The role of phenomenographic research in the design of instructional computer applications for number concepts.

Eva Ekeblad
Berner Lindström

Paper to be presented at the 6th European Conference on Learning and Instruction, University of Nijmegen, Department of Educational Sciences, August 26–31, 1995.

Kollegiet för
Inlärning, Kognition och Informationsteknologi,
Institutionen för pedagogik
Göteborgs universitet

The role of phenomenographic research in the design of instructional computer applications for number concepts.

Eva Ekeblad
Berner Lindström
Göteborg University
Department of Education and Educational Research

ABSTRACT

Four principles of design and one overarching subject matter related principle were the guidelines for a project aimed at developing instructional computer applications based on phenomenographic research in the area of young children's basic number concepts.

The overarching domain specific principle was to *promote structuring conceptions of numbers over counting conceptions*. The four design principles were that the computer would be used to present *crucial problems*, to introduce *variation* in some important dimension, to present analogies serving as *bridges* between the familiar and the unknown, and to *confront the children with the consequences of their conceptions*, in situations where a less functional conception would be revealed by the computer as yielding an incongruous representation of the world.

A number of educational games were developed in an iterative process of moving back and forth between software design and research aimed at exploring children's conceptions of numbers in connection with use of the developed software. This paper presents a reflective analysis of the design, the history of development and the documented use in classroom based research of some of our games, focusing on how principles turn out when applied in practice.

INTRODUCTION: THE APPLICATION OF PHENOMENOGRAPHY

A thorough understanding of the potential variation in the learners' understanding of a certain topic of instruction is invaluable in the design of interactive teaching/ /learning media. This means that for purposes of instructional design there is a demand for qualitative research on student learning. This symposium is devoted to a gathering and mobilisation of the experiences of researchers and instructional developers using a phenomenographic methodology for the "student learning" component of their diverse undertakings. Our specific contribution in this paper is a reflective (and retrospective) analysis of how a phenomenographic description of children's ways of understanding numbers and simple arithmetic were translated with the guidance of four pedagogic design principles, compatible with the general methodology of phenomenography, into a set of educational computer games aimed at promoting a s t r u c t u r i n g conception of numbers over a c o u n t i n g conception — an example of instructional design based on ideas of what the learners need to learn (Laurillard, 1993).

There are a number of questions: What kind of contribution can we see from a phenomenographic mapping, or outcome space, towards a specification of what students are to do with the computer? What is the relation between the phenomenographic description and the pedagogic principles? How are principles put into practice, i.e. how does the computer medium constrain their implementation and how does the actually produced software in its turn constrain the teaching/ /learning process? In the project we are referring to there was — as a third kind of principle, guiding the design process itself — an iterative process of moving back and forth between software design and research aimed at exploring children's conceptions of numbers in connection with use of the developed software. Thus we have had the opportunity to make a reflective analysis of the design, the history of development and the documented use in classroom based research of some of our games, focusing on how phenomenographic principles turn out when applied in practice.

PHENOMENOGRAPHIC RESEARCH IN A COMPUTER ENVIRONMENT

The IDM-project (Interactive Didactic Milieus)[1], was launched in 1986 and concluded in 1992. This phenomenographic foray into *using the computer to promote conceptual change* had two different branches, one of them devoted to exploring how people learn to understand the computer, and computer focused activities like programming (e.g. Booth, 1992), and the other one devoted to exploring how the computer could be used to promote understanding of other subject matter areas. In the present paper we will treat the second branch of the project: the one aiming at the computer as a mediator for understandings of some other (not computer related) subject matter area. Specifically, results from ongoing phenomenographic research (published in Neuman, 1987) in the subject matter area of basic number concepts (defined as the set of part-part-whole relations of numbers 1 to 10) were to be applied to the design of an *interactive didactic environment* — i.e. some kind of educational computer software (Marton, Gustafsson, & Lindström, 1986; Lindström, Ekeblad, & Neuman, 1987).

Following suggestions by Marton (1984) the general aim of this branch of the project was to find technical solutions to the problem of getting students to embrace some central ideas embodied in the culturally authorised system of knowledge. To the extent that this kind of ideas are taken for granted by qualified participants, they may not be easily available to the novice.

> The use of special research-based techniques in educational contexts has long been associated with an instructional process aiming at clearly defined *behavioral* objectives, but there are, of course, no reasons for claiming that there could never be any technical solutions to the problem of changing the student's ideas about some topic. (Marton, 1984, p. 66)

Conceptions of number and simple arithmetic computation of the type facing children in their first year of schooling were chosen as a case of particular interest. The phenomenography of Neuman (1987, 1990) on the origin of arithmetic skills provided a description of children's developing conceptions of numbers, and their confrontation with the cultural knowledge system. Her model included the developmentally fruitful areas of contact between initial and mature conceptions, as well as the potential areas of conflict with the conceptions taken for granted by teachers and the culture at large.

The general idea of the relation between phenomenographic research and its application in an educational technology was to first extract, through a phenomenographic investigation, the conceptions taken for granted in our cultural system and then to re-embed these conceptions in a research-based type of educational computer software (Marton, 1984). In the present case the extraction of educationally crucial conceptual variation was provided by Neuman's research. The re-embedding in a computer technology as undertaken in the IDM-project, on the other hand, would presumably be a matter quite different from the conceptual re-embedding performed by Neuman (1987) in her teaching experiment, due to differences in situations and media — where Neuman was continuously developing her teaching method on the basis of day-to-day work in the classroom, using media open to adaptations made on short notice, the development of research-based computer software narrows the scope for improvisation as so much of the method is already embedded in the machinery.

---

1 Funded by The bank of Sweden Tercentenary Foundation.

In terms of the conceptual variation Neuman made an overarching distinction between on the one hand fruitful, s t r u c t u r i n g and on the other hand potentially infertile, c o u n t i n g conceptions of numbers (Marton & Neuman, 1990). This distinction was the summary of a quite complex system of qualitatively different pre-numerical and numerical conceptions, based on a phenomenographic interview study and an extensive teaching experiment. The success of Neuman's teaching experiment provided evidence that the structured (part-whole) character of the culturally authorised system of numbers is educationally more important than the properties of countability by stepwise procedures. It is also more vulnerable, as it is less visible from the novice child's perspective. The prominence of counting as a feature of school arithmetic may easily come to obscure the part-whole character of the number system — i.e. if the cardinal, holistic aspect is taken for granted in teaching. The main task for the instructional design application was thus to enhance the structural aspect of small numbers (the 1 to 10 range).

The original plans for the development of interactive didactic environments, i.e. the computer implementation of Neuman's research results, included further refinement of the model of children's conceptions of number, as there was to be a development of related instruments for inital diagnosis as well as for a running diagnosis in the course of a child's interaction with the computer system (Marton et al., 1986). It was emphasised in the plans that the development of an educational technology like this, a research-based didactically interactive computer environment, would be a forceful contribution to the communicability of research results:

> Prominent teachers and/or researchers may reach amazing results, but when others try to make use of their contributions the application will often become something else than what is intended and the results do not appear. In this special respect our planned work means that some of the ideas that have been foundational for Dagmar Neuman's work will be embodied in a technology. This means that the same educational arrangement can be made available to as many students that one would wish. (Marton et al., 1986, p. 63, our translation)[2]

The educational computer software to be produced should use the unique computational and graphic capacities of the computer to avoid excessive use of number symbols in the teaching/ /learning activity centering on number concepts. The aim was to use graphics and synthesized speech on the interactive part of the computer and some kind of pointing tool on the part of the children (Marton et al., 1986). The general idea was, in accordance with Marton (1984), to invent computer based techniques for facilitating the desirable conceptualisations of numbers.

The most important target group was considered to be weak students with actual or potential mathematics difficulties, and so the aim for instructional software design was to produce as transparent and easily understood scenarios and interfaces as possible. Ideally, the computer would be a resource not only for *learning*, i.e. for coming to know numbers as they ought to be known, but also for *unlearning* habits of thinking obstructing fruitful ways of knowing numbers (Neuman, 1990).

---

2 Framstående lärare och/eller forskare kan nå häpnadsväckande resultat, men när andra försöker använda sig av deras insatser blir tillämpningen ofta något annat än den avsedda och resultaten uteblir. I detta speciella avseende innebär vårt planerade arbete att några av de idéer som har legat till grund för Dagmar Neumans arbete gestaltas i en teknologi. Detta betyder att samma pedagogiska arrangemang kan komma så många elever till del som man önskar. (Marton et al., 1986, p. 63)

THE DOMAIN SPECIFIC PRINCIPLE: WHAT CHILDREN NEED TO LEARN

Applying phenomenographich research to instructional software design first of all means using results from this type of research, which describe the variation in learner's potential ways of understanding and approaching the educationally central phenomenon in case, for the establishing some domain-specific principle of what is the educationally desirable way of approaching this specific phenomenon, and also for establishing what this means in terms of the less desirable ways of construing the same phenomenon.

At the time when the project plan was written the crucial aspects in the area of conceptions of small numbers and simple arithmetic were considered to be: first the establishment of units, second the insight in the difference in meaning between ordering numbers and summarising numbers, and third the ability to grasp the basic additive/ /subtractive structure of numbers. Although the theme of s t r u c t u r i n g versus c o u n t i n g is still implicit, these formulations clearly have a bearing on the distinction that was somewhat later to be more strongly articulated.

Neuman's complex phenomenographic system of categories (1987) contains the crucial distinction between fruitful s t r u c t u r i n g conceptions of numbers and potentially infertile c o u n t i n g conceptions. This great divide in Neuman's system makes good sense in terms of what kind of ideas of numbers that we want children to harbour. However it does not say much about how children come to take one developmental path or the other, or about how teaching can promote s t r u c t u r i n g conceptions of numbers over c o u n t i n g conceptions. It is in the finer structure of the category system that Neuman traces developmental change. In short, she argues that it is through f i n g e r n u m b e r s that children spontaneously come to a s t r u c t u r i n g conception of numbers, and that these also provide an excellent tool for teaching (Neuman 1987, 1990). The f i n g e r n u m b e r s provide a visual as well as tactile anchoring for whole-to-parts structuring of numbers. From this foundation more and more abstract experiences of the numbers in the basic 1 to 10 range can be undergone, until there is an understanding of abstract number relations (Neuman, 1990). Together with the considerations concerning the target group of weak students the principle of *promoting structuring over counting* means that activities and forms of numerical problem presentation enhancing visual structure while restraining unit by unit counting should be used, and additionally that number representation by arabic numerals is a dispreferred form.

THE PEDAGOGIC PRINCIPLES: HOW TO INDUCE LEARNING

For the purposes of instructional design, however, we need more than a description of the educationally desirable way of seeing the phenomenon in question (and how this way relates to other possible ways of seeing the matter). It is not enough just to embed the description in the computer, it must also be available for interaction in a teaching/ /learning activity, mobilising the concepts in question. To make the computer, or any educational technology, into a truly interactive didactic environment some pedagogic principles have to be outlined and applied. Needless to say, these principles should have some kind of agreement with the general research methodology, as their function is to provide the guidelines for the re-application of educational research to the practice of education, and it would be strange indeed to use radically different standards in different phases of the enterprise.

The domain specific goal was set in terms of children's conceptions of numbers, with a foundation in research based knowledge of the relations between the possible variation and the desirable outcomes among these conceptions. In order to apply this knowledge in an interactive educational technology there were in the original project plan a number of guidelines for how children were to be engaged through the computer in a conceptualising activity with the structure of numbers. Four general pedagogic principles for an interactive system were expressed:

1) The computer would be used to present *crucial problems*. Problems presented should be intelligible, but yet challenging enough to bring conceptual discrepancies to the student's awareness.
2) The computer would be used to present *variation* in some important problem dimension. The variation would point to relevant relationships in some aspect of importance, as well as serve to maintain focused attention to the activity.
3) The computer would be used to present analogies serving as *bridges* between the familiar and the unknown, between initial conceptions and mature ones.
4) The computer would be used to *confront the students with the consequences of their conceptions*, in situations where a less functional conception would be revealed by the computer as yielding an incongruous representation of the world. (Marton et al., 1986, p. 12)

Taken together, these principles meant that the computer would not just present the material to be worked with. It would also evaluate or diagnose the actions of the student and provide feedback on them, preferably a type of feedback that made clear the nature of the mismatch between the student's grounds for making a certain choice and the the domain as scientifically conceptualised. The prime example of this kind of feedback was given by the Newtonian microworlds used by Barbara White (1984) to confront physics students with the consequences of their everyday understandings of forces applied to an object in Newtonian space: before you realise that in a world without friction or gravitation a force, once applied to the spaceship you are controlling, will have permanently changed the velocity of the ship you will not be able to steer it without crashing.

In these four principles for the construction of an interactive didactic environment there is a strong resemblance to the recommendations for a teaching strategy in accordance with the phenomenographic focus on students' understandings given by Marton and Ramsden (1988):

1) Present the learner with new ways of seeing.
2) Focus on a few critical issues and show how they relate.
3) Integrate substantive and syntactic principles.
4) Make the learners' conceptions explicit to them.
5) Highlight the inconsistencies within and the consequences of learner's conceptions.
6) Create situations where learners centre attention on relevant aspects.
7) Test understanding of phenomena; use the results for diagnostic assessment and curriculum design.
8) Use reflective teaching strategies.

As pointed out by Laurillard (1993) recommendations 1, 2, 3, and 6, which basically are the ones corresponding to the first three IDM principles, focus more on how to *present* the content to students than on how to *interact* with them. This characteristic presumably indicates the suitability of embedding these principles in some kind of educational medium, whereas the more interactive recommendations (4, 5, 7 and 8) may well be taken as the proper responsibility of the teacher — or, at least before the era of teaching machines they would have been. Comparing the Marton & Ramsden recommendations to the IDM principles, it seems that it is only recommendation no 8 (mentioning reflective teaching strategies) that according to the IDM plans would still be the responsibility of a teacher. In order to confront the students with the consequences of their conceptions, these necessarily have to be in some sense made explicit, so principle 4 would correspond to recommendations 4 and 5, whereas recommendation 7 reads as an outline of the whole branch of the IDM project: the intention was to use Neuman's results as the basis for the instructional design of computer software which was also to include diagnostic facilities. Thus the original application in its vision of how Neuman's foundational ideas would be made generally available by being embodied in a technology does not seem to leave much scope for the teacher.

However, there was also at an early stage in the project an interest in viewing educational software as a *tool* for the teacher as well as for the learner. There were also ideas, inspired by Papert's creed that the child should be programming the computer rather than the other way around (Papert, 1980). The gist of these ideas was that an interactive learning environment dealing with numbers in the range 1 to 10 would be in some way a computerbased multipurpose tool for number related activities: some kind of "number lab," possibly analogous with the Newtonian microworlds described by White — a microworld of numbers where the child could intervene into the course of events and then be confronted with the effects of the intervention.

THE TRANSLATION OF PRINCIPLES INTO PRACTICE

The task for instructional design according to the four given principles applied in the specific area of conceptions of the set of part-part whole relations of numbers 1 to 10 would then be:

1) To use the computer for presenting challenging numerical problems in the given number range, and to present them in a way that promoted a perception of number s t r u c t u r e at the same time as suppressing opportunities for c o u n t i n g — presenting the children with new ways of seeing numbers.
2) To use the computer for presenting variation in important problem dimensions. This would mean on one hand that it was desirable to present a variation in problem types, (while making it at all times clear what was the actual problem at hand). On the other hand it would mean providing a variation in the numerical dimension within a problem type, and possibly in the visual structure of the problem presentation.
3) To use the computer to present links or bridges between different ways of understanding numbers, or numbers represented in different forms.
4) To use the computer to confront the students with the consequences of their ways of solving the numerical problems, preferably in terms of immediately perceivable effects in the problem domain, and preferably dependent on whether they approached the numbers by c o u n t i n g or by s t r u c t u r i n g.

The unique contribution of the computer to the first three, presentational, duties would be its capacity to provide animated graphics, and to produce a variation that was systematically randomised within given limits. However, the main contribution of the computer was indicated by the fourth principle: its capacity to provide directed feedback on the students' activity. In the first year of the project great efforts were devoted to the translation of Neuman's diagnostic insights into computerised diagnostics connected to a couple of problem environments of the type that were beginning to be developed.

In the course of the project an assortment of forms of number representation were produced. However, visually well-structured patterns like the dot patterns on dice or dominoes came at an early stage to be predominantly used for problem presentation, the prototypical problem being the numerical assessment of structured arrangements of objects (number monograms). In this way the learners would be enabled to *see* numerical s t r u c t u r e without c o u n t i n g. The patterns used were as a rule constituted by sets of pictorial objects. To further promote reliance upon pattern structure, and to discourage counting, short exposure times and/ /or limited response times were typically used. There would be time to *see* but not enough time to *do* anything that (like counting) takes time. In this way insight in the whole-to-parts structure of numbers would be made compelling.

To a limited extent a second lasting form of number (or magnitude) representation was employed in problem presentation: some form of measured or unitised distances. These were mainly used for “missing addend” problems, which played a small but important part in the system. The unique nature of missing addend problems as a touchstone for the understanding of numerical part-part-whole relations had been heavily stressed by Neuman (1987). These problems often involving desired future states seem, however, to be best presented in verbal form, as they often turned out to be hard to make sense of for the children (Ekeblad & Olstorpe, 1990; Ekeblad, 1992).

For the answering phase of the interaction between the child and the computer an assortment of number representations were explored. Numerical answers were most often to be given by the child through the choice of one out of a set of symbols on the screen. As has been indicated, arabic numerals were dispreferred, as they bear no pictorial/ /indexical resemblance to the numbers symbolised by them. They were, however, never totally avoided, although Roman numerals, dice patterns, and “finger numerals” were very often used in their place. Apart from input through screen symbols, there were from an early stage ideas about using a keyboard specially adapted for inputting a finger number, constituted by simultaneous keypresses. Due to mainly technical reasons the realisation of this paradigm was postponed for several years.

In the first set of games put on informal trial in a small number of classrooms in the school year -87 / -88 there were themes of ordering numbers versus summarising numbers, as well as themes of structured number patterns and one game aiming to present a problem analogue of a “missing addend” problem (Lindström et al., 1987). They all used simple versions of animated graphics to present their problem. Some of the structured number patterns, for example, were flashed on the screen for a brief moment, whereas in other games the scenario involved the child in interaction while objects were moving across the screen. Whereas visual structure was well provided for in the games using the mouse for input the tactile component considered as a vital aspect of f i n g e r n u m b e r s was not very well supported by the computer medium (Neuman, 1990). In order to amend this there were ideas for some kind of input device demanding the simultaneous placement of the right number of fingers on the keys. However, for technical reasons the implementation was postponed for several years.

As mentioned, there were ideas about creating some kind of “number lab” sustaining varied forms of intercation with a microworld of numbers. However, for reasons to be investigated below, most of the software produced within the project very soon adopted a rather simple and interactionally restricted game form. A problem or question was presented, and the student was provided with some means of answering the question. The computer evaluated the answer and provided some kind of feedback on this, then went on to present another problem of the same general kind, entirely in accordance with the pedagogic principles of the project: The first IDM principle, the one proposing a teaching strategy of crucial problems, proposed that in the interaction the computer should be the party responsible for presenting problems. The second principle, the one concerning variation in some important problem dimension, implied that it was a good idea to present several related but different problems in succession. The third principle, the one concerning bridging analogies between different ways of understanding numbers, could be taken as pointing to the link constituted by the necessity to present problems in one representational form while reserving another type of representation of the number(s) involved for the solution or answer. The fourth principle, the one mentioning that children should be confronted with the consequences of their conceptions, suggested that feedback should be provided by the computer in close connection with the learner's actions. That the computer feedback in practice never got much more advanced than simply indicating wheter the answer was right or wrong had to do with the difficulties of reliably diagnosing or interpreting the meaning of a single answer that was "off" in some way — or, for that matter, of what kind of understanding was expressed in a single correct answer.

PRACTICALITIES MATTER: WHY THINGS DID NOT GET DONE

The four main pedagogic principles of the project taken together with the subject matter principle of promoting s t r u c t u r i n g over c o u n t i n g were thus effective in guiding the software design. It is, however, also worth noting that practical matters exert a powerful influence indeed on the choices made in the course of research-based development of educational software.

At an early stage of the project the decision was made to conduct all software development in a Macintosh environment, as this seemed to fulfil the requirements for high-quality graphics and interaction facilities stated in the project plans. In 1986 this meant the original Macintosh with its small but high-quality black-and-white screen, and with keyboard and mouse (to point at and click on on-screen configurations) as the primary input devices. Once a basic choice like this is carried through, it sets some limits to the practical implementation of ideas in an educational technology.

Programming the Macintosh in these pre-Windows days was considered a real challenge as compared to programming in the PC environment. With the programming competence available to the IDM project this was painfully evident. A version of C was used for programming, but most every little thing to be done posed next to insurmountable problems. For example, graphical objects moving across the screen left traces, or erased objects that they passed. This problem was not really solved, but rather "worked around" by avoiding the use of overlapping graphics, which limited the use of animated graphics considerably, and made manipulable graphics more or less unthinkable. Also, for reasons of speed as well as with consideration to the small size of the screen, most pictorial objects (except for background rectangles and the like) were in the Macintosh "icon" format, i.e. 32*32 pixels in size. Even so, screens had a tendency to get crowded with information. Using alphanumeric input from the keyboard was, in our case, not a preferred mode of interaction, which was fortunate, as the use of keybord input would have demanded considerably more cumbersome programming than the mousedriven pointing and clicking which was actually used.

The idea of having the computer talk to the children, using synthetic or canned speech, never got worked into the developing software. The synthetic alphabet available for the Mac at the time the project was started had a heavy American accent, making Swedish number words more or less unintelligible. As for using recorded speech this was left aside first due to the lack of proper digitizing equipment and later due to the memory demands. Actually, in the early years any addition of illustrative sounds to some game had the tendency to crash the system.

In the first years of the project a great variety of games and other problem tasks were sketched (on paper and in the computer medium) and partially developed. For this there was a rich source of ideas available in the material from Neuman's teaching experiment (1987).

However, comparatively few of the possibilities explored came through the whole process as worthy of being brought into the classroom — game ideas were stymied by some technical problem, they did not present enough of a problem to be interesting, some aspect of them was found to conflict with some other pedagogic committment, or they simply did not seem likely to promote the conceptual change we were aiming for.

The m e a s u r e m e n t ideas intended to promote an insight of the need to *establish units* were dropped at some time during the first years of the project. This was partly due to the programming difficulties with handling manipulable graphics. Partly the ideas were abandoned as the favoured form of interaction between child and computer stabilised into the mentioned game form. So the exploratory use of manipulable “numerical objects” fell victim to a mixture of technical difficulties and adherence to the principles of presenting the child with crucial problems having diagnosable outcomes. On the other hand, any initial intentions to provide a running diagnosis of the structure of a child’s way of conceptualising a specific problem through computer analysis of her way of handling it fell victim to the narrow channels of interaction and input provided for in the game form, where, basically, the child’s contribution was an answer to a question, which could be evaluated as right or wrong, but for most of the time yielded very little additional information about what kind of reasoning had governed the answer.

The efforts to produce computerised diagnostics also foundered on the lack of any specific or explicit principles for how Neuman’s experience of interaction with children and her insightful interpretations of their understandings were to be transformed into computer heuristics for the sequencing of responses to various student inputs. Notably there was never any principle for drawing a limit, setting a maximal depth to the computer search — the problem was not even formulated in these terms, except by a frustrated programmer at the time when the diagnosis problem was reaching an impasse.

In 1988 there was a switch from a C-environment to HyperCard (and then, later to SuperCard), at least for sketching and development purposes. Whereas this facilitated the creation of variations on a theme for some of our problem types, and made the creation and management of sound and of game logs much easier, it also disrupted the use of objects moving on the screen in the games: interaction with the computer was more or less totally blocked while things were moving — with the available processing capacity it slowed things down far beyond the tolerable. As a result the collection of games was even more than before centered around a few problem types. Basically these were variations on numerical appreciation of structured patterns, generally shown for a limited time.

On the other hand the idea of using some input device capable of registering the simultaneous placement of several fingers were never abandoned. There was some experimentation with a MIDI-interface and the keyboard from a small synthesizer. However, the children were often unable to span the larger numbers with their hands. In 1991 a specially adapted keyboard appeared in the market, and a sample was ordered, arriving in the second half of 1992. A game where structured number patterns had to be coordinated with the placement of a matching number of fingers was developed in the following year (i.e. as one of the prolongations of the project).

RESEARCH — DESIGN — TEACHING

Reports on the project have consistently treated the computer as a tool for teaching as well as a tool for learning. It has been repeatedly emphasised that the computer cannot replace the teacher (Lindström & Ekeblad, 1989; Ekeblad & Olstorpe, 1990; Neuman, 1990). Neuman, in spite of her pessimism concerning the use of the computer to replace the child’s own fingers, acknowledges it as potentially useful in special cases. Under guidance from a teacher, the computer may serve as a motivating factor for somewhat older children with maths difficulties to start unlearning counting behaviors and constructing finger numbers.

As the IDM principles are better viewed as general pedagogic principles for the promotion of conceptual change than as explicitly aimed at the design of an educational technology, there will seem to be a distribution of responsibility for adherence to the principles between the stages of design and classroom use. The reflective analysis in the previous sections shows that it was at least attempted to satisfy all of the four pedagogic principles at the stage of software design. This turned the design towards the production of conceptually transparent game scenarios, intended to be more or less self-explanatory. The intention of promoting structuring over counting directed the production towards game scenarios more or less monopolising the attention of the student — by demanding a concentrated gazing at the screen — to the exclusion (or at least minimisation) of the interaction between human participants. On the other hand, as noted by Neuman (1990), the computer (at least when the mouse is used for input) cannot provide the kind of guidance towards the use of finger numbers that a teacher may. The introduction of the “finger-keyboard,” however, allows the computer to take over some of this responsibility as well.

The practical demands of the construction of educational software require the pre-packaging of a great number of decisions into the software before it can get into a running condition and be used. This is, of course, the reason why there is a need for a research foundation to the instructional design of interactive media. Due to the amount of decisions already made, and embodied in a piece of educational software, it seems reasonable to conclude that its design serves as a powerful limiting condition for what is subsequently taught and learned in connection with it, and also for how teaching and learning proceed. For this reason an iterative process of moving back and forth between software design and research aimed at exploring children's conceptions of numbers in connection with use of the developed software was practised within the IDM-project. Throughout the duration of the project the IDM-software was tried out in different ways: in smallscale, informally reported pilot studies, and by being used by teachers involved in school computer projects using the Macintosh. Also, educational software development was interleaved with studies of children using the games in situations more or less similar to a clinical teaching situation. Among these the ones concerning the use of the “finger number keyboard” for inputting assessments of structured numbers have not yet been reported, as the game did not exist in a running condition until in 1993. A study of pre-schoolers using this game was carried out in 1994, but has so far only been submitted to a preliminary analysis. An extensive study where the children of three first-grade classes were followed with reference to their ways of experiencing numbers through their first year in school, using among others three games with their roots in the IDM-project is the basis of an upcoming dissertation (Ekeblad, in preparation).

One of the early reported studies was conducted in a preschool setting, (Ekeblad, 1989), and involved 17 children (age 6) playing first one session of a game with structured number patterns, and then returning (14 of them) to play a session with a second game. The report records the conviction that used with some consideration as to what may be incidental and what seems to be more consistent in a child’s way of relating to numbers the IDM software may contribute to a diagnosis, made by the teacher. Using the number monogram games diagnostically it is possible to draw a line between children who are aware of the possibility to approach numbers by exploiting their structure (even if they *do* count in certain situations), and children who do not see any other possibility than to keep counting at any cost. Contrary to the project intentions, the computer did not automatically exclude counting. As noted by

Neuman (1990): if the game problems were to be solved by the use of finger numbers tutorial guidance from a teacher was needed.

The suggested transfer of diagnostic responsibility from the software to the teacher (Ekeblad, 1989) is a confirmation in print of the fact that most of the project ideas about creating some form of intelligent tutoring system had been abandoned. It may also be seen as an indication of the shift in emphasis from the creation of a simulacrum of the skills of a “prominent teacher” hinted at in the quote from the original application, to acknowledging the role of the “ordinary” teacher in the teaching/ /learning interaction. As for the games as promotors of conceptual change the report simply concludes that a single session with each game hardly provides enough material for the systematic observation of conceptual change. In the initial session with a game the child has primarily to begin making sense of its appearance as a whole, and there is ample opportunity for idiosyncratic interpretations of the various features of a scenario. These are expressed in ways that it would be very hard for the computer to catch, so there is a demand for a human teacher or tutor, serving as interpreter between the child and the computer in an introduction phase (Ekeblad, 1992).

The second study (reported in Ekeblad & Olstorpe, 1990) involved a wider range of games using structured number patterns but also incorporating numerical operations, and was conducted in a setting of lower primary remedial teaching. Seven children (ages 7-9) of varying ability participated in a more or less extended sequence of clinical teaching sessions centered around our games. Here it was possible to see in connection with the games employing number monograms, how these could indeed promote the establishment or consolidation of an understanding of numbers as a phenomenon affording structure as well as permitting counting procedures. Some of the participating seven children already had this understanding well established, others were able in working with the games to grasp more firmly what they had only glimpsed. One girl made a transition from perceiving anything numerical to do with objects in the world as a counting activity, and anything to do with computation as some kind of task with the purpose: “fill the pages neatly,” into an understanding that there is a meaning in numbers to which part-whole structure provides a key. The concluding remarks of this report point once more to the nature of the number games as a tool for teaching and learning taking place in the context of an interpersonal interaction between teacher and learner.

In these two studies we had been able to establish that games presenting number monograms for rapid assessment have a certain potential for promoting conceptual development; the representations as well as the scenarios are fairly easy to understand in their numerical sense. Concerning the part of the study involving games with computational themes, we found, however, a number of problems with the representations and scenarios required to pose intelligible and unambiguous questions about the relations between precisely stated parts of a precisely stated whole (Ekeblad & Olstorpe, 1990, Ekeblad, 1992). Following the prescription of avoiding the use of arabic numerals (and even more so the use of written problem presentations) and preferring graphical presentation of collections of discrete objects made it difficult to pose problems without giving at the same time the answer, or without encouraging the children to count objects on the screen. Especially the representations used for “missing addend” problems were often misunderstood by precisely those children who showed the least developed ways of understanding numbers.

DISCUSSION: HOW PRINCIPLES TURNED OUT

We have looked in this paper first at how the IDM design principles, as applied to the phenomenography of Neuman, were mutually constraining. The four pedagogic principles taken together seems to be what made the question-answer-feedback game form the natural choice satisfying them all. As it was specified that the computer should be used to present crucial problems, sketches for some kind of more tool like software, a number lab of some kind, easily were seen as too pointless, presenting too little of a challenge to the learner. As it was specified that the computer should make clear the consequences of less adapted conceptions of numbers, sketches were seen as incomplete as long as they did not contain facilities for evaluative feedback by the computer, but were only problem generators providing material for discussion between learners, or between teacher and children.

We have also observed how the practically available computer technology further constrained implementation. Technical constraints and obstacles facing the project also worked against the more tool-like or microworld-like applications. It turned out to be practically feasible to produce games in a delimited, interactionally narrow form, whereas the production of interactive environments with a greater scope of choice of activities for the teacher and the learners placed technical demands beyond our resources. Of course, a tool program leaving problem construction and evaluation to the users, might also have provided much less guarantee that project principles were adhered to in the classroom use of the software.

We have also seen something of how the actually produced software in its turn constrains the teaching/ /learning process — when there is little for the learner to do but to watch and to see properly, whatever interaction goes on is basically a business between the computer and the child watching the screen. It is evident that the dominant interaction sequence of the IDM games (question/ /answer/ /feedback) is a variant of the persistent “initiation/ /reply/ /evaluation” pattern repeatedly described in educational sociolinguistics as a central feature of classroom interaction (Sinclair & Coulthard, 1975; Lundgren, 1977; Stubbs, 1986). Lundgren (1977) labels the “recitation” pattern an artificial model of language, that appears in few other social situations than school. However, Heath (1982) reports on research that shows the cultural prevalence of this pattern in the interaction of parents with their young children not just in US “mainstream communities” but in a much wider range of literacy oriented communities, rooted in European traditions. Children in these communities are socialised, from the age when they learn to speak, into a conception of knowledge where the proper use of language is the recounting of factual events in a straightforward way, labelling objects and their properties correctly. Heath recounts how for the youngest children, picture-book reading activities together with an adult have precisely the “recitation” structure: The adult (who naturally knows the correct answer) points at an item, and asks the child what it is. The child provides the verbal label — or some vocalisation or even nonverbal sign of attention to the item, according to her mastery of language. The mother provides verbal feedback. This can be praise or an expansion of the child’s utterance. For the least competent the adult feedback may simply consist in giving the label, once the child has showed that she is attending.

So, in the pattern of interaction inscribed in the IDM-games the computer merely takes over the position of the teacher (or parent) in a pattern that is presumably well established in Western literate culture. The powerful regulation of the interaction provided by the prevailing game form basically makes the teacher a servant of the game, someone to function as its interpreter in the introduction phase, and as a supervisor of its proper use in later phases. This may seem to be in contradiction with the aims of the project, notably with the emphatically stated goal to create a tool for teaching, as well as a tool for learning. Indeed, there seems to have been an unacknowledged tension between ambitions to use the computer for producing insights by having it interact with learners in accordance with the methods of “prominent teachers” and ambitions to use the computer as a tool for teaching/ /learning by having it serve as material for collaborative teaching/ /learning activities. Both ambitions were in agreement with the pedagogic principles of the project, and it was our intention to produce software satisfying both requirements. However, in the day to day labours with practical and technical demands of design, aspects satisfying the first ambition were readily incorporated in the computer software, whereas the second ambition was left as a responsibility of the users, and also somewhat counteracted by the prevailing form of interaction inscribed in the game scenarios.

REFERENCES

Booth, S. (1992). *Learning to program. A phenomenographic perspective*. Göteborg: Acta Universitatis Gothoburgensis.

Ekeblad, E., & Olstorpe, K. (1990). *Kvadrater och Staplar: Datorn i specialundervisningen. [Squares and Stacks: The computer in special education.]* (Publikationer från institutionen för pedagogik No. 1990:26). Department of Education and Educational Research, University of Göteborg.

Ekeblad, E. (1992). Arithmetic without reading: Children interpreting representations of part-whole relations. *Computers in Human Behavior, 9*, 213-225.

Heath, S. B. (1982). What no bedtime story means: narrative skills at home and school. *Language in Society*. 11: 49-79.

Lindström, B., & Ekeblad, E. (1989). *The computer as a tool for developing basic arithmetic skills.* (Publikationer från institutionen för pedagogik No. 1989:14). Department of Education and Educational Research, University of Göteborg.

Lindström, B., Ekeblad, E., & Neuman, D. (1987). Using the computer to promote conceptual change: The case of elementary arithmetic. *Education and Computing, 3*(3-4), 223-230.

Lundgren, U. P. (1977). *Model analysis of pedagogical processes.* Stockholm, Stockholm Institute of Education.

Marton, F., & Neuman D. (1990). The perceptibility of numbers and the origin of arithmetic skills. *Report 1990:05,* Department of Education and Educational Research, University of Göteborg.

Marton, F. (1984) Towards a psychology beyond the individual. In K. M. J. Lagerspetz and P Niemi (Eds) *Psychology in the 1990:s.* Amsterdam: North-Holland

Marton, F., Gustafsson, J.-E. and Lindström, B. 1986. INTERAKTIVA DIDAKTISKA MILJÖER. Projektbeskrivning. (Projektansökan till Riksbankens Jubileumsfond.) Göteborgs universitet, Institutionen för pedagogik. [INTERACTIVE DIDACTIC MILIEU:S. A project description. Department of Education and Educational Research, University of Göteborg.]

Neuman, D. (1987). *The origin of arithmetic skills. A phenomenographic approach.* Göteborg: Acta Universitatis Gothoburgensis.

Neuman, D. (1990). Datorn som möjlighet för elever med matematiksvårigheter. *Report 1990:08,* Department of Education and Educational Research, University of Göteborg. [The computer as a resource for children with mathematics difficulties.]

Papert, S. (1980). *Mindstorms. Children, computers and powerful ideas.* New York: Basic books.

Sinclair, J. M., and Coulthard, R. M. (1975). *Toward an analysis of discourse*. New York: Oxford University Press.

Stubbs, M. (1986) *Educational linguistics*. Oxford: Basil Blackwell

White, B. Y. (1984). Designing computer activities to help physics students understand Newton's law of motion, *Cognition and Instruction.* 1: 69-108.