The Introduction of Computers to Primary and Secondary Schools

Arvid Staupe

The Introduction of Computers to Primary and Secondary Schools

Arvid Staupe

Department of Computer and Information Science
University of Science and Technology, Trondheim, Norway
arvid.staupe@idi.ntnu.no

Abstract. The paper presents a very first suggestion for a national plan to introduce computers into the education system in Norway in the early 1980s. Important goals was education of teachers, research in experimental schools, development of a new method for design of software, educate teachers for design software for use in education and developing software for education. Nordic cooperation, software development for education.

1 Introduction

White Paper no. 39 [8] with an action program was defined and accepted by the government at the beginning of 1984 and approved by Parliament in June 1984. The intention was to investigate whether this tool could improve education.

The major aims of the plan were: 1. Conduct experimental activity at selected schools. 2. Develop and test teaching programs or aids that provide pedagogical support for teaching in many subjects and are compatible with the school's social and cultural aims. 3. Establish mutual cooperation between schools. 4. Build up a national network of resource centers.

Important software actions included: 1. Buy good software, especially for subjects in which computer software is currently lagging behind. 2. Stimulate the build-up of national expertise and the development of software for use in education.

The implementation of special courses on developing software for education was one important action. The courses were named Grimstad courses, and the Grimstad model/Market model was used as the development model. The use of prototyping was central, as it is in agile development today.

2 The Action Program

The very first suggestion for a national plan to introduce computers into the education system in Norway was made in the early 1980s. White Paper no. 39 [1] with an action

program was defined and accepted by the government at the beginning of 1984 and approved by Parliament in June 1984. The intention was to investigate whether this tool could improve education.

The period prior to the action program is often described as a time of intensive development in the field of computer technology. Computer technology gained a stronger presence in an increasing number of areas of industry, business and other areas of society. With few exceptions the efforts made toward enhancing the use of computers in schools were concentrated on skills training, especially vocational training. Some schools offered programming courses. However, schools as a whole were totally unaware of the major developments in the field of computer technology.

Due to this situation the educational system was put under pressure. Groups of parents, professionals in the fields of industry and commerce, politicians and various institutions in society exerted various forms of pressure to incorporate computer technology and make it a part of everyday life in schools. The school system was confronted with both external and internal demands to do something.

This resulted in a white paper to Parliament in which the principles and valid guidelines for a Program of Action were laid out, and an extensive experiment with the use of computers in schools was recommended. It gave a broad survey of the possible areas in which computers could come into use and the areas of use that could be relevant in the future. The White Paper brought forward actual and fundamental questions with regard to hardware and software. It also strongly stressed the importance of a broad and varied development of competence in the field and especially in the education of new teachers and the qualification of teachers already in schools.


St. meld. nr. 39

DATATEKNOLOGI I SKOLEN

Fig. 1. St. meld. 39 (1983-84) DATATEKNOLOGI I SKOLEN. White Paper no. 39

The Action Plan concentrated on the following main areas:

- Vocational training – to improve education and make it more up-to-date and compatible with local trade and industry.
- Special education – to use computers as an aid/tool and as a remedy to reduce or overcome handicaps.
- The computer as an aid/tool in various subjects.
- Computer literacy.

These components could have considerable consequences for a series of factors in schools, such as teaching methods, curricula, relationship between teachers and students and relationship among students. In the long term, they could also have effects on the traditional role of teachers and of the school as an organization.

The following four areas were prioritized:

- Conducting experimental activities in 26 selected schools spread throughout the country in which the whole school participated. In addition, many project schools hosted small-scale experiments limited to one or a few subjects, and they only involved a small number of teachers and students.
- Establishing cooperation between different types of schools and between the schools and various computer milieus at colleges and universities.
- Developing national networks of resource centers and resource persons representing various relevant professions to give the educational system the necessary support in developing the necessary competencies.
- Creating an environment for the testing and development of educational software. Especial software for special education, the handicapped and vocational education.

The premises for these goals in the project were:

- New partitions of the students should be prevented.
- The social harmony within the school should be addressed.
- The introduction of new types of teaching problems should be avoided.
- The teaching materials should be varied and of such a quality that all the students should benefit from them.

After a long and engaged debate Parliament approved the Action Program for a period of four years. The majority of the Parliament wanted the establishment of a special project organization as a direct extension of the Ministry, and the necessary funds for establishing the Task Force were arranged. The four-year period of experimentation was intended to run from 1984-1987.

2.1 OECD Examiners´ report

At the end of the four-year period the Action Program was evaluated by the Organization for Economic Co-operation and Development (OECD), which states in the particular report entitled "The introduction of computers in schools: The Norwegian experience" [2]:

As in many countries, the initial focus in Norway for school use of computers was on teaching computer science, or on a vaguely defined attempt at computer literacy for teachers and students. What the program has achieved is a shift of emphasis to a more profitable direction: the use of the computer as a learning aid in all areas. This is a major consequence of the national programme.

The increased emphasis of viewing the computer as a teaching and/or learning toll has become accepted by a growing number of teachers. Furthermore, many more administrators are familiar with the possibilities. Increased exposure of this type is important for stimulating long-range changes in schools involving more use of computers.

The training of teachers has been recognized as the main problem to be solved for an effective use of computers in schools and great efforts have been made in that direction.

Attention has been given to the process of developing computer learning materials. Norway has advanced beyond the typical naïve approach, common in many parts of the world, where buying of hardware is considered as the main, if not the only, problem. This consideration and thought is important for extensive future developments.

The examiners noted with interest that there were instances where teachers had integrated the computer into a course, with the result that the curriculum had been extended and enhanced. Within the special education area, work was under way which contained an original idea of trying to develop hardware and software together in order to achieve a given objective. In the opinion of the examiners these activities should be encouraged in the future.

Both in the experimental schools and other schools stimulated by the existence of the national project, increased numbers of computers have made their way into schools. While these numbers are still below those in a few other countries, Norway now has a significant and growing number of computers in schools.

Managerial experience has been gained in the projects sponsored by the Task Force. This experience will be important for the future developments.

The national project has not solved all the problems associated with widespread introduction of computers in schools, but it has made good and reasonable headway in that direction.

Norway´s effort is impressive compared to many countries. At the beginning of this programme, Norway probably was lagging in this area, as compared to some other OECD countries. This is no longer the case.

2.2 Software Development

As mentioned earlier, one of the main goals was the development of expertise in software knowledge and development. Therefore, for several reasons, this had to be an area of top priority in the Action Program. The main reason for that was obvious: if computers should have any function in education, schools had to have sufficient educational software that was of high quality, both professionally and pedagogically. Another reason was to increase production of Norwegian software.

In the debate in Parliament, it was stressed that the Norwegian language and Norwegian culture had to be given special attention. Because computer languages and most of the available software were in English or other foreign languages, the importance of developing Norwegian software was particularly stressed by the Parliament.

Throughout the Action Program, very many efforts were used on qualifying courses in software design and software development, first on a Norwegian basis and then on a Nordic basis through the Nordic Council of Ministers and, to some extent, the Baltic countries. The participants of these courses were teachers, writers and representatives from publishing houses. One course also aimed at training instructors in software design and software development. They constituted a national network of competent persons and development centers, which were used in the in-service training of teachers and they were also used as advisors to those persons who developed software design and performed programming work over the country.

Approximately 100 educational software programs were developed in the period.

The Grimstad Model / The Market Model. One important task was the development of educational software. In this area, there was little tradition to build on in Norway. Most of what existed was just drills and practice. Along with IMTEC[1] the Ministry decided to build up this area. IMTEC linked the Ministry to international consultants through Les Green from Toronto, Canada and Dan Daniel from Houston, Texas. The Ministry itself had close contact with the Scottish development community through several visits and through contracts to access software from Scotland. The Scottish consultant was Alistair Fyfe from Edinburgh.

The first task was to develop an advisory team that could both directly help in the development of new software and guide schools in the use of new software. We

[1] The IMTEC Foundation, *International Management Training for Educational Change*

believed it was important that the advisory team be recruited from across the country and from different educational levels.

Naturally teachers knew best, how software could strengthen education in schools. It was therefore central to build up the skills of teachers in designing educational software.

Teachers were primarily trained to create new application ideas and to application design. They did not necessarily use programming and technical knowledge in practice. It was therefore also important to recruit and build centers for development around the country in which the programs could be completed.

It became clear in our discussion that we would have to move away from the structure of programs that existed for educational purposes, drills and practice, because these programs were either sequential/linear or built on branched tree structures. Both of these forms could have built-loop structures with repeated or alternative tasks, but the pupil could only follow certain given built-in paths. We saw the need for the development of interactive software in which the pupil could act more freely and without a given path to be followed in the program. The usual software development model at the time was the waterfall model/method in which where requirement specifications set at the start of the design and programming process.

We were fortunate that a telephone company had built a new school with dormitories and related facilities in the city of Grimstad. We were able to rent the school in the summer months of June, July and August. Grimstad on the Southern Coast of Norway is a very popular holiday resort city. For several summers, Grimstad was a center for the development of courses and teaching related to the design and development of educational software. Many hundreds of teachers from Norway and other Nordic countries participated in these programs.

The development of the Grimstad Model and the Grimstad courses began in June 1984 and were initially intended for supervisors, who were afterwards closely involved in the development of the Grimstad Model. The primary leader was Les Green, but it was also very much a team effort. Based on this work the Grimstad courses for program design and development were established. There were several courses each summer, and they were further developed, improved and repeated summer after summer over the course of several years.

A typical design course consisted of one hour of plenary/lectures/demonstrations in the morning and again in the evening, with the rest of the day dedicated to group/team work.

At the time we started the Grimstad courses, the discussion was concentrated mostly on purely technical problems related to computer hardware, operating systems and programming languages. Therefore, in the first year at Grimstad the technical staff did not speak at plenary sessions or demonstrations. The discussion comprised only pedagogy, didactics and program design, independent of hardware and software.

The central Grimstad Model dealt with metaphors, market, screen design and prototyping. The issues involved in the development of a typical program for teaching are illustrated in figure 2.

Teachers know the problems related to teaching and therefore know how to make a program strong. Based on the selected curricula, the teachers developed program ideas, design concepts.

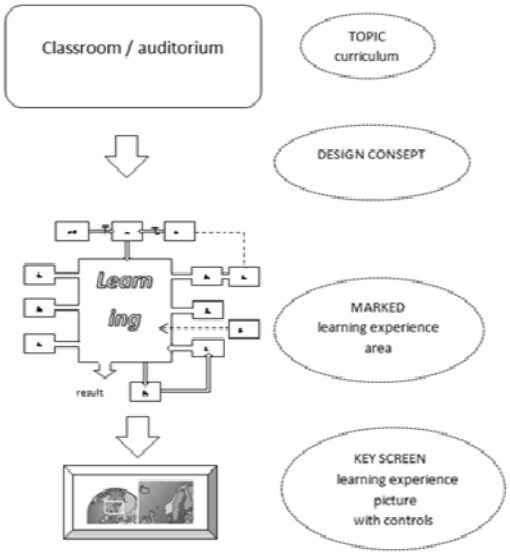

Fig 2. Grimstad model / Market model

Teachers know the problems related to teaching and therefore know how to make a program strong. Based on the selected curricula, the teachers developed program ideas, design concepts.

The problem dealt with selecting what we know from regular squares or the market, where we walk around the market, stopping by booths and adding the item we chose into our shopping baskets. In our metaphor the booths consists of places from which we can retrieve data, solve puzzles, etc. The market became our learning environment and was reproduced in a key screen, central screen. The key screen should be recognizable and show where you are and what the situation is just now. The key screen of the image is from a geography teaching in which we see the map of Europe and Scandinavia zoomed out and there could be added important information, such as: “Warning; Next day rain and stormy clouds in Eastern Norway.” The learner would normally have opportunities to perform different choices using buttons in the screen.

This was a rough description of the structure. To go deeper, we first worked with the idea and goal formulation. Who is the program planned for? What should the user obtain by using the program? Why use the computer as a medium? How should the program be used? At this stage, the preparation of a metaphor was important.

Metaphors /icons. Currently, we know that metaphors are widely used to quickly convey what a program does. In Windows 8, e.g., the screen is covered with metaphors representing different programs. It was emphasized that metaphors should quickly signal what a program could perform. It is far from easy to find good graphic metaphors that facilitate communication between a user and an application and motivate a user to work with the program. For some programs such as game programs, using cubes as metaphors can be a good choice. For archiving, we often find an archive cabinet used as a metaphor, etc. When designing metaphors, a starting point might be the virtual place where the user will be moved into; the role of the user; or the time: present, past, or future.

The design concept must clarify aspects such as the problem under study; the actions performed by the student, the program, and the teacher; the curriculum; and the program type.

The student may be able to move forward or back in time, e.g., through simulation of the future or the past in history programs depending on how the student believes he or she is placed in time or what role the student must play: a pilot, fisherman, actor, teacher, etc.

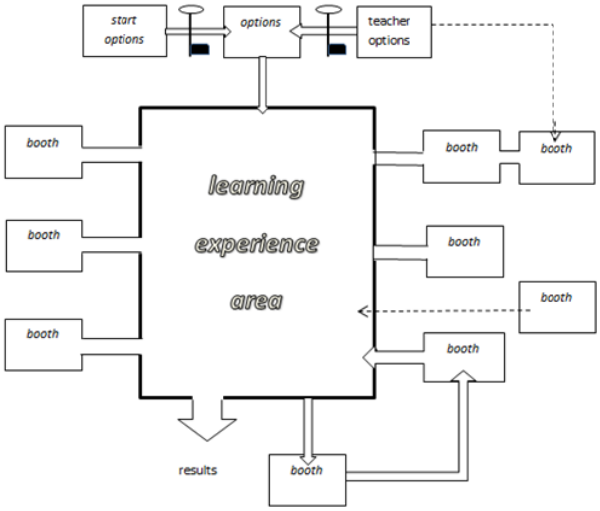

Fig. 3. The Market; “learning experience area”, booths and options

The Market Model facilitates the dynamics and interactivity wanted for well-functioning educational programs. As mentioned, in a market, one visits booths and

makes trades fairly freely. A sequential market in which a user always has to do the same trade route would make market shopping tedious. We cross back and forth from booth to booth with certain restrictions to meet our needs. A typical market has a structure as shown in figure 3.

The figure is a sketch of a generic market with booths. The arrow shows the direction in which one can move: in or out. Roads without arrows indicate both directions. The key shows conditions that cannot be changed during practice, for example, if the teacher sets certain parameters. Dotted lines indicate booths with closed entrances, which ensure that the student completes actions according to certain rules, e.g., that a student does not design a car with wheels on the roof or a person with eyes in the neck. Solid arrows indicate the ability to store and retrieve the design.

The key screen is developed from the concept of the market and becomes the part of the program to which the student will relate most. On the key screen, we find the educational issues built into the program by the designer. From the key screen the user can jump to other screens with explanatory details, but the key screen is always just a click away, which makes it easy to return to familiar terrain.

In the first Grimstad course, we facilitated the prototyping of designs under development. In practice, those who designed the programs could apply to be a part of a programming laboratory in which the programmers produced the prototypes and the designers were able to see their ideas on the screen. They could improve or reject the design and/or start over again. After a few years, we developed tools for prototyping that could be used directly by the designer. The prototyping tools gave designers a quick glance at how their ideas would work in practice on the computer. It was possible to go back and forth in the development of correction and creation until one was satisfied. We also developed several other tools for software development, which are described at the internet address:

{http://research.idi.ntnu.no/retrospect/}

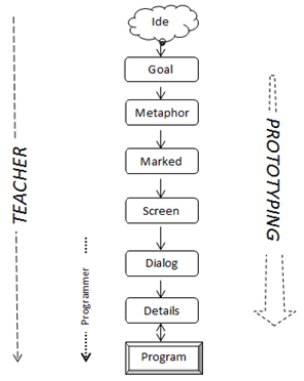

Fig 4. Steps in program design / development

Details. No special programming knowledge is needed to design metaphors, the market, the key screen or other screens and dialog boxes. However, the design process does reach a point in which the design must be programmed and implemented. Challenges in communication between the designer and programmer may arise, as designers generally cannot program but must communicate their intentions to the person who is an expert and is tasked with has to implement the program.

"*Once again, use your experience and knowledge as a teacher to make decisions – don´t let your assumptions about the nature of a computer program, or the advice of your programmer, influence your design. Compromise only when the limitation of the computer has been irrefutably demonstrated to you!*" [3] (Crossly & Green chap. 7, p 2).

One communication or specification method that is often used is a graphical representation of various states, called a state diagram. A state is a situation in which a program is waiting for user action, and it must therefore be made clear what must be performed prior to the transition state and what the subsequent state is. For example, what happens when you choose to place the cursor on an object and press the right mouse button, and when you release the button.

Another specification method is to use an action table. This is a tabulation of all possible opportunities for action that a user has, coupled with the impact each action will have on the program.

Agile Manifesto. The Grimstad method has similarities with what we now recognize as agile software development, or the Agile Manifesto [6]. The manifesto includes four core values of software development in addition to twelve principles that underline the manifesto.

Nordic cooperation. Nordic participants were invited to Grimstad courses in 1985. Only one non-Norwegian participant attended, coming from Denmark on his own initiative. In 1986, however, a good cooperation was established through the Nordic Council of Ministers. The course was set up in Denmark in 1987, Finland in 1988 and Iceland in 1989. We also gave courses in Estonia, with participants from the Baltics and Nordic countries.

The course in Denmark serves as an example of the typical participation, with the most members coming from the host country: 44 from Denmark, 17 from Finland, 3 from Iceland, 18 from Norway and 17 from Sweden. A total of 99 participants attended, 31 women and 68 men, from a pool of 180 who applied to join. In addition, 20 supervisors attended from the following countries: 6 from Denmark, 3 from Finland, 8 from Norway, and 3 from Sweden. Two programmers for prototyping came from Denmark, and two came from Norway.

In addition to the general design course that covered all subjects, courses were gradually developed related to specific disciplines. A special education course was

developed in Sweden in 1988 and in Denmark in 1989, and a social and industrial studies course was developed in Grimstad in 1990. [5]

Several advanced courses in design for the Nordic countries were arranged in Norway; 1987, 1988, 1989 and 1990.

There were a total of approximately 100 designs and programs developed under Nordic cooperation. Many projects were results of cooperation between participants from different Scandinavian countries.

2.3 Norwegian education programs

The Norwegian educational programs were distributed on topics as shown in figure 5 [7]. Types of program tools, simulations / games, information databases and illustration / instruction, are shown in figure 6 [7] .

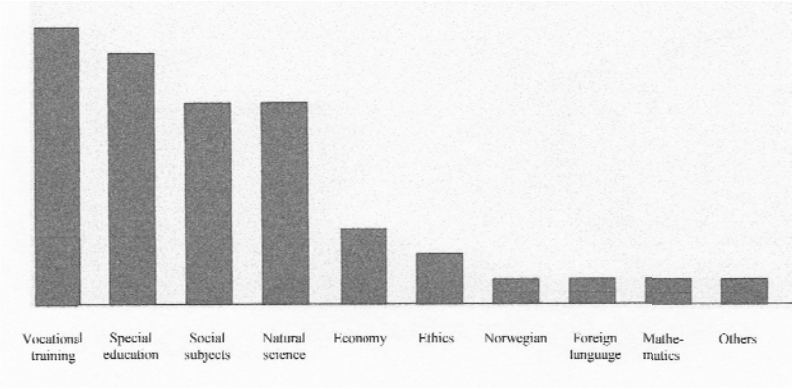

Fig. 5. Norwegian educational program divided into teaching areas

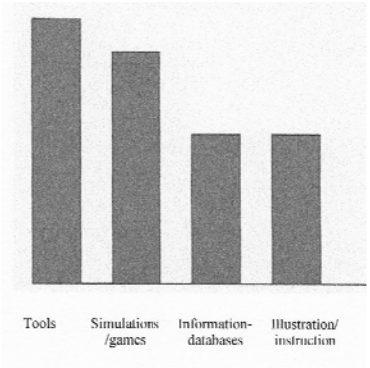

Fig. 6. Norwegian educational program by type of programs

2.4 Some selected Norwegian Educational Programs

AIDS – Simulation program providing information on how the AIDS epidemic may develop and information on how the virus has developed in Europe, including reasons for and patterns of the spread of the virus.

AKVAKULTUR (Aqua-culture) – Simulation program for the production of breeding fish in various coastal environments.

DATKUBEN (Datacube, "The Truck") - Help for children with learning difficulties and physical handicaps to gain some experiences that other children gain through play.

DYSLEKSI (Dyslexia) – Training in reading and writing for pupils with learning difficulties.

EKSPERTEN (The Expert) – Program for finding malfunctions in TV sets.

ESPEN I ASBJØRNSEN OG MOE – A fairy tale game based on some classical Norwegian fairy tales.

FULL FART MED NEWTON (Full speed ahead with Newton) – Experimental program for investigating Newton´s laws.

KOMPOSISJON/FARGE (Composition/Color) – A tool for balancing stripes and combinations of colors.

MØNSTER TIL KLÆR (Design of clothing) – A tool/program for clothing construction and design.

NAVIGARE – Simulation program for sea travel on boats – training on finding one´s way safely, depths, the rules of sailing, speed, time, distance, etc. It was developed commercially and sold worldwide.

SIM-SIM – A toolbox for making the development of dynamic simulation programs easier. The models described in Sim-Sim may be used as a part of applications written in other high-level languages, such as Pascal. It was developed commercially under a changed name and sold worldwide: http://www.powersim.com

SESAM – Statistics for analyzing demographic data.

VEVPLAN (Weaving design) – A tool program for designing, treating and analyzing woven fabric.

References

1. The Norwegian Action Program for the implementation of Computer technology in schools. A background paper for the OECD´s Hearing Committee. The Ministry of Church and Education. (1987)
2. The introduction of computers in schools: The Norwegian experience, Examiners´report, Centre for educational research and innovation (CERI) OCDE/OECD. (1987).
3. Crossley and Green: A Practical Guide For Teachers Designing Computer Lessonware. ISBN 0-9692382-0-7 (1985)
4. Ivar Minken & Børre Stenseth : Brukerorientert Program Design, User-oriented Program Design. The Ministry of Church and Education. (1992).
5. Nordiske ministerrådets Dataprogramgrupp, Verksamhet och resultat fram til 1991. Nordiska ministrådet. The Nordic Council of Ministers. Copenhagen (1992).
6. Fowler M. & Highsmith, J.: The Agile Manifesto. Software Development, 9(8). 28-35. (2001).
7. Status report, The Ministry of Church and Education[i]. (1986).
8. St. meld. 39 (1983-84) DATATEKNOLOGI I SKOLEN. White Paper no. 39. The Ministry of Church and Education. (1984)

Additional literature

9. Kolås L.: Implementation of pedagogical principles into the software design process of e-learning applications. Doctoral theses at NTNU. ISBN 978-82-471-2269-3 (2010)
10. Strand K. A. Concurrent Design Aproach to the Design of Customized Corporate E-Learning Doctoral theses at NTNU. ISBN 978-82-471-3692-8 (2012)

[i] Throughout the period, the Ministry has changed its name several times. It started with the Ministry of Church and Education (KUD) - an old name back to the first Norwegian government, 1814. Then the Ministry of Church-, Educations- and Research (KUF), today is the name is the Ministry of Education and Research (KD).