional learning sessions. The study probed deeply and analysed intensively the multifarious phenomenon that constituted OPL (Cohen & Manion, 2000). OPL was approached from a dialectical perspective meaning both the person and the environment were taken into consideration (Valsiner & Van de Veer, 2000). Data is strong in reality; hence the study examined an instant action within a bounded system (Cohen & Manion, 2000;). This involved OPL. The study was done to in-service teachers who have been facilitated to undertake OPL either by the course design or facilities support because it afforded an excellent opportunity to discover phenomena within the real world settings in which they occurred (Merriam, 1998).

The study was also explorative in nature, the first goal sought to seek information from the learner's experiences on OPL, benefits and constraints to develop a holistic understanding of the case. A set of open ended questions were used at the beginning. An evolving working hypothesis emerged (Merriam, 1998); new questions were asked based on the responses and inferences suggested. In the open interviews, relevant themes, descriptors and categories were identified through interpretation of raw data (Jansen, 2010). The open ended questions were posted on the in-service teachers WhatsApp page. The teachers were sent a prompted question in text, audio and video. The teacher would respond in the mode favourable to them. Semi structured, open-ended questions were used; (Wellington, 2000) asserts that in semi structured questions, the interviewer was flexible to ask questions in any order. The in-service teachers responded and the researcher asked follow-up question or clarification to get a deeper response to the questions. According to Hammersley-Fletcher, 2002 repeat interviews allows for refocusing and development of new questions based on the responses offered. The questions adopted a positive stance, with the main prompt to explore the experiences in OPL. The prompting was directed to derive the challenges educators were experiencing and how they were managing them. The series of questioning was uploaded or transcribed depending on data capture until the researchers were satisfied of the depth of the responses.

As naturalistic research continued, it was uncommon to change direction and ask new questions (Merriam, 1998). The study was an enquiry, in real life context, where deep investigation involved interdependence of parts and the patterns that emerged (Yin, 2009). In this study, the researcher explored a single phenomenon, which is contemporary with real-life context (Yin , 2009). Therefore, the study was a qualitative, instrumental, exploratory and interpretive case study.

The responses were transferred from WhatsApp to email. This allowed the extraction from email to a word processor. The video and audio responses were transcribed and all the text data were organized per aspect and tabulated. The tabulated data were posted on a spread sheet for coding. The codes began with two categories of challenges and approach to overcome the challenges. The second level of coding was based on the description of the response. The final coding was on the nature if the description. The responses were then sorted based on code 2 followed by code 1 and code 3. This allowed a deeper analysis of the responses base on the emerging codes.

FINDINGS AND DISCUSSION

One of the challenges was on attitude. The challenge on attitude focused on perspective and support (Zhang & Bhattacharyya, 2008). The in-service teachers felt that they did not have the required skills or were not supported adequately by their peers and facilitators (Selim, 2007). Furthermore some of the in-service teachers had thought that virtual environment was not effective in learning (Aixia & Wang, 2011). To overcome the attitude in-service teachers had to change their perspective by reflecting on the possible benefits of OPL.

The other challenge was on competency which brought out access, content, knowledge, skills and techniques. According to one respondent "…Insufficient skills to navigate the net, faulty machines that leads to inefficiency, lack training and resources, uncooperative skilful colleagues /facilitators who are never willing to assist ...." 6 January 2015. In-service teachers had difficulties in connecting, navigating and handling the computer while others felt that the online programs were difficult to comprehend (Burn & Thongprasert, 2005,Sweeney & Geer, 2010). The in-service teachers felt that they needed further training on how to operate the computer. To overcome this challenge the in-service teachers sought extra help from their peers and facilitator's necessary skills and techniques (Andersson, 2008).

The lack of competency influenced the learner's confidence (Simpson, 2004.). The learner could not open the laptops or press the keys as they thought they would tamper with the laptops (Rhema & Miliszewska, 2014). The attitude of them being "analogue" impacted negatively on their confidence (Agyei & Voogt, 2011). To overcome the challenge in-service teachers gained courage and confidence by using the computer for an extended period (Hussain, 2007).

Connectivity was a big challenge. Connectivity was seen through lack of access, lack of connectivity, cost, reliability, resources, instability and slow speed (Burn & Thongprasert, 2005). In-service teachers lacked internet at their homes and places of work so they had to travel all the way to the Academy to access OPL. Some went to the internet cafes so as to do the course, internet was also unreliable and slow so the in-service teachers could not be consistent with their work and most of the deadlines were not met. "I only manage to finish 1 course...because of network problem...thigh I can assess a laptop network was the biggest challenge.....to overcome that challenge I had to get a modem which had to use airtime..."noted a respondent, 6 Jan 2015. Some in-service teachers had no access to modems so doing the course outside the Academy was challenging. To overcome these challenge the in-service teachers decided to access the Academy where internet access was paid for and laptops are availed. At the Academy there were peers who assisted. The in-service teachers managed challenges by maximally utilizing the net when it was strong, and used other connections for example guest Wi-Fi. Other in-service teachers used a modem or visited cybercafé to overcome this challenge.

Although OPL was free there were costs involved in travelling to the Academy. This made it expensive, limiting and costly. In-service teachers had some materials to download and found it costly because they used modems which needed airtime and also use airtime on their phones to enable them complete assignments on the course, sometimes the bundles got exhausted before the in-service teachers were through with the assignments or when downloading videos or documents, to overcome this the in-service teachers opted to come to the Academy to gain free access to the net incurring the travelling cost.

Some courses were not offering certificates so in-service teachers felt that there was no need of doing such courses. The learner considered the certificate as a credibility indicator for their professional learning (Andersson & Grönlund, 2009). In some cases it was more of the certificate than the skills that they gained. This is because during promotion the certificates would be considered an indicator of professional growth. To manage the certificate issues the organiser only bookmarked OPL's offering certificates.

The other challenge was power fluctuation, blackouts, disruptions or poor connections (Hussain, 2007). This interrupted their pacing during the course. To overcome these challenges some in-service teachers ensured that their laptops were fully charged at all times, others used stabilizers or visited the Academy which had a standby generator.

Resources played a major part in such as access, availability, cost, reliability and support. Some in-service teachers had no laptops of their own, nor smart phones and the ones that were at the Academy were not enough for everyone, they were on first come first serve basis. Every time the in-service teachers came they were given different laptops this made it impossible to gain access to the work that was saved earlier (Papaioannou & Charalambous, 2011). Some in-service teachers had to purchase smart phones to enable them complete the course. To overcome this challenge some in-service teachers had to take loans to buy laptops or smart phones or saved work on the clouds, others saved to buy PC's.

Support was cited as a major challenge in terms of access, competency, delay, lack, quality, accommodation, availability, collaboration, differentiation and motivation. In-service teachers lacked support in terms of online facilitators not responding in good time when asked questions regarding the course, they also felt that it was hard to interact with the online facilitators. Some in-service teachers did not have transport to the Academy "…It's only that I sometime had to miss for some days due sometimes lack fare …" Respondent, 6 Jan 2015". In-service teachers also felt that they were at different skill level hence needed a differentiated approach to the course. To overcome these challenges they asked the facilitators to differentiate the course. They also sought help from their peers. The learner demanded support that responded to their needs. The in-service teachers compared the benefits of OPL against the costs involved and sought for funds.

Time presented a challenge to the in-service teachers due to alignment, deadlines, management, and quantity (Agyei & Voogt, 2011). The timings of the programmes were sometime not in line with in-service teachers schedules. This made the in-service teachers to miss deadlines. Most in-service teachers had issues with time management. They felt the time was short, limited, and inadequate or time consuming. This could have been a time management issue or it could have been due to the fact that they were also doing a full time job. The in-service teachers tried to manage this challenge by improving on their time management, aligned their schedule and being creative to access more time. This was done by creating time, being disciplined, scheduling work and requesting for more time.

A summary of the unique challenges brought a distribution as shown in figure 1. Connectivity, time and competency took prominence. Support and Resources had a mild presence. Cost, attitude, power and credibility was mentioned sparsely as a challenge.

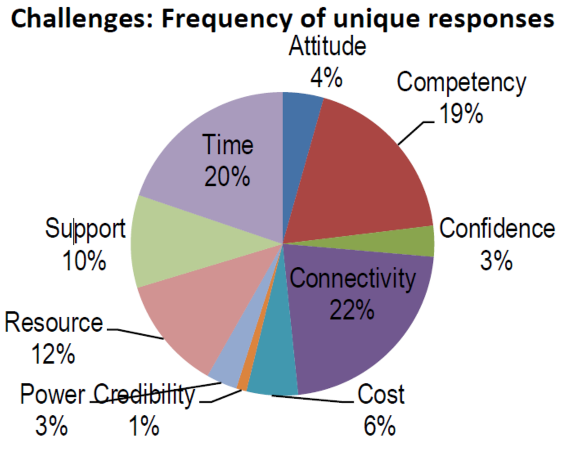


Figure 1 Distribution of Unique challenges

The teachers despite the challenges explore ways of overcoming the challenges. Table 2 presents the distribution of the unique solution implemented by the teachers.

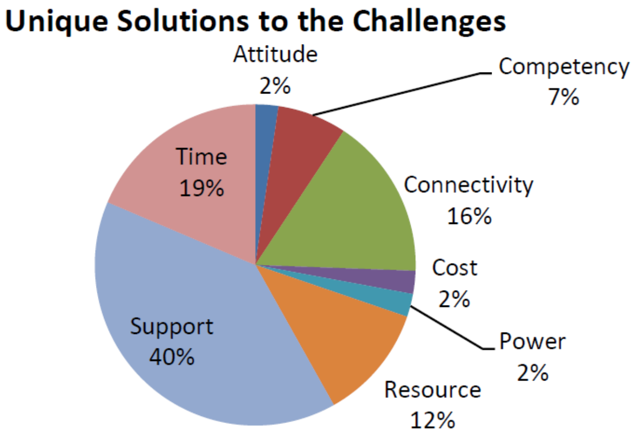


Figure 2 Distribution of Unique Solutions

Support took prominence as an aspect that the teachers sought to overcome the challenges. Time, connectivity and resources were solutions which we mentioned more after the support. Competency, attitude, cost and power were solutions that were least varied in the solutions that the teachers sought for help.

The challenges identified are very similar to the theoretical findings. The difference in the composition of the challenges is due to the contextual realities. The innovative responses to the challenges were driven by the need to access and complete OPL. The igniting aspects was the benefits offered by OPL and the weighting of the certificate earned. The innovative approaches were self-driven by the in-service teachers themselves. It is important to note the importance of intrinsic motivation as most in-service teachers try to find extrinsic motivation. More impactful approaches are contextually designed solutions to the challenges.

CONCLUSION

The findings in the Kenyan context are very similar to the literature findings. The approaches to overcome the challenges provide insights of what the in-service teachers undergo as they try to get the best out of OPL. Attitude, electrical power competency, connectivity, resources, support and time are critical aspects in OPL. OPL educators, designers and providers need to address these challenges if the OPL is to be beneficial to in-service teachers. The innovative approaches devised by the in-service teachers to address the challenge provide insights on how the OPL educators, designers and providers can support the in-service teachers using contextual solutions. The challenges and the innovative solutions the in-service teachers go through can be used during the design and implementation of the OPL learning environment to enhance access and utility effectively.

REFERENCES

Agyei, D. D., & Voogt, J. M. (2011). Exploring the potential of the will, skill, tool model in Ghana: Predicting prospective and practicing in-service teachers' use of technology. *Computers & Education, 56*, 91-100.

Aixia, D., & Wang, D. (2011). Factors influencing learner attitudes toward e-learning and development of e-learning environment based on the integrated e-learning platform. *International Journal of e-Education, e-Business, e-Management and e-Learning, 1*(3), 264-268.

Andersson, A. (2008). Seven Major Challenges for e-learning in Developing Countries: Case Study eBIT, Sri Lanka. *International Journal of Education and Development using ICT, 4*(3).

Andersson, A., & Grönlund, A. (2009). “A Conceptual Framework in Developing Countries: A Critical Review of Research Challenges,. *EJISDC, 38*(8), 1-16.
Bhuasiri, W., Xaymoungkhoun, O., Zo, H., Rho, J. J., & Ciganek, A. P. (2012). Critical success factors for e-learning in developing countries: A comparative analysis between ICT experts and faculty. Com-puters & Education, 58 (2012),. *58*(2012), 843-855.
Burn, J., & Thongprasert, N. (2005). A Culture-based model for strategic implementation of virtual education delivery. *International Journal of Education and Development using ICT, 1*(1).
Chu, R. J., & Chu, A. Z. (2010). Multi-level analysis of peer support, internet self-efficacy and e-learning outcomes – the contextual effects of collectivism and group potency. . *Computers & Education, 55* (1), 145-154.
Cohen, L., & Manion, L. (2000). *Research methods in education.* London: Routledge Falmer.
Felton, F. S. (2006). *The use of computers by primary school principals .* PhD Thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.
Hammersley-Fletcher, L. (2002). Becoming a subject leader: What’s in a name? Subject leadership in english primary schools. . *School Leadership and Management, 22*(4), 407–420.
Hussain, I. (2007). A study of student’s attitude towards virtual education in Pakistan. *Turkish Journal of Distance Learning, 8* (2), 69-79.
Jansen, H. (2010). The Logic of Qualitative Survey Research and its Position in the Field of Social Research Methods . *Forum: Qualitative Social Research, 11*(2).
Karim, M., & Hashim, Y. (2004). The Experience of e-learning Implementation at the Universiti Pendidikan Sultan Idris, Malaysia. *Malaysian Online Journal of Instructional Technology, 1*(1), 50-59.
Kocur, D., & Kosc, P. (2009). E-learning Implementation in Higher Education. *Acta_Electrotechnica et Informatica, 9*(1), 20-26.
Liaw, S. S., & Huang, H. M. (2011). A study of investigating in-service teachers’ attitudes toward e-learning. *2011 5th International Conference on Distance Learning and Education. 12*, pp. 28-32. Singapore: IACSIT Press.
Merriam, B. S. (1998). *Case study research in education. A qualitative approach.* San Francisco, CA: Jossey-Bass.
Olatoye, R. (2011). Levels of participation in ICT training programmes, computer anxiety and ICT utilization among selected professionals. *International Journal of Education and Development using Information and Communication Technology, 7*(2), 15-26.
Omidinia, S., Masrom, M., & Selamat, H. (2011). Review of e-learning and ICT infrastructure in developing countries (Case study of Iran). *American Journal of Economics and Business Administration, 3*(1), 120-125.
Pagram, P., & Pagram, J. (2006). Issues in e-learning: A Thai Case Study. *The Electronic Journal of Information Systems in Developing Countries, 26*(6), 1-8.
Papaioannou, P., & Charalambous, K. (2011). Principals’ attitudes towards ICT and their perceptions about the factors that facilitate or inhibit ICT integration in primary schools of Cyprus. . *Journal of Information Technology Education, 10*, 349-369.
Paris, P. G. (2004). E-learning: A study on secondary students’ attitudes towards online web assisted learn-ing. *International Education Journal, 5*(1), 98-112.
Rahamat, R., Shah, P. M., Din, R., Puteh, S. N., Aziz, J. A., Norman, H., et al. (2012). Measuring in-service teachers’ perceived satisfaction towards e-learning material and environment. *WSEAS Transactions on Advances in Engineering Education, 3*(9), 72-83.
Rhema, A., & Miliszewska, I. .. (2014). Analysis of student attitudes towards e-learning: The case of engineering students in Libya. . *Issues in Informing Science and Information Technology, 11*, 169-190.

Selim, H. M. (2007). Critical success factors for e-learning acceptance: Confirmatory factor models. *Computers and Education, 49* (2007), 396–413.
Simpson, O. (2004.). The impact on retention of interventions to support distance learning students. *Open Learning, 19*(1).
Sweeney, T., & Geer, R. (2010). Student capabilities and attitudes towards ICT in the early years. . *Australian Educational Computing,, 25*(2010), 18-24.
Trucano, M., Hawkins, R., & Iglesias, C. J. (2012). *Ten trends in technology use in education in developing countries that you may not have heard about. A World Bank Blog on ICT use in Education*. Retrieved 2014 29, December, from Edutech: http://blogs.worldbank.org/edutech/some-more-trends
UNESCO. (2006). *in-service teachers amd Educational Quality: Monitoring Global Needs for 2015.* UNESCO.
Valsiner, J., & Van de Veer, R. (2000). *The social mind. Construction of the idea. .* Cambridge UK: Cambridge university Press.
Wellington, J. (2000). *Educational research: Contemporary issues and practical approaches.* London: Continuum.
Yin, Y. K. (2009). *Case Study Research: Design and Methods.* Thousands Oaks, CA: Sage.
Zhang, P., & Bhattacharyya, S. (2008). Students' views of a learning management system: A longitudinal qualitative study. *The Communications of the Association for Information Systems, 23*(2008), 351-374.

Gioko Maina is Vice Principal Aga Khan Academy Mombasa. And Project Manage with SESEA programs. Gioko is the Head of the Professional Development Centre which provide professional learning to in-service teachers within the academy and through outreach. The programmes under the centre are Professional Learning for Educators Series, Educating Girls in Science, Teacher Preparation Programme and Learning Tool Kit

Rosemary Waga is SESEA Literacy and Numeracy Coach Aga Khan Education Service, Mombasa. Rosemary supports practicum in-service teachers and mentor in-service teachers as they work together. She also evaluates the learning environments to explore how best it supports teaching, learning and assessment. She is a member of English in-service teachers Language association and a Master trainer and Ambassador with the Learning Tool Kit

Gamification in business process management for a mobile contet: Industry and Higher Education Viewpoint

Nicholas Mavengere and Mikko Ruohonen
University of Tampere, Finland
{nicholas.mavengere, mikko.j.ruohonen} @uta.fi

Abstract

Current business environment characterized by factors, such as, globalization and technology advances makes companies constantly improve their business processes. Therefore, business process Management (BPM) is increasing gaining prominence because of the need by companies to understand and optimize their business processes. In fact, companies are basing their strategic advantages on improved and innovative business processes. Because of this BPM is critical to companies and academia is constantly also contributing to evolvement of BPM. Business Process Management (BPM) is a holistic approach to management, which is gaining popularity and is considered as key for enabling business evolution. Over the years, there have been broad initiatives and measures taken in both academia and practice to enhance BPM. In this research position paper we seek to engage gamification. We seek to develop a novel e-learning technology driven by gamification on mobile technologies.

1. Introduction

Business process management (BPM) is increasing gaining prominence because of the need by companies to understand and optimize their business processes. In fact, strategic advantages are gained by companies based on their efforts in developing business processes. Moreover, business strategy is enforced by processes which align company resources in line with set objectives. Because of this BPM is critical to companies and academia is also contributing to evolvement of BPM. Business Process Management (BPM) is a holistic approach to management, which is gaining popularity and is considered as key for enabling business evolution In general terms, "Business Process Management (BPM) is a management approach and methodology leveraged to address strategic business issues by allowing businesses to grow or operate in a cost effective manner with desired visibility, flexibility and speed" (Ujvagi, 2014). Some of the drivers of rapid growth in BPM include globalization, regulations, technological changes, stakeholders' actions and erosion of business boundaries (Lee and Dale, 1998). Over the years, there have been broad initiatives and measures taken in both academia and practice to enhance BPM. In this research position paper we seek to engage gamification as explained below. We seek to develop a novel e-learning technology driven by gamification on mobile technologies.

Deterding et al. (2011, 9) noted that "the idea of using game design elements in non-game contexts to motivate and increase user activity and retention has rapidly gained traction in interaction design". They noted that this idea, termed "gamification" has rapidly been adopted in several applications in education, health, finance, productivity, sustainability, entertainment and news media. In this research seek to initiate the application of gamification in BPM. That is, to adopt gamification in both industry and academia as a tool to advance BPM. "Gamification is an informal umbrella term for the use of game elements in non-gaming systems to improve user experience (UX) and user engagement" (Neeli, 2012). Gamification involves the use of game mechanics in non-game circumstances and in this case, learning of BPM in both academics and practice. Hamari et al. (2014) suggested the growing interest in gamification in an academic context by noting the growing number of published papers and thus argued that gamification has become a popular subject for academic inquiry.

Moreover, mobile technologies environment is in focus of this research because of the growing adoption of mobile devices and advances in mobile technologies. In doing so, this research also seeks to integrate learning processes between classroom/formal and mobile/informal/workplace learning. This paper makes an attempt to highlight what has been covered in literature and done in industry in line with gamification in to promote BPM learning. Therefore this research seeks to establish plausible industry oriented future research on the application of gamification on BPM. Hence the objectives and guidelines of potential results of this research paper are:

- Literature review on gamification in BPM incorporating mobile technologies,
- based on the literature review, notes on what has been covered and done in practices on the research topic and,
- Based on the two previous inquiries, highlighting plausible future research areas

Technology enhanced learning is considered as a key for the future education and lifelong learning. Gamified approaches to learning are considered as highly potential future technologies. Mobile apps business is one of the fastest growing industries.

This paper is comprised of five sections. Next section is the background which includes an elaboration of the key concepts of the research. Then gamification in BPM is highlighted in Section 3. The past related empirical works are noted in Section 5 before the last section which highlights future possible research areas.

1. Background

In this section we will precisely explained the key terms of the research, such as, business process management, gamification and the integration of informal and formal learning. We will do that in highlighting the research context and application of these terms in the research.

2.1 Business Process Management (BPM)

Balzert et al. (2012) noted the current and ongoing processoriented intense debate in academia and Balzert et al. (2012) noted the current and ongoing process-oriented intense debate in academia and industry because of the dynamic business environment which includes technological advances. The changes in the environment make it critical to constantly improve the business processes to take advantages of, for example, new innovations and practices. Therefore, this research seeks to make an input on BPM by engaging gamification as a technique to foster the learning process. There is constant pressure on the BPM community, that is, employees, management, students and researchers to learn the ever present BPM improvements. Thus, this research seeks to reduce this pressure by offers tools which make learning easier and fun. This is supported in the literature, for example, Balzert et al. (2012) argues that learning management is critical success factor in the current dynamic environment.

Processes are integral for organizations in the dynamic business environment. Thus making process management a crucial issue in industry and there has been broad efforts in academia to research on BPM. There are several issues which needs to be addressed in BPM. For example, Wensley (2003, 217) noted some of the issues as;

- Process modelling approaches that facilitate the integration of knowledge and knowledge management

- Research on and approaches to facilitating the integration of knowledge within and between organizations and, increasingly, between cultures
- The part technology can play in enabling and supporting knowledge and process management
- Approaches to integrating the visioning and revisioning of organizational strategies and process perspectives
- Research and practice relating to analyzing, designing, implementing, and managing interorganizational processes.

BPM involves the main aspects of business operations which brings value and these are governed by rules, Lee and Dale (1998, 216) proposed the following:

- Major activities have to be properly mapped and documented
- BPM creates a focus on customers through horizontal linkages between key activities
- BPM relies on systems and documented procedures to ensure discipline, consistency and repeatability of quality performance
- BPM relies on measurement activity to assess the performance of each individual process, set targets and deliver output levels which can meet corporate objectives
- BPM has to be based on a continuous approach to optimisation through problem solving and reaping out extra benefits
- BPM has to be inspired by best practices to ensure that superior competitiveness is achieved

2.2 Gamification

In simple terms, gamification could be described as the use of game techniques and mechanics to influence the people involved. This is essential in order to motivate people to undertake tasks which could be boring and challenging. In business settings gamification is useful, for instance, for business improvement managers to make use to make involvement more engaging and fun. “Gamification is an informal umbrella term for the use of game elements in non-gaming systems to improve user experience (UX) and user engagement” (Neeli, 2012). In clarifying Deterding et al. (2011, p13) suggested gamification to refer to:

- the use (rather than the extension) of design (rather than game-based technology or other game-related practices)
- elements (rather than full-fledged games)
- characteristic for games (rather than play or playfulness)
- in non-game contexts (regardless of specific usage intentions, contexts, or media of implementation).

This definition of gamification differentiates it from games and positions it in relation to gaming and playing as illustrated in Figure 1 below.

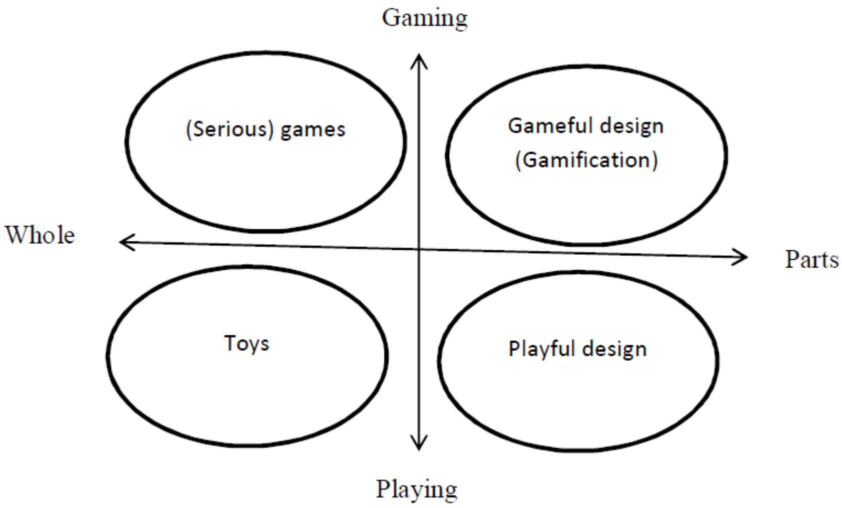


Figure 1. Gamification between game and play, whole and parts
(Deterding et al., 2011)

Basically in simple terms, gamification is the use of the enjoyable playing aspects of the game for work purposes. This seeks to make learning of new work processes easier and faster. This is important because of the challenges to motivate employees to learn the constantly changing BPM environment. Gartner (2011) highlighted "the goals of gamification are to achieve higher levels of engagement, change behaviors and stimulate innovation. The opportunities for businesses are great – from having more engaged customers, to crowdsourcing innovation or improving employee performance". It has been argued in past research, for example Czikszentmihalyi (1991), that there are certain conditions that as to be fulfilled for effective apply gamification, such as, a balance between the employee skills and challenges at hand as illustrated in Figure 2.

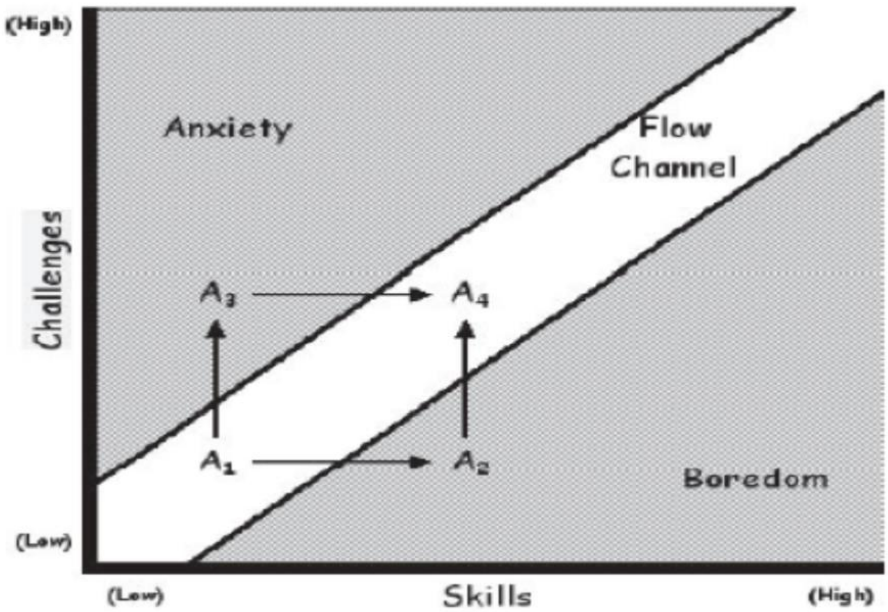


Figure 2. Skills and challenges level for successful
gamification application (Neeli, 2012)

Boredom is derived from tasks that are not challenging, that is, not requiring excessive time and thought to complete and hence players loose interest. On the contrary, a task that is too hard frustrates the players and they also lose interest. Hence, as the skills level improves so should, simultaneously be the level of challenges as illustrates in Figure 2. Therefore, it is important to match the skills level of employees to level of challenges of tasks. The flow channel illustrated in Figure 2 proposes the level of skills and challenges that should be provided to make the tasks interesting. Another option to keep employees in flow channel requires making the challenges tougher as an employee gets closer to the "goal" as this mechanism creates a sense of achievement and learning for the employee and also creates an impression to other employees that they are still in the "game" and have a chance of "win" (Neeli, 2012).

Gartner (2011) noted four principal means of driving engagement using gamification as;

1. Accelerated feedback cycles. In the real world, feedback loops are slow (e.g., annual performance appraisals) with long periods between milestones. Gamification increases the velocity of feedback loops to maintain engagement.
2. Clear goals and rules of play. In the real world, where goals are fuzzy and rules selectively applied, gamification provides clear goals and well-defined rules of play to ensure players feel empowered to achieve goals.
3. A compelling narrative. While real-world activities are rarely compelling, gamification builds a narrative that engages players to participate and achieve the goals of the activity.
4. Tasks that are challenging but achievable. While there is no shortage of challenges in the real world, they tend to be large and long-term. Gamification provides many short-term, achievable goals to maintain engagement.

2.3 Integration of learning processes between classroom and formal contexct with mobile, informal and workplace learning

Learning at different levels such as organizational, group and individual levels has been drastically changing in line with the changes in the global community. For example, technological advances especially mobile technologies and internet increased learning opportunities and channels. This led to many innovative learning solutions and models, for instance, massive online open courses (MOOCs). Balzert et al. (2012, p3642) argues that "the successful management of learning and knowledge has become a critical success factor for organizations in today's knowledge-intensive business world". Thus this research seeks to contribute to the learning process by adopting gamification as a tool to foster learning.

Companies need to adapt to the changing business environment, this is done by drawing lessons from the past and take predictive and responsive measures to the future. That is, a firm should act and react to the business environment. But how to companies do that? Balzert et al. (2012) argues the need for experienced and knowledgeable employees. Therefore, it is important for companies, in this case BPM organizations, to invest in the learning process of the employees and the organization as a whole. In fact, a continuous professional development which includes both formal and informal learning is advocated for in this research. Balzert et al. (2012) notes that undertaking formal learning alone is inadequate and also unproductive because separate daily working routine and learning initiatives.

3. Gamification in BPM

The use of game concept to foster learning has been widely recommended as Habgood and Ainsworth (2011) noted, “educational software has traditionally attempted to harness games as extrinsic motivation by using them as a sugar coating for learning content”. The use of gamification to foster BPM adoption has been suggested in literature (Liyakasa, 2012 - INSIGHT). However, before starting gamification for BPM it is important to clarify and set the objectives. This is important in order to use the right techniques, set achievable goals and collect relevant data in the process. There are several reasons gamification could be used for PBM such as:

Behavioral change – employees could be more engaged and enjoy the business processes thus making them work harder and at the same time enjoying their work productively.

- Social aspects – gamification makes BPM seem as fun and thus encouraging employees to interact in sharing experience and competing. This also offers platform to gather feedback.
- New entrants training – gamification is a great way for employees to understand new positions. In addition, it could be used to test after training sessions rather than rigid and unpleasant formal assessments.
- Data collection – critical data could be collected for business process improvement initiatives.
- Business process managers could get data from both the BPM systems as well as feedback from participants which is useful for process improvements.
- Change management – new processes and systems adoption in organization is most of the cases a difficult exercise and thus gamification could facilitate such events.

(PROCESSpedia, 2014)

Gartner (2011) noted that the full potential of gamification for BPM still remains relatively untapped but there are potential opportunities to be seized. They reported three key findings derived from the design of a gamification systems namely (Mak, 2012);

- Use of appropriate game mechanics – BPM challenges, for example, creating a culture of continuous improvements could be tackled by using appropriate game technics and mechanics.
- Quantifiable measurements – BPM quantifiable measurements for work activities ensures real-time feedback useful for management and employees.
- Strategy – gamification could be used as a strategy to achieve set business goals

However, we would like to emphasize that gamification is not an easy and quick solution to all BPM challenges. Moreover applying gamification to BPM should not be taken for granted. In fact, poor application of gamification may not fulfill desired process improvements and business outcomes.Factors to be considered ((Mak, 2012) in introducing gamification for a BPM environment include

1. Define process objectives, metrics and desired outcomes – after clarifying the behavioral change targeted by the gamification it is important to set goals and objectives to be done to achieve the change effort. When the targets are clearly defined the BPM and gamification designers needs to include the appropriate game mechanics and tools to achieve the set learning objectives. In doing so, designers need to understand what the tools are, how they work and how to map required game technique to the correct action for the task. At this initial step, it is important to verify if gamification is the correct procedure to engage in order to meet the set goals as it has been noted that not all processes require gamification.

2. Cultural awareness and context – Gamification strongly relates to the culture and context in which it is being applied. Thus it is very important that BPM and gamification designers are aware of the culture and context of the environment in which gamification will be applied. For example, organizational culture could be defined by different groups which are, for example, aggressive, collaborative, competitive or passive. And hence, gamification needs to incorporate the specific group culture.

3. Assign value to each task – To motivate adoption and continuity there is a need to assign value to each task or activity. The participants need to understand how an action will be rewarded. This is a delicate procedure as a poor reward system could have undesirable effect.

4. Plan for interactions and increasing the challenges – gamification should include plan for interaction and increasing the challenges in order to minimize fatigue and boredom. It should be noted that usually there are strengths, flaws and inconsistences with the first design which would need improved versions.

5. Find needed skills – there are several different skills required to setting up gamification, such as, user interface design, game design and implementation. It is important to involve all required skills in order to develop effective gamification.

4. Past related empirical research

There is vast and broad, in terms of focus, empirical studies on gamification (Carignan and Kennedy, 2013). Hamari et al. (2014) after conducting a literature review of empirical studies suggested that gamification provides positive effects but the effects depends on context and users. They suggested that context of gamification, that is, core services / activities being gamified, such as, commerce, learning, health/exercise, intra-organizational systems, sharing, innovation and data gathering has an effect on the result and gamification process. Moreover, the empirical value of gamification is emphasized, for instance, Gartner (2011) estimated that above 50 percent of organizations managing innovation processes will gamify the innovation processes by 2015. Gamification has been empirically used for learning purposes, for instance, Ibáñez et al. (2013) assessed the engagement appeal and learning effectiveness of a gamified learning activity. This is much related to the research this position paper is advocating for, but goes a step further by including the organizational learning, that is, industry perspective. Ibáñez et al. (2013) concluded that there are "positive effects on the engagement of students toward the gamified learning activities and a moderate improvement in learning outcomes". This is a motivational factor in engaging gamification to enhance BPM learning in different settings, these are, academia and industry.

5. Proposed future research areas

Hamari et al. (2014) correctly noted that understanding gamification effectiveness is a pertinent practical issue. This is because a large number of firms have invested large sums of money into gamification efforts. Moreover, there is a growing number of firms providing gamification services. Thus, gamification is discussed in several outlets, such as, industry chatter in wide ranging perspectives; these are, extremely positive and extremely negative. Therefore, it is noted in this research that there is practical demand for empirical results on the effectiveness of gamification. This research includes industry that will play a key role on providing the empirical industry perspective to the research. Specifically, this research includes empirical gamification for BPM improvement purposes because of the noted drivers of rapid growth in BPM, such as, globalization, regulations, technological changes, stakeholders' actions and erosion of business boundaries (Lee and Dale, 1998).

This study embraces the educational/learning aspects related to gamification. This is proposed as a future research focus because of the dynamic nature of the BPM subject. Hamari et al. (2014) from their review of empirical studies on gamification noted that “all of the studies in education/learning contexts considered the learning outcomes of gamification as mostly positive, for example, in terms of increased motivation and engagement in the learning tasks as well as enjoyment over them”. Therefore, this research seeks to engage gamification to enhance BPM learning both in industry and higher education settings.

The application of mobile technologies in different context, such as banking, educational and development, has advocated for in research and practice due to high level of mobile adoption, as well as, mobile technological advances. This research also seeks to take advantages of the benefits gained from with use of mobile applications. This gamification for BPM would also be considered in mobile environment.

The plausible outcomes of the proposed research includes

1. Business process modeling and simulation for a virtual organization
2. BPM Industry oriented and based gamification solutions
3. Publications in both academic and industr journals on gamification in BPM

The epected impacts include

- New learning approaches to be adopted by:
    - Higher education institutions (especially business/management schools)
    - Industry: organizational training using gamification for BPM
    - Consultants: individuals and BPM practitioner communities
- Enabler for lifelong learning
- Enabler for young adults to build BPM skills early
- Better integration and growth of BPM practitioner communities
- A new business started as a spin off from the project
- Readiness to commercialize prototped gaming products

REFERENCES

Balzert S., Fettke P. and Loos P. (2012) A Framework for Reflective Business Process Management, 45th Hawaii International Conference on System Sciences

Carignan, J., Kennedy S.L. (2013) Case Study: Identifying Gamification Opportunities in Sales Applications, DUXU/HCII 2013, Part II, LNCS 8013, pp. 501–507.

Csikszentmihalyi. M., (1991) “Flow: The psychology of optimal experience: Steps toward enhancing the quality of life”, Harper Collins Publishers.

Deterding S, Dixon D., Khaled R. and Nacke L. (2011) From Game Design Elements to Gamefulness: Defining “Gamification”, MindTrek’11 conference, Tampere, Finland 2011.

Gartner (2011) “Gartner says by 2015, more than 50 percent of organizations that manage innovation processes will gamify those processes”, http://www.gartner.com/newsroom/id/1629214 (checked July 05, 2014).

Habgood, MPJ & Ainsworth, SE (2011). Motivating children to learn effectively: exploring the value of intrinsic integration in educational games. Journal of the Learning Sciences, 20(2°, 169- 206.

Hamari. J, Koivisto J. and Sarsa, H. (2014) Does Gamification Work? — A Literature Review of Empirical Studies on Gamification, 47th Hawaii International Conference on System Sciences.

Ibáñez, M., Di-Serio A. and Delgado-Kloos C. (2013) Gamification for Engaging Computer Science Students in Learning Activities: A Case Study, IEEE Transactions on Learning Technologies.
Lee R.G. and Dale B.G. (1998) Business process management: a review and evaluation. BusinessProcess Management Journa 4(3), pp. 214-225.
Liyakasa, K. (2012) Game Mechanics could steer BPM adoption. Insight, CRM Magazine.
http://www.destinationcrm.com/Articles/Columns-Departments/Insight/Game-Mechanics-Could- Steer-BPM-Adoption-81613.aspx (checked 7 January 2015).
Mak, H. W. (2012) Gartner Report: Gamification Potential for Business Process Management
http://www.gamification.co/2012/09/24/gartner-bpm-gamification/ (checked, July 07, 2014).
Neeli, B.K. (2012) A Method to Engage Employees using Gamification in BPO Industry, The 3rd International Conference on Services in Emerging Markets (ICSEM). December 12-15, 2012.
PROCESSpedia (2014) What is Gamification http://processpedia.com.au/featurespage/gamification, (checked, July 02, 2014).
Ujvagi, T. (2014) Why Business Process Management, http://centricconsulting.com/why-businessprocess- management-bpm/ (Checked 7 January 2015).
Wensley, A. (2003) Business Process Management Revisited. Knowledge and Process Management 10(4), p 217.

Books

Kapp, Karl M. (2012). The Gamification of Learning and Instruction: Game-Based Methods and Strategies for Training and Education. Wiley & Sons.
Kapp, Karl M., Blair, Lucas and Mesch, Rich (2014). The Gamification of Learning and Instruction Fieldbook: Ideas into Practice. Wiley & Sons.
Werbach, Kevin and Hunter, Dan (2012) For the Win: How Game Thinking Can Revolutionize Your Business. Wharton Digital Press.

MeRV: A Scaffold to Promote Creating 2D Map of Method Call Structure in Block-based Programming Language

Takashi Ohata, *ohata@sakailab.info*
Shizuoka University, Graduate School of Informatics, Japan

Yoshiaki Matsuzawa, matsuzawa@inf.shizuoka.ac.jp
Shizuoka University, Faculty of Informatics, Japan

Sanshiro Sakai, sakai@inf.shizuoka.ac.jp
Shizuoka University, Faculty of Informatics, Japan

Abstract

Although Block-based language has been widely used in introductory programming education, current languages have little support for promoting understanding about method (function). In particular, current languages have only functionality for avoiding grammatical errors, whereas we think the visualization in the two-dimensional canvas has the potential to support high-level method structural design. In this research, we proposed the “method calling relationship visualizer” (MeRV) that visualizes the calling relationships between methods in Block-based programming language. We hypothesized that the following two effects were expected by this scaffold function: (1) The scaffold encourages learners’ clear understanding regarding method calling system. (2) The scaffold directs learners to create a two-dimensional map that clearly shows the entire method call structure. We conducted an empirical study in an introductory programming course for approximately one hundred non-CS undergraduate students. The results of qualitative analysis for the students’ outcomes clearly showed that the two hypotheses were supported.

Keywords

Programming education, Visual programming language, Block language, Java, Me thod

INTRODUCTION

In the past decade, visual programming languages have been used in introductory programming education. In particular, languages based on the building-block approach (Maloney, 2004) are the most popular visual programming style in this field such as Squeak (DAN, 1997) or Scratch (http://scratch.mit.edu), and we call these Block-based languages.

Block-based language is considered to work as a scaffold for learning the computational problem-solving skills called “computational thinking" (Wing, 2006) through programming activities. The students can focus on high-level design of programming because the Block-based language eliminates any grammatical errors. Matsuzawa et al., 2015 showed that the scaffolded programming skills in students were fading in a migration procedure from Block-based to text-based language (Matsuzawa, 2015).

However, Block-based languages currently proposed for the beginner of programming (Harvey, 2010; Scratch Team, 2009; Cooper, 2000; Google, 2011) have little support for assisting students in understanding method (or function). For example, although Build Your Own Blocks (Harvey, 2010), OpenBlocks (Roque, 2007), or BlockEditor(Matsuzawa, 2015) have a functionality to support creating methods, they are limited in their ability to avoid grammatical errors. Understanding how methods work in programming involves not only grammatical knowledge, but knowledge concerning how they can be designed, how the designed methods communicate with each other, and how each method is a part of the whole method structure. At this point, current Block-based languages are less powerful for supporting such high-level design activities, whereas some other visual programming methods have tried to use the two-dimensional space effectively for the design of method structure (e.g. Yahoo Pipes at https://pipes.yahoo.com/pipes/).

In this study, we propose a software environment which assists students in creating a two-dimensional map of method calling structure in a Block-based programming language. We have named it "method calling relationship visualizer" (MeRV), since it visualizes method calling relationships. We implemented it on a BlockEditor as an extension of the tool, and it was examined in an introductory programming course for approximately one hundred non-CS undergraduate students.

RELATED WORK

This research was built on the BlockEditor (Matsuzawa et al., 2015). BlockEditor offers mutual language translation between Block to Java. The authors claimed that as BlockEditor can translate between block and Java mutually, students would gradually migrate Block to Java on their own schedule. BlockEditor has a typical Block-based language user interface which supports basic capabilities in programming language such as branch, repeat, methods, or object-oriented. However, its functionality was limited in supporting grammatical problems.

AppInventor (http://appinventor.mit.edu/explore/) is a visual programming language that can define methods on the canvas as well as BlockEditor. The expression of its method definitions and calls are shown in **Figure 16**. In the figure, "toCloseScreen" method is defined and called by the "call CloseScreen" block. This example indicates that it is not easy for learners to trace program codes.

Harvey proposed "Build Your Own Blocks" which is implemented as an extension of Scratch, and has functionality for defining methods. Alice (Cooper, 2000) has a feature of creating procedures. Although there are no procedural blocks, it can create and manage procedures in the class tab. RAPTOR (Carlisle, 2009) is a flow chart based language and also creates and manages methods by tab. The functionalities that were implemented on those tools are limited in supporting grammatical problems as well.

Figure 16: Method definition at AppInventor

MERV：METHOD CALLING RELATIONSHIP VISUALIZER

BlockEditor

MeRV was implemented as an extension of the BlockEditor. BlockEditor has the capability of mutual language translation between Block to Java as shown in **Figure 17**. BlockEditor was developed using the OpenBlocks Framework (Roque, 2007). The mutual translation between Block to Java in BlockEditor covers enough grammatical elements for introductory programming such as branch, repeat, methods, or object-oriented, though it does not support all grammar in Java.

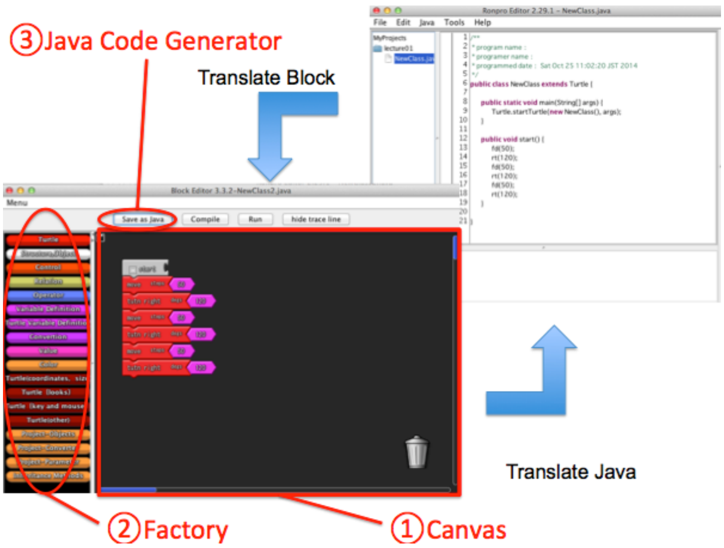


Figure 17: Overview of BlockEditor

MeRV

We developed the function to visualize relationships between method's callers and callees. We named the function MeRV (Method caller Relationship Visualizer). Our philosophical motivation behind this function is not only that the visualization makes students to trace program easily, but it also promotes the student's autonomy to layout methods to create a "map" which clearly shows the method calling structure.

The relationships between methods are drawn by a yellow arrow from caller to callee. A concrete example is shown in **Figure 3**. In the example, an arrow was drawn from the "drawTriangle" block in the "start" method to the "drawTriangle" method definition block. Students can understand a relationship between a caller and a callee in a straightforward visualization. A user can turn on and off the state of showing arrows by clicking a toggle button.

The color of the arrow turns gray when the method calling relationship is invalid. An example of this state is shown in **Figure 4**. In the example, a parameter has not been set for the method calling block. Students can easily find the method calling block which is invalid by watching the line color.

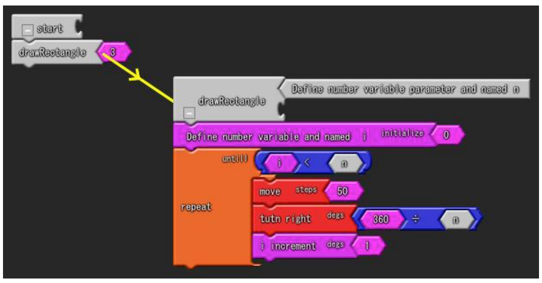


Figure 18: Method call relation

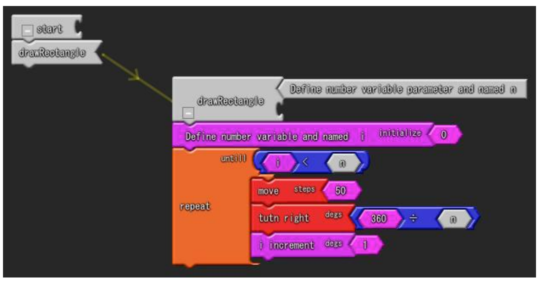


Figure 19: Illegal method call state

EXPERIMENTAL STUDY METHOD

Hypothesis

We conducted an experimental study for MeRV in our introductory programming course. The research question of the study was to confirm the effects of MeRV using the following two hypotheses:

Hypothesis I:

The scaffold encourages learners' clear understanding regarding the method calling system. Particularly, it clearly shows relationships between callers and callee, and students are able to traverse a method call structure easily.

Hypothesis II:

The scaffold encourages learners to create a two-dimensional map which clearly shows the entire method call structure. Students manually make a layout of blocks on the map which involves highly cognitive work, but it would be done by students with their personal motivation.

Questionnaire

Hypothesis I was examined by a questionnaire survey. The students indicated that by using MeRV they could recognize merits of methods by defining method and laying out their own program; it was especially easy to understand logical structure by using methods.

Analysis Method of the Block Layout

Hypothesis II was examined by analyzing layouts of the blocks of students' block programs. In this section we describe the two analysis categories of the program.

(S-I) Structured Layout (Type I - Basic Layout)

The laid out program such as (S-I) in **Figure 20** assists students in understanding method structure and their relationship. We named such layout types Structured type.

(S-II) Structured Layout (Type II - Hierarchical Layout)

Hierarchical Layout is one of the StructuredLayout. An example of this layout is shown in **Figure 20** (S-II). The layout defined the callee's method on the right of its caller block so that the student could trace the method call process left to right. The layout such as (S-II), makes it easier to understand method structure and trace program than other layout types. Therefore, we named the layout such as (S-II) hierarchy type.

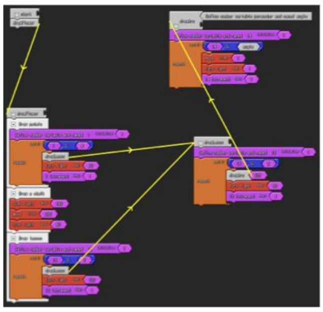
(NS) Non-Structured Layout

In the poor layout program such as (NS) in **Figure 20**, there are no rules among caller and callee so that students have to cross left to right several times when they trace program.

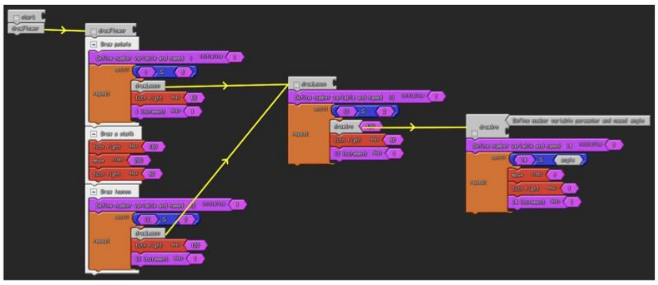
A non-structured layout makes it more difficult for students to understand processing order and caller and callee relationships as well as their own layout. We believe that if it is hard for students to understand method relation, they could make a program without using method definition, or hierarchic method call also. We called such layout type non-structured type.

Experimental Study Environment

We conducted an experimental study in our introductory programming course for non-CS students. This experiment involved two classes, one from 2013 and one from 2014. There were few differences between the classes; both classes participated in the same lectures and were required to perform the same tasks. The class which did not use MeRV (Non-control group) was 2013 (99 students), and the class that used MeRV (Control group) was 2014 (105 students). The difference between classes was the first task condition. 2013 is ANY, but 2014 is Block which was used to introduce students to MeRV.



(S-I)Structured Layout Type
(Basic Layout)



(S-II)Structured Layout
(Hierarchical Layout)

(NS)Non-structure Layout Type

Figure 20: Structure and Non-Structure Layout

Task Subject

Number	Contents	Purpose
1	Create draw circle method	To understand the method definition and call.
2	Create draw house method	To understand the method definition using another method.
3	Create draw flower method	To understand the method definition using hierarchical method call.
4	Create draw squares method	To understand the method definition using some parameters.
5	Create draw free square	To define any methods students want.

Table 9: Lecture tasks and conditions

Number 3 in **Error! Reference source not found.** is the task which was required to efine hierarchical method call. **Figure 21** shows the result of the task's execution, and **Figure 22** shows the correct answer example of the task.

The task has three steps: Step one was to define the method which draws an arc using a number parameter, Step two was to define the method which draws a lemon calling Step one's

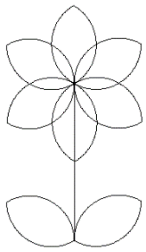


Figure 21: Execution result of the task

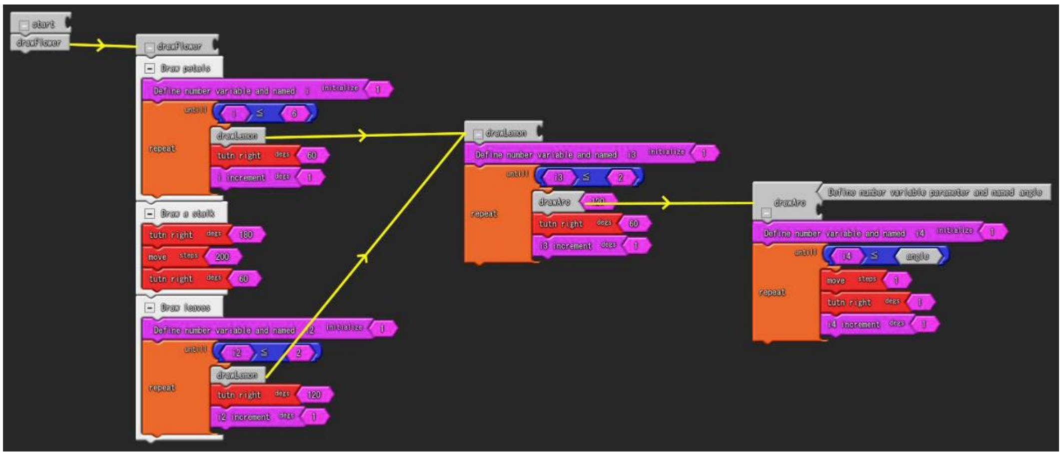


Figure 22: Correct answer structure of the task

method, and Step three was to define the method which draws the flower calling Step two's method. Hence, students' programs would be almost the same structure as **Figure 22**, although there would be some difference depending on whether the students define methods to draw a stem, leaves, and/or petals.

The program which did not use the hierarchic method would not be of value to test the hypotheses because the layout of one method block would not present any layout problems. Therefore, we test those using number 3 in **Error! Reference source not ound.**.

Method understanding data and the analysis

We focused on whether students could understand the merit of defining and using methods.

One of the method's merits is the ease in which students are able to recognize logical structure. Therefore, we did not ask students whether they could define methods but whether they would recognize program structure easier by defining methods well.

We completed the questionnaire survey at the end of the term, and we compared students who used BlockEditor with those who did not. We received 84 answers, and we analyzed all of the answers as available data.

Layout type analysis

We conducted the layout types analysis for programs which satisfied the following conditions:

- Programmed using BlockEditor
- Called method hierarchically

The number of programs satisfying those condition was 48 in 2013, 25 in 2014.

We classified layout type as the three types below:

- Structured type (Hierarchy type)
- Structured type (Non-Hierarchy type)
- Non-structured type

All data were coded independently by two qualified researchers on the basis of the category descriptions. Inter-rater reliability for the 2013 (2014) class was 82% (96%). For each item on which raters disagreed, the item was discussed and consensus was reached on appropriate classification.

RESULT

Questionnaire survey result

Query	Group	Strongly agree	Agree	Neutral	Disagree	Strongly disagree
Method definition is difficult. But, it is easy to recognize programs' structure if it would be defined well.	Block	2 (9%)	19 (83%)	1 (4%)	1 (4%)	0 (0%)
	Other	10 (17%)	38 (62%)	5 (8%)	5 (8%)	3 (5%)

Table 10: Understanding of method's merit (Easy to recognize logical structure)

Query	Strongly agree	Agree	Neutral	Disagree	Strongly disagree
It is easy to trace program if you make layout for method blocks.	3 (14%)	17 (81%)	0 (0%)	0 (0%)	1 (5%)

Table 11: Whether it is easy to trace program if blocks are organized well

Query	Awareness	Number(percentage)
Please answer awareness when you make layout for method blocks.	To trace program process easier.	13(62%)
	Not to overlap a line and other blocks.	14(67%)
	Not to overlap a line and the other lines.	9(43%)
	Not to overlap a block and the other blocks.	11(52%)
	To program more easier.	3(14%)
	Other awareness.	0(0%)

Table 12: Awareness when students laid out blocks

Table 10 shows the questionnaire survey results which asked students whether it was easy to recognize the program's structure by defining and calling methods if it was defined properly. 92% of students who laid out blocks "strongly agreed" or "agree" with the question and the other group was 78%. This result suggests students who laid out blocks recognize the method's merits more frequently than the students who didn't layout blocks.Table 10: Understanding of method's merit (Easy to recognize logical structure)

shows the result of whether it would be easy to trace process by laying out blocks. 95% of students indicated they "strongly agree" or "agree" with the question. This result shows that almost all students thought if they lay out blocks orderly, it would be easier to trace programs.

Table 12 shows the result of students' awareness when they laid out blocks. The questionnaire answer allowed multiple answers and the number of answers for this questionnaire was twenty-one. Thirteen students answered "To trace program easier.". Fourteen students answered "Not to overlap a line and blocks.", nine students answered "Not to overlap a line and the other lines", and eleven students answered "Not to overlap a block and the other blocks". Only three students answered "To program more easy", and no students answered other awareness.

Layout Type Distribution

Figure 23 shows the classification result after we classified students' programs into structure/non-structure type. In **Figure 23**, S expresses the number of structure-type layout, and NS expresses the number of non-structured layout. In Figure 8, there are no students who were classified non-structured layout in 2014, although there were seventeen students classified non-structured type in 2013.

There is a difference concerning the number of students between 2013 and 2014. The average number of Block Editor users in the 2014 class was less than the 2013 class due to a lower number of students in 2014. As a result, we do not believe MeRV caused the reduction.

Furthermore, we classified structural layout into hierarchy-layout and non-hierarchy layout. The result is **Figure 24**. In **Figure 24**, S-I expresses the number of hierarchy type layout, and A-II expresses the number of the other layouts in structure-layout.

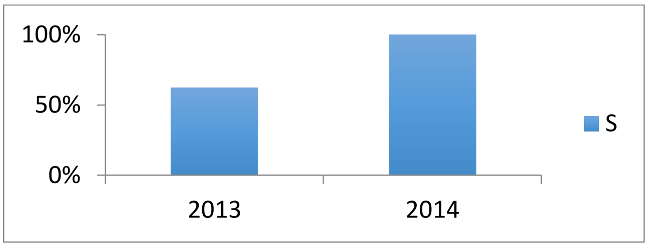


Figure 23: Result of layout analysis I: the percentage of the structured layout t

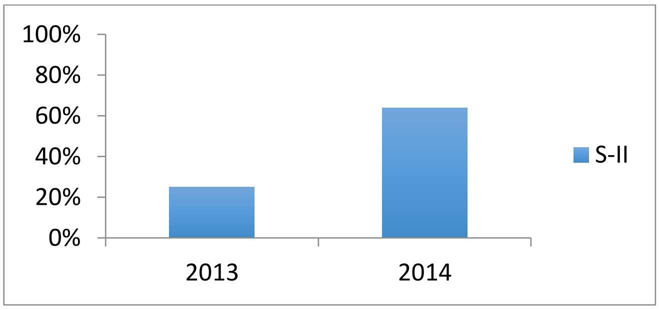


Figure 24: Result of layout analysis II: the percentage of the hierarchical layout(S-II) in whole structured layout (S)

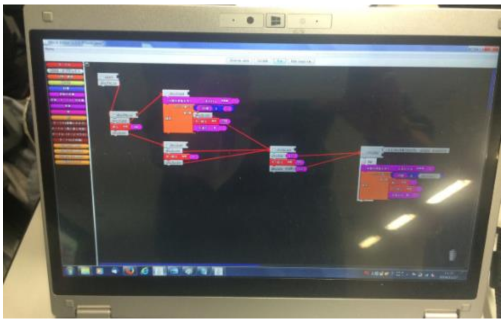


Figure 25: Teaching scene in our course

Figure 25 shows the teaching scene in our course. In this picture, the version of MeRV was a little old. This is the reason why the line color is red. Even though the teacher didn't explain about layout, we could observe almost all students lay out their blocks orderly. Students were especially careful to ensure that lines did not overlap. There were few students who disposed lines.

DISCUSSION

Table 10 shows that the percentage of students in the Block group who answered "strongly agree" or "agree" was more than the other group. The result slightly indicates that those who experienced laying out blocks could better understand the merits of method.

We obtained 95% agreement as the result of Table 10: Understanding of method's merit (Easy to recognize logical structure)

. The result of this question indicated that MeRV could assist in the understanding of merits of methods and process and the result supported hypothesis I.

Figure 23 shows there were no students who make a non-structured layout by visualizing method caller and callee relationship. In addition, **Figure 24** shows that students select more hierarchical way to organize their "two dimensional map". All these works have done by students' spontaneous motivation, without any advice from teaching stuff. That infers they thought creating the map made to trace programs more easily and made their programs more coherent. Hence, hypothesis II was supported by the results.

We confirmed that MeRV made students sence the methods' merit and made them lay out blocks more hierarchically. We believe that MeRV can promote creating a 2D map of method call structure. There would be two major advantages for students by laying out blocks: 1) students can recognize the logical structure of their own program, 2) it is easier for them to trace a process.

In the previous block-base languages, there was no consideration about block layout, so some problems occurred because of a free layout. A free layout made understanding methods and trace, process, and logical structure more difficult for students. In our study, we were able to increase the percentage of students who understood method and were able to create a 2D map of method call structure by visualizing the relationship between methods. We assess that we could support students to lay out blocks and their understanding of method by lay outs that had not been considered previously.

With regard to limitations in our study, we could not distinguish a clear difference between students who laid out blocks and those who did not by our questionnaire or through analyzing student's programs. MeRV could support students in understanding logical structure and guiding layout. In our examination, we required students to define four or five methods. It was assumed that we would gain different layout classification maps if those programs were defined with more methods.

CONCLUSION

In block-type language, there is the problem that if students lay out blocks disorderly, it could be more difficult to trace programs and their comprehension regarding method is effected by their block layout. In this study, we implemented the Method call Relationship Visualizer(MeRV) in BlockEditor to guide students in laying out blocks on their own. The research was conducted in our introductory programming course. As a result, students using MeRV laid out blocks in an orderly fashion themselves, and students were able to gain significant understanding of the merits of method by using MeRV. Therefore, we believe that MeRV can be a scaffold to promote creating 2D maps of method call structure in block-based programming language.

ACKNOWLEDGEMENT

This work was supported by JSPS KAKENHI Grant Numbers 25730203, 26280129.

REFERENCES

Carlisle C. Martin. (2009). Raptor: a visual programming environment for teaching object-oriented programming. Journal of Computing Sciences in Colleges archive Volume 24 Issue 4, 275-281.

Google (2011). Blockly. Retrieved from https://developers.google.com/blockly/ (accessed 2015-02-17)

Google (2012). MIT App Inventor. Retrieved from http://appinventor.mit.edu/explore/ (accessed 2015-02-17)

Harvey Brian and Monig Jens. (2010). Bringing “no ceiling” to scratch: Can one language serve kids and computer scientists? Constructionism 2010, Paris, 2010.

Matsuzawa Yoshiaki, Ohata Takashi, Sugiura Manabu, Sakai Sanshiro. (2015). Effects of Simultaneous and Sequential Work Structures on Distributed Collaborative Interdependent Tasks. In CHI’14, 139-148.

MIT Media lab(2006). Scratch - imegine.program.share-. Retrieved from http://scratch.mit.edu. (accessed 2015-02-17).

Roque Vallarta Ricarose. (2007). Openblocks: An extendable framework for graphical block programming systems. Master thesis at MIT.

Yahoo! (2014). Pipes. Retrieved from https://pipes.yahoo.com/pipes/ (accessed 2015-02-17).

Takashi Ohata received the bachelor degree in Informatics from Shizuoka University (Japan) in 2014. He is currently a graduate student of Informatics. His research interests include programming education using visual programming language.

Yoshiaki Matsuzawa received his PhD in Media and Governance in 2008 from Keio University, Japan. He is an assistant professor in the Faculty of Informatics at Shizuoka University. His research interests include software-engineering education, educational software development, modeling and simulation of complex systems, and learning sciences.

Sanshiro Sakai received his PhD in engineering in 1984 from Shizuoka University, Japan. He is a professor in the Faculty of Informatics at Shizuoka University. His research interests include computer supported collaborative learning, software development environment and programming learning support system.

Practice of Tablet Device Classes in Keio YochishaPrimary School
ICT Education From Primary School First Grade

Tsugumasa Suzuki, *deniro@yochisha.keio.ac.jp*
Keio YochishaPrimary School, Japan

Keiko Okawa, *keiko@sfc.wide.ad.jp*
Keio University Graduate School of Media Design, Japan

Jun Murai, *jun@sfc.wide.ad.jp*
Keio University Faculty of Environment and Information Studies, Japan

Abstract:

This research is an ongoing project started in September 2013 at Keio Yochisha Primary School, involving a continuing series of classes taught to a group of thirty-six second grade students, utilizing tablet devices which have been provided for each student in class from the first grade until the present. These students have already been taking classes that use tablet devices for a total of eighteen hours during their first grade. The purpose of this research is not only limited to development of certain skills, such as ICT literacy and twenty-first century skills, but the research also focuses on the introduction of tablet devices as new stationery. Most standard and general tablet functions were selected and tested for the classes. Drawing applications, digital cameras, movie shooting applications and drills have been utilized over the course of the project. Students continue to take classes on how to use tablet devices after moving up to second grade (thirty-one hours total). Results from these classes and questionnaires show changes in students' understanding, as well as improvements in their personal ICT skills. To students, ICT is no longer a set of special gadgets, but has started to become part of their familiar, every day stationery. In this presentation, the process of how such change occurs is described.

Keywords:

first grade, second grade, ICT skill, tablet device, stationery, learning style

1. INTRODUCTION

The purpose of this researchis to develop ICT skills and literacy at students' early age. This attempt was made through providing a tablet device and a set of digital learning materials for each first grade student at the time of their entry into regular primary school classes, and continuing to provide such educational practice in their regular courses.

In recent years, introduction of ICT in educational environments has become very popular. Furthermore, with respect to computerization of education, as well as education using ICT such as electronic blackboards or tablet devices, Japan is actively promoting these advancements as national policy, and their importance is increasing. On April 28, 2011, "A Vision for Education Informatization" [1], a summary of comprehensive policy and strategy on 'informatization' of school education (for primary and secondary school education) was presented. There are also experimental studies by individual ministries such as "Future School Promotion Project" initiated by the Ministry of Internal Affairs and Communications in 2010, [2] and "Business Innovation of Learning" by the Ministry of Education, Culture, Sports, and Technology in 2011.[3] Currently, there are various action researches on: electronic blackboards, wireless LANenvironments, education clouds, and education through the distribution of one information terminal per person.It should be noted that while many of the ICT related action researches in these high-technology projects are targeted at upper primary school grades and older, the target of this research isa lower grade group, and therefore has significance in that respect.

2. EDUCATION SYSTEM AT KEIOYOCHISHA PRIMARY SCHOOL

One of the characteristics of the education system at Keio Yochisha Primary School is that there is no class change: both students and homeroom teachers remain the same throughout their entire six years of schooling. This means that while the homeroom teachers' responsibilities are great, their discretion is also great.

The teacher (this author) of the second grade class (thirty-six students) has not changed since first grade, and as a general rule, will continue to teach the students until their graduation at the end of the sixth grade. In this class, each student was provided a tablet device during their first grade year from September 2013, as part of a joint research project with Keio University Graduate School. The goal is to familiarize the students with the use of tablet devices from the early primary school stage, and to have them actively utilize various types of ICT instruments in various learning situations as a new type of stationery during their six years in the school.

It should be noted that, more than ten years ago, a specialized department called "Department of Information Technology" was created, with full-time teachers at Keio Yochisha; however, this research is not performed by the Department ofIT.

3. SYSTEM ENVIRONMENT

ASUSTeK Computer Inc.'s Android products were chosen as the students' tablet devices. A special storage cabinet and charging cabinet have been hand-made.

The devices are configured so that iPad, Android, PC and other screens can be projected onto electronic blackboards or projector screens by a single-touch operation using AppleTV and Miracast compatible wireless HDMI adapters. Cisco Systems, Inc.'s wireless APwere also installed in order to create a stress-free, high-speed wireless LAN environment in the classroom.For the purpose of nurturing an awareness that use of tablet devices is not special, but ordinary, only inexpensive or free applications with minimal functions that are very general or standard were chosen.Learning methods that fully utilize the advantages of tablet devices, such as intuitive operation and light weight (mobility), are currently being explored: for example using drawing applications, digital cameras, movie shooting applications, and workbooks.

4. PRACTICE

As stated earlier, the class taught by the author already had a total of eighteen hours of classes using tablet devices during the students' first grade of primary school (September 2013 – March 2014).[4] Table 1 shows the description of each class.

Instructions on how to use ICT devices were kept to a minimum, so that the students could develop ICT skills while working through the coursework. As expected, students gradually became more familiar with tablet device operation, and became creative in the use of the devices without teachers having to teach them how.

Class no.Date	Class description
No. 1 Sept. 17	Rules and confirmation concerning use of tablet devices, how to put devices away, how to turn power on/off and other basic operation
No. 2 Sept. 18	How to take photos using camera, how to look at photos
No. 3 Sept. 27	Arithmetic: "Number Check"; Taking photos of favourite animals
No. 4 Oct. 4	Arithmetic : Creating Addition Quiz using blackboard app.
No. 5 Oct. 11	Japanese:Drawing pictures using blackboard app.
No. 6 Oct. 18	Drawing app.: Free drawing
No. 7 Oct. 25	Drawing app: Importing photos
No. 8 Nov. 15	Calculation app.: Drill (1)
No. 9 Nov. 27	Calculation app.: Drill (2)
No. 10 Dec. 4	Creating stories in groups (1): Photo taking and drawing
No. 11 Dec. 11	Creating stories in groups (2): Presentation
No. 12 Jan. 31	Studying newspaper web site (1)
No. 13 Feb. 5	Studying newspaper web site (2)
No. 14 Feb. 7	Studying newspaper web site (3)
No. 15 Feb. 19	Creating stories using sound app. (1)
No. 16 Feb. 26	Creating stories using sound app. (2)
No. 17 Mar. 5	Learning how to write using kanji app.
No. 18 Mar. 14	Reading exercises using kanji app.

Table 1:First grade class

During students' second grade, in the two-semester period from April to December 2014, a total of thirteen hours of classes were taught using tablet devices.

Table 2 shows the description of each class.

Class no. Date	Class description
No. 19May 9	Arithmetic:learning "time": using time learning app. "KumaDokei"
No. 20May 19	Arithmetic: learning "time": using time learning app. "HanpukuDokei"
No. 21June 5	Japanese learning "who does what"(1): using tablet's movie app. for shooting "who does what to what"
No. 22June 13	Japanese learning "who does what" (2): continue shooting "who does what to what" using tablet's movie app. and presentation
No. 23June 19	Japaneselearning using text "Swimmy": creating flip cartoon animation (using Mozilla Web-based site for creating animation on Swimmy)
No. 24June 26	Japanese learning using text "Swimmy" (2): continue creating flip cartoon animation (using Mozilla Web-based site)
No. 25July 3	Exercises using kanji app.
No. 26Oct. 23	Practicing flick input (1)"TypGage""TYPROID"
No. 27Oct. 29	Practicing flick input (2)"Flick Master""My Typing"
No. 28Nov. 12	Self-photo shooting and posting, and inputtingself-introduction using flick input method and sending to education SNS "Edmodo"
No. 29Nov. 13	Photo shooting and posting and writing description using flick input method on topic of autumn and sending to SNS "Edmodo"
No. 30Dec. 12	in Japanese class, to create original rakugo (Japanese comical stories) and sending to education SNS "Edmodo"
No. 31Dec. 15	creating further rakugo stories using "reply" function of "Edmodo".

Table 2: Second grade class

5. ICT SKILLSSURVEY

In this chapter, I will discuss the teaching practices of the tablet devices from three perspectives: - performance based assessment, questionnaire, and observation.Students began to take advantage of the tablet device as new stationery, as they gain more experiences. The students were able to learn a variety of skills and knowledge.Students learned ICT skills andliteracy as summarized following: (1)How to use the tablet's built-in camera,(2) How to take photos using the camera,(3) How to view the photos, how to delete them,(4) Importing, modifying, and editing photos,(5) Sharing of image data,(6) How to share and present the data already edited and modified,(7) Use of application for arithmetic drills and Kanji learning,(8) Working on arithmetic and Kanji study exercises,(9) Browsing internet websites,(10) Composition of newspaper article summary from web sites,(11) Selection of media using the web site and the Japanese of notes,(12) Presentations and information sharing of newspaper summary using the web sites,(13) Sound recording and playing audiodata,(14) Making a story using a sound app.

In addition, resultsof the questionnaires show how students changed their attitudesin their use of tablets in class.We have conducted the survey three times and we asked the same question every time. "Would you like to use a tablet in class?"Following is a summary of the surveys conducted at three different stages of the project. 1) prior to the introduction of tables, 2) after 11 class sessions over three month period, 3) after 18 class sessions over six months. The summary below shows the transformation of students' consciousness (table 3).

Would you like to use a tablet in class?			
Number	Yes	Neither	No
No.1 Sept.13 / 2013	25	0	11
No.2 Dec.11 / 2013	36	0	0
No.3 Mar.13 / 2014	35	1	0

Table 3 Awareness survey “Would you like to use a tablet in class?”

In the surveys, we also asked another question every time."What would you like to do with a tablet?" Before the tabletwere introduced, students could only indicate subject names with no specific preference of activities.In the second survey, the Math app was the most popular.We could also hear their wish to start using a sound application. In the third survey, more than half of the class indicated their wish to use of applications to study other subjects, more advanced challenges of upper grades. They also indicated their wish to use communication applications, research tools and other advanced and new use of the device (table 4). The number of students increased who could propose more specific ways of using tablets after six months period of the practice.

What would you like to do with a tablet?		
Number	Answer	Number of student
No.1 Sept.13 / 2013	IT, Japanese, Math	21(in 36 students) response(58%)
No.2 Dec.11 / 2013	Math App. , Sound App.	13(in 36 students) response(13%)
No.3 Mar.13 / 2014	Kanji, English, Investigation learningcommunication, e-mailing and studying at home , to do new things	18(in 36 students) response(50%)

Table 4 Awareness survey “What would you like to do with a tablet?”

6. SUMMARY AND AGENDA

As stated earlier, a tablet device was used per student for subjects such as Japanese and arithmetic in classes for the thirty-six youngest (first grade) students in the primary school, during the fifteen month period from September 2013 to December 2014, for a total of thirty-one hours.Data such as questionnaire surveys, class records, and students’ responses were used in order to address issues such as: how to find suitable scenarios for the most effective use of electronic blackboards and ICT instruments like tablet devices; and how to develop and improve teaching methods.Learning using tablet devices was not only shown to be possible in first and secondgrade lower primary school classes, but there were also improvements in these students’ understanding and activeness.On the other hand, questions concerning students’ interest toward tablet devices as a new form of stationery, and the more essential question of whether there has been progress in understanding learning, will require a longer period of observation and research.Furthermore, there is also a necessity for a change in attitude by the teacher (the author) himself on educational guidelines for a lower elementary class group: that teachers must also change from a one-directional,lecture-style transmission of knowledge to students, to a more interactive learning style, where teachers actively participate in the teaching process.

During these classes, tablet devices were always placed in the classrooms for students to use freely. By establishing strict rules on usage, problems that might have occurred, such as students taking away these devices without permission for purposes other than learning, were avoided. The students have started to learn how to efficiently take out and put away their tablet devices at the appropriate times during classes.

Further, the students are starting to become more knowledgeable about tablet device operation and are able to creatively utilize the devices without the teacher's instructions. This is one of the achievements of this project. In some classes, the students became the center of teaching rather than teachers, and teachers sometimes became facilitators, or, in some cases, learners along with the students.

Ideas on how to use tablet devices in school learning were invented by students themselves. Such ideas are perhaps the result of their inexhaustible desire that emerges during the learning process.Suchdesireof these students should not be stopped or changed forcefully, but rather, methods to support and promote better learning should be sought. Based on class data as stated in this report, further development of education materials and studies on teaching methods should be set as a goal, as well as the continuation of data gathering and analysis of the educational effects of conducting active classes that utilize tablet devices as learning tools.

REFERENCES

[1] Ministry of Education, Culture, Sports, Science, and Technology (2011) "The Vision for ICT in Education", (Date of Reference 2015.01.10)http://www.mext.go.jp/b_menu/houdou/23/04/__icsFiles/afieldfile/2012/08/03/1305484_14_1.pdf

[2] Ministry of Internal Affairs and Communications (2010), "Future School Promotion Project", (Date of Reference 2015.01.10) (in Japanese) http://www.soumu.go.jp/main_sosiki/joho_tsusin/kyouiku_joho-ka/future_school.html

[3] Ministry of Education, Culture, Sports, Science, and Technology, "Experimental Research Report on Business Innovation of Learning", (Date of Reference 2015.01.10)(in Japanese) http://www.mext.go.jp/b_menu/shingi/chousa/shougai/030/toushin/1346504.htm

[4] Suzuki, T., Nishiyama, Y., Okawa, K (2014), "Practice of Tablet Device Classes in Keio Primary School: ICT Education From Primary School First Grade", The Japan Society of Digital Textbook, 3: 21-22 (in Japanese)

Tsugumasa Suzuki (born in Tokyo, Japan) B.S. and M.S. degrees in Media and Governance from the Keio University, Japan, in 1995 and 1998, respectively. Since 1998, he has been a teacher (2nd grade) at Keio Yochisha Primary School, Japan. Member of the Information Processing Society of Japan and the Institute of Electronics, Information and Communication Engineers.

POSTERS

Understanding Operating Systems: Considering alternative pedagogies and tools which may be utilised by teachers

Lynne Dagg, lynnedagg@tiscali.co.uk,
University of Sunderland, United Kingdom of Great Britain and Northern Ireland

Steven Haswell, steven.haswell@sunderland.ac.uk
University of Sunderland, United Kingdom of Great Britain and Northern Ireland

Abstract

Pupils within the UK have had the opportunity to learn about Operating Systems within their GCSE and A Level Study for many years but the new National Curriculum (Department for Education, 2013) provides the opportunity for pupils to gain knowledge of the topic throughout their school career. Much of what will be taught will be based on theory but if we want to grasp the attention of pupils the question arises of the alternative approaches which may be available to teachers. The purpose of this paper is to consider whether these approaches should be based on the past or the present or the future.

Keywords

Operating Systems, Windows, Linux, Android, Virtualisation, School, Pupils, Pedagogy

INTRODUCTION

The National Curriculum for England 2014 provides three pages of guidance for Secondary School Computing Teachers on what they are to teach to pupils. The first page provides some guidance on the “Purpose of Study”, “Aims” and “Attainment Targets” (Department for Education, 2013). The second page provides the Subject Content for Key Stage 3 and the third page provides for Key Stage 4. The Subject Content for Key Stage 3 contains the following bullet point “understand the hardware and software components that make up computer systems and how they communicate with one another and with other systems”. This could be taken to include Operating Systems although it is possible that not all teachers will read it in this way.

There was no requirement in the AQA A Level (AQA, 2013) or GCSE ICT (AQA, 2012) syllabi to teach Operating Systems, but the topic is present in the OCR equivalent examinations (OCR, 2013a)(OCR, 2013b). In almost all of the Computing Syllabi considered, the topic is present (see for example (Pearson Edexcel, 2014) and (OCR, 2012). Hence the growth of Computing within the National Curriculum suggests that this is a topic which all teachers need to know about but that some teachers may not have taught in previous years. It is possible to teach the topic without allowing pupils the opportunity to experience different types of Operating System although the question arises of whether it is desirable as this experience may help the pupil to visualise the theoretical context

Teachers do not operate in a vacuum. They operate in a school which has cost constraints, an overarching requirement for child protection as well as a need for Computer Security. It is simply not feasible for schools to buy in a wide variety of computers with different operating systems just to allow pupils to use them for a few lessons while they are shelved for the rest of the year. Schools owe a duty to keep children safe both in the physical borders of the school and within their ephemeral use of the Internet. Hence if we are encouraging pupils to use alternative operating systems we must try to ensure that they are as safe utilising them as they are on existing systems; and that requires that computer safety is maintained.

DISCUSSION

The question arises of how we could teach Operating Systems. In order to do this we need to consider what we need to teach, how we may teach it (and how teachers relate they teach it), and the constraints placed upon teachers.

What do we need to teach?

The first point we need to consider is the subject itself. Operating Systems is a large topic with a plethora of books available for Lecturers to utilise on University modules or for technicians to employ to gain an improved understanding of the systems they service (see for example (White & Davisson, 2015), (Fleishman, 2015),(Wilson, 2014),(Harrington, 2015)and (Mapp, 2014)). The focus of this paper is on the more academic qualifications which pupils may undertake which is relatively superficial - although it is recognized that modules within vocational qualifications may lead pupils to a more detailed study of the topic.

As has already been stated the Key Stage 3 contains a phrase which should point teachers towards the importance of Operating Systems as a topic pupils are to “understand the hardware and software components that make up computer systems and how they communicate with one another and with other systems” (Department for Education, 2013, p. 2).

Let us break this down a little. Pupils are to “understand hardware and software components” and hence will need to understand both the computer and any peripheral devices they may use as well as the software available in the computer system. It is possible to do this and mention operating systems only very briefly. The second part of the phrase may also be unpicked further: operating systems allow the hardware and the software to communicate with each other and have a role in the networking which enables computer systems to communicate with other systems.

As computer professionals we know that at the heart of this process is the operating system which is the most fundamental systems software program available of computers as it ensures that the hardware and the software communicate with each other. In some cases there may be intermediaries (eg device drivers), but in many cases the communication is direct.

In GCSE ICT, one board requires candidates to “have knowledge and understanding of systems software: operating systems, utility software, drivers” (OCR, 2013b, p. 6). While there is little detail present as to exactly what candidates are required to know, it is at least a topic to be studied by candidates - which is not always the case in that GCSE subject.

Within GCSE Computing/Computer Science the requirements are rather more focused “Candidates should be able to: explain the need for the following functions of an operating system: user interface, memory management, peripheral management, multi-tasking and security” (OCR, 2012, p. 8). Another board states that pupils are to “Understand what an operating system is and the functions of an operating system [file management, input/output, resource allocation, process management, network management, user management]” (Pearson Edexcel, 2014, p. 8) It is clear that this is much more explicit than was the case at GCSE ICT or in the National Curriculum but the two lists are not identical in terms of what needs to be studied. The AQA board makes no explicit mention of the term operating system within its GCSE Computer Science syllabus although it does mention areas such as Computer Hardware.

Pupils within England are now required to be engaged in education or training until they attain 18 years of age (note that this does not have to be full time) (Gov.uk, 2014). Hence an increased number of students may remain in education and may pursue A Level studies. At A Level the AQA Computer Science Syllabus requires pupils to “Understand the need for, and functions of the following system software: • operating systems (OSs) • utility programs • libraries • translators (compiler, assembler, interpreter).” (AQA, 2015, p. 26). This does distinguish between different types of system software but isn’t explicit about what elements need to be taught. In contrast the OCR Computer Science Syllabus (a) The need for, Function and purpose of operating systems.(b) Memory Management (paging, segmentation and virtual memory).(c) Interrupts, the role of interrupts and Interrupt Service Routines (ISR), role within the Fetch- Decode-Execute Cycle.(d) Scheduling: round robin, first come first served, multi-level feedback queues, shortest job first and shortest remaining time.(e) Distributed, embedded, multi-tasking, multi-user and Real Time operating systems.(f) BIOS.(g) Device drivers.(h) Virtual machines, any instance where software is used to take on the function of a machine, including executing intermediate code or running an operating system within another.” (OCR, 2014, p. 7). It can clearly be seen that the OCR requirements are more detailed than the AQA equivalent or the earlier syllabi considered. Hence it is this list which will be considered in terms of what pupils need to know about Operating Systems rather than any other.

As Teachers and Lecturers we also know many types of operating systems which are available for us to use and we know that operating systems are types of systems software. But pupils will need this to be explicitly explained to them. These explanations may be practical demonstrations, theoretical demonstrations, practical activities or small workshops.

The first item to be learned is the function and purpose of operating systems. In reality they have a number of functions and purposes, some of these may be explained in a manner which does not involve using a computer. For example explaining the functions of operating systems could be done relatively straightforwardly in practice without recourse to a computer, but the use of a computer may help this teaching to be more effective.

The second item of memory management may be done practically where we allow pupils to change settings (this is feasible in some circumstances we will consider rather later in the paper.

We can explain the role of interrupts and Interrupt Service Routines but they are difficult to demonstrate in practice by allowing pupils to access the Operating System. It is also difficult to demonstrate scheduling in a practical manner, while some types of operating system may not be easy to demonstrate as they may be restricted in their application. Although BIOS is something we should demonstrate to pupils it may not be practicable for pupils to be given access – sometimes passwords are placed on BIOS to prevent access. While it may be beneficial for children to gain access, care must be taken when allowing them to do so as alterations (eg to a password) may occur and leave the computer open to security issues.

Utilising an operating system, pupils can see device drivers in practice, and indeed install one for a new peripheral. However device drivers need to be seen in context and pupils will need to learn the theory as well as to learn through practical activities.

The final item – virtual machines is perhaps the most interesting of these items as, although an operating system provides access to an operating system such machines provide a means of allowing pupils to learn about different operating systems without huge resource constraints, security issues or interruptions to future users of the machine.

Having considered what is to be taught it is clear that some topics lend themselves to theoretical study, while others lend themselves to more interactive, hands on experience. Hence it is important to consider how teachers may approach teaching the topic.

Possible Approaches

Having established that teachers in the UK need to ensure that the children they teach know about operating systems in order to meet their educational needs, it is pertinent to consider how teachers may approach it in a practical way before considering the implications of each method for the school.

The first method is one which contains no practical experience with operating systems at all. Many of the items outlined as being part of the curriculum in schools can be done without recourse to computers. The topics could be taught didactically or through workshops.

A second method would be to use online websites with emulators. There are many example of these One member of Computing at School created a list of such emulators for UK teachers to use. (Tranchino & Powell, 2014). Emulators work within the guest Operating System to allow the user to see how particular computers operate (or did operate). Many of those available are based on older computers and these may be useful in helping pupils to see some systems where fewer items are hidden.

A third method would be to have computers without user restrictions preventing them from doing some of the things they need to explore within operating systems. Some computers may not have been tied down to security settings in the first place as technicians may be unfamiliar with them (eg linux machines or Macs). Teacher A indicated that they used both Linux computers and a room with Macs to teach about User Interfaces for Operating Systems.

A fourth method would be to alter the security settings on existing computers for the lessons where practical activities are being undertaken. We will go into this further later in the paper.

The next method teachers could use is to request additional resources – for example teachers may request that a class set of Raspberry Pi computers are purchased. Teacher A noted that they had Raspberry Pis at their School – these may have been purchased for a different purpose but could be utilised.

A fifth approach is the utilisation of Virtualisation opportunities and amongst teachers who participated in an opportunistic survey, this was done most frequently. For example, one respondent (Teacher B) stated that “I have taught this using virtualisation. Allowing the students to install different types of operating systems, adjusting settings and completing installs. This allows me to look at system utilities etc with them also”. Virtualbox was the most common Virtualisation software mentioned with one Teacher C mentioning that he used it to “give students access to Windows 10, Windows 7, Windows XP, Ubuntu, Android. Here they can configure, compare HCI install software etc”. There is some question about whether it was used to really learn about the operating system itself or for other purposes. – perhaps to talk about the User Interface, or to teach about other system software. Nevertheless, it may be that some aspects of operating systems are discussed when changing settings or configuring the operating system.

The sixth approach is relatively straightforward – using a pen drive version of software and booting up a computer by changing its bios settings. The Operating system could be placed on a flash drive using software such as Universal USB Installer (Pendrivelinux.com, 2015) which will boot the operating system so that it can be used/tested as if it were the main system for the computer. This was not mentioned by any of the teachers taking part in the survey.

Resource, Security and Safeguarding Considerations

Let us look at each of the possible techniques in terms of practicality.

Firstly we considered an approach which does not make use of computers at all. Making the topic more theoretical does mean that there are no real resource, security or safeguarding considerations which may impact on the school. However it may not be the best way to engage pupils in the topic or to provide information about the real world use of operating systems within computers. Pupils may come to the topic and be unable to link their real life experience of an operating system with what is done in class.

The second possibility was the use of websites. There are also some web based OS for pupils to explore and again these have no resource implications (eg (Silversoft, 2015). The advantage is that many of the older operating systems are more basic and the functions are rather more apparent than would be the case in sleeker, rather more modern systems. The disadvantage is that children are in class to learn about operating systems rather than to learn about history. It should also be remembered that because they are dealing with emulations they are learning about what was built rather than on what exists. At first sight available to all pupils in school. However the reality is not as straightforward. There are some simple questions such as

- How many people can use the emulator at a time – will the numbers of pupils trying to access these sites simply cause them to crash.
- Who runs the site – this may be a safeguarding issue
- Are the sites free from viruses and other malware?
- Will we need to unblock sites?
- Is there advertising on the site and if so, for what?

The advantage is that many of the older operating systems are more basic and the functions are rather more apparent than would be the case in sleaker, rather more modern systems. The disadvantage is that children are in class to learn about operating systems rather than to learn about history. It should also be remembered that because they are dealing with emulations they are learning about what was built rather than on what exists.

A third possibility is that the computer settings are available for pupils to change during lessons on this topic. This may be done through technicians altering computer settings or user settings or both (unless this is the normal state for these computers). This method has the advantage the pupils are using software with which they are familiar but has some major disadvantages. Firstly, the technique will take up large quantities of technician time – they will need to deal with problems caused by children who make errors on the computer (which may even lead them to rebuild the computer), as well as time spent setting up the exercise. The second problem is one of security because we are giving children access to areas (eg creating user accounts) which may lead to breaches of network security. The final problem is safeguarding as many of the controls placed on students may be considered to protect children from areas of the internet where they may be vulnerable. Hence if the settings allow pupils to alter their settings to enable them to access any site they wish it may lead to safeguarding issues.

The fourth possibility of purchasing new items with different Operating Systems (eg Chromebooks or the purchase of Raspberry Pi computers) has a number of benefits. The reason for choosing these items was their low cost relative to a normal PC. But the cost of purchasing a cheap Chromebook would be about £150 (200 Euro) while the cost of a Raspberry Pi is around £30 (40 Euro) (plus the accessories required) and when we multiply this for a class of 30 (assuming they will be re-used by other classes), we can see that this may be a sizable proportion of the School IT budget. Some Schools already have more than one type of operating system on computers within the school and hence these existing resources could be utilised. Any item purchased would need to be subjected to normal security policies within the school unless they were used outside the normal network (eg we could create a local network without access to the Wider Area Network used within the school). For the purposes of considering operating systems this may suffice. Where an item is to be used as part of the wider network, we need to consider the issue of safeguarding – will the controls placed on websites that can be accessed also operate within the newly purchased item.

The fifth approach outlined above was to use Virtualisation using software such as VMVirtualbox. This software is free and allows a different operating system to be hosted on the computer while the host operating system is in use. This allows users to work on both operating systems as though they were working normally in the host operating system with a program open in which they can work. The advantages of this technique are that it is possible to provide examples of a number of operating systems (many of them free such as some Linux distributions or Windows 10 Developer Edition), for children to explore the guest system as an administrator and, if they make mistakes and corrupt the system the host is unaffected – computers are not put out of action because of an error made by a child. There will be resource costs however – obtaining and copying the guest Operating System will take up technician time, as will installing the Virtualisation software. In other words some computer security would be maintained when using this technique.

A final approach that could be used is to boot a different operating system using the opportunities in BIOS to boot from a flash drive. . Again this has time implications for technical staff but only in terms of setting up the flash drives and changing the BIOS settings. There is a danger in this type of process – that the user attempts to install a system as this may overwrite the normal operating system for the computer and potentially make it unusable.

Pedagogical Considerations

CONCLUSIONS

There are many ways in which this topic may be taught and there is undoubtedly a need for much of the content cannot easily be taught through practical interactions with operating systems. However where opportunities arise to teach children that there are different types of Operating systems, they do need to be used effectively and in a practical manner.

If the school has a collection of machines including Mac OS, Windows, Android and Chrome, there are no resource constraints (resources are already present) and safeguarding is not an issue as concerns will have been addressed and school policies implemented before children utilise the machines. However, it is important to recognise that there may be difficulties in undertaking certain tasks (eg setting up users) due to restrictions on pupil (and probably staff) accounts.

If the school wants to invest in new hardware (eg Raspberry Pi computers) then this may allow pupils to engage in many practical activities. The cost of such computers is quite low (about £30 plus the cost of items such as MultiMedia Cards, Power Cables, HDMI etc.) which makes the purchase of a class set relatively cheap. Pupils can be encouraged to set up the MultiMedia cards, but this may cause difficulties as existing computers will need a Multi-Media Card Reader. The problems may arise when taking out plugs from existing computers and removing the HDMI cable to link the Raspberry Pi to the monitor. This is time consuming and may be impracticable;

hence it may be possible that more resources (eg a room and monitors with some benching for networking) may be required.

If we decide to consider MS-DOS, we can explain what MS-DOS was like, but pupils may find out more from engaging in practical activities using an emulator than they can from simple discourse or from PowerPoint slides, or even a classroom demonstration. There are good reasons for teaching pupils about historic systems – but perhaps the best reason is that they can see more clearly what the operating system does than is possible with some modern day equivalents. The drawback is that such emulators are often web based and may involve large time lags.

If we decide to utilise virtualization there are certain advantages:

- Safeguarding issues may be reduced as school policies can be retained
- Children may learn to install an operating system
- Children may create users
- Children can see detail about the computer resources they may not be able to see on their normal account in their normal operating system

However it isn't always the best or easiest way to demonstrate an operating system as creating a virtual machine may actually prove very time consuming and the machine may prove very slow because of the need to share system resources. In addition, the Technicians will be required to install virtualisation software and this will take up valuable resource time.

The final way, while probably the easiest is least secure: booting the computer using a pen drive. There are some operating systems (eg Sugar on a Stick (Sugar Labs, 2014)) which are designed to be run in this way, but others should be approached with greater caution. The benefits of using this technique include the ability to trial software in a way which will mimic its performance when installed on the computer and the ease in which it may be implemented by technicians (it will take only a few minutes per computer). However there are a number of drawbacks: For example we need to question:

- Whether pupils will attempt to use this technique to place new software which undermines computer security and safeguarding within the school
- Whether pupils may seek to use this outside the school environment which may place them in danger (eg if they use it to go around parental security blocks on their internet use)
- What happens if pupils attempt to install the software on the computer and it overwrites the normal boot procedures (this may cause great problems for the Technicians).

Hence we can conclude that there is no right way to allow pupils to gain experience of different operating systems and to see how the theoretical underpinnings relate to what they can see happening within the operating systems they use. The method utilised should be a decision for the school as staff have greater awareness of what is viable, however it is crucial that pupils are provided with the opportunity to see the way operating systems work in practice if at all practicable.

The question of pedagogy is one for the teacher who is taking a group learning about operating systems. However, believers in an active pedagogy will employ a practical approach to teaching elements of Operating Systems.

REFERENCES

AQA. (2012). GCSE Specification ICT. AQA. Retrieved from http://filestore.aqa.org.uk/subjects/AQA-4520-W-SP-14.PDF

AQA. (2013). GCE AS and A Level Specification Information and Communication Technology. AQA. Retrieved from http://filestore.aqa.org.uk/subjects/specifications/alevel/AQA-2520-W-SP-14.PDF

AQA. (2015). AS and A-Level Computer Science AS (7516) A-level (7517) Specifications For teaching from September 2015 onwards For AS exams in May/June 2016 onwards For A-level exams in May/June 2017 onwards. AQA. Retrieved from http://filestore.aqa.org.uk/resources/computing/specifications/AQA-7516-7517-SP-2015-V1-1.PDF

Department for Education. (2013). Computing programmes of study: key stages 3 and 4. National curriculum in England. Retrieved 1 March 2015, from https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/239067/SECONDARY_national_curriculum_-_Computing.pdf

Fleishman, G. (2015). *A Practical Guide to Networking and Security in iOS 8*. Seattle WA: Aperiodical LLC.

Gov.uk. (2014, November 19). School leaving age. Retrieved 1 March 2015, from https://www.gov.uk/know-when-you-can-leave-school

Harrington, W. (2015). *Learning Raspbian*. Birmingham, UK: Packt Publishing.

Mapp, G. (2014). *Operating Systems and Networked Environments: From Multics to the Cloud* (1 edition). London: Chapman and Hall/CRC.

OCR. (2012). GCSE 2012 Computing Specification J275 Version 1 April 2012. OCR. Retrieved from http://www.ocr.org.uk/Images/72936-specification.pdf

OCR. (2013a). AS/A Level GCE, GCE ICT: OCR Advanced Subsidiary GCE in ICT H117: OCR Advanced GCE in ICT H517 version 5 -September 2013 specification. OCR. Retrieved from http://www.ocr.org.uk/Images/81815-specification.pdf

OCR. (2013b). GCSE 2012 ICT Specification J461 – Full course J061 – Short course. OCR. Retrieved from http://www.ocr.org.uk/Images/74376-specification.pdf

OCR. (2014). A LEVEL Specification Computer Science H446 For first assessment in 2017. OCR. Retrieved from http://www.ocr.org.uk/Images/170844-specification-accredited-a-level-gce-computer-science-h446.pdf

Pearson Edexcel. (2014). Pearson Edexcel GCSE in Computer Science. Pearson Edexcel. Retrieved from http://qualifications.pearson.com/content/dam/pdf/GCSE/Computer%20Science/2013/Specification%20and%20sample%20assessments/9781446914229_GCSE_Lin_CompScie_Issue2.pdf

Pendrivelinux.com. (2015). Universal USB Installer – Easy as 1 2 3 | USB Pen Drive Linux. Retrieved 2 March 2015, from http://www.pendrivelinux.com/universal-usb-installer-easy-as-1-2-3/

Silversoft. (2015). SilveOS.com - Silverlight operating system. Retrieved 2 March 2015, from http://www.silveos.com/

Sugar Labs. (2014, December 9). Sugar on a Stick - Sugar Labs. Retrieved 2 March 2015, from http://wiki.sugarlabs.org/go/Sugar_on_a_Stick

Tranchino, M., & Powell, P. (2014, November 16). CAS Community | List of Operating Systems emulators [Education]. Retrieved 1 March 2015, from http://community.computingatschool.org.uk/resources/2775

White, K. M., & Davisson, G. (2015). *Apple Pro Training Series: OS X Support Essentials 10.10: Supporting and Troubleshooting OS X Yosemite* (1 edition). San Francisco, CA: Peachpit Press.

Wilson, K. (2014). *Everyday Computing with Windows 8.1* (1 edition). New York, NY: Apress.

Flipped classroom: Can it motivate digital natives to learn?

R. Robert Gajewski, *rg@il.pw.edu.pl*
Warsaw University of Technology, Poland

Abstract

Use of Technology Enhanced Learning in the course of Applied Computer Science did not improve learning outcomes in the expected way. Students equipped with different learning materials enabling them to learn in more flexible and effective way did not have better grades mainly due to the fact that they were not enough motivated to learn. The paper tries to answer on the fundamental question – can flipped classroom motivate digital natives to learn.

Keywords

Flipped classroom, motivation to learn, multimedia, screencasting

Information technologies course

Curricula of Information Technologies (Applied Computer Science) course is based on compromise between the level of students' knowledge and foreseen needs of other subjects. While first point can be easily measured by questionnaires second one is hard to tackle with because of conservative attitude of many teachers to the role of Information Systems and Computer Sciences in engineering. Questionnaires regarding level of students' knowledge has been conducted regularly since 2011. Their results are worse than expected. Students know how to run software like text editor or spreadsheet but they do not know how to use it in order to solve particular problem in effective way. This means that material from Information Technologies on the level of European Computer Driving Licence still should be present in curricula of studies.

FLIPPING THE CLASSROOM

The idea of flipping the classroom (Bergmann & Sams, 2012) for IT course was presented by (R. R. Gajewski, 2013) and then developed in subsequent papers. Motivation to start flipping the classroom was a question raised by four young teaching assistants – how to motivate digital natives to learn (Wlasak et al., 2013)? In order to have a chance to compare FC with traditional classes it was used only for the third block of classes devoted to engineering calculations and their programming.

SURVEY

In January 2014 and in January 2015 surveys concerning students' attitude to flipped classroom were performed. In order to have a possibility to compare results with similar investigations questions were based on surveys which were conducted in Canada (Johnson, 2013) and US in which Likert scale was used (Likert, 1932). Due to this fact questionnaire which was used in the survey was divided into two parts. Detailed analysis on the answers from the survey will be presented during the conference on the poster.

Preliminary discussion

All students were equipped with all possible educational materials suitable for different learning styles like: book in Polish or book in English language, software animations (podcastings) in Polish and in English languages, supplementary portal for the book in Polish language, office hours at the University (three times per week) and in the form of webinars (twice per week), access to software with tutorials and help and auxiliary educational materials stored on educational portal. The most popular kind of educational materials were software animations.

Quantitative results

Quantitative results show that Polish respondents avoid using extreme response categories which is called central tendency bias. Respondents may naturally tend to provide "middle-of-the-road" responses or ratings for multiple items. They behave in such a way unless they already have very strong opinions toward the question topics. Perhaps students from Canada and US are more familiar with flipped classroom so their answers have no central tendency bias.

Qualitative results

There were hundreds of answers on open questions. Sample answer on the question what are the worst elements of Flipped Classroom are as follows: time used for preparation to classes, students have to learn at home, laboratories are useless for students which did not prepare at home, more time which should be spent at home, lack of contact with teachers.

CONCLUSION

Why students having all kinds of materials suitable for different learning styles have still so poor results? Perhaps mainly because they spend not enough time on learning in order to obtain better grade. There is quite different attitude to Flipped Classroom observed among Polish, Canadian and American Students. Polish students have definitely negative attitude to it maybe because it forces them to learn.

REFERENCES

Bergmann, J., & Sams, A. (2012). *Flip Your Classroom: Reach Every Student in Every Class Every Day*. International Society for Technology in Education.

Gajewski, R. R. (2013). Towards a new look at streaming media (Vol. Practice papers, pp. 98–103). Presented at the 10 th IFIP World Conference on Computers in Education, Learning while we are connected, Toruń.

Johnson, G. B. (2013). Student Perceptions on the Flipped Classroom. The University of British Columbia.

Likert, R. (1932). A Technique for the Measurement of Attitudes. *Archives of Psychology*, (140), 1–55.

Wlasak, L., Jaczewski, M., Dubilis, T., & Warda, T. (2013). How to Motivate Digital Natives to Learn? In *WCCE 2013 10th IFIP World Conference on Computers in Education* (Vol. 3: Book of Abstracts, pp. 78–79). Torun: IFIP.

R. Robert Gajewski works for Warsaw University of Technology. He is head of Applied Computer Science at the Faculty of Civil Engineering of Warsaw University of Technology. For more than twenty years his scientific interests and research have been focused on Technology Enhanced Learning. He published more than fifty papers devoted to this subject. His last fields of interest cover such areas as podcasting, webcasting technologies, flipped classroom and m-Learning.

Bring Computer Science Competences into Italian Secondary Schools

Silvio Giaffredo, *silvio.giaffredo@unitn.it*
University of Trento, Italy

Luisa Mich, *luisa.mich@unitn.it*
University of Trento, Italy

Abstract

Many educational systems adopt the competence-based approach, with the aim of enabling a more effective learning process. As a consequence, many definitions have been conceived, at different levels and in different countries, both of general competences for common citizens, and of specific competences, related to individual disciplines, as computer science and others. But the practical application of the competence-based approach is not widespread among the Italian teachers. A first case-study is proposed here, in which the informatics secondary teachers are involved to construct a set of teaching tools, according to the competence approach.

Keywords

Computer science education, informatics education, secondary schools, education by competence

INTRODUCTION

The competence-based approach is occupying a wider and wider space in the field of education, in different areas and at various levels. Initially developed by the professional and vocational sectors, this learning and teaching approach has been acquired by international institutions. Its relevant evolution produced the definition of a standard level in the general competences, standard required to the citizens of the knowledge society. For the educational systems, it is important also the definition of specific competences, characterising the individual disciplines. But, whereas the described trend is steering towards a more precise definition of the expected competences for the learners, sometimes the teachers struggle to interpret this change and adopt a competence-based approach, requested to all the actors of the system. In the following sections we describe a research, which main goal is to support the teachers in adopting a competence-based approach. The "Discussion" and the "Specific competences" sections give several references on competences. The section "Adopting Computer science competences in Italy" focuses on the teaching of Informatics in the Italian secondary technical schools and introduces the area of interest of our research, further illustrated in the following section, "A preliminary analysis". Finally, "A case-study with computer science teachers" gives more details on the ongoing research. A conclusion closes the paper.

DISCUSSION

The many pedagogical theories have been grouped into four different learning schools by Anderson (Anderson, 2000): the behaviourism, the cognitivism, the constructivism and the connectivism. Before the latter and most recent had been defined, the constructivism has been widely developed, generating various related approaches, methods, theories, and definitions. The competence-based approach flourished in this stream of ideas and concepts. According to this approach, learning is a personal evolution along one or more of the four dimensions of the competence: cognitive, functional, social or meta-competence (Le Deist, Winterton, 2005). The relevance of the competence approach has been acquired by the international institutions, with the effect of further attracting the attention of the educational bodies. The definition of the eight key competences for European citizens, contained in the Recommendation of the European Parliament and of the Council (2006), started to promote a wide range of initiatives, often driven by the Ministries of Education of the member states. The key competences, also called general competences, have been coherently adopted as a base to redefine the curricula for K-10 school grades, that is for the mandatory level of education. Besides the definition of the general competences, requested to all the European citizens, similar solutions have been attempted also for the single disciplines of study, which are interesting mainly for learners and teachers, at different levels and sectors of schools. Apart the renewing or integration of the member states curricula, international initiatives have been started with the aim of defining subject-specific competences.

Specific competences

The Tuning project (2014) copes with the specific competences for various disciplines of tertiary education. The effective methods adopted to develop the activity and the good results achieved by this European project, contributed to export this kind of initiative into the rest of the world. Nowadays we can count important extra-European projects, named Tuning Latin- America (http://www.tuningal.org/), United States of America (http://degreeprofile.org/), China (http://tuningchina.org/), Russia (http://www.tuningrussia.org/), Africa (http://www.tuningafrica.org/), and TuCAHEA (http://www.tucahea.org/), for Central Asian countries, which are grouping local Universities. On the specific field of Computer Science, the FETCH project (2014) is trying to build a consistent framework of competences for the discipline.

Also at the national level of secondary school institutions, the Ministries for Education defined some curricular supports for education by competences. In this sense, the situation of secondary school in Italy is not different. After the reform in 2011, Italian schools are gathered into three groups: Licei, with a strong base of general cultural education; Tecnici, technical schools which are divided into the Economical and the Technological sectors; Professionali, vocational schools with a shorter path of 4 years. For the first 2 school-years, the definition of competences has been derived mainly from the European key competences for lifelong learning (Chiozzi, Giaffredo, Gris & Ronchetti, 2014). With the aim at adapting the communitarian definition to the specifically Italian situation, the national revision introduced other characteristics. The most evident is the grouping of disciplines into clusters, named axes, and the definition of Italian Citizenship key competences, a revised version of European key competences. Thus, this kind of national interpretation added some issues to the system (Chiozzi, Giaffredo, Gris & Ronchetti, 2014). In the case of the three final years of the secondary schools, the reference to an European definition is less evident. For the different branches of the secondary education, some general competences have also been defined. In some cases, they are connected directly to specific disciplines, considered of election for those competences. In addition, every specific subject has been enriched with a list of discipline-specific competences. In some countries, a competence is explicitly developed by more than one discipline. In other cases, only one discipline is devoted to an individual competence.

Adopting Computer Science competences in Italy

Restricting the analysis to the field of Computer Science, also named Informatics, the situation is not homogeneous among the different European member states. In some countries, there are institutional definitions of competences, specific for Informatics. But competences defined with a normative origin often suffer for an important limitation. As claimed by Dörge (2014) referring to the German situation, the catalogue created by a normative process contains only the competences “important to the experts compiling the list” (Dörge, 2014 p. 202). And the proposed solution (Dörge, 2014) is to extend the research by a qualitative content analysis of the textbooks on informatics teaching, limiting the risk to miss competences relevant for the literature. The results of the analysis process will help to form a complete set of long lasting competences, with a non-normative origin.

Another issue, which has not be addressed yet, is the limited adoption of the competence-based approach by the teachers, at least in the Italian secondary schools. This is an issue that has not been thoroughly investigated. There are no research data describing how popular and widespread is the competence-based approach in the Italian secondary schools. In fact, to manage a detailed survey there are a lot of difficulties to cope with, and the best way is to ask the teachers. Only to refer to one of the first issues, we would like to remind here the wide range of meanings related to the word “competence” (Winterton, Delamare - Le Deist & Stringfellow, 2006). This reflects a situation of wide variety and richness in the interpretation of a relevant educational approach, but it is quite discouraging with respect to start a complete and detailed analysis of the educational system, even limiting the study to the computer science discipline and for a small area, like a province of Italy. Nevertheless, as a first step of our research, we tried to collect some data asking the computer science teachers. The aim was to collect a preliminary set of evidences and attitudes on the adoption of the competence-based approach to teaching.

A preliminary analysis

We started interviewing a sample of seven computer science teachers, in four different schools of three town of the analysed province. The interviewees have different roles, as some of them always teach in laboratory; the Italian name of this role is ITP, which stands for technical-practical teacher. The total number of the two kinds of teachers in the province is around sixty. Moreover, our research, related to the technical sectors, is specifically interested in the three final course years of secondary schools, excluding the teachers engaged only in the first two course-years. As a consequence, the dimension of the sample is representative enough of the local population of computer science teachers. The individual interviews were recorded by the interviewer and authorised by the interviewed, who are kept anonymous. The analysis of the collected data does not allow us to say a final word on the acceptance of the competence-based approach among computer science teachers. Nevertheless, several interesting elements are emerging as recurrent from the interviews. Relevant issues identified are: the students involvement/motivation, the needed time, and the distance from institutional efforts and the definitions of competences. The interviewed teachers are aware of the relevance of the competence-based approach, to develop the knowledge, the skills and the attitudes of the students. In order to overtake the traditional priority of the instructional content and aim at a wider and deeper involvement of the students, the computer science teachers often use the methods of the project (or problem) based learning (Blumenfeld, Soloway, Marx, Krajcik, Guzdial & Palincsar, 1991). As a critical factor, this approach has been outlined by interviewed as time-consuming in terms of work: with the students; of the students by their-own; to prepare the presentation of the problem to address in class; in some cases with teachers of other subjects; and to accomplish the bureaucratic mission. Some teachers are trying to follow a competence-based approach, also applying some didactical methods well fitting it. But in these cases, they do not find an useful scaffolding in the normative definition of the competence. And the training activities on competence approach, attended by some of the teachers during the last years, do not seem to be adequate to support the transfer of the approach into the daily work with the classes.

A case-study with computer science teachers

Our case study intends to explore a different way to make competences a real application of a more effective approach to education. An action-research will support the engaged teachers in constructing their personal learning path towards the competence-based approach. In our prospective case-study, the teachers will be trained applying to themselves the constructivist approach to learning, which they often follow in their plans with classes: they will construct learning artefacts focusing to competences for their students, in this way building up their knowledge. The teachers will be sustained to form their competence on competence-based teaching in the same way the students often construct their competence through constructive activities.

We are going to start with a very small number of three computer science teachers, in two technical secondary schools. They will be involved in defining several Learning Units (in Italian: Unita' di Apprendimento – UdA) to develop with the classes. The work will be also supported by a software system, enabling teachers to manage the design and definition of the Learning Unit, the assignment to the students and the assessment of the resulting learning outcomes, and helping them to cope with competences. The units will be built according to the problem based learning approach, because PBL can "enhance motivation and thought as students attempt to learn in classrooms": the students can "better understand subject matter content" and the teachers can sustain this development process, shaping to the students "opportunities for learning, guiding students' thinking, and helping them construct new understandings" (Blumenfeld, Soloway, Marx, Krajcik, Guzdial & Palincsar, 1991, p. 392 and 393). This aspect of PBL fits the "learning to learn" competence, a key competence which could, in turn, help to base the work of the project on competences, considering the competences as the learning objectives for the students. The teachers will design the UdAs, freely choosing weather they do it according to the institutional competences or not. The first instances of UdAs will probably have the same shape of previous projects developed by the teachers, and will be based on their reliable experiences. At the same time the research, also supported by the software, will provide to the teachers the right scaffold to identify and define the competences, all along the life-cycle of the UdA: from the design, through the development in classes, and during the assessment activities. Completed the single UdA, the teachers will be asked to observe the students' evolution, with respect to the development process of the competences, highlighted during the activities included in the UdA. At the end of the work, the teachers will have a repertory of teaching and learning tools. These tools support the educational activities, activities designed and performed in the classes following the competence-based approach. The experience will help the engaged teachers to be aware of the importance of the competence-based approach, since from the phase of teaching/learning design. The teachers will achieve this result not in order to be compliant with a normative suggestion, defined far from the everyday classes; but because they have been supported in the autonomous construction of Learning Units, then applied with the students, following the competence-based approach.

The goal of the action-research is to plan, to manage and to analyse a case-study, directly involving computer science teachers through a process of construction of professional awareness, oriented towards the competence approach to learning and teaching. This process will help teachers to intend the competence approach as a possible and effective one, not too far from reliable activities of practitioners. And it is a constructivist process to build up their practical knowledge of competence. The next step of the research will start on the next school-year. It will be the cloning of the experience done with the small sample, and will extend it to other small groups of computer science teachers, starting in the two schools hosting the case-study but involving also more schools. The development of the second step will be supported also by the software system and moreover by the results of the first round: the Learning Units, the competences, and the discussions around their definitions. Nevertheless, in the second round the teachers will be request also to build up a repertory of learning tools, according to the competence-based approach. There will be new processes to construct professional knowledge with different teachers.

CONCLUSION

The competence-based approach is spreading through European and international educational systems. But in some cases, teachers could be found who are not applying it. We gave an overview of the introduction of the competences approach in Europe, then focusing on Italian situation, and on the specific subject of computer science in technical secondary schools. As a first result of an introductory survey, some evidences have been shown to highlight the biggest challenges met by teachers in developing this approach in daily classes. All the same, the Informatics teachers seem to be eager for project-based learning, which can be adopted as an effective driver to introduce the competence-based approach into the classes. Supported by a software system, specifically developed for this research, we plan a case-study involving a small sample, formed by secondary teachers of Italian technical schools. An extension of the case-study has been also suggested.

REFERENCES

Anderson, T. (Ed.). 2008. The theory and practice of online learning. Athabasca University Press. ISBN 978-1-897425-08-4

Blumenfeld, P. C., Soloway, E., Marx, R. W., Krajcik, J. S., Guzdial, M., & Palincsar, A. (1991). Motivating project-based learning: Sustaining the doing, supporting the learning. Educational psychologist, 26(3-4), 369-398.

Chiozzi, G., Giaffredo, S., Gris, R. & Ronchetti, M. (2014). Helping educators to teach competences with the support of technology in the Italian context. In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2014* (pp. 106-113). Chesapeake, VA: Association for the Advancement of Computing in Education (AACE).

Dörge, C. (2014, June). A methodological approach to key competencies in informatics. In Proceedings of the 2014 conference on Innovation & technology in computer science education (pp. 201-206). ACM

FETCH Project – Future Education and Training in Computing: How to support learning at anytime anywhere. Retrieved 3 November, 2014 on http://fetch.ecs.uni-ruse.bg/? cmd=gsIndex

Le Deist, F. D., Winterton, J. (2005) What Is Competence? Human Resource Development International, vol. 8, No. 1, pp. 27–46

Recommendation of the European Parliament and of the Council of 18 December 2006 on key competences for lifelong learning. Official Journal of the European Union 06.05.2008 (2008/C111/01) pp-10-18. Retrieved 4 November, 2014 on http://eurlex. europa.eu/LexUriServ/ LexUriServ.do?uri=OJ:L:2006:394:0010:0018:en:PDF

TUNING Project - Educational Structure in Europe. Retrieved 3 November, 2014 on http://www.unideusto.org/tuningeu/home.html

Winterton, J., Delamare - Le Deist, F., Stringfellow, E., (2006), Typology of knowledge, skills and competences: clarification of the concept and prototype, CEDEFOP Reference series; 64 Luxembourg: Office for Official Publications of the European Communities, 2006

Computer Business Game – a Tool for Increasing Systematic Thinking of Learners

Irena Patašienė, Irena.Patasiene@ktu.lt
Kaunas University of Technology, Lithuania

Martynas Patašius, Martynas.Patasius@ktu.lt
Kaunas University of Technology, Lithuania

Gražvidas Zaukas, Grazvidas.Zaukas@ktu.lt
Kaunas University of Technology, Lithuania

Abstract

Usually the main task of Business Games is to help participants to understand something related to business. It means that computer business games used for teaching subjects are related with business-related fields like accounting, marketing, economics, management, finance. Achievements of each enterprise depend on optimal made decisions. In order to make a decision, it is necessary to use data analysis, understand data base structure, possibilities of installed information system and have competencies to choose the best tool for visualization data. In order to make optimal decisions the manager of the enterprise has to understand what software can clarify the current situation. Systematic thinking can help manager to make the best decision that will be beneficial not only in short run, but also in long run. The paper describes architecture and use of business game “Hard Nut” that would promote such thinking of the learners.

Keywords

Business Game, education, information systems, GIS

INTRODUCTION

Designing and Computer Business Game (BG) requires multidisciplinary knowledge related with Business management, education and ICT. Education systems usually suggest that learner should processes in the training environment systematically (step by step). Designers are able to design very complex games, but they would be difficult to understand for our learners (XXXX, 2010; XXXX, 2014). Designer of Global Business Game J. Wolfe (2005) noticed the BG paradox problem. J. Wolfe, S. Gold, R. Teach, E. Murff (Wolfe & Gold, 2007; Teach & Murff, 2008) analysed in detail the advantages and disadvantages of complexity of BG as well as the relation between complexity and realism.

Most business games that have been developed for several years are rather complex (Kriz, & Auchter, 2011). The complexity of the game also increases with its closeness to reality (Meijer et a., 2014). Research of R.Teach and E. Murff shows that students find it hard to understand a complex game, while teachers find the requirements hard to explain. Thus they prefer the use of a series of small business simulations as such series conveys knowledge better than a single, large scale business game (Teach and Murff, 2008). Other authors consider use of real data advantageous (Lainema and Makkonen, 2003), that, in turn, increases the complexity of the game.

The university student has to know the environment more deeply, we propose to use the game in various modules starting from its basic form and extending it (according to the module) later (XXXX, 2014). Such an extension also further adapts the business game to multidisciplinary studies (Klabbers, 2006).

The objective of this paper is to help the student to understand the situations in real life and to make rational decision for choosing needed software, which helps quickly to make right managerial decision. Thus the game is extended so, that some decisions are made using real data, while the rest use the basic model. Such gradual and didactic solution limits the difficulties for the student both in understanding the environment and decision making.

The player surveys after the game and analysis of the database of results have shown that the players find the purpose of advertising expenses rather hard to understand and explain. Experience has shown that students find it hard to understand the advertising expenses and their distribution. It is not a peculiarity of this business game as advertising strategy is known to be a major concern in simulation play (Goosen, 2009). Simulating a realistic strategy in a formalized way is difficult, thus we decided to avoid implementing this model as a computer business game. Instead, we ask the students to make a tool, gather the data and explain their advertising expenses. As it is going to take a lot of time, both individual and group assignments can be used. Some authors consider that to be important (Rahn, 2009).

METHODS

Experienced managers make decisions based on data analysis. During their studies the students generally get enough theoretical knowledge, but they find it hard to choose the right method, indices, software in the actual practical situation.

In order to give the students such knowledge, simulation and gaming can be used. In this paper business game “Hard nut” was used to help the students from Kaunas university of technology, Faculty of Economics and business, study programs of “Business administration” and “Leadership”.

Students have to understand that the use of chosen software can require modification of the database. They also have to estimate the time and knowledge required for this work.

The basic version of business game “Hard Nut” uses neither localisation of sales, advertising localisation, logistical problems, bankruptcy probability estimation nor calculation of financial ratios. As 5 financial years are being simulated in the usual case, the students have to plan the use of tools for 5 simulated years, as the decision making gets limited time.

Common schema of involving Business Game in education process is shown in Figure 1.

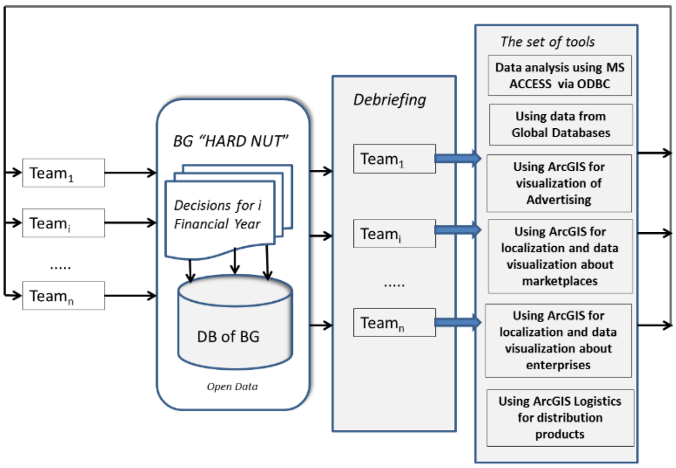


Figure 1: Common schema of involving Business Game in education process

The schema shows that every team of students is offered a choice of tools that can help in making further decisions:

- Data analysis using DB via ODBC,
- Using data from Global Databases,
- Using ArcGIS online for visualization data about advertising,
- Using ArcGIS for localization and data visualization about achievements of marketplaces,
- Using ArcGIS for localization enterprises and visualization summarized data about competitors,
- Using ArcGIS Logistics for distribution products, that requires special preparation.

Each suggested tool demands a different level of IT knowledge. Students should understand their possibilities to use it. If they do not have enough knowledge they could plan to study it in the future.

BG architecture allows students to change DB structure. Students asked to design MS Access tool for data analysis using BG DB via ODBC.

In order to find out the effectiveness of use of business game in the study process, a survey was prepared. Students were asked about effectiveness of business game in gaining of knowledge of business-related topics and IT skills, priority of possible improvements of business game.

First and second year students of Kaunas university of technology (“Business administration” and “Leadership” study programs) participated in survey. The sample included 15 students.

Finally, the relationship between use of tools (given in Fig. 1) and financial results of the enterprises have been compared.

RESULTS

As the students were from “Business administration” and “Leadership” study programs, representing the social sciences, and the time was limited, very advanced use of tools was not expected. Still, they have used

Sometimes the students found it necessary to add some additional tables in common relationship schema. For example, the basic version ignores the geography. Figure 2 shows an example of modification of relationships.

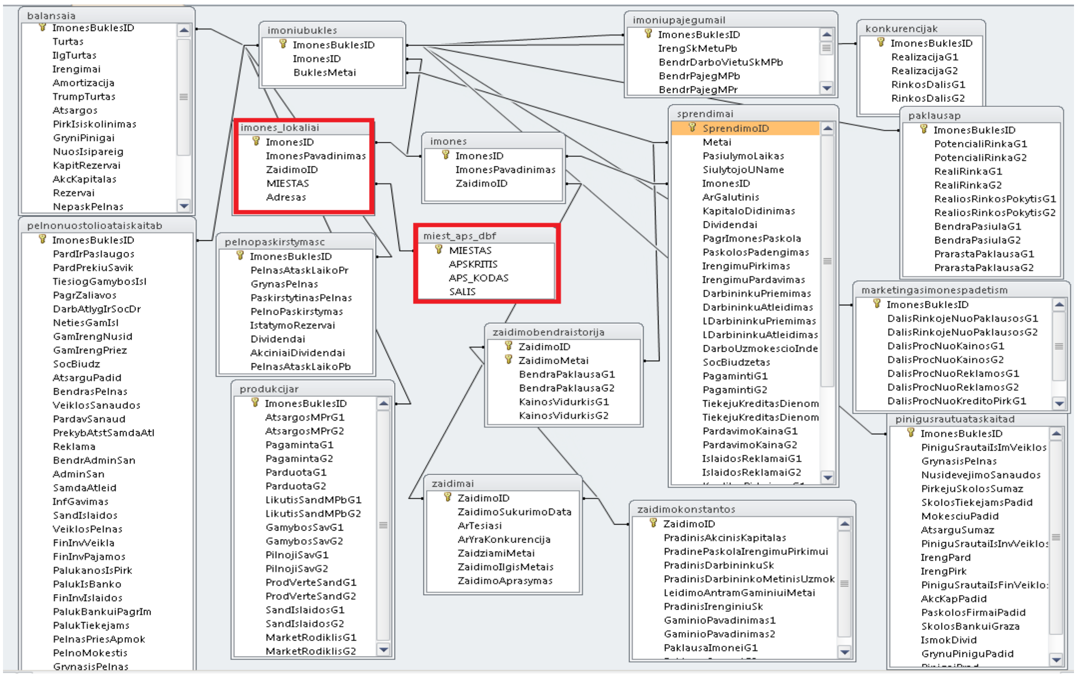


Figure 2: Example of involving additional tables

The relationship schema can also integrate data from other global open databases, for example, database of Department of Statistics or EUROSTAT.

Geographical Information System (GIS) can be used for data visualisation. ArcGIS online was found to be the most suitable option, because of user-friendliness. Businessmen find not just localisation of enterprises or sales useful. For example, Figure 3 shows visualisation of financial ratios on the map.

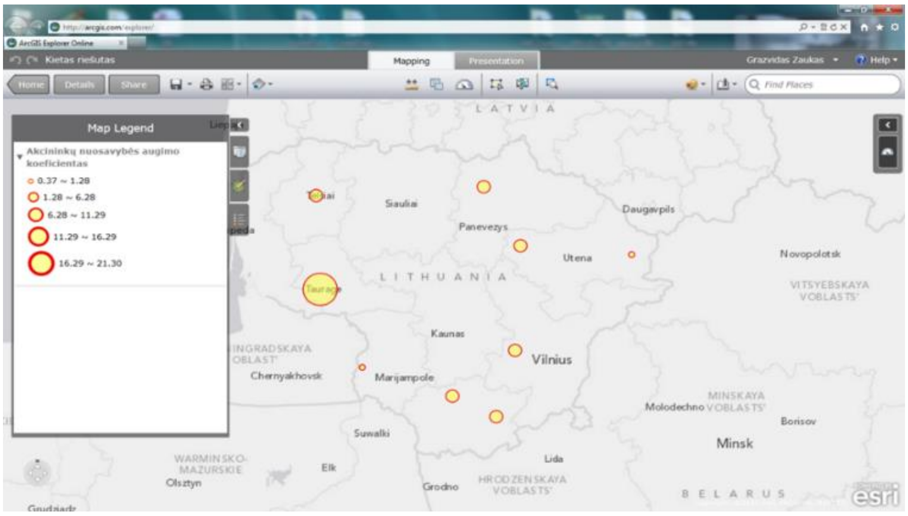


Figure 3: Example of using ArcGis online for visualization data

Figure 4 shows additional visualisation using charts.

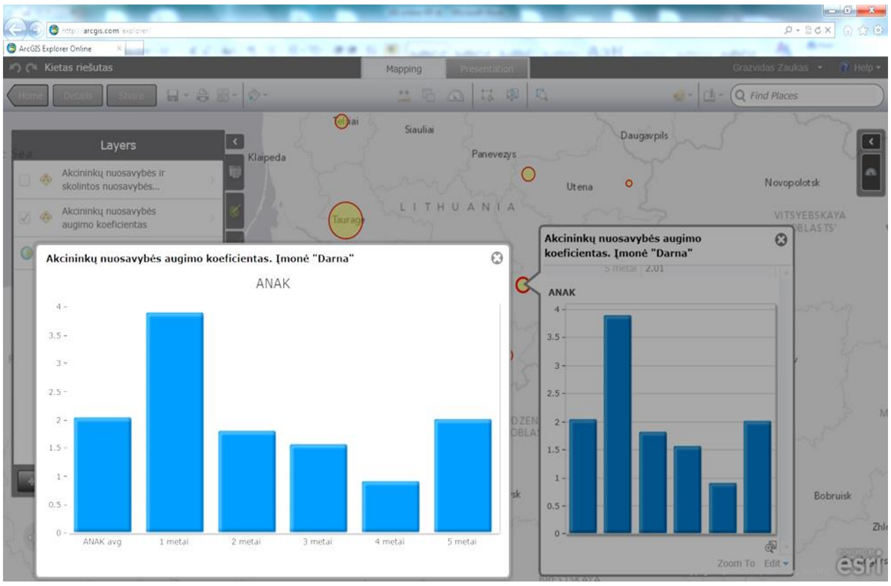


Figure 4: Example of using ArcGis online for visualization data

Fig. 5 shows the geographical positions of the simulated enterprises, chosen by the players. In various cases (while choosing the place for a new enterprise, forecast of sales etc.) the density of enterprises can be important. Thus two rings are shown, one at 100 km radius, another – at 50 km.

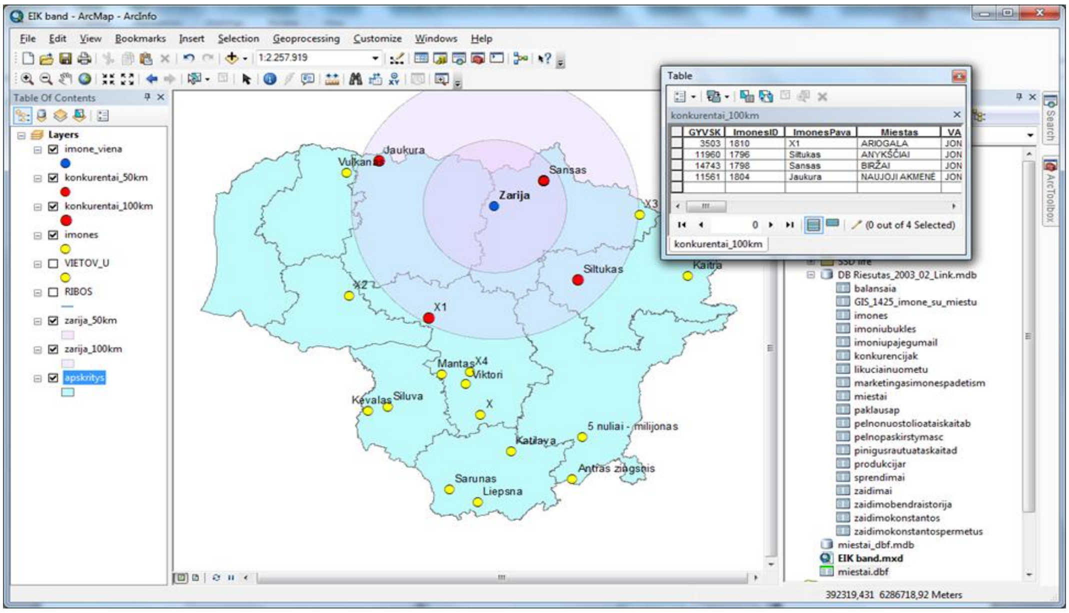


Figure 5: Example of using ArcGIS Desktop for visualization data

Many students are not able to use ArcGIS Desktop, as this version requires special preparation, but during debriefing, they do see the value of this choice. It motivates them to learn to use this program.

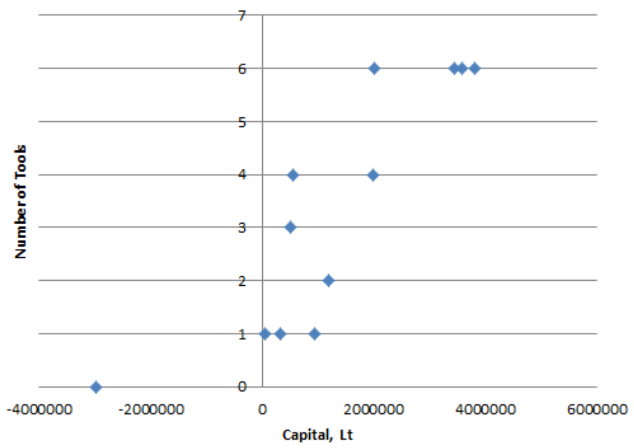


Figure 6: Dependence between resulting capital and number of tools (from Fig. 1) used by the team

Since different students used different tools, we how those choices correlate with financial results (Capital that the students are supposed to maximise). And Fig. 6 shows that more extensive use of tools really has been correlated with better financial results. While the use of most of those tools cannot be expected to affect the results directly, it is likely that the effect was made through making the experience more immersive (for example, even if use of GIS to choose the places for advertisements is not strongly connected with the inputs and outputs of the game, it enables the students to see the realistic level of advertising expenses).

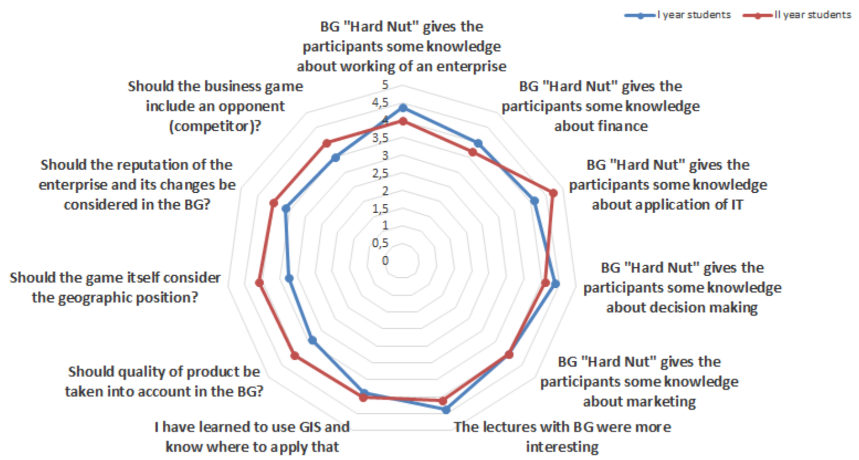


Figure 7: Summarizing opinion of respondents after running the BG

Figure 7 shows that opinion of first year and second year students differed. It can be explained by change of perception of understanding.

CONCLUSION

Multifunctional and multidisciplinary use of business game has shown its universality in study process.

Further data analysis by geographical elements makes it possible to make decisions that are more rational.

Each simulation tool helps users improve knowledge and skills not only in the field of business but in other related subjects as well. That is supported by correlation between use of tools and financial results. Using simulation in study programs at universities helps students improve practice experience in real-life situations requiring decision-making.

Integrated combined application of different technologies allows students to learn analysing data better and make well-supported and more rational decisions more quickly.

While the students say that their understanding of business environment has improved, some of them still struggle to balance the decisions. In the future, we should look for tools that could help those students as well.

REFERENCES

XXXX (2010). Use of simulation and gaming to enhance entrepreneurship. Electronics and Electrical Engineering. Kaunas: Technologija. Vol. 6 (102), p.p. 155-158.

Goosen, K. (2009) An Experimental Analysis of Advertising Strategies and Advertising Functions. Developments in Business Simulation and Experiential Learning, Volume 36, pp. 58-74.

Klabbers, J. H. G. (2008). The magic circle: principles of gaming and simulation. Rotterdam: Sense publishers, p. 373.

Klabbers, J. H. G. (2014). Gaming as Language for dealing with comlex systems in general, Back to the Future of Gaming, p.p. 10-11.

Kriz, W.; Auchter E.; Wittenzellner, H. (2011) Summative evaluation of the nationwide German “Start-Up-Game” competition for entrepreneurship education. Changing the world through meaningful play, Spokane: Eastern Washington University. pp. 98-102.

Lainema, T.; Makkonen, P. (2003). Applying constuctivist approach to educational business games: Case REALGAME.Simulation & Gaming,vol. 34, p.p. 138-149.

Lainema, T.; Nurmi S. (2006). Applying an authentic, dynamic learning environment in real world business. Computers & Education,vol. 47, p.p. 94-115.

Meijer, S., Reich, Y., Subrahmanian E. (2014). The Future of Gaming for Design of Complex Systems, Back to the Future of Gaming, p.p. 154-167.

XXXX (2014). Increasing complexity of Business Simulations and Games is expected in the Future, Back to the Future of Gaming, p.p. 228-241.

Rahn D. *(2009). Enchancing Web-based Simulations with Game Elements for increasing engagement. Developments in Business Simulation and Experiential Learning*, Volume, pp. 303-311.

Teach, R., Murff, E. (2008). Are the business simulations we play too complex? Developments in Business Simulations and Experimential Learning, vol. 35. p.p. 205-211.

Vasconelos F.C.; Ramirez, R., (2011). Complexity in Business environments. Journal of Business Reserarch vol. 64, p.p. 236-241.

Wardaszko, M. (2014). Global domination of Games, Back to the Future of Gaming, p.p. 187-197.

Wolfe, J. (2005). Overcoming the business game complexity Paradox. Developments in Business Simulations and Experimential Learning, vol. 32, p.p. 329-330.

Wolfe, J.; Castroviovanni, G. (2006). Business Games as Strategic Management Laboratories, Developments in Business Simulations and Experimential Learning, vol. 33, p.p. 31-40.

Wolfe, J.; Gold, S., (2007). A study of business game stock price algoritms. Simulation & Gaming, vol. 38, p.p. 153-167.

Teaching Programming in Terms of Supporting Socially Vulnerable Youths: A Qualitative Study of Capability Expansion and Approaching Digital Equity through Computing Education

Toshinori Saito, t-saito@kyoiku-u.jp
Japan Professional School of Education, Japan

Abstract

A study on teaching programming in terms of supporting socially vulnerable youths at risk of social exclusion has been conducted in corporation with a supporting group helping such youths in a small city around Tokyo, Japan. We present a summary of the qualitative analysis of an interview taken from one of the participants of the programming workshop we held and an outline of the relevant information capabilities.

Keywords

Teaching programming, capability expansion, non-formal education, digital equity

BACKGROUND

We consider that there is strong demand for computing education not only for the students in school educations but also for those who are excluded from and outside of school educations. To empower and support such socially vulnerable youths who have missed for various reasons the opportunities to participate in school and also in labour market, and therefore been at risk of being excluded from the process of social engagement, is urgent especially in information-rich societies.

Based on the interest shown above we held a programming workshop for beginners using Scratch in cooperation with a supporting group run by a non-profit organization placed in Yokosuka, a small city around Tokyo, helping socially withdrawn youths, known internationally as Japanese term 'Hikikomori', and also truant students. We made observations and interviews on how the experience of learning programming was interpreted by the participants of the workshop.

In this paper we are going to present a summary of the qualitative analyses of the interview data, which was taken from one of the three participants of the workshop, and theoretical considerations of the analyses from viewpoint of the 'expansion of capabilities' (Saito, 2003), which are derived from the theoretical framework of the Capability Approach (Sen, 1993).

Purpose

The purpose of the research is to derive from the learners' narratives a set of hypothesises about the notion of information capabilities, which should be applied for the exploration into the normative and practical significance of computing education for the purpose of promoting digital equity in information-rich societies. Our main concern is on the potential of teaching programming to provide general understandings of computing, which may expand the youths' capabilities relating the use of information and computers indispensable for achieving well-being.

FINDINGS AND CONCLUSIONS

The workshop consisted of four days of lectures and practices from the 10th of December in 2013 to the 14th of January in 2014, and then in each of the four days we had the time for lectures and practices using Scratch for about three hours including brakes. There were three participants of the workshop who were completely novice for programming. Two of them were adults in the twenties who had experiences of social withdrawal behaviour and the rest was a truant student of junior high school in the middle teens. The author participated the workshop as the main facilitator, and the staffs of the supporting group partly participated the workshop as assistant facilitators. We conducted a semi-structured interview after the workshop to all of the participants for the investigation into the meaning they had found in learning programming at the workshop. Below is a summary of the analysis of the interview taken from a participant, who is one of the twenties.

- From the practices of the workshop, the participant has learned that acquisition of programming may provide an entrance to comprehensive understanding of computing through transcendation of usual manners of human daily thinking.
- Programming was experienced by the participant as the thoroughness of being active toward the computer, which may be the reflection of heteronomous nature of computers.
- The participant has fostered imagination about how computers work, which led the relativization of his usual manners of thinking and the reconsideration on usual ways of using computers.
- For the participant learning programming voluntarily means something like a 'treasure hunt done at one’s own discretion', that is to say, an exploration driven only by one’s intellectual interest seeking for something found by accident without any plans and directions.

We consider that the results show above may indicate the possibility of expansion of the capabilities which may allow us to choose the functionings of, namely, (1)living humanly achieving worthwhile purposes within given informational environment, (2) continuing learning about the knowledge and skills of informatics field to establish autonomy in the society through obtaining and securing economic background using such knowledge and skills, and (3)establishing the awareness as and continuing to be a lifelong learner in each of the contexts of daily lives within certain informational environment.

REFERENCES

Saito, M. (2003). Amartya Sen’s Capability Approach to Education: A Critical Exploration. Journal of Philosophy of Education, 37(1), 17–33. doi:10.1111/1467-9752.3701002

Sen, A. (1993). Capability and well-being. In Nussbaum, M. C. & Sen, A. (Eds.), The quality of life (pp. 30-54). Oxford University Press.

Toshinori Saito is an associate professor in the Professional Program of School Education at Japan Professional School of Education, also is a member of Information Processing Society of Japan, Japan Society for Educational Technology, Community for Innovation of Education and Learning through Computers and Communication Networks, and Asia-Pacific Media and Information Literacy Education Centre.

Facing the upcoming challenges in vocational training with mobile learning

Adrian Wilke, *adrian.wilke@uni-paderborn.de*
University of Paderborn, Institute of Computer Science, Germany

Marina Kowalewski, *dr.kowalewski@vdw-nachwuchsstiftung.de*
VDW-Nachwuchsstiftung GmbH, Germany

Johannes Magenheim, *jsm@uni-paderborn.de*
University of Paderborn, Institute of Computer Science, Germany

Melanie Margaritis, *melanie.margaritis@uni-paderborn.de*
University of Paderborn, Institute of Computer Science, Germany

Poster abstract

This poster presents an approach of supporting a learning project in the field of vocational training in mechanical engineering. In the three-year project Mobile Learning in Smart Factories (MLS), a mobile application for context-sensitive learning will be developed. Apprentices will have the possibility to scan barcodes on machines and receive relevant data directly at their workspace. Trainers have to support the apprentices and need relevant competencies. With regard to the creation and evaluation of workshops for the trainers, we provide insight into the planned project evaluation. The MLS project is funded by the Federal Ministry of Education and Research (Germany, abbr. BMBF).

The changing demands on the vocational education are the starting point of our project. It is funded in the focus area Digital Media in Vocational Training (DIMEBB), in which Web 2.0 technologies and mobile applications are promoted. The Federal Institute for Vocational Education and Training, Germany (n.d.) describes the objective of MLS as "*the development of a context-sensitive mobile application for learning and working [...] to initialize sustainable learning*". This includes both of the target groups, apprentices and trainers. Apprentices will be enabled to obtain new competencies in the working process and to reflect their state of learning. Additionally, trainers will be enabled to enhance their media competency and to integrate digital media into the daily work routine. To achieve this, the learning process of the trainers will be supported by additional workshops, which are integrated in the development process of the project and will take place simultaneously with the development of the software and didactically prepared learning scenarios.

The MLS project comprises the development of a software, the creation of learning materials, the realization of related workshops, and an evaluation of the individual components. We present the project plan and its objectives, the application features, the suspected knowledge fields of trainers and researchers, and a major project property, the context sensitivity. The project is in the initial phase. We started with first observations at vocational schools and a company.

Acknowledgement

The project Mobile Learning in Smart Factories (MLS) is funded by the Federal Ministry of Education and Research (BMBF, Germany) in the focus area Digital Media in Vocational Training (DIMEBB). Funding code: 01PD14009B.

REFERENCES

Federal Institute for Vocational Education and Training, Germany. (n.d.). *MLS - Mobile Learning in Smart Factories*. Retrieved February 17, 2015, from http://www.qualifizierungdigital.de/projekte/laufende-projekte/mls-mobile-learning-in-smart-factories.html.

Federal Institute for Vocational Education and Training, Germany. (2014a). *Datenreport zum Berufsbildungsbericht 2014*. Bonn. (ISBN 978-3-88555-961-1)

Federal Institute for Vocational Education and Training, Germany. (2014b). *Training regulations and how they come about*. Bonn. (ISBN: 978-3-88555-959-7)

Mishra, P., & Koehler, M. (2006). Technological Pedagogical Content Knowledge: A Framework for Teacher Knowledge. *Teachers College Record* , 108 (6), 1017-1054.

VDW-Nachwuchsstiftung. (2012). *Jahresbericht 2012*. Retrieved February 9, 2015, from http://www.vdw-nachwuchsstiftung.de/index.php/medien.html.

Adrian Wilke is a research assistant in the department of Computer Science Education at the University of Paderborn. He holds a master´s degree in computer science. His minor field of study was psychology. His research is related to the project Mobile Learning in Smart Factories.

Dr. Marina Kowalewski is working at the VDW-Nachwuchsstiftung since 2009. From 2012 to 2014 she was head of the project WIKOM (knowledge and communication platform for computer-aided manufacturing processes), which was funded by the german BMBF and ESF.

Johannes Magenheim is a Professor for Computer Science Education at the Institute of Computer Science at the University of Paderborn. His primary areas of research and teaching are CSE and E-Learning. He is member of different working groups of the German Informatics Society (GI) and member of IFIP WGs 3.1 & 3.3 and national German representative to TC 3.

Melanie Margaritis is a research assistant in the department of Computer Science Education at the University of Paderborn. Her research interests are competence development, Mobile Learning and research in computer science for primary school. (http://ddi.uni-paderborn.de/personen/melanie-margaritis.html)

PANELS/SYMPOSIUMS

Promoting a new culture of learning with EDUsummIT and UNESCO

Niki Davis, *Niki.Davis@canterbury.ac.nz*
University of Canterbury, Christchurch, New Zealan

Mary Webb, *Mary.Webb@kcl.ac.uk*
University of London, Kings' College, UK

Nick Reynolds, *nreyn@unimelb.edu.au*
The University of Melbourne, Melbourne Graduate School of Education, Australia

David Gibson, David.c.Gibson@curtin.edu.au
Curtin University, Perth, Australia

Abstract

This panel session enables input from IFIP to strengthen work already underway within the themes for EDUsummIT 2015 to inform UNESCO and IFIP's equitable goals post World Summit on the Information Society. Leaders of the EDUsummIT address the themes of the conference to promote the flow of information and support. Papers to inform UNESCO Institute of Statistics global ICT survey are presented as part of this work.

Keywords

ICT in education, informatics education, change with ICT, research in ICT education

INTRODUCTION

IFIP TC3 working group for research is a sponsor of a related initiative called EduSummIT which, like IFIP TC3, is a global community of policy-makers, researchers, and educators working together to inform evolution of education in the digital age. The EDUsummIT community recognizes the need to respond to the challenges of a world transformed by globalization and economic transformation, caused to a large degree by the development of digitally networked technologies.

The EDUsummIT community first met in The Hague in 2009 to define action steps following the publication of the 'Handbook on Information Technology in Primary and Secondary Education' (Voogt & Knezek, 2008). IFIP became a sponsoring partner early on along with UNESCO. The 2015 EDUsummIT takes place in Bangkok in collaboration with UNESCO Bangkok, which is well known for its equitable work in the area of ICT in education. This panel proposes to inform the 2015 IFIP TC3 conference theme in Vilnius 'A New Culture of Learning: Computing and Next Generations' and its themes and be informed by a panel with leaders in the EDUsummIT.

PANELISTS & IFIP THEMES

Introduced by EDUsummIT co-chair David Gibson three panellists will take at least themes of the 2015 IFIP TC3 conference, as follows:

IFIP 2015 theme Researching educational change: Davis contributions started with a chapter on co-evolutionary change the 2008 Handbook. In 2011 with Birgit Eikelmann she led the working group on educational change with digital technologies (Davis, Eickelmann & Zaka, 2013). In 2015 she is a leader of TGW1 on Smart Partnerships, which continues to focus on organisational change. In addition Niki, drawing on her collaboration with Tim Bell, will also comment on a second theme 'Promoting and supporting Informatics education'.

IFIP 2015 theme: New approaches to learning (collaboration, informalisation, personalisation of learning: Webb as chair of IFIP's working group for educational research and a co-chair of this conference is well known. She has also been a leader within the EDUsummIT for some years and in 2015 she is a leader of the working group Group 9: Curriculum - Advancing Understanding of the Roles of CS/Informatics in the Curriculum". Previous analysis of challenges for information technology supporting assessment Included Webb, Gibson, and Forkosh-Baruch (2013) following discussions at EDUsummIT 2011 identified student involvement in assessment and digitally-enhanced assessment as critical for 21st century learning.

Finally, panellist, Nick Reynolds, as co-team leader of the Creativity working group for EDUsummit2015, will discuss the aims and directions of that group and invite input and cooperation from IFIP members to identify areas of interest and concern with regard to Creativity in a technology enhanced curriculum.

EDUsummIT 2015

Like IFIP working conferences EDUsummITs have led to Call to Action, an Action Agenda, position papers, journal articles, conference presentations, international working group meetings, and white papers aimed to assist people who are working for transformative change in education powered by technology. Information about EDUsummIT can be found at http://www.curtin.edu.au/edusummit/index.cfm

Nine themes have been defined for discussion at EDUsummIT2015, partly based on input from the UNESCO Bangkok office. For the 2015 EDUsummIT working groups are also working up relevant research to discuss at the meeting. For example, the production of working papers to inform UNESCO Institute of Statistics future surveys of ICT in education (Twining & Davis, 2015; Gibson, Downey & Bradley, In Press).

CONCLUSION

From the start of the International Handbook and the EDUsummIT IFIP has been involved. In this panel session will seek input from IFIP participants to strengthen work already underway within the themes for EDUsummIT 2015 to inform UNESCO and IFIP's equitable goals post World Summit on the Information Society.

REFERENCES

Davis, Eickelmann, B. & Zaka, P. (2013). A co-evolutionary perspective on the restructuring of schooling systems in the digital age. *Journal for Computer-Assisted Learning*, 29(5), 438-450.

Gibson, Downie, J. & Broadley, T. (In Press 2015). *Evolving learning paradigms.* Qubec: UNESCO Institute of Statistics.

Twining, P., Davis & Charnia, A. (In Press 2015). *The evolving nature of digital technology and related infrastructure for the school sector.* Qubec: UNESCO Institute of Statistics.

Voogt, J. & Knezek, G. (Eds.) (2008). *International Handbook of Information Technology in Primary and Secondary Education*. New York: Springer.

Webb, M., Gibson, D., & Forkosh-Baruch, A. (2013). Challenges for information technology supporting educational assessment. *Journal for Computer-Assisted Learning,* 29(5), 451–462.

Niki Davis is Distinguished Professor of e-Learning in the University of Canterbury, New Zealand, where she directs its e-Learning Lab. She was awarded the IFIP TC3 Award for Outstanding Service in 2005 and is a past chair of educational research working group. Niki is a leader of EDUsummIT working group on “Smart Partnerships”, informed by her theory of the co-evolution of education and ICT.

Mary Webb is Senior Lecturer in Information Technology in Education at King’s College London and chair of IFIP Working Group 3.3 on Research. Mary co-chaired the working group "Assessment as, for and of learning in the 21st century" in EDUSummIT 2013 and is leading the EDUSummIT 2015 working group on "Curriculum - Advancing Understanding of the Roles of CS/Informatics in the Curriculum".

Nick Reynolds is a Senior Lecturer in ICT, Curriculum and Teaching at The University of Melbourne. He is a Vice-Chair of IFIP TC3 WG 3.3 and is the Australian representative to TC3. Nick is a co-team leader for the Creativity Thematic Group for EDUsummIT2015. He was awarded the IFIP Silver Core Award in 2014

David Gibson is Director of Learning Engagement at Curtin University in Australia and a co-director of the 2015 EdusummIT. His interdisciplinary interests cover a wide range of the learning sciences as well as in the scholarship of teaching and learning, including technology-enhanced pedagogical content knowledge.

Symposium: Scaling up digital pedagogy and innovation in the classroom: research challenges

Symposium co-ordinators:

Cathy Lewin, *c.lewin@mmu.ac.uk*
Manchester Metropolitan University, UK

Sue Cranmer, *s.cranmer@lancaster.ac.uk*
Lancaster University, Lancaster, UK

Background

This symposium draws on our work supporting teachers to change their practice through the adoption of digital technologies in the classroom. Whilst many have theorised about how to achieve educational change (eg Fullan, 2007; Kampylis et al., 2013; OECD, 2008) and undoubtedly some teachers develop innovative digital pedagogies (Law, 2014; Luckin et al., 2012), mainstreaming digital technologies in the classroom remains a challenge for policy makers and practitioners (Kozma, 2003; Brečko et al., 2014). Moreover, innovation in the classroom is a contested and slippery concept, and a teacher's starting point and context must be taken into account (Fullan, 2007; Somekh, 2007). In practice, teasing out what counts as innovative is complex and challenging. The aims of this symposium are to foreground and debate key issues in relation to researching innovation and educational change in digital pedagogy, both in relation to methodological challenges and theoretical perspectives.

Four papers (Turvey & Pachler, Davis & Mackey, Reynolds, Cranmer & Lewin) reflecting different perspectives and experiences will be presented. Each presenter will respond to three key questions addressed by the symposium. The final section of the session (20-30 minutes) will be led by our discussant, Don Passey, who will firstly respond to the presentations and then facilitate a wider discussion with the audience members.

Symposium questions are:

- What does it mean to be innovative? How can we define and describe innovative digital pedagogies? How do other stakeholders perceive innovation in the classroom?
- How can we judge what counts as innovative? What are the methodological challenges of addressing this in research contexts? How might these challenges be overcome?
- How can we best support innovation and change? Which approaches might be most effective?

Keywords

Innovation, digital pedagogy, methodology, change management

A PROBLEM SPACES APPROACH TO PEDAGOGICAL INNOVATION THROUGH DIGITAL TECHNOLOGIES

Keith Turvey, *kt6@brighton.ac.uk*
University of Brighton, UK

Norbert Pachler, *n.pachler@ioe.ac.uk*
UCL Institute

A 'problem spaces' approach to pedagogical innovation through digital technologies (Turvey & Pachler, in press) draws on a framework for constructing a coherent and detailed analysis of technology interventions, concurring with Klafki's (1995/1958, p.21) analysis of pedagogy as rooted in "a particular human, historical situation and with specific groups of children in mind." We concur with others that digital technologies can act as a catalyst for pedagogical innovation but sustained and effective pedagogical innovation through digital technologies is necessarily incremental and teacher led (Rogers, 2003; Law, 2008; Lewin & McNicol, 2014).

The concept of "problem spaces" (Crook et. al., 2010, p. 6) is adopted to explore the genealogy of a framework for pedagogical innovation. Problem spaces are defined as those areas that need to be routinely problematized in the critical analysis of digital technologies because they are conceived as bringing tensions but also opportunities. We identify three significant problem spaces; pedagogy and learning design; teacher professional development, and; convergence of web 2.0 and school cultures. The framework seeks to move beyond the superficial dichotomy of learning technologies being viewed as either instrumental or deterministic and seeks to develop a more coherent and authentic method of analyzing *for* pedagogical innovation with digital technologies.

CO-EVOLUTIONARY PERSPECTIVES ON INNOVATION WITH DIGITAL TECHNOLOGIES IN EDUCATION

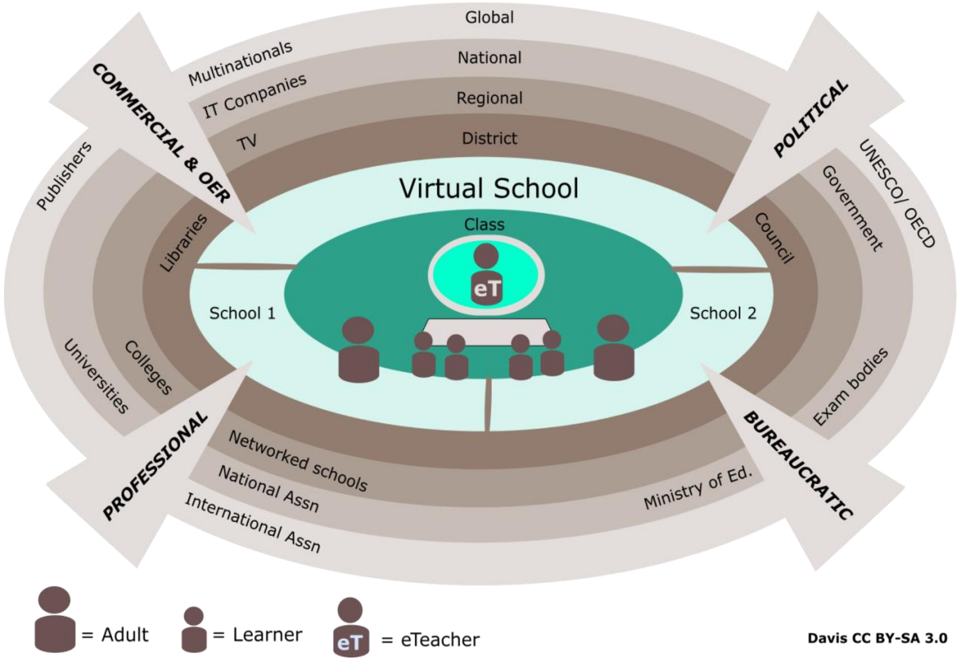
Niki Davis, *niki.davis@canterbury.ac.nz*
University of Canterbury e-Learning Lab, New Zealand

Julie Mackey, *Julie.mackey@canterbury.ac.nz*
University of Canterbury e-Learning Lab, New Zealand

Innovations emerge in many ways from the chaos of the evolution and co-evolution of education and digital technologies in local and global educational systems and services. Some of these innovations can be seen to serve valid educational purposes and careful leadership and management work to sustain them; others are more disruptive. Niki Davis' arena of change with digital technologies in education illustrated in Figure 1 provides this co-evolutionary framework to enable practitioners and researchers to conceptualise the complex chaos that is part of schooling in the 21st century with a teacher and her networked class at the centre of the interacting ecologies. This year Davis' arena is being used to frame a UNESCO publication to guide the updating of UNESCO's survey on ICT infrastructure for schools (Twining, Davis & Charnia, 2015; this version is described more fully in Davis, Eikelmann & Zaka, 2013). Although the teacher is the central keystone species in her class, research into school leadership is identifying the importance of that role in leading and managing change.

The challenge of researching such complex and chaotic systems can be approached by case study, informed by eco-cultural theories. Stakeholders and participants can be recruited to assist in data gathering in ways that improve equity and interpretation (Davis, Mackey & Stuart, 2015). This will be illustrated with research of the case study of the leadership of a future focussed school in New Zealand, undertaken in collaboration with the principal (Stuart) and four additional primary school principals. In common with action research, this case study naturally begins its dissemination near the start of the research as the principals reflect on leadership and interpret it into action in their own schools. The research was strongly informed by Levin and Schrum's case studies of exemplary schools in the USA.

Figure 1. Davis' global arena of change with ICT in schooling showing a distant teacher working with students in at least two schools, as well as the ecosystems within which a teacher enabled by ICT is nested at the centre (developed from Davis, Eickelmann & Zaka, 2013, p.442; first published in Twining, Davis & Charnia, 2015, in press).



RECOGNISING, CELEBRATING AND DEFINING INNOVATION: THE TELEMATICS CHARITABLE TRUST'S APPROACH TO GRANT ALLOCAION.

Nicholas Reynolds, *nreyn@unimelb.edu.au*
The University of Melbourne, Melbourne Graduate School of Education, Australia

While society calls for innovative practices in many areas, identifying what innovation might actually look like is a rather difficult task. What is innovative for one individual or group might easily appear to be mundane to another individual or group. The task of recognising innovative technological practice in social, community, educational and health areas in order to award grant moneys is one that requires a certain degree of flexibility of definition. This presentation will focus on the practices of the Telematics Charitable Trust and the challenges it faces in supporting and celebrating innovation while at the same time acknowledging that innovation might happen at very different times and might look very different for different organisations and for different audiences and purposes.

The author is a long-term member of the Telematics Trust Grants Advisory Committee and uses data from annual reports and reviews of the Telematics Trust and some anecdotal record of thinking and approaches to making awards, to suggest that defining innovation is fraught with danger and that a flexible approach that recognises the contextual nature of innovation, especially when technologies are involved, is essential in encouraging innovative practice and in developing skills and knowledge at a community wide level.

ITEC: ADDRESSING THE CHALLENGE TO DEVELOP AND IDENTIFY INNOVATION IN EUROPEAN CLASSROOMS

Sue Cranmer, *s.cranmer@lancaster.ac.uk*
Lancaster University, Lancaster, UK

Cathy Lewin, *c.lewin@mmu.ac.uk*
Manchester Metropolitan University, UK

Innovation, a complex concept, underpinned Innovative Technologies for an Engaging Classroom (iTEC). This was a 4-year pan-European project (2010-2014) designed to increase the effective use of technology in school classrooms. The project developed resources and digital tools which were piloted at scale in over 2500 classrooms across 20 European countries (Van Asshe et al., 2015).

This paper describes how innovation was conceptualised in the project, resulting in numerous debates and discussions amongst the team. It proved to be impossible to tightly define this complex concept; instead we worked with a collection of ideas which informed our thinking. Such ideas included change (in practice), added value, dependency on context, and degree of change (incremental, radical). The project team agreed that innovation should be based on teacher perceptions (relative to individual's starting points and their particular context) and that supporting incremental rather than radical changes in teachers' practices would be more sustainable and effective (Lewin & McNicol, 2014).

iTEC developed a process to support such incremental changes with a particular focus on digital pedagogy. This paper presents the resulting iTEC approach to learning design enabling teachers and other stakeholders to develop educational scenarios which were intended to be sources of inspiration for pedagogic change (Cranmer & Ulicsak, 2015). We revisit case study data from the iTEC evaluation to exemplify how teachers and other stakeholders applied the learning design approach and how they perceived innovation in practice within their own countries. We will discuss enduring methodological challenges of large-scale comparative studies such as the issue of defining innovation in different cultural contexts and, in turn, the complexity of identifying its occurrence. The paper concludes by proposing the next steps for similar research endeavours.

REFERENCES

Brečko, B.N., Kampylis, P., & Punie, Y. (2014). *Mainstreaming ICT-enabled Innovation in Education and Training in Europe: Policy actions for sustainability, scalability and impact at system level*. JRC Scientific and Policy Reports. Seville: JRC-IPTS.

Cranmer, S., & Ulicsak, M. (2015, in press). Development of the Future Classroom Toolkit. In F. Van Asshe et al (Eds), *Re-engineering the Uptake of ICT in Schools: The iTEC project*. Springer.

Crook, C., Harrison, C., Farrington-Flint, L., Tomas, C., & Underwood, J. (2010). *The impact of technology: value-added classroom practice. Final report*. Coventry: Becta. Available at: http://dera.ioe.ac.uk/1771/

Davis, N.E., Eikelmann, B., & Zaka, P. (2013). Restructuring of educational systems in the digital age from a co-evolutionary perspective. *Journal of Computer Assisted Learning*, 29 (5), 438–450.

Davis, N. E., Mackey, J., & Stuart, C. (2015). *Leadership strategies for a future focused intermediate school – A case study*. Paper presented at the American Education Research Association Annual Meeting, Chicago, 16-20 April.

Fullan, M. (2007). *The New Meaning of Educational Change* (4th edn). New York, NY: Teachers College Press.
Kampylis, P., Law, N., Punie, Y., Bocconi, S., Brečko, B., Han, S., Looi, C.-K., & Miyake, N. (2013). *ICT-enabled innovation for learning in Europe and Asia: Exploring conditions for sustainability, scalability and impact at system level.* Luxembourg: Publications Office of the European Union.
Klafki, W. (1995/1958) *Didactic analysis as the core of preparation of instruction* (Didaktische Analyse als Kern der Unterrichtsvorbereitung). *Journal of Curriculum Studies, 27(1)*, 13-30.
Kozma, R.B. (Ed) (2003). *Technology, Innovation and Educational Change: a Global Perspective*. Eugene, OR: International Association for Technology in Education.
Law, N. (2008). Teacher learning beyond knowledge for pedagogical innovations with ICT. In J. Voogt & G. Knezek (Eds), International handbook of information technology in primary and secondary education (425-433). Springer: New York.
Law, N. (2014). Comparing pedagogical innovations. In M. Bray, B. Adamson & M. Mason (Eds), *Comparative Education Research: Approaches and methods* (2nd edn). Hong Kong: CERC and Springer, 333-364.
Lewin, C., & McNicol, S. (2014). *Creating the Future Classroom: Evidence from the iTEC project. Full Report.* Available at: http://itec.eun.org/web/guest/deliverables
Luckin, R., Blight, B., Manches, A., Ainsworth, S., Crook, C., & Noss, R. (2011). *Decoding Learning: The proof, promise and potential of digital learning*. London: Nesta.
OECD (2008). *Innovating to Learn, Learning to Innovate*. Paris: OECD Publishing.
Somekh, B. (2007). *Pedagogy and Learning with ICT: Researching the Art of Innovation*. London: Routledge.
Turvey, K., & Pachler, N. (in press). 'Problem spaces': a framework and questions for critical engagement with learning technologies in formal educational contexts. In N. Rushby & D. Surry (Eds), *The Wiley Handbook of Learning Technology*, Wiley-Blackwell.
Twining, P., Davis, N.E., & Charnia, A. (2015). *The evolving nature of digital technology and related infrastructure for the school sector*. Montreal: UNESCO Institute of Statistics. Available at: http://www.uis.unesco.org/StatisticalCapacityBuilding/Workshop%20Documents/Communication%20workshop%20dox/Paris%202014/ICT-infrastructure-schools-EN.pdf
Rogers, E.M. (2003). *Diffusion of Innovations* (5th edn). New York: Free Press.
Van Asshe, F., Anido, L., Griffiths, D., Lewin, C., & McNicol, S. (2015, in press). *Re-engineering the Uptake of ICT in Schools: The iTEC project*. New York: Springer.

Sue Cranmer is a lecturer in technology-enhanced learning, Department of Educational Research, Lancaster University. Her research interests include digital media and learning; designing innovative approaches; the development of digital skills and competences; and digital inclusion. Recent work has included a study of how visually impaired young people use digital technologies for learning.

Niki Davis is Distinguished Professor of e-Learning in the University of Canterbury, New Zealand, where she directs its e-Learning Lab. In 2005 IFIP TC3 Award for Outstanding Service, and past chair of educational research working group. Niki is a leader of EDUsummIT working group on Smart partnerships, informed by her theory of the co-evolution of education and ICT.

Cathy Lewin is a Professorial Research Fellow and Director of the Technology and Learning research group. Her research interests concern young people and ICT, in relation to both formal and informal learning. She has conducted research for Becta, the UK government, ESRC and European Schoolnet including ImpaCT2, ICT TestBed, the Primary Schools Whiteboard Expansion Project and iTEC.

Julie Mackey is Dean of the University of Canterbury College of Education, Health and Human Development in New Zealand. She a senior research fellow in the University of Canterbury e-Learning Lab. Research interests include leadership of future focussed schools, modern learning pedagogies, and the design of student-centred online learning.

Don Passey is Professor of Technology Enhanced Learning and Director of the Centre for Technology Enhanced Learning in the Department of Educational Research at Lancaster University, UK. He is vice-chair of IFIP TC3 and WG3.7. His research spans uses of digital technologies to support teaching and learning, in compulsory education, focusing on those who find it hard to learn, and those seeking employment and training

Nick Reynolds is a senior lecturer in ICT, Curriculum and Teaching at the University of Melbourne. He is the Australian representative to TC3 and in 2014 was awarded the IFIP Silver Core Award. Nick has been a member of the Telematics Trust Grants Advisory Committee for more than five years.

Keith Turvey is Principal Lecturer in education at the Education Research Centre, School of Education, University of Brighton, UK. His research interests focus on qualitative research and in particular, the potential of narrative methodologies as a participatory model for researching educators' pedagogical practice with digital technologies.

Towards deeper understanding of the roles of CS/ Informatics in the curriculum

Mary Webb, *mary.webb@kcl.ac.uk*
King's College London, UK.

Niki Davis, *Niki.Davis@canterbury.ac.nz*
University of Canterbury, Christchurch, New Zealand

Yaacov J Katz, yaacov.katz@biu.ac.il
Bar-Ilan University, Israel

Nicholas Reynolds, nreyn@unimelb.edu.au
Melbourne Graduate School of Education, The University of Melbourne, Australia

Maciej M. Sysło, syslo@mat.umk.pl
UMK Toruń, University of Wrocław, Poland

Abstract

This panel will be led by members of the TC3 Task Force: *Curriculum- deeper understanding of roles of CS/ Informatics.* Through brief presentations by panel members as well as audience discussion this panel will examine the position in the school curriculum of Computer Science or Informatics as a key academic discipline that is important in the education of all citizens. It is expected that the outcome will lead to a TC3 briefing paper.

Keywords

curriculum, Computer Science, Informatics

INTRODUCTION

In the light of a recent resurgence of interest in Computer Science or Informatics as a key academic discipline that is important in the education of all citizens, TC3 as the Education Committee of the International Federation of Information Processing (IFIP) needs to continue to take a lead in this important area by reviewing recent developments, identifying key issues and dilemmas and proposing ways forward. This panel which will be led by members of the TC3 Task Force: *Curriculum- deeper understanding of roles of CS/ Informatics* aims to take forward this aim.

PANEL FOCUS

Previous discussions at IFIP Conferences since 2012 have established that within the IFIP Community there is strong agreement that new technologies are enabling new ways of thinking in education, new cultures of learning and that curriculum and assessment needs to change to accommodate new opportunities. The need for development of "digital literacy" and the continued development of the use of technologies for learning is undeniable. The relative importance of CS/ Informatics as the underlying academic discipline and the need for all students to develop key understanding, skills and thinking approaches that emerge from Computer Science/Informatics is less clear. This uncertainty is mirrored in curriculum developments across the globe. In some countries a recent resurgence of focus on Computer Science is driven by economic imperative to create more computer scientists. At the same time many educators see the importance of people understanding the capabilities of computers so that they can exploit technologies for their own use as well as take their place as informed citizens. This new thinking and understanding is not the basic digital literacy whose importance is already well-established but a set of skills, understanding and thinking that can only be developed by engaging with understanding Computer science, how computers work and designing and creating computer-based solutions, including through programming.

Key questions emerging are:

1) What is the range of skills and understanding that should be developed?
2) Are such skills and understanding necessary for everyone?
3) At what age should such education commence and to what extent should it be/remain compulsory?
4) What pedagogical approaches are likely to be appropriate?

PANELISTS & FORMAT OF THE SESSION

Each panel member will talk for approximately 10 minutes on one or more of these questions and then the discussion will be open to the audience. Panel members will talk from their varying experience of their own national situation as well as their internationally-based research.

CONCLUSION

A key outcome of this panel will be a position paper which is currently in draft form. This paper will then be reviewed by the TC3 Committee with a view to it being ratified as a TC3 briefing paper.

Views and perspectives

A VIEW FROM THE UK

Mary Webb

Review of the ICT curriculum in the UK (The Royal Society, 2012) identified a need for fundamental reform: a major concern was that the curriculum had become unbalanced with too much emphasis on basic digital skills at the expense of deeper understanding of concepts. The new curriculum in England for Computing introduced in 2014 recognises the value of Computer Science as the underlying academic discipline and expects pupils to "understand and apply the fundamental principles and concepts of computer science" as well as being able to "analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems". This does not mean that all of the "old ICT curriculum" has been dropped completely: students are still expected to be able to evaluate and apply information technology and to be "responsible, competent, confident and creative users of ICT".

Those of us who have been working in Computing Education for many years are delighted by these curriculum changes: in recent years many of the teachers, with degrees in computer science, whom I have trained have been deeply frustrated by constraints of the school curriculum. So we are all working very hard now to make this new curriculum work and to support teachers in developing their knowledge and pedagogy. We recognise that there are high risks and major challenges to address, in particular: insufficient Computing teachers with the appropriate knowledge of the subject matter; persuading schools to find sufficient time for the new curriculum; ensuring assessment supports and enables exciting and challenging learning in

Computer Science and identifying appropriate pedagogical approaches to achieve this.

One of the important and more general issues arising from the experience in England is how to avoid such curriculum degeneration in future. There are several important factors that contributed to the demise of Computer Science in the curriculum including problems with our assessment and accountability system; weak specification of the curriculum leading to too much ambiguity; lack of understanding of most people of the importance of Computer Science. This last factor remains a significant problem: even people in the IT industry are prone to state that Computing is just a practical subject and there is no need for the underlying theory. There is also a current problem with the perception of programming as just coding. As Computer Science educators we need to not only examine the rationale and content of curriculum but also how to communicate the importance of Computer Science to all learners and the general public.

A consideration of curriculum theory (Webb 2014) and experiences of curriculum design in other subjects suggest that we need to live with uncertainty and to accept the need for a dynamic and continually renegotiated curriculum. However there are epistemological considerations and constraints which can guide curriculum design. A key consideration should be learners' entitlement (Young, 2013) and identifying the knowledge that might constitute at least part of this entitlement must be an important part of the endeavours of those in the discipline that understand the current state of disciplinary knowledge. While the identification of such knowledge is no doubt made more complex by the changing nature of knowledge brought about by the knowledge society, it is clear that in computer science, some knowledge is sufficiently stable to be classified as “powerful knowledge” (Young, 2013).

The approach we are taking in England is to specify a strong and challenging curriculum content of knowledge, understanding and skills underpinned by computer science and emphasising computational thinking up to the age of 14. From 14 to 16 students are still expected to develop their understanding of computing but have the opportunity to specialise in computer science or IT as well as continuing to develop their digital literacy. Developing appropriate pedagogical approaches for students of a range of ability remains challenging. On the Teaching London Computing Project (TLC.org.uk), a professional development programme for existing ICT teachers across London, we emphasise the importance of combinations of “unplugged “ activities as well as practical hands-on experience.

References

Webb, M. E. (2014, July 1st – 4th, 2014). Considerations for the design of Computing curricula. Paper presented at the KEYCIT 2014 – Key Competencies in Informatics and ICT, University of Potsdam, Germany.

Young, M. (2013). Overcoming the crisis in curriculum theory: a knowledge-based approach. Journal of Curriculum Studies, 45(2), 101-118.

A VIEW FROM NEW ZEALAND

Niki Davis

Aotearoa New Zealand, especially the region of Canterbury which is home to a number of software companies with a global reach, provides an interesting perspective on Computer Science education. Canterbury is also home to Tim Bell leads a worldwide collaboration to produce strategicly important research and developmetn with online ‘text books’ that are used world wide, namely Computer Science Unplugged (Bell 2013) and the Computer Science Fieldguide for Teachers, which we also use and develop as part of our postgraduate programme for teachers of Computer Science education. Computer Science disappeared from the high school curriculum for a number of years and was introduced again in 2012 as an examination subject for seniors in high schools (NCEA Digital Technologies standards; NCEA is the summative final examination system in New Zealand) (Bell 2014). The reintroduction for high school only provides evidence that delaying the introduction of computing concepts to the last 2-3 years of school is not effective. The acquisition of programming knowledge and skills appears to be particularly challenging for students, and even more so for teachers. Duncan, Bell and other colleagues in the University of Canterbury are resarching when and how Compuer Science is best introduced into schools (e.g. Duncan, Bell & Tanimoto, 2014).

Bell and Computer Science colleagues in other New Zealand universities have recognised the importance of professional development for school teachers and their difficulties in aquring enough knowledge and skills as mature adults (Bell et al., 2013). Therefore we have developed a course for practicing teachers that involves them activly developing curriulum resources and we are looking to develop more professional development opportunities, including most recently a course in programming and an industry placement. This is being developed in response to a national initiative informed by a report from the Institute of IT Professionals NZ (2014) that proposes “proposes the creation of a new Digital Sciences Learning Area, incorporating Digital Literacy-related topics at primary and secondary school levels as well as digital practice, and Digital Technologies focused on Electronics and Infrastructure, Programming and Computer Science and Digital Media” (Institute of IT Professionals NZ, 2014: 5).

REFERENCES

Bell, T. (2014) Ubiquity Symposium: The science in computer science: unplugging computer science to find the science. *Ubiquity*, March, 1-6. http://dx.doi.org/10.1145/2590528.2590531.

Bell, T. (2014). Establishing a nationwide CS curriculum in New Zealand high schools. *Communications of the ACM*, **57**(2), 28-30.

Bell, T. (2013). Unplugging computer science. In D.M. Kadijevich (ed) *Improving Computer Science Education.* (66-81) New York/Abingdon: Routledge.

Duncan, C., Bell, T. and Tanimoto, S. (2014) *Should your 8-year-old learn coding?* Berlin, Germany: Proceedings of the 9th Workshop in Primary and Secondary Computing Education - WiPSCE '14, 5-7 Nov 2014. 60-69

Thompson, D., Bell, T., Andreae, P. & Robins, A. (2013). *The role of teachers in implementing curriculum changes.* Denver, CO, USA: 44th Association for Computing Machinery Technical Symposium on Computer Science Education (SIGCSE '13), 6-9 Mar 2013. In Proceedings 245-250. http://dx.doi.org/10.1145/2445196.2445272.

Institute of IT Professionals NZ. (2014). *Digital Sciences Learning Area A possible model for the future of Digital Technologies education in New Zealand schools.* Wellington: IIPNZ. Retrieved from http://iitp.nz/files/IITP%20Digital%20Sciences.pdf

Digital Technologies Curriculum in Australia

Nicholas Reynolds,
Dianne P. Chambers,
The University of Melbourne, Melbourne Graduate School of Education

An Australian Curriculum containing the compulsory subject, Digital Technologies, was about to be launched in 2015. A change of Federal Government and its review of the Australian Curriculum has resulted in a delay or even the possible cancelling of that curriculum component. The reviewers were of the opinion that the curriculum was too hard for teachers and not really necessary for young children. It was proposed that it either be offered only from Year 7 (the first year of secondary schooling) or at Year 9. The State of Victoria has decided independently that the subject Digital Technologies, should be a compulsory component of schooling from the first years of schooling (Foundation) to the last years of compulsory schooling (Year 10). This curriculum is built on the notion that computing is its own specialised learning area (different from ICT) that needs its own curriculum and a way of assessing achievement through mandated Achievement Standards; something that had not occurred in previous curricula where ICT was seen as integrated and could often be neglected.

A pilot project to investigate the ways in which teachers went about implementing the new curriculum has been reported by the authors (Reynolds and Chambers, 2014; 2015). It indicated that teachers were capable of understanding the complexities of the new curriculum with adequate support. It also showed that teachers adopted three broad approaches to curriculum implementation. The project showed that teachers can build appropriate learning contexts for young children and for older students.

This Victorian Curriculum (ACARA) is built on five Key Concepts of Abstraction, Data collection (representing and interpreting), Specification, Digital systems, and Interactions and impacts (p.26). The use of these five Key Concepts as the basis for the curriculum moves this curriculum from a purely Computer Science curriculum (although it does contain aspects of Computer Science and coding) to one that attempts to address Digital Technologies as a whole discipline, a discipline where the actions and interactions of humans and computers is of as much importance as the specific knowledge and skills required to think computationally. This curriculum assumes that teachers will be competent, or can be supported to competence, to teach this learning area and that students, even as young as five, have the capacity and the right to develop the skills and knowledge required to operate effectively and ethically in a digital world.

ACARA, (2014) http://www.australiancurriculum.edu.au/technologies/digital-technologies/Curriculum/F-10?layout=1

Reynolds, N. & Chambers, D. P. (2014) Digital Technologies: Teachers interpreting a new curriculum (in French) Revue Internationale de l’Education de Sèvres – International Journal of Education of Sevres *Pédagogie et révolution numérique*. No 67.

A VIEW FROM ISRAEL

Yaacov J Katz

"Computer Science" is a major subject offered in a small but significant number of Israeli high schools that are located at the upper end of the high school technology track. These are usually elite institutions where very talented students study computer science, physics, chemistry and biology (biotechnology) (following Barak, Waks and Doppelt, 2000). However, the mainstream of high school students in public high schools do not study computer science per se as a stand-alone subject but are instructed in technology literacy as a major medium and methodology that contributes to their learning in all the subjects that they study (following Cheema and Zhang, 2013). The Ministry of Education in Israel has a well organised and detailed strategy that directs high schools in both the enhancement of "Computer Science" as a stand-alone high school subject as well as the promotion of "Computer Literacy" as a must for any successful high school student in all subject areas under study ((Naim, 2010).

In my panel presentation I will emphasize the fact that computer use in education is one of the most potent and significant means and platforms for instruction and learning in the 21st century (Alsafran and Brown, 2012). It is imperative that teachers develop knowledge and computer skills that allow them to indulge in teaching based on the transferal of knowledge within their sphere of expertise while utilising available digital databases. Thus, teachers need to be less dependent on their own knowledge of subject matter acquired during their pre- and in-service training, and more dependent on their ability to master the technological alternatives that refer them to digital sites where relevant, updated subject matter can be accessed. In addition teachers need to be technologically literate facilitators who enhance their students' ability to become autonomous learners who possess the computer skills necessary for accessing data relevant to the subject matter they are studying. Teachers need to develop psycho-pedagogical strategies that can motivate their students to acquire the necessary technological capabilities that will allow them to accurately and efficiently become autonomous learners with the ability to access as well as master relevant subject matter in all areas of the school curriculum (following Katz, 2014).

REFERENCES

Alsafran, E. and Brown, D.S. (2012). The relationship between classroom computer technology and students' academic achievement. Research in Higher Education Journal,15 (Mar), 1-19.

Barak, M, Waks, S. and Doppelt, Y. (2000). Majoring in technology studies at high school and fostering learning. Learning Environments Research, 3(2), 135-158.

Cheema, J.R. and Zhang, B. (2013). Quantity and quality of computer use and academic achievement: Evidence from a large-scale international test program. International Journal of Education and Development using Information and Communication Technology, 9(2), 95-106.

Katz, Y.J. (2014) The historical relationship between affective variables and ICT based learning and instruction and achievement in the Israeli school system. In A. Tatnall & W. Davey (Eds.) Reflections on the history of computers in education: early use of computers and teaching about computing in schools. Heidelberg: Springer-Verlag, 324-338.

Naim, Y. (2010). Attitudes of elementary school pupils and teachers towards a technology-based learning environment. Ramat-Gan, Israel: Unpublished M.A. Dissertation, School of Education, Bar-Ilan University.

A NEW INFORMATICS CURRICULUM FOR ALL STUDENTS IN POLAND

Maciej M. Sysło,
University of Wrocław, Faculty of Mathematics and Computer Science, Poland
syslo@mat.umk.pl; syslo@ii.uni.wroc.pl; http://mmsyslo.pl/

The new informatics (here informatics stands for computer science, computing) curriculum benefits from our experience in teaching informatics in schools in Poland for almost 30 years – the first curriculum of informatics as an elective subject in high schools was approved by the ministry of education in 1985 and this subject has never been removed from the high school curricula. It happened 20 years after the first regular classes on informatics were held in two high schools in Wrocław and in Warsaw in 1965. Today, informatics is a compulsory subject in middle (7-9 grades) and in high (10-12 grades) schools and it will replace computer lessons (mainly on ICT) in elementary schools (1-6 grades). The new informatics curriculum is also addressed to vocational high schools.

The new unified informatics curriculum is addressed to **ALL students in K-12** and its main goal is to motivate students to use computational thinking and to engage in solving problems in various school subjects. Moreover its aim is also to encourage and prepare students from early school years to consider computing and related fields as disciplines of their future study and professional career. To this end, the curriculum allows teachers and schools to personalize learning and teaching according to students' interests, abilities, and needs.

The new curriculum is built by describing concepts, activities and personal goals in five areas: (1) **Understanding and analysis of problems** based on logical and abstract thinking, algorithmic thinking, and information representations; (2) **Programing and problem solving by using computers and other digital devices** – designing algorithms and programs; organizing, searching and sharing information; using computer applications; (3) **Using computers, digital devices, and computer networks** – principles of functioning of computers, digital devices, and computer networks; performing calculations and executing programs; (4) **Developing social competences** – communication and cooperation, in particular in virtual environments; project based learning; taking various roles in group projects. (5) **Observing law and security principles and regulations** – respecting privacy of personal information, intellectual property, data security, netiquette, and social norms; positive and negative impact of technology on culture, social live and security. In each of these areas, learning objectives are defined that identify the specific informatics concepts and skills students should achieve in a spiral fashion through the four levels (primary 1-3 and 4-6, middle 7-9, and high 10-12).

Preparation standards for informatics teachers at each school level, teacher evaluation criteria and certificates, teaching and learning materials for students and for teachers accompany the new curriculum. Moreover, all topics in the curricula for other school subject, appropriate for including and using informatics concepts and skills, have been annotated with comments how to apply computational thinking to enhance knowledge and skills in the other subjects.

Mary Webb is Senior Lecturer in Information Technology in Education at King's College London, on the TC3 Executive and Chair of IFIP Working Group 3.3 on Research, co-director of "Teaching London Computing Project". Mary's research over many years has examined pedagogy and formative assessment across both Computer Science as a subject and the uses of new technologies for learning.

Niki Davis is Distinguished Professor of e-Learning in the University of Canterbury, New Zealand, where she directs its e-Learning Lab. Niki studies Computer Science and Psychology in Edinburgh in 1970s and applies that to her postgraduate teaching with Tim Bell for teachers of Computer Science.

Yaacov J Katz is Professor Emeritus at the School of Education, Bar-Ilan University in Israel and is also President of Michlala Jerusalem Academic College. Prof Katz served as Head of the School of Education at Bar-Ilan University and as Chief Pedagogic Officer of the Israel Ministry of Education where he was responsible for all subject matter taught throughout the Israeli educational system. He served as Chairperson of the Israel UNESCO Educational Committee.

Maciej M. Sysło, is a mathematician and computer scientist – academic and school teacher, instructor at in-service courses for teachers, organizer of the Bebras Competition, author of informatics and ICT curricula, educational software, school textbooks and guidebooks for teachers, member of several national committees on education, representative of Poland in IFIP TC3, organizer of the 10th WCCE 2013 in Toruń, recipient of national and international awards and grants: Mombusho (Japan, 1974-1976), Humboldt (Germany, 1982-1984), Steinhaus (Poland, 1986), Car (Poland, 2010), Fulbright (USA, 1996-1997), Best Practices in Education Award (Informatics Europe, 2013), IFIP Outstanding Service Award (2014).

Nicholas Reynolds is a Senior Lecturer in ICT, Curriculum and Teaching at The University of Melbourne. He is a Vice-Chair of IFIP TC3 WG 3.3 and is the Australian representative to TC3. Nicholas is a co-team leader for the Creativity Thematic Group for EDUsummIT2015. He was awarded the IFIP Silver Core Award in 2014

DOCTORAL CONSORTIUM

Informatics in Italian Secondary Schools: competences re-engineered

Silvio Giaffredo, *silvio.giaffredo@unitn.it*
Università di Trento, Italy

Abstract

The adoption of the competence-based approach, required by the educational institutions, sometimes is not realised at school. The action research here proposed involves the Informatics teachers, to construct a meaningful repertory of learning and teaching tools. Only at the conclusion of this training process the teachers will be asked to cope with an adequate set of competences, which are derived rather than passively accepted by the teachers.

Keywords

Informatics education, computer science education, education by competence

QUESTIONS AND MOTIVATION

We focus on the adoption of the competence approach, in Italian Informatics secondary classes of technical schools, grade 11 to 13. There is a gap between the institutional definition of competences and their limited adoption in teaching. Addressing this problem, our general research question is: how can we help the Informatics, also named Computer Science (CS for short) teachers to adopt the competence-based education? This question was split into two sub-questions: which repertory of competences is suitable for CS teachers? And which methods can help the adoption of the competence-based CS teaching?

BACKGROUND AND FRAMEWORK

The constructivism is the background of the competence approach (White, 1959). The concept has been further detailed through a four dimensions model, highlighting functional, social, and cognitive competences, along with the meta-competence (Le Deist, Winterton, 2005). From this holistic model derives the definition of competence as a set of knowledge, skills, and attitudes, declined in the eight key competences by the Recommendation of the European Parliament and of the Council (2006). Besides the general competences, definitions of competences have been produced for many disciplines of study. The individual European member states have defined other supports to the education by competences. Referring to the three final course-years of the Italian secondary technical schools, a list of discipline-specific competences has been defined for every subject, Informatics included.

RESEARCH PLAN

The two main goals of the research are creating a teaching environment and defining a teachers training plan. The teaching environment will be a tool-box, containing a repertory of competences, several related Learning Units, and a set of guidelines. We plan to achieve this first goal on June 2016. From May to November 2016, the training plan will be defined.

We started with a small number of Informatics teachers, involved in defining several Learning Units, according to the project-based learning (PBL) method (Blumenfeld, Soloway, Marx, Krajcik, Guzdial & Palincsar, 1991). The work will be supported by a software system, developed during the research, enabling teachers to manage the Learning Unit design, the assignment to the students, and the assessment of the resulting learning outcomes.

Research approach and methodology

The teachers are involved in an action-research, constructing learning artefacts which will be used in classes. The teacher training methodology applied is the constructivist approach to learning: the teachers form their competence on competence-based teaching through activities which construct real learning units, with no preliminary reference to the competences. During the work, the teachers will observe the students' evolution, with respect to the development process of competences, identified together with the researchers.

Findings and remaining work

There is a lack of quantitative analysis, able to measure how common is the use of the competence-based approach in teaching. So, we applied two qualitative research techniques. Some focus-groups involved sixty teachers of different disciplines, then we interviewed seven secondary CS teachers. The overall result reveals that the teachers are not eager to follow the institutional definitions of competences. The CS teachers use commonly the PBL method. As a consequence, we required the teachers to design several project-based learning units, with no reference to the institutional competences. The learning units have been applied in the classes. The results of the experiences will help to understand, ex-post, which competences have been developed during these activities, even if the competences were not explicit initially. The re-engineering of the learning units will explain the competences related to the activities, supporting the building of a competences repertory.

EXPECTED CONTRIBUTION OF PHD WORK

The action-research process of building a repertory of competence-based teaching tools represents a training course template, which can be used to plan teachers training courses.

CURRENT CHALLENGES AND QUESTIONS

The school-years and the years of the PhD research are out of phase. The current school-year will last only two more months, a short time to collect a big amount of data.

REFERENCES

Blumenfeld, P. C., Soloway, E., Marx, R. W., Krajcik, J. S., Guzdial, M., & Palincsar, A. (1991). Motivating project-based learning: Sustaining the doing, supporting the learning. Educational psychologist, 26(3-4), 369-398.

Chiozzi, G., Giaffredo, S., Gris, R. & Ronchetti, M. (2014). Helping educators to teach competences with the support of technology in the Italian context. In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2014* (pp. 106-113). Chesapeake, VA: Association for the Advancement of Computing in Education (AACE).

Le Deist, F. D., Winterton, J. (2005) What Is Competence? Human Resource Development International, vol. 8, No. 1, pp. 27–46

Recommendation of the European Parliament and of the Council of 18 December 2006 on key competences for lifelong learning. Official Journal of the European Union 06.05.2008 (2008/C111/01) pp-10-18. Retrieved 4 November, 2014 on http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2006:394:0010:0018:en:PDF

White, R.H. (1959) Motivation reconsidered: the concept of competence. Psychological Review, Vol. 66 (5), 279-333

Silvio Giaffredo is a PhD student at the Doctoral School in Information and Communication Technology of the University of Trento, Italy, where he received the Diploma of the Special School in Informatics and the degree in Economics. He worked in a software house, then in high schools as a computer science teacher.

Delivering and Assessing Computational Thinking Skills through Integration to National Curricula

Volkan Kukul
Gazi University, Turkey

INTRODUCTION

Computational thinking (CT) is a concept that has been gaining importance for the past few years. Especially after Wing’s definition in 2006, CT gained more attention. According to Wing (2006), CT is a way of "solving problems, designing systems and understanding human behaviour by drawing on the concepts fundamental to computer science". It is perceived as a fundamental skill, similar to reading, writing, etc. as the importance is also mentioned by Wing (2006). Wing (2006) also says that “Computational Thinking will be a fundamental skill used by everyone in the World by the middle of the 21st Century”. Another point within CT is 21st century skills and standards. On the other hand, one of the skills that students should have in 21st century has been pointed by Trilling and Fedal (2009) is “Problem Solving”. They mean real-life problems and solutions with “Problem Solving”. According to them, it is important to apply knowledge to real-life problems. So, it is similar with the scope of CT.

The International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA) have collaborated for an operational definition of CT;

Computational thinking (CT) is a problem-solving process that includes (but is not limited to) the following characteristics:

- formulating problems in a way that enables us to use a computer and other tools to help solve them,
- logically organizing and analyzing data,
- representing data through abstractions such as models and simulations,
- automating solutions through algorithmic thinking (a series of ordered steps),
- identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources, and

- generalizing and transferring this problem solving process to a wide variety of problems (CSTA & ISTE, 2011).

When we examine CT definition of CSTA and ISTE, problem solving process stands out. While trying to integrate CT to curriculas, problem solving process can be a framework for us. In recent years, researchers and educators try to investigate how to integrate CT to classrooms or curriculas. According to Weinberg (2013) , there are 4 main ways to up skill CT to students in K-12 classrooms. These ways are;

- Technology Free Computational Thinking (The activities like cs-unplugged)
- Programming Game and Robotics
- Initial Learning Environments (Like Scratch, Alice, AppInventor)
- Integrating Computational Thinking with Other Disciplines (Like STEM fields)

In Turkey the name of the Information Technologies course is changed to Information Technologies and Software (ITS). New technologies and 21st student standards were integrated to new curricula. Moreover, programming is integrated into the curricula too. In this research, one of the Initial Learning Environment AppInventor will be used to donate students with CT skills.

The Aim of This Research

In the light of these facts, the aim of this research is to develop a CT integrated model for teaching programming to secondary school students. In this model CT will be handled as a problem solving process and model will be based on problem solving.

Research Questions

- How can CT be integrated into the ITS curricula?
- What kind of activities can be used to help students gaining CT skills?
- How can students' CT skills be assessed?
- What kind of evidence can be sought to explore whether students adopt CT skills into real-life situations?

Methodology

In this research, it is planned to use convergent parallel design which is one of the mixed method designs. In this design, qualitative data and quantitative data is collected synchronous. A simple motivation for this design is that one data collection form supplies powers to offset the weaknesses of the other form, and that a more complete understanding of a research problem results from collecting both quantitative and qualitative data (Creswell, 2012, p540). In this research design the priority is equal for both qualitative and quantitative data. Collected data is analysed separately but they are interpreted together (Creswell ve Plano Klark, 2015, p79).

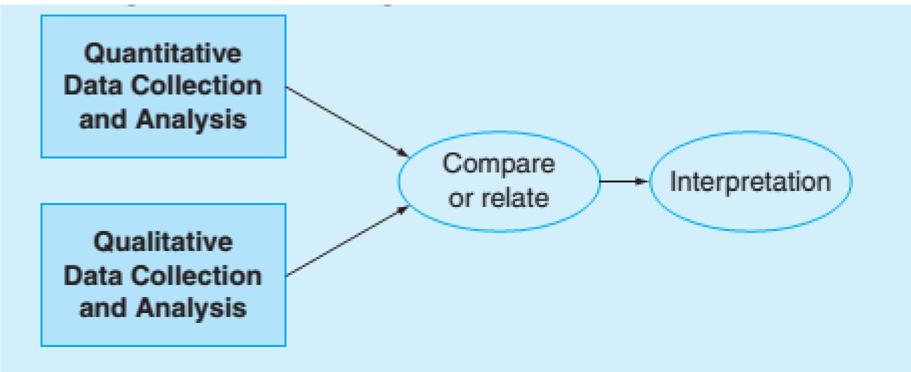


Figure 26. Convergent Parallel Design (Creswell, 2012)

Data Collection

The research is will base both quantitative and qualitative measures. In the qualitative part of the research, students will be observed by the researcher with an observation form, interviews will be conducted for evaluating the process from students' aspects and a checklist will be developed for evaluating students' products about the problems.

For the quantitative part, it is planned to develop a survey for evaluating the students' CT skills. For the survey the progression chart of Computer Science Teacher Association (CSTA) can be used, or can be added more steps.

References

Creswell, J. W. (2012). Educational Research Planning, Conducting, and Evaluating Quantitative and Qualitative Research. Boston: Pearson.

Creswell, J. W., & Plano Clark, V. L. (2015). Karma Yöntem Araştırmaları Tasarımı ve Yürütülmesi. (Y. Dede, & S. B. Demir, Çev.) Ankara: Anı Yayıncılık.

CSTA, & ISTE. (2011). Computational Thinking in K–12 Education leadership toolkit. URL: http://csta.acm.org/Curriculum/sub/CurrFiles/471.11CTLeadershiptToolkit-SP-vF.pdf

Trilling, B., & Fadel, C. (2009)."21st Century Skills Learning for Life in Our Times". Jossey-Bass, *San Fancisco.*

Weinberg, A.E. (2013). Computational Thinking: An Investigation Of The Existing Scholarship And Research. *Colorado State University, Colorado, USA.*

Wing, J. M. (2006). "Computational thinking". *Communications of the ACM*, *49*(3), 33. doi:10.1145/1118178.1118215.

Parallel Analysis- Accuracy in factor retention

Seeta Jaikaran-Doe seetam@utas.edu.au
University of Tasmania, Australia

Abstract

Parallel analysis (Horn, 1965) is the most accurate method to confirm the number of factors/ components to retain in instruments such as surveys, however, the method is infrequently used. This paper describes the process of utilising parallel analysis with Monte Carlo simulation techniques (Watkins, 2000) as the final process to correctly establish factors after the following is completed: Principal Component Analysis (PCA); Varimax with Kaiser Normalization; an examination of the eigenvalues greater than 1; and Catell's screeplot . A simple survey instrument which investigates teachers' confidence to use ICT devices for their teaching and learning demonstrates how parallel analysis was implemented to generate eigenvalues from randomly generated correlation matrices. These were compared with the eigenvalues extracted from the researcher's dataset. The number of factors retained was the number of eigenvalues larger than the corresponding generated random eigenvalues.

Keywords:

Parallel Analysis, principal component analysis, factor retention, Monte Carlo, eigenvalues

INTRODUCTION

Parallel Analysis is gaining popularity, especially in the social sciences literature (Choi, Fuqua, & Griffin, 2001; Horn, 1965), however, it is infrequently used because it is not included as an analysis option in many statistical packages, such as IBM SPSS. According to Costello and Osborne (2005) the most important decision in exploratory factor analysis is the number of factors to retain, however, extracting too few or too many factors, may lead to the wrong conclusions in the analyses. Different simulations are used to conduct Parallel Analysis. The ViSta-PARAN program is a plug-in of ViSta, the Visual Statistics System (Young, 1996). Young, Valero-Mora, and Friendly (2006) explained ViSta focuses on exploratory data analysis and statistical visualization methods. Most importantly it is free and provides basic, intermediate and advanced statistical methods. Its statistical language is mainly LispStat (Tierney, 1990). ViSta Paran provides for both Principal Component Analysis and Principal Axis Factor Analysis and Parallel Analysis.
Another method is O'Connor (2000) Parallel analysis and Velicer's minimum average partial (MAP) test using SPSS, SAS, and MATLAB. These are validated procedures that are commonly utilised by statisticians. The program, "rawpar" conducts parallel analyses after the raw data matrix is first read. The cases/individuals in a survey instrument are found in rows of the data matrix; the variables are located in the columns. Parallel analyses which focus on either normally distributed random data generation or on permutations of the original raw data set are done by the "rawpar" program.

A method that is very simple and most accurate to use is the Monte Carlo simulation technique to determine the number of factors to retain in principal component analysis. This method is described step by step using an example of a survey instrument. This technique was guided by (Pallant, 2013).

METHODOLOGY AND RESULTS

Exploratory factor analysis with IBM SPSS was performed on a survey instrument which sought to explore teachers' confidence to use 12 ICT devices for teaching and students' learning. Principal Component Analysis (PCA) and Varimax with Kaiser Normalization as the rotation method was conducted. An examination of the results of PCA indicated there were two factors with eigenvalues greater than 1, explaining 60.66% and 10.34% of the variance as shown in Table 1. This was followed by an examination of the scree plot.

Components	Initial Eigenvalues			Extraction Sums of Squared Loading			Rotation Sums of Squared loadings
	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %	Total
1	7.28	60.66	60.66	7.28	60.66	60.66	6.44
2	1.24	10.34	71.00	1.24	10.34	71.00	6.06
3	.84	6.96	77.97				
4	.51	4.23	82.20				
5	.44	3.64	85.83				
6	.41	3.44	89.27				
7	.34	2.81	92.08				
8	.30	2.54	94.61				
9	.25	2.12	96.73				
10	.17	1.37	98.10				
11	.15	1.24	99.34				
12	.08	.66	100.00				

Table 1: *Total Variance Explained*

Extraction Method: Principal Component Analysis and Varimax with Kaiser Normalization

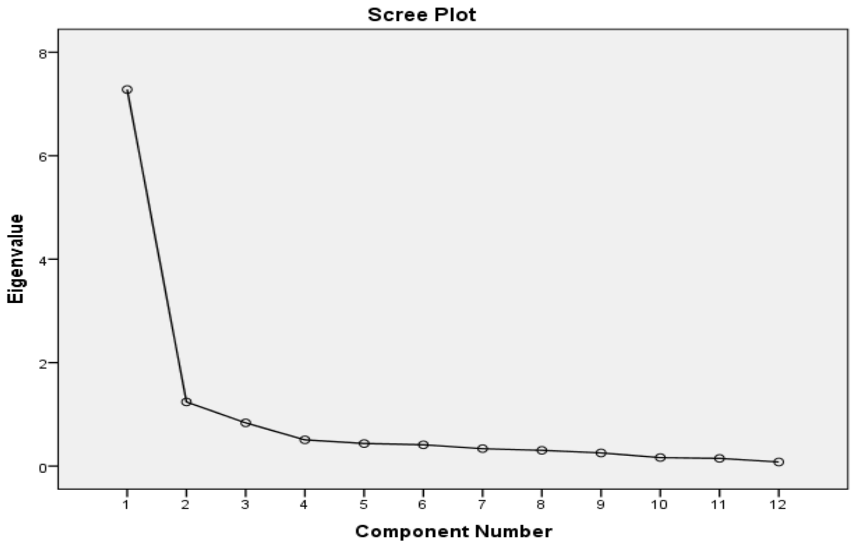


Figure 1: Scree plot

An inspection of the scree plot indicated a clear break after the first factor. To determine how many factors will be retained, parallel analysis will be conducted. Using the website, www.allenandunwin.com/spss, click on further resources, then scroll down the page to MonteCarlo.PA.zip. Right click and save to your desktop. Open and unzip the file. The following screen will be displayed:

Monte Carlo PCA
for Parallel Analysis
by Marley W. Watkins

Insert the number of variables in the first box, (for my survey I used 12), then the number of participants in your survey in the second box (226 participants completed my survey), followed by the number, 100 in the third box. Then click on calculate on the below left icon. This program will produce 100 sets of random data as the same size as your real data file. The output is shown in Table 2.

Eigenvalue #	Random Eigenvalue	Standard Dev
1	1.3946	.0590
2	1.2845	.0393
3	1.2021	.0309
4	1.1367	.0315
5	1.0773	.0259
6	1.0178	.0261
7	0.9584	.0230
8	0.8999	.0243
9	0.8517	.0268
10	0.7923	.0299
11	0.7268	.0301
12	0.6578	.0399

Table 2: *Random Eigen values generated from Parallel Analysis*

The first eigenvalue here, Table 2, (1.39) will be compared with the first eigenvalue obtained with IBM SPSS in Table 1 (7.28). Since this is larger, the factor is retained. Again the second eigenvalue from Table 2 (1.28) is compared with the second eigenvalue in Table 1 (1.24). Since this is less, the second factor is rejected. Therefore one factor solution is accepted and is accurate. The factor was labelled, 'Confidence to use ICT'.

Now examine Table 3. This output was generated with IBM SPSS during the Principal Component Analysis (PCA) and Varimax with Kaiser Normalization as the rotation method. Most of the items load strongly above .4 on the first component. This suggests that a one-factor solution is most appropriate. It is now possible to go back to IBM SPSS and force a one factor solution.

How confident are you to use the following ICT devices for your teaching and students' learning?	Component 1	Component 2
Internal software	.82	.41
External software packages	.81	.37
Word processing	.80	
Multimedia devices	.80	.36
Digital video for production and editing	.79	
Spread Sheet	.79	
Databases	.79	
Computer	.79	.46
Digital camera/document camera	.77	
Webpage design	.74	.34
World Wide Web	.73	.43
Interactive whiteboard	.63	
Cronbach's Reliability Coefficient .94		

Table 3 *Component Matrix*

CONCLUSION

This paper has explained the importance of parallel analysis as an accurate method of confirming factors present in a simple survey. Eigenvalues greater than 1 and Catell's scree plot were inspected to aid in identifying the possible number of factors that may be present in the survey. Comparing the eigenvalues obtained with IBM SPSS and the values of the random results generated by parallel analysis, confirmation of the number of factors in the survey was made. This paper also discussed two other methods of parallel analysis: ViSta-PARAN and Velicer's minimum average partial (MAP) test using SPSS, SAS, and MATLAB. It was not the intention of this paper to use all three methods discussed but to select the simplest method of Monte Carlo technique.

REFERENCES

Choi, N., Fuqua, D. R., & Griffin, B. W. (2001). Exploratory analysis of the structure of scores for the multidimensional scales of perceived self efficacy. *Educational and psychological Measurement, 61*(3), 475-489.

Costello, A. B., & Osborne, J. W. (2005). Best practices in exploratory factor analysis: four recommendations for getting the most from your analysis. *Practical Assessment Research & Evaluation, 10*(7).

Horn, J. L. (1965). A rationale and test for the number of factors in factor analysis *Psychometrika, 30*(2), 179-185.

O'Connor, B. P. (2000). SPSS and SAS programs for determining the number of components using parallel analysis and Velicer's MAP test. *Behavior Research Methods, Instrumentation, and Computers, 32*(3), 396-402.

Pallant, J. (2013). *SPSS Survival Manual: A step by step guide to data analysis using IBM SPSS (5th edn)*. sdyney. Melbourne. Aukland. London: Allen & Unwin

Tierney, L. (1990). *Lisp-Stat An Object-Oriented Environment for Statistical Computing and Dynamic Graphics* New York: John Wiley & Sons.

Watkins, M. W. (2000). *Monte Carlo PCA for parallel analysis*. PA: State College. Ed & Psych Associates

Young, F. W., Valero-Mora, P., & Friendly, M. (2006). *Visual Statistics Seeing Data with Dynamic Interactive Graphics*. NJ Wiley and Sons

A Case Study of Undergraduate Students' Perception of Passion and Creativity in Science and Technology in Taiwan

Jo-Yu Lee, jo-yu.lee@kcl.ac.uk
Kings College, University of London, UK

Abstract

The aim of this research is to investigate undergraduate students' perception of passion and creativity in science and technology in Taiwan and to examine how passion affects their creative processes. In this day and age of the knowledge economy, many companies are focusing on developing innovation and creativity in order to compete with rivals in the global market (Carayannis & Gonzalez, 2003; Egan, 2005). Because of the increasing importance of creativity in education, in the year 2000, the Ministry of Education (MOE) published a "White Paper on Creative Education" (MOE, 2002) and with the final aim of building Taiwan as a "Republic of Creativity" (MOE, 2002). Closely related to the development of both individuals and the economies of countries, creativity research has begun to include an investigation into passion (Seligman & Csikzentmihalyi, 2000). The findings of some recent studies (e.g. Chen, 2012; Hsueh, 2012; Lin, 2010) in Taiwan show that students' motivation and various creative strategies or skills can be cultivated by creating a supportive environment, and further enhance their creativity.

This research proposes that passion may inspire students' creative motivation which further enhances their creativity. Specifically, I focus on Amabile's (1996) componential theory of creativity and Vallerand, Blanchard, Mageau, Koestner, Ratelle, Leonard, Gagne, Marsolaiss' (2003) dualistic model of passion, as well as their inter-relationships is reviewed in this research. The two theories are applied because both theories have obtained empirical support in various domains in Taiwan, for example, arts and humanities (Lai, 2006; Yao, 2006), natural science (Yen, 2007), information management (Shih, 2010) sports (Shih, 2013), gambling (Wang, 2012) and internet-related activities (Chu, 2007). However, to date, studies have not been done in Taiwanese context of exploring undergraduate students' "passion" and "science and technology creativity" specifically from the perspective of Amabile's (1996) componential theory of creativity and Vallerand, Blanchard, Mageau, Koestner, Ratelle, Leonard, Gagne, Marsolaiss' (2003) dualistic model of passion.

This research is designed as a sequential mixed method approach, survey, interviews and a documentary analyses are adopted for data collection, and thematic analysis is applied for data analysis. 110 students from the faculty of computer science in Taiwan are completed the passion scale (Vallerand et al., 2003). Based on the administered test, students who show the highest, the middle and the lowest level (two students at each level) passion are selected for the interviews. Additionally, a series of interviews are completed with one academic leader, one work team member and one teacher for triangulation. Hopefully, this research will lead to outcomes that inform the development of practical ways to develop creativity in education as well as science and technology education for higher education in Taiwan.

Keywords

Passion, creativity, science and technology, undergraduate students

REFERENCES

Amabile, T. M. (1996). *Creativity in context: Update to "the social psychology of creativity."* (p. 113). Boulder, CO, US: Westview Press.

Carayannis, E. G., & Gonzalez, E. (2003). Creativity and innovation competitiveness?
When, how, and why? In L. V. Shavinina (Ed.), *The international handbook on innovation* (p. 587-606). New York: Elsevier Science.

Chu, Y. S. (2007). Passion, addiction and aggressive behaviour of online game players. Master Thesis, Graduate Institute of Information Management,
National Taipei University, Taipei.

Chen, H. I. (2012). An exploration study of the experiences of flow on the Taiko team students. Master Thesis, Graduate Institute of Education, Taiwan Shoufu University, Tainan.

Egan, T. M. (2005). Creativity in the context of team diversity: Team leader perspectives. *Advances in Developing Human Resources*, *7*, 207-225.

Hsueh, M. C. (2012). Effects of the different training approach to enhance passion of boxer and the mediating effect of motivational climate in sport with coach and peer. Master Thesis, Graduate Institute of Sports Pedagogy, Taipei Physical Education College, Taipei.

Lai, C. C. (2006). The confirmation of Amabile's component model of creativity and consensual assessment technique: The evidence of children's drawing. Master Thesis, Institute of Education, National Sun Yat-sen University, Kaohsiung.

Lin, J. H. (2010). A study of success models about what psychological characteristics adolescent athletes possess towards top-notch athletic achievements. PhD Thesis, Department of Special Education, National Kauhsiung Normal University, Kauhsiung.

MOE. (2002). *White paper on creative education*. Taipei, Taiwan: Ministry of Education (MOE).

Seligman, M. E. P., & CsikZentmihalyi, M. (2000). Positive psychology: An introduction. *American Psychologist, 55*, 5-14.

Shih, C. H. (2010). Exploring the factors affecting IM students' creativity: View of Amabile's componential theory of creativity and theory of reasoned action. Master Thesis, Department of Information Management, Ming Chuan University, Taoyuan.

Shih, P. J. (2013). Passion, intrinsic motivation and anxiety among collegiate athletes. Master Thesis, Department of Recreational Sport Management, University of Taipei, Taipei.

Vallerand, R. J., Blanchard, C., Mageau, G. A., Koestner, R., Ratelle, C., Leonard, M., Gagne, M., & Marsolais, J. (2003). Les passion de 1'Ame: on obsessive and harmonious passion. *Journal of Personality and Social Psychology, 85*(4), 756-767.

Wang, P. Y. (2012). A study of gambling passion and passion for sports viewerships of sports bettors. Master Thesis, Department of Recreational Sport Management, University of Taipei, Taipei.

Yao, T. H. (2006). A confirmation Of Amabile's creativity componential model and the consensual assessment: Writing of six grade in primary school. Master Thesis, Institute of Education, National Sun Yat-sen University, Kaohsiung.

Yen, C. L. (2007). The study of the development process on innovative science teaching material and its instruction effect in the sixth grade. Master Thesis, Institute of Education, National Sun Yat-sen University, Kaohsiung.

COLLABORATIVE LEARNING USING ICT. Creation, implementation and evaluation of pedagogical scenario

Justina Naujokaitienė
1st year PhD student, Vytautas Magnus university

RESEARCH PROBLEM

Today's students generation is called net generation - at an early age they perfectly use various digital technologies. In other words, technologies had already become an integral part of every student's daily life. It also stimulates teaching and learning environment and requires making changes. Network -generation students are ready to learn how to collaborate and to use different ICT applications, particularly in different online. In responding to this schools are well equipped with new ICT tools (computers with access to the Internet, multimedia, smart boards, etc.). However, are these technological tools are always used properly in educational process? Do teachers have adequate competencies to organize collaborative learning and purposefully select ICT tools and appropriately integrate it in the learning and teaching process?

Research object – collaborative learning using ICT.

Aim of the research - Substantiate collaborative learning using ICT in secondary schools and for development of collaborative learning create a pedagogical scenario implement and evaluate it.

Main research goals:

To make a contribution to the disclosure and holistic conceptualization of collaborative learning based on analysis of developing collaborative learning using ICT theories.

To reveal what kind of pedagogical scenario teachers are using while engaging to ICT and how the cooperative learning takes place in the lesson.

To describe tendencies of collaborative learning using ICT by interviewing secondary school teachers.

To create pedagogical scenarios based on collaborative learning methods and effective usage of ICT.

To integrate created pedagogical scenarios in the nature and social science lessons.

To evaluate the effectiveness of created and integrated educational scenarios.

Researches based on social constructivism approach often tries to expose interaction processes between individuals (Creswell, 2008). This will be done in this research also; in order to reveal collaborative learning processes in different pedagogical scenarios. Objective of the social constructivism study is to rely on the research participants in terms of the situation that is analyzed. Quantitative research design was selected, which will be based on social constructivism ontological and epistemological access. In order to reveal collaborative learning process the David W. Johnson and Frank P. Johanson (1991) group dynamic theory and activity theory will be analyzed.

METHODOLOGY OF THE RESEARCH

Following the analysis of the scientific literature, the questionnaire for the survey will be developed; aimed to find out teachers preparation to use ICT for collaborative learning and what kind of educational scenarios are used during learning.

In order to assess reliability and validity of created questionnaire there will be pilot study carried, about 200 teachers will be surveyed. With the pilot study there will be tested psychometric properties of the instrument - reliability, validity. To evaluate reliability of the scale two methods will be used: the re-evaluation (test - retests reliability) and verification of internal consistency (calculated Cronbach alpha coefficient). By calculating Cronbach alpha coefficient there will be removed the reliability (internal consistency) reducing claims.

There will be factor analysis carried out to evaluate the validity. For the validity of the method there will be content validity assessment reviewing how the content of the statements reflects on the assessment task. Once the scale psychometric properties meet the requirements a quantitative study of the teachers sample will be done.

According to the statistical data of Lithuanian Ministry of Education and Science and Education Centre of Information Technologies (2014) there are 30,552 teachers in Lithuania. The estimated sample size for quantitative study (State of art study) should be 379 participants (sampling error - 5% confidence level - 95 %). Research participants will be selected using nested sampling. Using this method of sampling there are selected not individual members but all groups, for example - teachers community in secondary school.

The aim of the state of art research is to find out whether the teachers in secondary education schools apply collaborative learning and are they able to use ICT in this context. Also what kind of pedagogical scenarios are used for implementation of this educational activities. The sample of this research will be 500 secondary school teachers. Using the larger sample there is intended to maintain the required sample size, considering that the part of the questionnaires can be filled improperly.

The results of the state of the art research will be used for development of methodological tool for teachers that offers different pedagogical scenarios for implementation of collaborative learning using ICT. Teachers will be trained how to use methodological tool.

In order to assess the effectiveness of developed methodological tool there will be quasi - experiment implemented. The participants will be teachers who were trained to use methodological tool. After a quasi - experiment, findings summarized and recommendations will be made.

Model of Learning Computational Thinking

Tauno Palts, *tauno.palts@ut.ee*
University of Tartu, ESTONIA
Margus Pedaste, *margus.pedaste@ut.ee*
University of Tartu, ESTONIA

Abstract

There is a high demand for qualified information and communication technology (ICT) practitioners in the European labour market, but the problem at many universities is a high dropout rate among ICT students, especially during the first study year. The solution might be to focus more on improving students' computational thinking (CT) before starting university studies. Therefore, research is needed to find the best methods for learning CT already at comprehensive school level to raise the interest in and awareness of studying computer science. Doing so requires a clear understanding of CT and a model to improve it at comprehensive schools. Through the analysis of the articles found in EBSCO Discovery Search, this study gives an overview of the definition of CT and presents three models of CT. The models are analysed to find out their similarities and differences in order to gather together the core elements of CT and form a revised model of learning CT in comprehensive school ICT lessons or integrating CT in other subjects.

Keywords

Computational thinking, models, comprehensive school, learning

INTRODUCTION

Computational thinking (CT) is the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form which can be effectively carried out by an information processing agent. The information processing agent can be a computer, a machine or a human being (Wing, 2006).
The research focuses within the general framework for teaching programming and computational thinking in different formats (separate subjects, embedded in science studies, visual and syntax-based programming languages) in comprehensive school and higher education institutions. Gathering together the advantages of the models of CT, a new model for learning CT can be created which includes the various dimensions of CT for learning CT at comprehensive school level.

REVISED MODEL OF LEARNING COMPUTATIONAL THINKING

This paper presents the models of CT found in the academic journals from three different perspectives. Research was done through a systematic search with the EBSCO Discovery Search tool using the following steps: The search term "computational thinking" was used to make a search in the EBSCO Discovery Search. All of the three most often applied CT approaches (Interaction between a Human and a Computer, Conceptual Model, and Engineering Design) have advantages and disadvantages. When gathering together the advantages of all the models, a new model for learning CT can be created which includes the various dimensions of CT for learning CT at comprehensive school level (Figure 4).
The revised model of learning CT includes three main components derived from previous models of CT:

1. Interaction between a Human and a Computer in the model of learning CT has a vital role and is presented in Figure 4 as the centre of the model (Cooper et al., 2010).

2. Five core elements of CT (structuralization, formalization, optimization, association and interaction, reuse or sharing) are included in the model because those five components of CT can be taught in various lessons, in various key stages and involve the core elements of CT (Wenchong et al., 2014). In Figure 4 those five elements can be seen surrounding the centre of interaction of humans and computers. The five core elements can be rotated around the centre and used dynamically without any fixed order.

3. 8 steps are included in the model to go through all the steps that occur during the process of learning CT (Massachusetts Department of Education, 2006). The first step in the process of learning CT should be identifying the problem. The arrow pointing from step 1 to step 2 in Figure 4 indicates that after the problem is identified, research needs to be done, and after that, possible solutions are developed. The steps are following each other in a linear way and the various steps more or less include the core elements of CT. The circular arrows indicate that when one problem is solved (step 8), a new one can be started from the beginning (step 1).

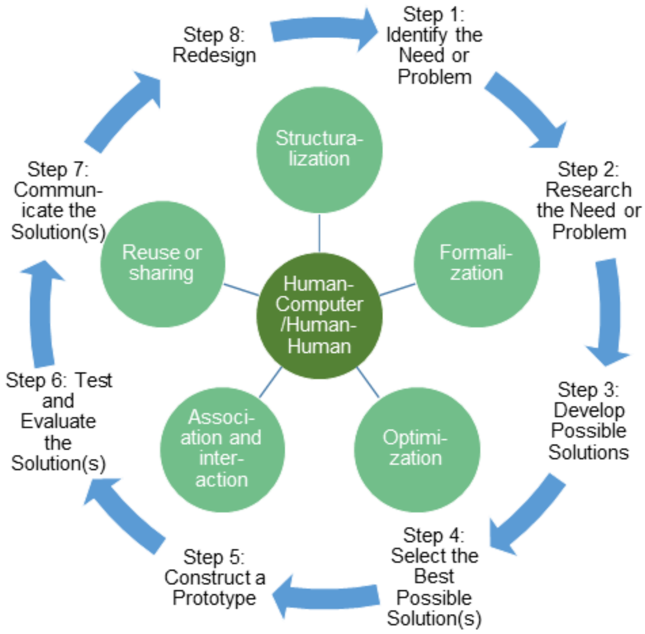


Figure 4. Revised Model of Learning CT

CONCLUSION

In this study a model of learning CT was designed for comprehensive school level ICT lessons or for integrating CT in other subjects. Through the analysis of the articles found through EBSCO Discovery Search, this study gave an overview of the definition of CT and presented three models of CT. The core concepts from the models were integrated into a revised model for learning CT in comprehensive school ICT lessons or for integrating CT in other subjects.

REFERENCES

Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3):33 – 35.

Cooper, S., Prez, L. C., and Rainey, D. (2010). Education k-12 computational learning. Communications of the ACM, 53(11):27.

Massachusetts Department of Education. (2006). Massachusetts science and technology/engineering curriculum framework. Massachusetts Department of Education http://www.doe.mass.edu/frameworks/scitech/1006.pdf.

Wenchong, S., Maohua, L., & Hendler, P. (2014). Computational Features of the Thinking and the Thinking Attributes of Computing: on Computational Thinking. Journal Of Software (1796217X), 9(10), 2507. doi:10.4304/jsw.9.10.2507-2513

Tauno Palts is an assistant in didactics of informatics and PhD student at the University of Tartu.

Intelligent personalized learning

Airina Savickaitė
Vilnius University, Lithuania

Abstract

The World Wide Web could be named particular provide a unique platform to connect learners with educational resources. Educational material in hypermedia form in a Web-based educational system makes learning a task-driven process. Consequently, many researchers have focused on developing e-learning systems with personalized learning mechanisms to assist on-line. Aalthough most personalized systems consider learner preferences, interests and browsing behaviors when providing personalized curriculum sequencing services. Systems usually neglect to consider whether learner ability and the difficulty level of the recommended curricula are matched to each other. It is therefore worthwhile to continue researches in this area of learning. Find solutions to personalized e-learning services to increasingly higher quality of education

Research questions and motivation

“Lifelong learning”, “real time studying”, “personalized learning”, is the most popular terms for our life necessity. It is appropriate to improve it, which helps to create a more educated generation. E-learning development, this is one of many ways to improve the learning environment. The aim of this article is to set the changing of e-learning algorithm in relation to the learner. For this aim are fallow tasks:

1) To review the literature of e-learning development personalized.
2) Compare the two e-learning models.
3) Create the direction of the exchange algorithm.

BACKGROUND

It is appropriate to analyze and improve the e-learning area, it is important for us and our generation. We are entering the e-World. One special part of it is an e-University that supports e-Learning and provides e-Teaching services. Graduates must be well prepared for working in, with and through the e-World. e-Learning and e-Teaching services are necessary to support modern students learning and help academics to provide excellent teaching. (Goscinski, AM 2003) On the other hand e-learning is in serious doubt and makes discuss the quality of knowledge and skills. This is another aspect of why e-learning need to be more research. The increased use of e-learning techniques as an accepted form of teaching has resulted in a growing volume of academic research dedicated to their assessment. Despite the importance of the technique, there is little comprehensive knowledge on e-learning, especially in non-educational fields. (Chen, Liang-Chu 2008) Focus on individuality and personal skills currently very successful emphasis on teaching (and learning) form for learning new knowledge and skills. Customizing a learning environment to optimize personal learning has recently become a popular trend in e-learning. Because creativity has become an essential skill in the current e-learning epoch, this study aims to develop a personalized creativity learning system that is based on the data mining technique of decision trees to provide personalized learning paths for optimizing the performance of creativity (Lin, Chun Fu 2008) Personalization is becoming a key issue in designing effective e-learning systems and, in this context, a promising solution is represented by software agents. Usually, these systems provide the student with a student agent that interacts with a site agent associated with each e-learning site.(Rosaci, 2014) One way to personalize the e-learning environment, adapting it to different learner devices. E-learning has been a revolution in recent years in training field. This, combined with the increased use of mobile devices has caused the emergence of m-learning. Hence new problems have appeared in the training field, such as displaying correctly learning contents in a mobile device that has restricted features or taking into account the learner's context in the learning process, who could be anywhere. (Cabot, 2000)

No matter how different devices are available for e-learning environment, it is appropriate to emphasize the standards by which quality is maintained. According. (Iraklis, Varlamis 2006) the roadmap to achieve standardization of e-learning technologies comprises the following steps:

1. First we should overview the e-learning process as a whole. We must define the operations included in the e-learning process, the information exchanged (input, results etc). In this step we should stabilize the existing practices and record the existing standards and needs.

2. The second step is to locate the main standardization bodies and have them work for the common aim. International boards must decide on the standards by taking into account the needs reported by the national form.

3. The third step concerns the thoughtful definition of specifications. The specifications should cover all possible needs of e-learning systems and avoid redundancies.

4. The final step comprises the dissemination of specifications and their stabilization into standards. Once they are defined, the specifications are communicated to the community for testing. Additional requirements or modifications are covered in this step. Once approved, the specifications become standards.

In order to understand the future evolution of e-learning algorithm for learner, it is appropriate to highlight what is clear about the future of E-Systems. According (Tom Vender 2005) it is equally important to know what is already clear for future decisions. This advanced form of digital learning is designed to provide:

1. Intelligent analysis of a student's solutions: The system interacts with the student by analyzing the data from the student's actions in real time as he solves problems, explores concepts, and makes decisions
2. Interactive problem solving support: Extensive, detailed feedback provides prompts to the student that encourage him to rethink his strategies and solutions, and ultimately correct his own misunderstandings or mistakes by furthering his understanding; instead of simply "telling" a novice student what the "next step" of an expert's strategy would be, the system emulates the questions and feedback of an effective teacher Beyond passively delivering previously recorded lessons, presenting digitized textbooks, or providing memorization drills, the best educational use of technology involves assessments and levels of interactivity that approximate human levels of coaching.
3. Curriculum sequencing: The system sequences each student's progression through the modularized curriculum by providing an optimal planned sequence of lessons and curriculum units as the student demonstrates readiness
4. Multiple learning experiences: The intelligent adaptive learning system uses pedagogically sound approaches that support students as they learn important concepts and skills; tasks are meaningful, at an optimal level of difficulty for the student, and contextualized in ways that enable students to build schemas so they can make sense of the concept within the world around them
5. Customized presentation and pace: Diagnostic, adaptive assessments are embedded within each lesson to assess mastery in a fluid, transparent way that doesn't create anxiety for students; as a student progresses through the system,

his pace is determined by how quickly he demonstrates mastery of a concept, thus allowing pace to vary between learners

E-learning systems comparison

About e-learning systems personalization steps are referred to in the (Cabot, AG 2000) article. The adaptively employed in educational hypermedia systems via a student overlay model provides the mechanism for personalization at the content level. System provides annotated associative links within the system That trial can address the WWW in total, or the local file space. Media rich items can be stored locally to avoid problems associated with delay. It also highlights the student grouping. Link hiding (Brusilovsky 1996) Stereotyping (Kay 2000) within the student modeling structure, appears to be an appropriate technique where distinct student groups with similar characteristics can be identified. The coupling of different content presentations to the stereotyping system allows for variance in the content presented (adaptive presentation (Brusilovsky 1996)).

It is also mentioned student task analysis and evaluation capabilities. That feedback is achieved by a student in an appropriate form. Assessment at the concept level is based on a competency model which is administered online; where a concept is a skill or task, students are asked to complete a task and have it registered manually. For example, the concept we are dealing with might be the scanning of an image. The concept might be demonstrated and the student asked to perform the task; completion of this task is manually registered. Suitability should also pertain to both the course model (pre-requisites, outcomes) and the formal assessment, and students. (Kay 2000) The following hints about e-system opportunities for a role within a number of theory. The fact that a learning objects can have a role within a number of theories, constitutes a network. The fact that some phases precede others is analogous to the pre-requisite relationship of the concepts within AMLE. With more instructional methodologies defined in terms of phases with corresponding learning objects in the LE, the student too might select a particular methodology; here the LE would automate the delivery by searching the network of objects (Dr. Jill Slay, 2000) The analysis of modern learning should be akcentas motivation to learn. According (Vassileva, J 2008) generally, the motivation for learning is to satisfy a short-term goal; it is “solution-driven” rather than “learning on principle.” Often, the motivation for learning is social, e.g., finds a curious fact to impress peers or to help with a task that a group in which the learner is a member at the moment has undertaken together (e.g., finding a strategy for a particular type of attack in an online multiplayer game or a group project in class)

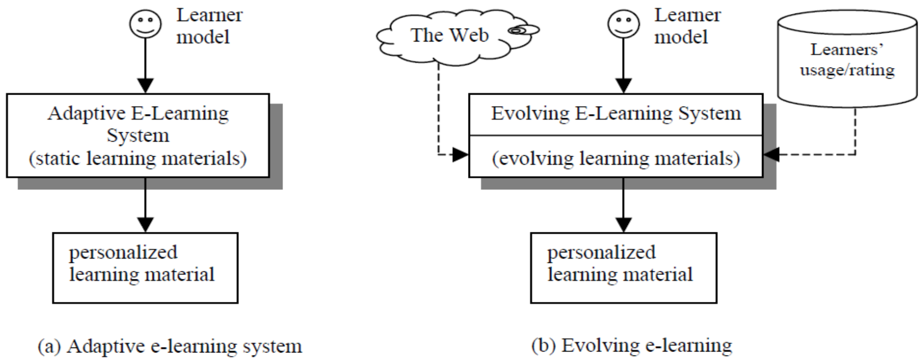


Figure 2. A comparison of evolving e-learning system vs. adaptive e-learning system. (Tiffany, YaTANG 2010)

In Figure 2. are two kinds of collaboration in the system. One is the collaboration between the system and the user, another is the collaboration between the system and the open Web (Vassileva, J 2008). Users do not have direct interactions with the open Web. Learners are free to learn whatever they want, which is different from the majority of other web-based learning systems. The system shown in Figure 3 diversification (Vassileva, J 2008) is bility to make smart, adaptive recommendations based on the system's Observations of learners' activities throughout their learning, and theme cumulated ratings given by the learners. Each paper is tagged based on its content and technical aspects. Learners are required to give feedback (ratings) towards the papers recommended to them. Therefore, according to both the usage and ratings of a paper, the system will adaptively change a paper's tags, and determine whether or not the paper should be kept, deleted or put into a backup list. Two of the major techniques that would be adopted include collaborative filtering and data clustering which have seldom been reported in the artificial intelligence in education literature.(Tiffany,YaTANG 2010)

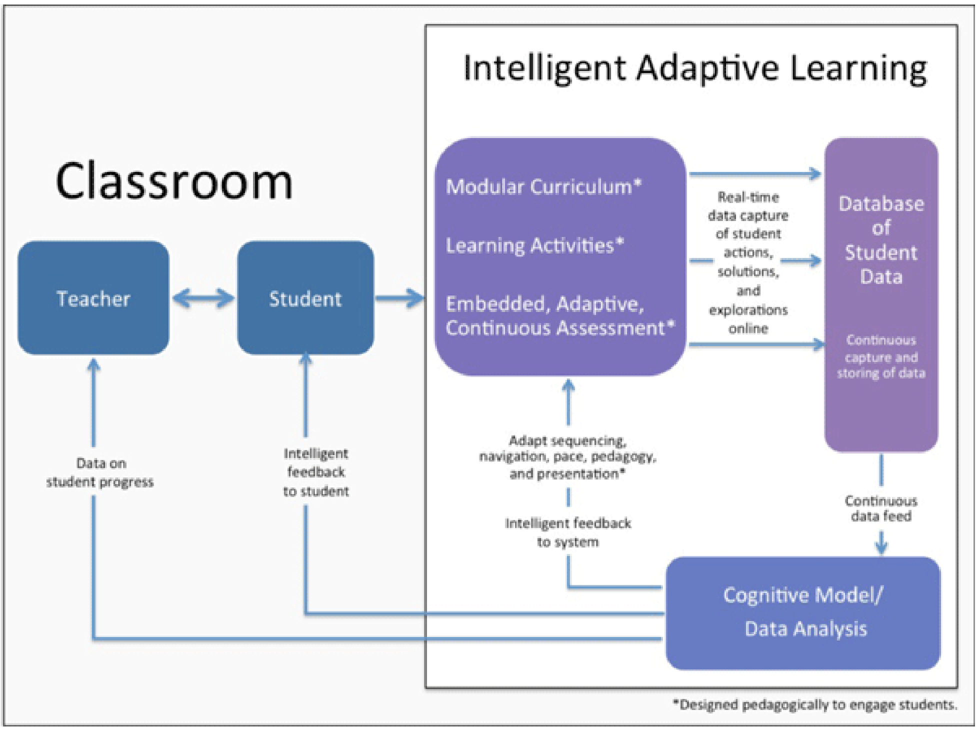


Figure 3. E-learning Zone of Optimized Learning .(Tiffany,YaTANG 2010)

One of today’s next generation technology innovations now available to schools is intelligent adaptive learning (IAL), which serves to individualize and, to some extent, personalize learning for each student. Intelligent adaptive learning is defined as digital learning that immerses students in modular learning environments where every decision a student makes is captured, considered in the context of sound learning theory, and then used to guide the student’s learning experiences, to adjust the student’s path and pace within and between lessons, and to provide formative and summative data to the student’s teacher. This type of learning tailors instruction to each student’s unique needs, current understandings, and interests, while ensuring that all responses subscribe to sound pedagogy. (Tiffany,YaTANG 2010)

Prognostication

The following is a comparison of the described previously in the e-learning algorithms Systems projections into the future. It should be noted that the older algorithm predictions of the future in line with the current e-learning system algorithm. Under this situation is the present-future prediction of e-learning algorithm.

1. how learners can be *actively engaged* in the learning process.	1.1. Adapt the sequencing of the curriculum and associated learning experiences. 1.2. Individualize the pace of learning. 1.3. In A Cost Effective Way?
2. tools and activities that promote the *construction* of new student knowledge in student centered ways;	2.4. Student engagement through gaming. 2.5. The effectiveness of tutoring through a continuous stream of intelligent feedback to the user, the system, and the teacher.
3. how learners can work *collaboratively*;	3.1. Student engagement through gaming
4. ensuring engagement in complex and ill-structured contextual problems;	4.1. The effectiveness of tutoring through a continuous stream of intelligent feedback to the user, the system, and the teacher. 4.2. Individualized sequencing of the curriculum and associated learning experiences for each student. 4.3. The regulation of cognitive load. 4.4. The individualized pace of learning. 4.5. Get The Learning Results.
5. how best to provide for *student reflection* of learning;	5.1. Individualized sequencing of the curriculum and associated learning experiences for each student 5.2. The regulation of cognitive load. 5.3. Attainable Through Tutoring.

Tale.1 E-learning algorithm identity in 2000 and 2014 years

2014 years	**2017 years**
1.1 Adapt the sequencing of the curriculum and associated learning experiences. 1.2 Individualize the pace of learning. 1.3 Cost Effective Way?	Some of the e-learning environment combined with other e-sources. E-health systems, e-banking, e-sports, ... and so on.
2.1 Student engagement through gaming. 2.2 The effectiveness of tutoring through a continuous stream of intelligent feedback to the user, the system, and the teacher.	E-learning system combined to social networking
3.1 The effectiveness of tutoring through a continuous stream of intelligent feedback to the user, the system, and the teacher. 3.2 Individualized sequencing of the curriculum and associated learning experiences for each student. 3.3 The regulation of cognitive load. 3.4 The individualized pace of learning. 3.5 Get The Learning Results.	Other e-learning system participants (different status, age, social group is very good) after seeing the results of the same tasks.
4.1 Individualized sequencing of the curriculum and associated learning experiences for each student 4.2 The regulation of cognitive load. 4.3Attainable Through Tutoring.	Other e-learning system participants (different status, age, social group is very good) after seeing the results of the same tasks

Table. 2 E-learning algorithm identity in 2000 and 2010 years

The first algorithm shows e-learning environment components. Interface, data sources, platforms between sources of learning and learners.

Comparing the first (Figure.4) and the second (Figure 4) algorithm could be clear that the extended options environment. You can choose not only e-learning course, supporting groups, communication methods test methods, detailed reporting.

The final algorithm indicated in the appropriate connection of the other elements of the environment. This connection is intentional so that e-learning systems will use the same environmental tools (coding languages, databases, etc.), as well as other platforms listed in Figure 6. In order to e-learning systems personalization are significant environmental connection to other social environments and their consumer data.

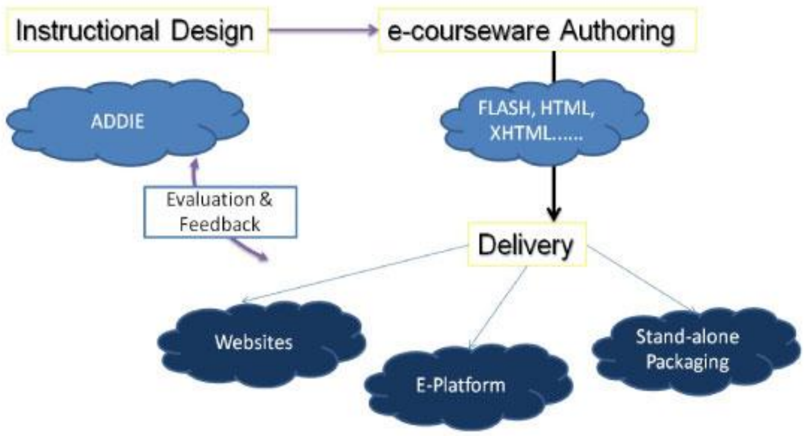


Figure 4. E-learning system content algorithm in 2000

In review both of algorithms is clear that the personalization of e-learning system environment appear to need the total internet users information and database from different environment, community information. Therefore, it is appropriate to say that will be carried out in different systems or parts connection.

Figure 5. E-learning algorithm in 2014

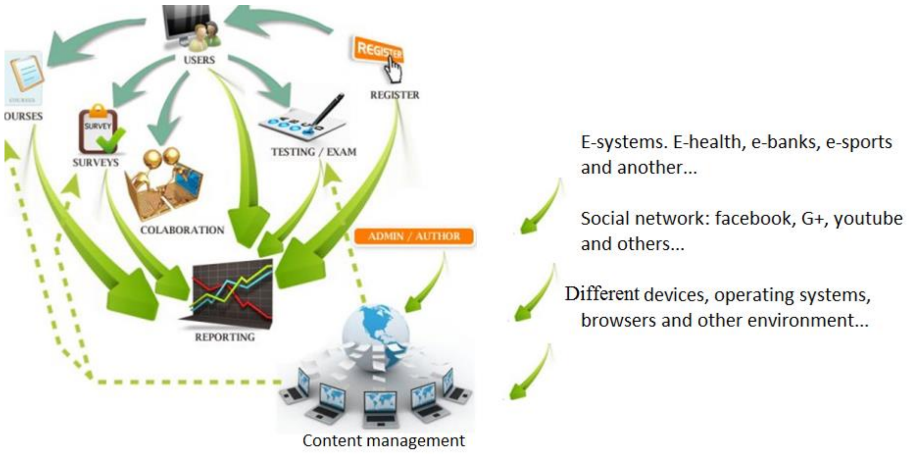


Figure 6. E-learning algorithm with plug-in from another environment

Research plan

1. objective and tasks;
2. review of the literature;
3. e-learning personalizing comparison;
4. e-learning changes prognostication;
5. findings;

Research approach and methodology

Comparison and prognostication

Findings

1. According to the material of theoretical background, the e-learning systems are created for a modern person who is seeking for a lifelong learning. Which means that the e-Learning System to be adaptable to different software, the, different browsers, different devices

2. Compared to a decade difference with e-learning algorithms systemes is clear that the trend does not change. The focus is on the learner's ability to make it easier to use e-learning system to motivate learners, to deepen the understanding of knowledge. Knowledge must be available regardless of time or a social group.

3. Forecast to e-learning algorithm variation of it can be said that there will be integration with other E-Systems and social networks. It should be noted that to see ourselves through others is one of the most important methods of learning, so it is necessary for the high youth and consumer base, and data flows

Main expected contributions of your PhD work

This article is part of the research literature review dissertation "Intellectual multi-agent e-learning system to personalize learning study" Purpose of literature review is to set (identification) scientific problems posed challenges related to intellectual multi agent development of e-learning system to personalize learning

Current challenges and questions

1. Lector's role is also very important. Lectors must have excellent skills for communication and provide their experience and knowledge. Training classroom

teacher (or lector) should be able to present information in a clear way, have a lot of sources of information, real examples for problem salvations in practice tasks. Not only to provide knowledge. It is appropriate to carry out environmental studies lecturer personalization of e-learning environment to a higher quality of learners' education.
2. Learner assessment of knowledge is also important to investigate how changing knowledge assessment algorithm. Knowledge assessment methods, data and other calculations are also affected personalized e-learning to build.

LITERATURE

1. Goscinski, AM (Goscinski, AM); Silcock, JJ (Silcock, JJ) Edited by:Zhou, W; Nicholson, P; Corbitt, B; Fong, Toward supporting e-Learning and providing e-Teaching services for e-World J ADVANCES IN WEB-BASED LEARNING - ICWL 2003, PROCEEDINGS Book Series: LECTURE NOTES IN COMPUTER SCIENCE
2. Dalia BAZIUKAITE Making Virtual Learning Environment more intelligent: the problem of software agent's mental state recognition Liet. matem. rink, 45, spec. nr., 2005, 459–464 [interactive]. [accessed on 1-07-2015] <ftp://ftp.science.mii.lt/pub/publications/45_TOMAS(2005)/taik_matematika/baziuk.pdf>
3. Lin, CF (Lin, Chun Fu); Yeh, YC (Yeh, Yu-chu) ; Hung, YH (Hung, Yu Hsin) hang, RI (Chang, Ray I.) Data mining for providing a personalized learning path in creativity: An application of decision trees ByCOMPUTERS & EDUCATION Volume: 68 Pages: 199-210 DOI: 10.1016/j.compedu.2013.05.009
4. Cabot, AG (Cabot, Antonio Garcia); Garcia-Lopez, E (Garcia-Lopez, Eva) ; De-Marcos, L (de-Marcos, Luis); Abraham-Curto, J (Abraham-Curto, Javier)Adapting Learning Contents to Mobile Devices and Context to Improve Students' Learning Performance: A Case Study By: nJOURNAL OF UNIVERSAL COMPUTER SCIENCEmVolume: 20 Issue: 15 Pages: 2032-2042 Published: 2014 TIONAL INTELLIGENCE Volume: 26 Issue: 2 Pages: 121-141 Published: MAY 2010
5. Iraklis, Varlamis; The Present and Future of Standards for E-Learning Technologies Athens, Greece 2006 [interactive]. [accessed on 1-07-2015] <http://ijklo.org/Volume2/v2p059-076Varlamis.pdf>
6. Tom Vander Ark CEO The Future of Learning: Personalized, Adaptive, and Competency-Based, Getting Smart [interactive]. [accessed on 1-07-2015] <http://cdno3.gettingsmart.com/wp-content/uploads/2011/08/D1-white-paper-the-future-of-learning-personalized-adaptive-and-competency-based.pdf>
7. Narciss, S (Narciss, Susanne) ; Sosnovsky, S (Sosnovsky, Sergey) ; Schnaubert, L (Schnaubert, Lenka) ; Andres, E (Andres, Eric) ; Eichelmann, A (Eichelmann, Anja) ; Goguadze, G (Goguadze, George) ; Melis, E (Melis, Erica) Exploring feedback and student characteristics relevant for personalizing feedback strategies COMPUTERS & EDUCATIOR
8. Spalter, AM (Spalter, AM); van Dam, A (van Dam, A) COMPUTERS & GRAPHICS-UK Volume: 27 Issue: 3 Pages: 329-337 DOI: 10.1016/S0097-8493(03)00027-X Published: JUN 2003View Journal Information
9. Zeying Wan[a], Deborah Compeau[b] & Nicole Haggerty[cn]pages 307-n40Publishing models and article dates explained Published online: 08 Dec 2014
10. Peng, J (Peng, Jun) MH (Wang, Minhong); Peng, J (Peng, Jun); Cheng, B (Cheng, Bo); Zhou, H (Zhou, Hance); Liu, J (Liu, Jie). Knowledge Visualization for Self-Regulated Learning Wang, 2008 EDUCATIONAL TECHNOLOGY & SOCIETY
11. Garcia-Cabot, A (Garcia-Cabot, Antonio)[; de-Marcos, L (de-Marcos, Luis); Garcia-Lopez, E (Garcia-Lopez, Eva) An empirical study on m-learning

adaptation: Learning performance and learning contexts COMPUTERS & EDUCATION)

12. Rosaci, D (Rosaci, Domenico); Sarne, GML (Sarne, Giuseppe M. L.)(EFFICIENT PERSONALIZATION OF E-LEARNING ACTIVITIES USING A MULTI-DEVICE DECENTRALIZED RECOMMENDER SYSTEM COMPUTATIONAL INTELLIGENCE) Volume: 13 Issue: 1-2 Pages: 67-86 Published: MAR 2014
13. Hammami, S (Hammami, Salah); Mathkour, H (Mathkour, Hassan) WEB APPLICATION FOR AN ADAPTIVE MULTI-AGENT E-LEARNING SYSTEM: A CONTINUOUS IMPROVEMENT OF E-CONTENT JOURNAL OF WEB ENGINEERING Volume: 13 Issue: 1-2 Pages: 67-86 Published: MAR 2014
14. Vassileva, J. Toward Social Learning Environments Learning Technologies, IEEE Transactions on Year: 2008, Volume: 1, Issue: 4 Pages: 199 - 214, DOI: 10.1109/TLT.2009.4 Cited by: Papers (34) IEEE Journals & Magazines.
15. Lina Yang; Sch. of Educ. Technol. & Inf., Tianjin Foreign Studies Univ., Tianjin, China ; Zhijun Yan Published in: Electrical and Control Engineering (ICECE), 2011 International Conference on Date of Conference:16-18 Sept. 2011 Page(s):6689 - 6692 Print ISBN:978-1-4244-8162-0 INSPEC Accession Number:12355273 Conference Location :Yichang DOI: 10.1109/ICECENG.2011.6056936 Publisher: IEEE
16. Tiffany Ya TANG; Gordon MCCALLA Smart Recommendation for an Evolving E-Learning System [interactive]. [accessed on 1-07-2015] <http://sydney.edu.au/engineering/it/~aied/vol10/vol10_TangMcCalla.pdf>
17. Dr. Jill Slay; Frank. Kurze Towards a personalised Learning Environment Frank Kurzel, 2000 [interactive]. [accessed on 1-07-2015] <http://ausweb.scu.edu.au/aw03/papers/kurzel/paper.html>
18. Brusilovsky, P. (1996). "Methods and Techniques of Adaptive Hypermedia." User Modeling and User-Adapted Interaction **Vol 6**(2-3): 87-129.
19. Kay, J. Stereotypes, Student Models and Scrutability. ITS2000, Montreal, Canada, (2000).Springer-Verlag.
20. Oppermann, R., Rashev, R. & Kinshuk (1997). Adaptability and Adaptivity in Learning Systems.Knowledge Transfer. A. Behrooz. London, UK, pAce.**11:**173-179
21. Wiley, D. A., C.Learning objects need instructional design theory, IN, The 2001/2002 ASTD Distance Learning Yearbook, McGraw-Hill, New York.
22. Sabine, Graf; Fuhua, Lin; Kinshuk, Rory; cGreal Intelligent and Adaptive Learning Systems: Technology Enhanced Support for Learners and Teachers ISBN-13:978-1609608422 ISBN-10:1609608429

An AR Tool for Understanding Quicksort Algorithm

Maiko Shimabuku, Kazuto Tsuchida, Seiichi Tani, Tomohiro Nishida,Yoshiaki Nakano, Susumu Kanemune

Abstract

Computer Science Unplugged (CSU) is an effective educational method to learn concepts of Computer Science through hands-on activities that do not require the use of a computer. However, in some cases, it is difficult for teachers to prepare enough numbers of tools for CSU activities. Therefore, we developed AR balance scale (AR-BS) for a CSU activity to learn sorting algorithms. We also propose some worksheets for understanding quicksort programs using arrays. In our experiment, AR-BS with the worksheets helps to improve learners' understanding quicksort programs.

Keywords

AR, CS Unplugged, Quicksort, Programming Education

INTRODUCTION

Computer Science Unplugged (CSU) is an effective educational method to learn concepts of Computer Science through hands-on activities that do not require the use of a computer. For example, learners can understand sorting algorithms by using balance scales and weights (see the right of Figure 1). Computers usually only compare two values at once. The restriction is realised by using a set of balance scales in the activity.

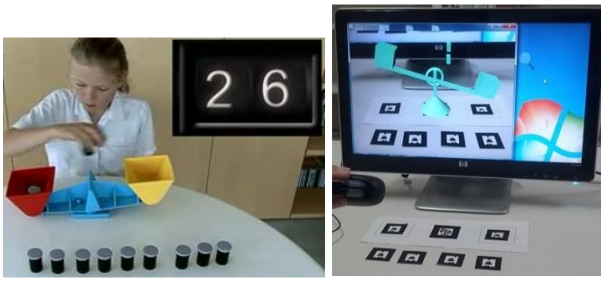


Figure 1: CSU Sorting Activity (Original and Proposal)

CSU is effective for learning CS. However, it takes labour for teachers to prepare hands-on materials such as required number of sets of balance scales and weights for students in the class. So we have developed an AR (Augmented Reality) Balance Scale (AR-BS). We expect that AR-BS has the same effect as real scale and weights. Additionally, we have developed worksheets intended to hep learners to understand quicksort programs.

AR BALANCE SCALE

As shown in the right of Figure 1, in AR-BS we provide a paper balance scale and paper weights and show a set of virtual balance scale on a computer screen in AR-BS. AR-BS is suitable for large class because teachers have to prepare computers, papers and web cameras and do not have to prepare real scales and weights. We also display how the number of times of the comparisons on the scales. Therefore, learners can easily confirm the performance of the algorithm that they are trying. When learners use real weights, there is a possibility to guess those weights by their hand. On the other hand, there is no possibility to guess because learners use paper weights with AR markers, which have same a shape but different weights. Marker indicates which weight can be decided at random each time. Therefore, the same paper weights can be used next time. We use Java and OpenGL to develop the system.

WORKSHEETS TO LEARN QUICKSORT PROGRAMS

CSU is effective to learn sorting algorithms. The following is an outline of the quicksort algorithm: (1) Choosing a “pivot” from the values in the array; (2) Dividing the values into two partitions: less or greater than the pivot; (3) Sorting each partition recursively. Learners would understand the outline through CSU activities. However, there something difficult to understand sorting programs with an array. According to the experience of authors, the cause of the difficulty seems to lie in the complexity of manipulating data in an array recursively. Therefore, we propose two kinds of worksheets for learning quicksort programs (see Figure 2). The sheet at the left side of Figure 2 helps to understand how to divide the values into two partitions based on the pivot. The sheet at the right of Figure 2 shows the whole recursion process.

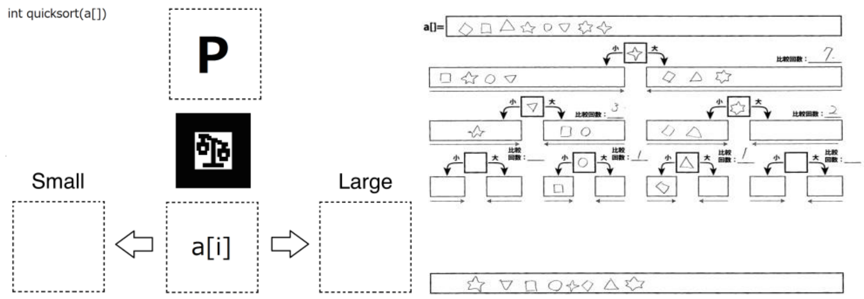


Figure 2: Worksheets of quicksort.

EVALUATION

We carried out an evaluation experiment, the subject of which are 16 undergraduate students who major information science. We divided the subjects into three groups. Group A (n=7) used AR-BS and worksheets to learn quicksort programs. Group B (n=4) used only AR-BS. Group C (n=5) used only an algorithm textbook. They also had pre- and port- test. We asked how values changed in an array in Question 1, and the role of variables in Question 2. We found a significant difference between pre- and post-test in Question 1 for group A (Table 1) and no significant difference in other questions. The results suggest that using a combination of AR-BS and the worksheets is effective for improving learners to understand quicksort programs.

	Pre-test	Post-test	T-test
Question 1	1 (14.2%)	5 (71.4%)	p=0.03176
Question 2	4 (57.1%)	4 (57.1%)	

Table 1: Comparison of the test result of the learning group A (n=7)

CONCLUSION

CSU is effective for learning of sorting algorithms. However, it is sometimes difficult that teachers prepare real scales and weights. Therefore, we have developed AR-BS, which can be used for learning sorting algorithms on the computer screen same as CSU and the worksheets. We confirmed that it is effective to use AR-BS and the worksheets for understanding quicksort programs.

ACKNOWLEDGEMENT

This work was supported by JSPS KAKENHI (25350214).

REFERENCES

Tim Bell, Ian H. Witten, Mike Fellows: Computer Science Unplugged - An enrichment and extension programme for primary-aged children (2005).

Seiichi Tani, Susumu Kanemune, Yoshiaki Nakano. Current Status of Information Study in senior high school in Japan. Open Conference on Informatics Fundamental and Computer Fluency, Druskininkai, Lithuania (2011).

Tomohiro Nishida, Susumu Kanemune, Yukio Idosaka, Mitaro Namiki, Tim Bell, Yasushi Kuno. A CS Unplugged Design Pattern. SIGCSE2009, Chattanooga, Tennessee USA, (2009).

Computational modelling of the fundamental informatics concepts through interactive tasks

Gabrielė Stupurienė
Vilnius University, Institute of Mathematics and Informatics, Lithuania

- Brief Background
  Concepts of informatics play a central role in all curricula and standards for informatics education at secondary schools. In practice at schools however very often the training of skills in application software is given much more room than the understanding of fundamental concepts of informatics. The international Bebras contest on informatics and computer fluency has the goal to convey informatics concepts to as many school students as possible in a way that can motivate them to be more interested in informatics (*Dagienė V., Futschek, G. (2010) Introducing Informatics Concepts through a Contest. // IFIP working conference: New developments in ICT and education. ISBN 9782953728514*).
  Interactive tasks are more attractive and very necessary type of tasks. There are problems with tasks representation in different operating systems, lack of export tools.
  "Bebras Lodge" is a tool for creating and developing interactive tasks. Its aim is to make the process of interactive task creation quicker and simpler. The drag & drop interface makes the tool intuitive, usable by non-professional programmers, and the Javascript framework with SVG support gives you freedom to create something more complex.
- The aim of research – to create recommendation lists of the fundamental informatics concepts according to the age group of school students. To prepare methodology of interactive tasks development and create (or adapt) a computer model.

- The preliminary research plan is:
  - To look over the fundamental informatics concepts and to identify the problems related with them.
  - To create classifications of the fundamental informatics concepts.
  - To prepare methodology of interactive tasks development.
  - To create or adapt a computer model for creating an interactive tasks.
  - To perform expert assessment of the computer model.
- Main expected contributions of your PhD work

  First of all, it is very important to know what we need to teach during informatics (computer science) lessons. In Lithuania, a lot of time is spending for teaching word processor or spreadsheet programs.

  Interactive task creation tool will help to teach students the fundamental informatics concepts in very interesting way. To know and understand fundamental concepts are very useful for every school student.
- **Current challenges and questions**
  A popular approach to verify classification of the fundamental concepts is to ask informatics education experts. So now (in April), I am creating a questionnaire for experts.

Investigating students' engagement in an engineering degree through technology enhanced learning

Luke Turner, *lturner1@utas.edu.au*
University of Tasmania, Australia

David Wood, *D.J.Wood@utas.edu.au*
University of Tasmania, Australia

Seeta Jaikaran-Doe, *Seetam@utas.edu.au*
University of Tasmania, Australia

Peter Doe, *Peter.Doe@utas.edu.au*
University of Tasmania, Australia

Abstract

This paper presents the methodology and results of a student engagement survey administered to staff and students involved in an undergraduate engineering degree course from a regional Australian University. This survey investigated the effects of lecturer's instructional methods; the utilisation of online materials; and preference to online or face-to-face lectures relating to student motivation to attend class and their engagement. The paper also explored lecturers' opinions of changes necessary to enhance learning and improve engagement. Findings revealed although 95% of the students used on-line materials, 59% favoured face-to-face classes. The study recommends student engagement needs to be improved through better preparation of lectures, real-life examples, practical techniques such as 'hands on' activities, and the use of audio-visual materials.

Keywords

Engineering education; student engagement; instructional methods; online delivery; Australian regional university

INTRODUCTION

Engagement and Disengagement

Student engagement is an integral component in teaching and learning in the education system from pre-school to tertiary level. It is defined as a broad positive phenomenon related to active participation and attraction to tasks, time spent discussing lecture materials, persistence in the face of challenges, and enjoyment and satisfaction gained from the completion of work by university students in an academic as well as in a non-academic social environment (Markwell, 2007; Strong et al., 1997; Krause and Coates, 2008). On the other hand according to Baron and Corbin (2012), there was a perceived trend towards student disengagement which was of great concern to those directly involved in educational organization and related institutions. Not only do disengaged students suffer academically but lecturers too are discouraged and find teaching difficult when attempts are made to teach bored or inattentive classes which are not fully engaged.

Assessing engagement

Previously traditional methods of assessing engagement among students focused on student outcomes such as test scores, graduation rates, drop-out rates and attendance (Parsons and Taylor, 2011; Krause, 2005; Carini et al, 2006). Commonly, in modern research, student engagement is assessed through: attendance, participation and academic results; student/staff surveys; extracurricular participation; hours studying and time on task, and observed behaviours and enjoyment (Parsons and Taylor, 2011).

METHODOLOGY

In this study a student/staff survey was designed to assess student engagement/ disengagement in an undergraduate engineering degree program at a regional Australian university. A total of 106 students and 11 lecturers responded to the survey (Turner, 2014) which investigated: To what extent the utilisation of informatics in teaching and student learning in an engineering degree course facilitate student engagement? Items were constructed differently for students and lecturers. Students responded to both closed-ended and open-ended items whereas lecturers responded to only open-ended ones. Students were asked to select the most appropriate response from among five choices for closed-ended questions: lecturers' preparation to keep materials interesting and engaging; purpose of utilising online materials; and preference to attend face to face lectures as outlined in Table 1.

Survey items	Frequency of responses				
Lecturers keep the subject materials interesting and engaging	All the time	Most of the time	Sometimes	Rarely	Never
I use online materials to …	Support what I learn in lectures	Help study for exams	Avoid going to class	I don't use online materials	Other
Would you prefer to watch lectures online or attend class on campus?	Yes, for all classes	Yes for some classes	Maybe	No, but I would watch them later for revision	No, I learn more in class

Table 13: Survey items and Frequency of Responses (Closed-ended items)

Students were also asked to respond to following open-ended items which examined engagement from three perspectives: motivation to attend classes, changes at the university to improve learning and engagement, and examples of good learning or engagement. The questions were:

1. *I would be more excited and eager to attend class if….*
2. *What do you believe could be changed at your school to improve your learning and engagement?*
3. *List any examples of good learning or engagement at your school*

Lecturers were asked to respond to two open-ended items which focused on issues relating to learning, engagement and improvement of student engagement. The questions were:

1. *What are the most important issues for you relating to student engagement and learning?*
2. *What could be done to improve student engagement?*

RESULTS

Student responses to the three closed-ended survey items

The following bar graphs present the frequency of responses to the three closed-ended items on the student survey.

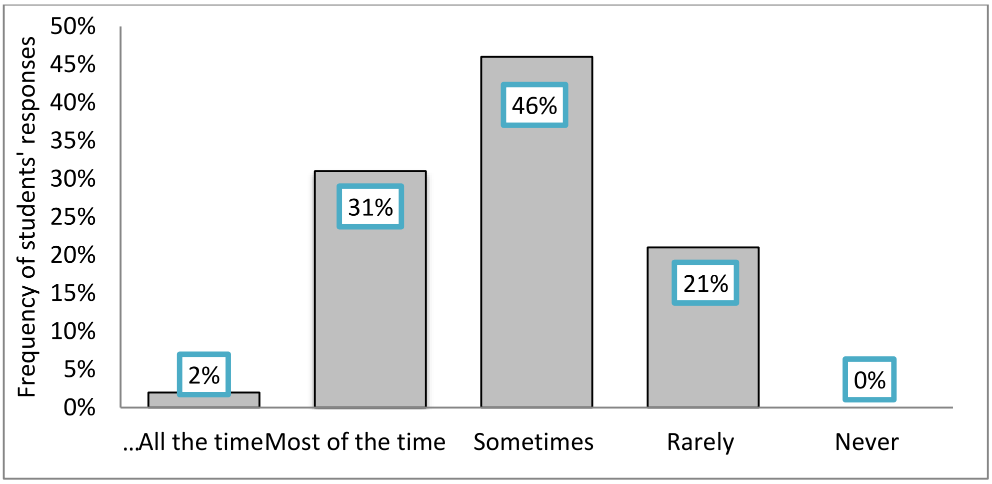


Figure 1: Frequency of responses to reflect interest and engagement of subject materials

An examination of the bar graph shows that about one third (31%) of the respondents perceived the lecturers kept the content interesting and engaging most of the time, whereas less than half (46%) believe this may be so sometimes. What is interesting; only 2% reported the content is interesting all the time and 21% responded rarely.

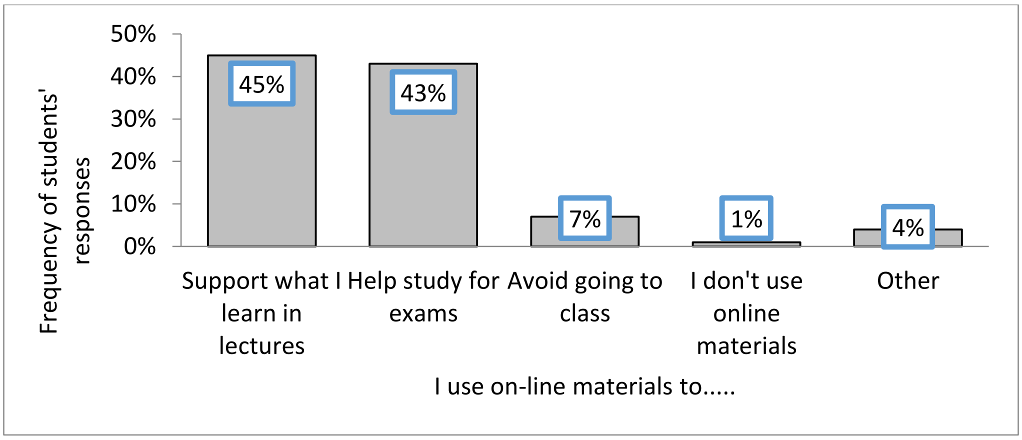


Figure 2: Frequency of responses for the purpose of on-line materials

Figure 2 captures the purposes of using online materials. A total of 95% of the students used the available on-line materials for supporting what they learned in the classroom or to help them study for exams. Only 7% reported they utilised the materials to avoid going to class, whereas 5% either did not make use of any of the materials or used them for other purposes.

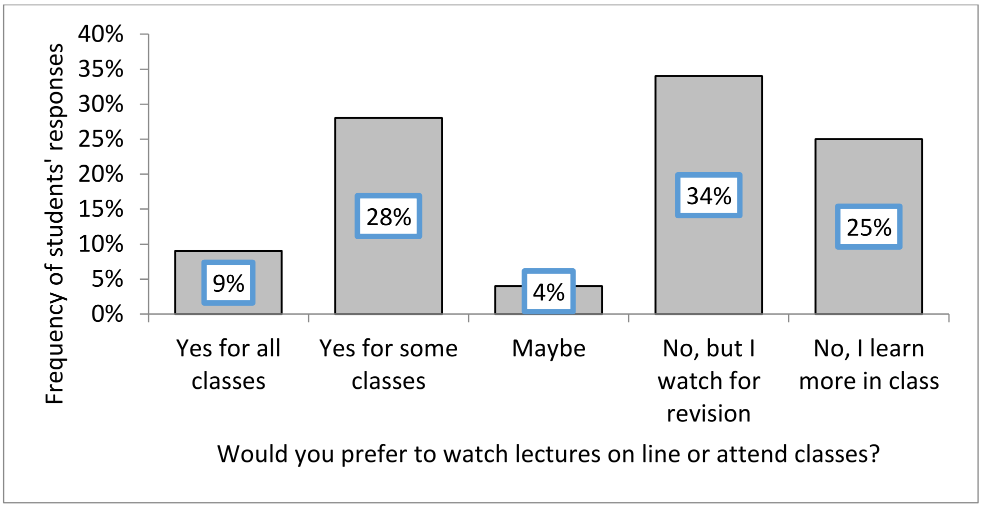


Figure 3: Students' preferences to their mode of learning: watch lectures online or attend face-to-face classes.

Figure 3 reflects a total of 34 % of the respondent preferred to watch on-line lectures only for revision. 37% indicated they preferred to watch all/some lectures on line, whereas 25% showed a preference for face-to-face classes. A total of 4% was unsure.

Student responses to open-ended survey

Students' responses to first open-ended item *"I would be more excited and eager to attend class if…."* were fairly consistent throughout the surveyed cohort to the areas and practices that would make them more eager and excited to attend classes. A common trend was the need for more real-life or relevant examples. 15% of students felt more linkages to industry, physical examples and samples of how to utilise the theory to real-world applications would make lectures more desirable. One student (S1) articulated: *"Experience is more important than writing notes."*

Twenty-one students were critical of the preparation and delivery of lectures, ~~of~~ among these 5 were of the opinion that the overuse of PowerPoint presentations negatively affected their learning with a common response along the line: (S2) *"I'd be more eager if lecturers didn't use PowerPoint so much".* They preferred lectures where the lecturer wrote notes on a board.

Also in the first open-ended item, eighteen students responded that they would be more likely to attend classes if the lecturers were more interested and engaging.

Some comment relevant to the use of informatics teaching and students learning included: *I would be more excited and eager to attend class if….*

S3 *"lecturers were better trained as actual teachers which obviously isn't really their actual job."*

S4 *"the lecture did not just consist of someone standing in front reading the slides. It appears almost all engineering lectures are all about the lecturer reading his notes. If they show videos or models or other things that give us an idea of how it works, it makes it a lot more interesting. If I wanted to read some notes about a subject I would use Wikipedia."*

S5 *"there was less use of PowerPoint slides. It is easy for lecturers to cover the material or work through examples too quickly when they are using slides. If they have to draw the whole diagram on the board, they have to slow down and explain each component they are drawing. I personally learn better when I work* *through an example myself, at the same time as the lecturer is doing it on the board. When PowerPoint is involved, I usually can't keep up."*

Student comments relevant to the second open-ended survey item *"What do you believe could be changed at your school that would improve your learning and engagement?"* included:

S6 "the only update to their teaching method over the last hundred years has been the colour of the board… it will always be more difficult to write while focusing on what a lecturer is saying, scrawling, pointing to, standing in front of or erasing before you even got to read it."

S7 "Online quizzes are under-utilised in engineering subjects."

There were very few examples of "*good learning or engagement at your school"* reported in the survey, one was:

S8 *"Interactive demonstrations and examples of concepts. Visual/graphical examples".*

Most students felt that competitions (such as the 'rat-trap boat race' and 'pasta-bridge' held among first and second year students) were beneficial to their learning.

Lecturers' responses

In response to the survey item *"What are the most important issues for you relating to student engagement and learning?"* one lecturer (L1) commented: *"There needs to be a high level of practical work in all subjects. Without that many students just do not grasp what the teaching is about and cannot link analysis to real applications."* Another (L2) stated *"….achieving specific learning outcomes, particularly the skills needed to be an engineer".*

One lecturer (L3) had clearly taken an interest and had some knowledge of integrating ICT for teaching and students' learning. This lecturer gave as an example of what could be done to improve student engagement: *"Having students prepare and present 3-5 minute video group reports in two units…The videos were assessed and contributed to the final mark; students enjoyed the experience."*

Lecturers' suggestions for improving student engagement included: *"Showing current news clips or video segments as illustrations" (L5); "Replacing lectures with more interactive teaching methods" (L6); and "lecturetools.com is nice for big classes."(L7).* These comments reflect some lecturers are aware of modern, informatics based, teaching methods that could improve students' engagement. However the fact that these were listed as suggestions indicates these techniques are not being currently employed.

There were suggestions supporting student's (S1) appeal for more hands-on and industry relevant experience: *"More teaching assistance to prepare demonstrations"(L6),* and *"Have lecturers undertake workshops to improve their technological pedagogical content knowledge"(L7).*

DISCUSSION

Students were disengaged by lecturers' unwillingness or inability to adopt modern teaching and learning techniques. Only about a third of the students surveyed found the lecturers made their subject materials interesting and engaging most or all of the time. Surprisingly, PowerPoint presentations were not favoured by a few students who preferred the pace of the traditional 'chalk-and–talk' delivery. On-line facilities were almost universally used by students, but only a narrow majority indicated a preference for face-to-face instruction over on-line. This could be a result of the lecturers' lack of interest and/or their lack of pedagogical knowledge and training in modern lecturing techniques. There were few reports of lecturers employing modern informatics-based teaching techniques, although several gave suggestions for improving student engagement. It can be concluded that more use of informatics by lecturers would improve student engagement at this university.

REFERENCES

Baron, P. & Corbin, L. 2012. Student engagement: Rhetoric and reality. *Higher Education Research and Development,* 31**,** 759-772.

Carini, R. M., Kuh, G. D. & Klein, S. P. 2006. Student Engagement and Student Learning: Testing the Linkages. Research in Higher Education, 47.

Krause, K.-L. 2005. Understanding and promoting student engagement in university learning communities. The University of Melbourne.

Krause, K.-L. & Coates, H. 2008. Students' engagement in first-year university. Assessment & Evaluation in Higher Education, 33.

Markwell, D. 2007. The Challenge of Student Engagement. Teaching and Learning Forum. University of Western Australia.

Parsons, J. & Taylor, L. 2011. Student Engagement: What do we know and what should we do? University of Alberta.

Strong, R., Silver, H. F. & Robinson, A. 1995. What do students want (and what really motivates them)? Educational Leadership, 53, 8.

Turner, L 2014, Student Engagement, Bachelor of Engineering with Honours Thesis, University of Tasmania.

WhatsApp in lower secondary school. Can technology support inclusion?

Maria Ventura, mariavent25@hotmail.com
Trade UIL-SGK Bolzano-Bozen Italy

Abstract

Mobile phones have become a regular part of daily life of teenagers. WhatsApp is a smartphone application for instant messaging. I chose to conduct a qualitative research in a German speaking inclusive class of lower secondary school, testing this new application in and outside of the classroom, analyzing the impact of WhatsApp on school performance. This research also allows a teacher behavior analysis.

Keywords

Inclusion, Lower Secondary School, Whatsapp, Smartphone application, school performance, teacher behavior

INTRODUCTION

The italian law 517/1977 abolished special classes. Nowadays our classes are provided with additional support by special needs teachers working in the class with the main teachers. In Italy class sizes are smaller than in the rest of Europe, we have a maximum of 20-25 children. Our goal is the full inclusion.

Inclusion means not only the presence of pupils with special needs in class, but also the daily work with pupils with different family structures, different traditions and different languages. Inclusion is not easy to realize. In lower secondary school. Teachers try to do their best in class, but often they use teacher-centred instruction forms, a good method for classes without big differences inside. How can the use of WhatsApp support inclusive education in lower secondary school?

DISCUSSION:

In September 2014 I created two WhatsApp groups with one class of the lower secondary school in Neumarkt. WhatsApp is a Smartphone application that operates on nearly all current types of devices and operating systems. WhatsApp replaces the existing SMS platform for a system that is free of charge. This class has fourteen-year old pupils with special needs and I have been working there since 2012. During the last two years I observed the 20 pupils of this class. Below is a table with their characters traits and the performances

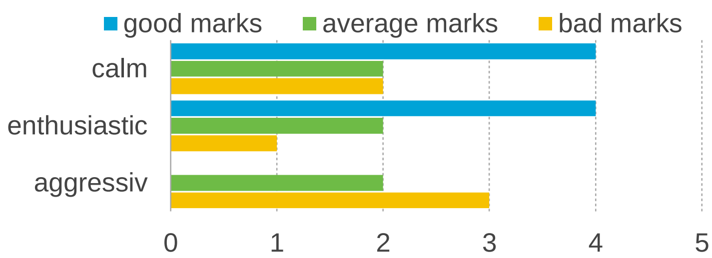


Figure 1: school marks and general features of the pupils of one class in lower secondary school of Neumarktschoolyear 2012-2014

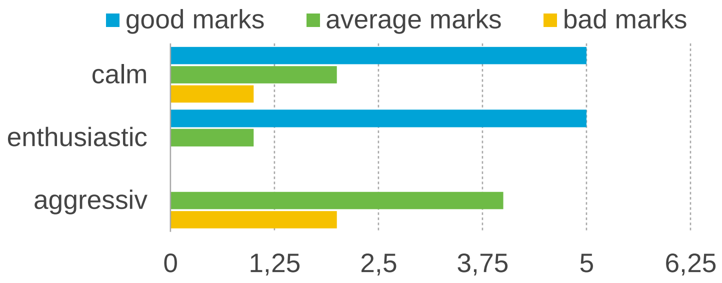


Figure 1: school marks and general features of the pupils of one class in lower secondary school of Neumarkt with the Whatsapp work schoolyear 2014-2015

In contrast to the first two years, I noticed that also the very shy and calm pupils enriched the lessons with valuable observations. And the pupils with bad marks were much more motivated to study.

Discussion sub-point 1Can Whatapp replace some classic teaching methods? I gave some homework every month to do with WhatsApp but the result was shocking.

Only 25% of the class worked at home with technology. The problem was not the smartphone or the internet. Pupils love to work with technology, but they prefer to stay in the same room while doing it. It seemed that they need both, the class situation and the presence of the teacher.

CONCLUSION

Many schools banned mobil phones from classrooms, but teachers have to find new ways to interact with students, especially in an inclusive context.

The use of technology can help to create an environment of cooperation for students who may not know exactly how to participate to the class activities. It supports the learning process of shy pupils and increases the sense of belonging. WhatsApp can support the teachers' work and improve the connective knowledge.

REFERENCES

Bouhnik, D., & Deshen, M. (2014). WhatsApp goes to school: Mobile instant messaging between teachers and students. Journal of Information Technology Education: Research, 13, 217-231. Retrieved from http://www.jite.org/documents/Vol13/JITEv13ResearchP217-231Bouhnik0601.pdf

Downer ,S. (2012). Connectivism and connective knowledge. National Research Council Canada http://www.downes.ca/files/books/Connective_Knowledge-19May2012.pdf

Maria Ventura is a German and history teacher at lower secondary school in South Tyrol (Italy). She received her Ph.D in Education from the University of Innsbruck. She is the co-author of the 3 schoolbooks *Zeitreise* (South Tyrol edition), the winner of the School Book award 2012 at the Leipzig Book Fair.