Ten Design Lessons from the Literature on Child Development and Children’s Use of Technology

Heléne Gelderblom
School of Computing, University of South Africa,
PO Box 392 UNISA, 0003, South Africa
+27 12 429 6631
geldejh@unisa.ac.za

Paula Kotzé
Meraka Institute, CSIR and School of Computing, University of South Africa
PO Box 395, 0001, Pretoria, South Africa
+27 12 841 4791
paula.kotze@meraka.org.za

ABSTRACT
The existing knowledge base on child development offers a wealth of information that can be useful for the design of children’s technology. Furthermore, academic journals and conference proceedings provide us with a constant stream of new research papers on child-computer interaction and interaction design for children. It will require some effort from designers to gather and digest the scattered research results and theoretical knowledge applicable to their products. We conducted an extended research project whereby the existing knowledge relating to the design of technology for children aged five to eight have been gathered and presented in a way that makes it accessible and useful to designers in practice. This paper provides and extract from that research, focusing on ten useful lessons learnt from existing literature.

Categories and Subject Descriptors
D.2.2 [**Design Tools and Techniques**]: User interfaces; H.1.2 [**User/Machine Systems**]: Human, factors, Human information processing, Software psychology; J.4 [**Social and Behavioral Sciences**]: Psychology; K.3.1 [**Computer Uses in Education**]: CAI.

General Terms
Design, Human Factors, Theory.

Keywords
Interaction design for children, Children’s technology, Design guidelines, Cognitive skill development.

1. INTRODUCTION
Working from the assumption that knowledge available in the literature provides sufficient information, we conducted a study through which a dependable and useful set of guidelines for the design of technology for children aged five to eight years was derived from an existing body of knowledge. In doing this, we appreciate the value of user input and usability testing in the design process and admit that no set of guidelines alone will guarantee design success. We do, however, believe that proper guidelines can substantially reduce the amount of usability testing required and hence the cost of design and development.

Development of the guidelines firstly involved research into the psychological theories of children’s development to identify those elements of development and the characteristics of children that may have bearing on children’s use of technology. Secondly, the literature on children’s development of specific skills such as literacy and mathematics was investigated. The available literature on young children’s use of technology was studied next and, finally, the applicability of existing design guidelines and principles for children’s products evaluated. Throughout this literature investigation we gathered design-relevant factors that were systematically coded, processed, analysed and categorised. The result is three hundred and fifty guidelines organised into a framework that integrates the relevant theoretical fields and provides practical support for designers.

This paper provides an extract from that research. Many of the guidelines that emerged are obvious, and experienced designers will probably apply them naturally. Some are less obvious and provide insights that need to be made explicit. Five to eight-year-old users have needs and preferences that differ from that of other user groups. They can, for example, be entertained by the same sequence of events over and over, they care little about the efficiency of an input device, and they have limited short-term memory capacity. For this paper we have chosen a subset of guidelines that we formulate here as lessons learnt from the literature. For each lesson we discuss the particular literature sources it is grounded in and we sometimes illustrate their importance through practical examples.

The aim with this paper is thus to share with the interaction design community some of the things we have learnt through and extensive study of the literature on cognitive skill development and children’s technology, showing that knowledge that exists in the literature can provide designers of young children’s technology with valuable insights.

The broader research on which this paper is based can be described as descriptive, applied and qualitative [34]. It was *descriptive* in the sense that we aimed to give narrative-like developmental characteristics of children aged five to eight that may have some bearing on their relationship with technology. It was *applied* as the results of our research will assist people with problem-solving and decision-making in the context of designing technology. It was *qualitative* because the data was in the form of

IDC 2009, June 3–5, 2009, Como, Italy

written language that we analysed by identifying elements, themes or patterns that helped us in organising the information into a framework of design guidelines.

Research methodology has three elements, namely sampling, data collection and data analysis [15]. We used multiple case sampling [28] to establish a trustworthy profile of five to eight year old children and reputational case selection [28] to choose the relevant theories of development, research papers, and existing design guidelines for the literature investigation. There was no clear distinction between data collection and data analysis in our study. We started with a process of familiarisation with and immersion in the chosen texts followed by the identification and coding of elements that potentially relate to design guidelines. We explored these elements further, translating them into guidelines and finally identified themes and categories according to which we organised the guidelines into a useful framework.

2. SCOPE OF THIS PAPER

We chose to do this research from a cognitive development point of view rather than from an educational point of view. The developmental domains that we focused on are literacy, mathematics and thought. Literacy and mathematics are the two learning areas where timely acquisition of the skills is necessary for a solid foundation and a positive attitude toward the learning content. A large percentage of the available technology for children addresses these two skill domains and many of the studies on young children and technology are based on experiments with children using software that support literacy or mathematics. A number of the guidelines included in this paper refer specifically to mathematics skill development. We could not cover all the skill domains in the space available.

Since cognitive development and education are inseparable, the resulting guidelines naturally apply to the design of educational technology.

Our focus is on informing the design of technology aimed at children aged five to eight years. Since we emphasise the importance of age-appropriateness of technology, the results are not intended for generalisation to other age groups. While focusing on a specific age group and trying to establish a user profile of five to eight-year-old computer users, we do acknowledge the variability of children's development. We strongly support the idea of adaptation to individual variation, as some of the lessons presented here will confirm.

For this paper we selected, from the complete framework, ten themes to focus on. We present the guidelines associated with these themes as ten 'lessons' grounded in respected theoretical or empirical research results. The restriction on the length of this paper did not allow us to include more. Our main reason for choosing these particular lessons is that they each offer an interesting example of how existing theory or research can be translated into a useful guideline for the design of technology for young children. For the complete set of guidelines the reader is referred to [19].

To place the research in context, we next review a fraction of the literature on young children and technology. The rest of the paper presents the ten lessons with their justification from the literature and we end with a short conclusion.

3. YOUNG CHILDREN AND TECHNOLOGY

Computers will never replace important play and learning material such as paint, blocks, sand, water and books [29], but technology does provide new and exciting opportunities for childhood activities. Computer technology makes it possible for children to easily apply concepts in a variety of contexts [31]. It exposes them to activities and knowledge that would not be possible without computers. A literature survey by Roschelle et al. [31] revealed the general finding that the use of computer aided instruction or, in some cases, ordinary applications such as word processors, improve achievement in the following areas: writing skills, remedial writing, verbal and nonverbal creativity, mathematics, phonological awareness, learning time, positive attitude to learning, auditory skills, language skills, story telling, meta-cognition, reasoning skills and independent thinking.

People opposed to the use of computers by young children have warned against some potential dangers such as keeping children from other essential activities, causing social isolation and reduced social skills, and reducing creativity. There is general agreement that young children should not spend long hours at a computer, but computers do stimulate interaction rather than stifle it [21]. Current advances in technology make it possible to create applications that offer highly stimulating environments and opportunities for physical interaction. New tangible interfaces are changing the way children play with computers [30]. An argument against early computer use is that children are being 'rushed'. Clements [8] responded that the possibility that children can be pushed to learn to write to soon do not make us keep pencils and paper away from them until they are ready. The important thing is to allow children to perform activities on the computer that are at their level of development. For Clements [9] developmental appropriateness means 'challenging but attainable for most children of a given age range, flexible enough to respond to inevitable individual variation, and, most important, consistent with children's ways of thinking and learning' (p.161). According to Haugland and Wright [21] the benefits of developmentally appropriate computer experiences for young children are:

- It provides opportunities to acquire and construct knowledge through active participation.
- It provides a holistic learning environment in the sense that by exploring virtual environments they acquire knowledge and skills in different domains of development.
- It promotes intrinsic motivation to learn by providing children with challenge, control, fantasy and feeding their curiosity.
- It provides children with scaffolding that enables them to acquire skills faster. (For example, children can type letters on a keyboard before they can make proper letters with a pen and this makes it possible for them to communicate through writing earlier.)
- It connects children to the world by providing access to people and resources throughout the world.
- It gives them access to a huge amount of information.

In general, technology is not regarded as a threat any longer and the potential benefits of young children's exposure to it are generally accepted.

Clearly, technology has become an important element of the context in which today's children grow up and it is important to understand its impact on children and their development. According to Druin [13] we should use this understanding to improve technology so that it supports children optimally. The development of any technology can only be successful if the designers truly understand the target user group. Knowledge of children's development and familiarity with the theories of children's cognitive development is thus essential when designing for them. Also, a lot of research has been done over the past thirty years on children's use of technology that designers can learn from.

4. TEN DESIGN LESSONS

The framework of three hundred and fifty guidelines presented in [19] organises the guidelines into six broad categories: developmental appropriateness; development of specific skills; design of built-in support; catering for diversity of users; interaction environments and devices; and support of collaborative use of technology. The first five are represented in the ten lessons below. Lessons 1 to 4 relate to a combination of developmental appropriateness and the development of mathematics skills. Lesson 5 relates to built-in support and lesson 6 to addressing diversity of users. Lessons 7 and 8 refer to speech input and output, while lesson 9 deals specifically with mouse use. Lesson 10 shows how well-known design principles get special meaning when applied to children's products.

As we said before, a consequence our focus on the support of cognitive skill development is that many of the resulting guidelines relate to the design of educational products. The first six lessons involve such guidelines while the last four refer to general interaction design.

Lesson 1: If a child can solve a specific kind of problem in one domain that they cannot necessarily transfer that skill to a different domain

In his neo-Piagetian theory of development, Case identified knowledge and control structures that transpire in the child's mind as categories, event scripts, strategies, rules and plans [17]. Whether a child is capable of a particular activity depends on the structures that they have available that relate to that activity [5]. According to Case, children construct a specific cognitive structure independently of any other structure. How they do this depends on the context within which they find themselves as well as on their prior learning history [6]. So, if a child develops a specific knowledge structure in the context of playing a computer game and there is no clear link between the way the knowledge transpires in the game and the way that it is used in real life, the child may not be able to apply it in real life.

The opposite is also true. If a child learns something in a real life situation, they cannot necessarily transfer the underlying concepts to a general, abstract level. For example, when children have accomplished simple division problems in a scenario where they have to help a character share a specific number of biscuits fairly with a number of friends, a program can present similar problems in a purely mathematical context. To help them generalise the division skill, the program should make the link between the two contexts explicit.

Fischer's dynamic skill development theory has a similar view of skill transfer [16]. It regards 'skill' as a concept that includes both person and environment. Skill is task-specific, context-specific and dependent on factors such as emotion, memory, culture, experience and biological maturation. Through constructive generalisation and repeated rebuilding, a skill that begins as task or context-specific can gradually be extended to other contexts [17].

Following these views, designers should support independent development of skills in different domains, while at the same time, considering how a skill is applicable across domains. Designers should also make any skill's connection with real life explicit. An activity chosen to develop a skill must preferably be one that can be naturally associated with that skill.

Using mathematics as an example, any product that supports the development of mathematics skills should encourage children to apply their mathematical skills and knowledge in real-world situations and provide links between different cognitive domains.

TimezAttack [1] succeeds in integrating fun with practicing multiplication tables but it fails to demonstrate how the things children memorise by playing the game are used outside the game environment.

One author's daughter recently provided a good example of how strongly a child can link the mathematical content of a game to the game environment. It was suggested that she use a thirty minute break to study the 'times eight' table for a test the following day. When asked whether she did this she said that she could only study three sums in the thirty minutes. Later the author found the piece of paper on which she practiced the 'times eight' table. For each sum she had drawn a detailed recollection of a scene from TimezAttack containing one 'times eight' sum as the program would present it. Figure 1 shows one of the drawings. Clearly she could not detach the multiplication tables from the game.

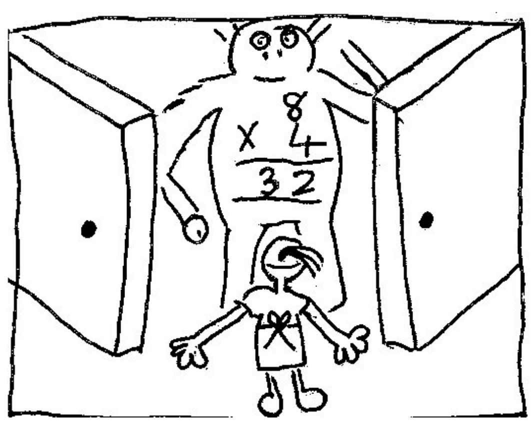

Figure 1 Child's drawing of a scene from TimezAttack

Lesson 2: Young children find it difficult to translate between the formal system of mathematics and the quantities, operations and concepts it represents

Given that preschool children can already reason coherently about number, Hughes [22] investigated why young children find school arithmetic so difficult. He attributes the problem to the formal code of arithmetic system which is like a foreign language to young children. In order to use this symbol system they need to translate from the symbols to the concepts they already have. Young children can answer a question such as 'how many is two lollipops and one more?' correctly, but when asked 'what does one and two make?' they do not interpret it as 'what does one object and two objects make?'. There lacks a connection between our number words and the numbers that they represent [22].

Children learn number words while they are still learning their language and have to distinguish the number words from hundreds of other words. The language they are brought up with and the way numbers are used by the people around them will therefore influence their perception of number [14]. Hughes refers to an early Indian system where the word for 'one' means moon, that for 'two' means eyes, and so on. It may be that children would find it easier to do simple arithmetic with number words of which the meaning are in some way connected to the numbers they represent. To aggravate this problem for English-speaking children, 'two' sounds like 'to', 'four' sounds like 'for' and the word 'one' is used as a number or a pronoun. Other examples of words that may be confusing are 'table', 'odd', 'even' and 'volume'. When presenting such terms to children, designers must make sure that the children attach the correct meaning to the word [14].

In accordance with Hughes' theory, Griffin [20] views mathematics as comprising of three worlds, namely the world of real quantities that exist in space and time, the world of spoken counting numbers and the world of formal symbols that consists of numerals and operation signs. Competence in mathematics depends fundamentally on the development of relationships among these worlds, and thus, the ability to translate between concrete situations and formal code [20].

In most countries children start formal schooling at the age of five or six and are then first introduced to written arithmetical symbolism [22]. Generally they use workbooks to learn arithmetic and have to complete sums such as 2 + 4 = ? and 5 - 2 = ?. Hughes and colleagues conducted an experiment to determine how natural it is for children to write down numbers and sums in this form [22]. When asked to write down the number of objects (e.g. bricks) displayed to them, only 38% of children from five to seven used conventional numerals. 45% drew the required number of objects, others used vertical strokes or blob-like shapes and some drew the appropriate number of some object (other than the one displayed). When asked to write down a simple subtraction or addition sum (e.g. suppose we have ten bricks and take away five), 69% of the children represented only the final number of objects. Only eleven of seventy-two children could differentiate between addition and subtraction and only four of these did it in a way that could be understood by others. One wrote 'took a away' and 'add 3', one superimposed the added bricks on the others, one drew a hand adding bricks and one drew dashes through the bricks that had to be removed. Not one child used the conventional + and – signs to represent the operations despite the fact that they were using these regularly in their workbooks. These findings suggest that many children do not realise that the symbols they use in their workbooks can be used to represent quantities of real objects or the operations on these quantities.

To summarise the implications for designers: Keep in mind that children find it difficult to translate between the formal system of mathematics and the quantities, operations and concepts they represent. Do not assume that they can correctly associate even the most basic number or operator symbols with real quantities or operations. Only introduce operator symbols such as + and –, and their associated operations, when children can use the number symbols confidently. Help children make the connection between formal representations and dynamic visual representations, thereby supporting construction of mathematical ideas through visual approaches. Make sure children will interpret activities correctly – when they follow audio instructions ensure that they understand numbers and concepts that sound like other words (e.g. two, table, odd) correctly. Always consider how the context may influence their understanding.

Lesson 3: Do not separate the instructional part and the fun part of a product

We discuss this in the context of mathematics skill development. Recent products that support the development of mathematics skills often still include drill-and-practice activities but much improved graphic capabilities make it possible to embed them in a storyline or an engaging on-screen environment. Soloway and Norris [33] believe that the mere fact that computer-based drill and practice is more fun makes it superior to paper-based practice. What they have against such software is that the reward for success is often unrelated to the mathematics just mastered. When children achieve success in such a system they often get the award of appealing feedback or the chance to play some delightful game. If these rewards are not linked to what they have just learnt it sends the message that fun comes after mathematics and thus that mathematics itself is not fun [33].

TimezAttack [1] is a good example of a product that successfully integrates the game module and the instructional module. Practice of the multiplication tables is disguised in repeated conquering of ogres or other nasty characters. All learning happens in the game.

Lesson 4: Allow children to use different strategies in problem-solving activities

We again use mathematics as an example. Children have different preferences with regard to strategy use. The variability in strategy use depends on factors such as brain maturation, different levels of understanding of mathematical concepts and procedures, as well as on experience.

The initial addition and subtraction strategies that preschool and grade R (kindergarten) children use are external representations such as fingers or counters [4]. For addition they use the 'counting-all' strategy where each number set is counted and then the combined set to get the answer. Some may use the 'count-on' strategy where one number is represented with the fingers or counters and the second number is counted on from there. For subtraction, they represent all of the numbers from which to subtract, remove the number of objects that need to be subtracted and count what remain to get the answer. They can also represent the number to be subtracted and then 'add-on' counters, counting until they get to the subtrahend. Here they need to realise that the counters that were added on represent the answer. A third way to do subtraction is by 'matching'. The subtrahend and the number to be subtracted are lined up so that the unmatched counters give the answer.

In the first grade children start to mentally 'count-on', 'add-on' and 'count-back', without having to represent the numbers physically. By grade two (around eight years of age), some children begin solving subtraction problems by turning the problem around and using addition to get to the answer [4].

For multiplication, young children use 'direct counting', by counting out the required number of sets of counters and then counting them all to get the answer. The second strategy is 'repeated-addition' and the last one 'multiplicative calculation' where the answer is drawn from memory or through derived facts [4].

To support the variability in children's strategy preferences, Carr and Hettinger [4] say that teachers (and, by implication, technology that supports mathematics skill development) should be flexible in the strategies they require and provide opportunities for using different strategies to solve different kinds of problems. Children should be provided with as many views as possible on a specific problem. Successful strategy use requires good conceptual understanding of the strategy, therefore, when teaching children to use a strategy, teachers (and technology) have to emphasize the underlying concepts. Practice and play with different kinds of mathematical problems lead to the development of more sophisticated strategy use. Manipulatives should only be used in the earliest stages of mathematics development or in situations where older children do not understand the underlying concepts. The use of manipulatives requires a lot of working memory capacity and may hinder the acquisition of more complex mathematics skills [4].

Lesson 5: Promote reflective thinking and skill development by making children aware of the processes underlying success or failure

Providing encouragement is not sufficient – children need to be told what specific actions or choices led to the correct or incorrect result to help them to generalise from an experience to future ones. Although given sixteen years ago, Klein and Nir Gal's [25] opinion still holds: children should be provided with immediate feedback about their performance and give specific reasons for success or failure. This is what teachers and parents do when teaching a child a new skill. For technology to be successful in supporting development it should too. Klein and Nir Gal [25] suggest different ways in which mediation can be incorporated into the construction of software. Specific mediation variables that can potentially be incorporated into software are focusing (to ensure that the child focuses on the correct interface element through actions such as selecting, exaggerating, accentuating, scheduling, grouping, sequencing or pacing), affecting (to attract the child's attention to the concepts he or she used to solve the problem through verbal or non-verbal affect), expanding (through behaviour that may improve cognitive awareness) and encouraging (through verbal or non-verbal expression of satisfaction with specific components of a child's behaviour). To these, Klein, Nir Gal and Darom [26] added mediated regulation of behaviour.

Klein et al. [26] report on several studies that found adult mediation during computer activities improves children's abstract reasoning, logical thinking, and analogical and reflective thinking. The aim of their research was to identify the specific characteristics of successful adult-child mediation with reference to children's computer use. They compared three levels of adult guidance, namely mediation (including the variables listed above), accompaniment (availability of an adult to answer children's questions) and no assistance (with only technical or basic instructions provided at the beginning of a new activity). They found that children interacting with trained adult mediators scored significantly higher than other children on measures of abstract thinking, planning, vocabulary, visio-motor coordination and responsiveness (including reflective thinking). No difference was found between the children who had the accompaniment of an adult to answer questions and those who only received initial technical assistance [26].

An interesting observation during the study was that children who used computers at home scored lower on visual association tests than children who did not have computers at home. An explanation for this is that since children do not receive mediation at home, they internalise a trial-and-error way of solving problems without any conceptualisation [26]. Klein et al.'s findings suggest that children using computers without sufficient mediation do not get the full advantage of computer technology for development. Through mediation they learn to focus on a problem, to seek and receive precise information, to compare different perceptions and to plan their actions.

Lesson 6: A skill may be taught or acquired differently by children from different cultural groups

If a product is aimed at children from different cultural groups, designers should first investigate how the different groups use and teach the skills that the product will support.

Case recognised domain-specific developmental changes that are influenced by the nature of the tasks and children's varying experience [27]. For example, a child who often listens to or tells stories, but never draws will have more advanced conceptual structures in the story domain than in the drawing domain. He further acknowledged the way culture presents children with opportunities for development and how different cultures provide different tools for problem solving [5]. Case attributed variations in children's patterns of development to cultural and sub cultural differences, specific problems that are typical within that culture and with which they are confronted frequently, and the models that the culture provides for solving those problems. Designers should therefore acknowledge the culture and sub-culture of the intended users and identify particular problems that are important in that culture and the tools used to solve that kind of problem.

According to Case, a child's cognitive development depends on the structures they have available that relate to their current task or situation, what they can do with that information and their mind's capacity for information processing [27]. Presenting a mathematical problem in the context of paying a restaurant bill may be suitable for some cultures, but many children may not have scripts for 'eating in a restaurant'.

Vygotsky was probably the first modern psychologist who regarded culture and society as defining factors in human development [10]. His theory is based on the belief that children's mental development is closely tied to the social context in which they grow up. This context is made up of the people that interact with the child, as well as the child's experiences with art, language and culture [27]. Vygotsky regards play as an important part of children's growth and sees children's games and the things they use as toys as the means by which culture is integrated with development [24]. Cognition does not happen only in the mind, but in the interaction between the mind and material artifacts and social practices. Through these cultural elements knowledge is transferred from one generation to the next [11].

Designers should, however, be careful not to overestimate the importance of cultural differences. Chimbo and Gelderblom [7] challenge the general assumption that technology should either be adaptable to or cater for diverse cultural backgrounds. In an experiment with a culturally diverse group of seven and eight-year-old South African children using an American storytelling application, they found that gender was a stronger influencing factor in the interaction than culture. The only culture-related recommendations that resulted from the study are that a storytelling application should ideally be adaptable to the user's language, allowing second language English speakers to choose a country-specific English, spelling and pronunciation, and the selection of story characters and objects should be representative of as many ethnic groups and nationalities as possible, so that any child can find a character that resembles him or herself. A large percentage of children in non-western developing countries have access to television and they have a lot of exposure to American and British television programs – most of what appears in American or British computer games will therefore be quite familiar to them.

Finally, while considering their target users' cultural variability, designers must also acknowledge their own culture and context and how that may consciously or subconsciously influence their design practice.

Lesson 7: Do not rely on children's accurate recall of audio instructions

Dix et al.'s [12] principle of persistence is very relevant in children's products. It refers to the duration of the effect of a communication act and the ability of the user to make use of that effect. Audio communication persists only in the user's short-term memory while visual communication remains available as long as the user can see the display. Phonological short-term memory improves dramatically during early and middle childhood with the memory span (number of items that can be held) doubling from age five to age fourteen [18]. The short term memory capacity of a young child is thus more limiting in terms of design than that of an adult. Providing young children with audio instructions in the beginning of a game may not be adequate as they may not be able to remember the instructions until they are needed. The fact that pre-reading children rely on audio instructions makes this an important consideration.

The phonological store keeps information for only a few seconds [18], so designers should not expect a child to act on audio cues that occurred some seconds ago. Verbal instructions should be short.

Lesson 8: Give children control over the level and frequency of speech feedback

Speech feedback is very useful in applications that teach children to write, but it comes with some implementation difficulties. Shilling [32] questions the value of speech-synthesised feedback when used by beginning writers who invent their own spelling. Their phonetic way of spelling, often leaving out vowels, can produce confusing feedback from the speech-synthesiser. For example, when a child writes 'DVD' for 'David', the synthesiser responds by naming each letter [32]. Shilling believes that speech-synthesised feedback is most effective when children already have some reading ability. Designers should keep in mind that speech feedback may be confusing if given on misspelled words. This can, however, be used in a way that makes the interaction fun. Ideally a product should give users the option to set up the system so that the feedback supports the user in the best possible way, and according to the taste and requirements of the individual user.

Different speech-software handles speech feedback differently [2]. In some applications the user can select feedback on letter, phoneme, word or sentence level. The word-processing software that Borgh and Dickson [2] used for their research activated the Vortax Personal Speech System whenever the user typed a period, question mark or exclamation point. At this point it 'reads' the last sentence. The user is given a choice to listen again, change the sentence or continue, or they can ask to listen to the whole text being read back. They found that children using the speech feedback system edited their writing significantly more than children who used a normal word-processor [2]. Children responded positively to the speech-synthesiser, although some indicated that they would prefer to have more control over when feedback was given. Children should thus be able to change the level of speech feedback – that is, whether it should be at letter, phoneme or word level or even be switched off completely.

In his discussion of children's software evaluation, Buckleitner [3] emphasises the importance of the intended function of the software as well as the context in which it will be used. Software that teaches a specific skill, such as letter recognition, in a quiet classroom setting should be measured against different criteria than a software game that requires a child to race a car in a noisy games room context. So, when designing software, it is important to have a clear idea of the context in which the software will be used. In a classroom setting audio feedback may disturb classmates. Beeps, music or agent voices coming from ten or twenty computers in a media room may be unbearable, but the same sounds may be acceptable for home use or in a shopping mall games room. Adaptability to the context of use is therefore also important.

Lesson 9: Young children perform point-and-click quicker and more accurately than drag-and-drop

Inkpen [23] conducted extensive research on the effect of mouse interaction style on children's performance and motivation when playing a computer game. Her experiments involved a puzzle-solving computer game of which some subjects used the IBM-version and some the Macintosh version. The former uses a point-and-click interaction style. An object is moved by first clicking on it, then clicking on the point to where is should move and finally clicking again to make the move. When the object is first clicked it disappears and an iconified picture of it is attached to the cursor. The Macintosh version uses the drag-and-drop style, whereby the mouse is kept down on the object while dragging it to the desired position and then releasing the mouse. In this version, if the object is dropped in a position that is already occupied it causes an error and the object is returned to its original position where the user must pick it up again. The IBM version gives the user another chance to pick a location. Another task involves connecting two screen objects with a connector object. This requires a more complex sequence of mouse actions than merely moving an object. The IBM version still uses mouse clicks only, while the Macintosh version here uses a combination of mouse click and drag actions.

Inkpen's [23] first experiment found girls more successful at playing the IBM version (where success was measured by the number of puzzles they could solve in the game). Girls using the IBM version were more motivated as demonstrated by the fact that 21% of the drag-and-droppers left early as opposed to 6% of the point-and-clickers. The overall conclusion of the first experiment was that point-and-click is the more effective option in terms of performance and motivation.

The second experiment involved girls and boys using both versions of the game. According to the results of this experiment, children who used the drag-and-drop style first were more likely to state a preference for point-and-click than those who did the point-and-click task first. Of all children, 66% preferred point-and-click, 28% preferred drag-and-drop, while 6% had no preference. Point-and-click proved to be the more effective interaction style in terms of speed as well as accuracy. The reasons children gave for preferring the point-and-click style are that their fingers became tired of holding the mouse button down. Children who preferred the drag-and-drop style said that they were more familiar with the style or that the tactile feedback helped them in the task. Other research confirms that kinaesthetic connectivity of holding the mouse down to hold on to the object can reinforce the conceptual connectivity [23].

The conclusion is that point-and-click interaction is faster than drag-and-drop and leads to fewer errors. Inkpen acknowledges the possibility that the results are task dependent, but the results have been confirmed by other research in this regard. The impact of the chosen style also depends on the size of the objects and the distance the objects need to be moved.

Lesson 10: Reachability, familiarity and substitutivity have special meaning for children's products

Most of the existing design guidelines or principles have been formulated with products for adults in mind and need reinterpretation when applied to products for young children.

Dix, Finlay, Abowd and Beale's [12] principle of *reachability* refers to the possibility of navigation through the observable system states. Young children often discover an aspect of a system that they particularly enjoy and will want to get to that part easily to play it repeatedly. Designers should allow children to go directly to their favourite parts of an application and should avoid forcing children to go through a fixed series of activities. Unless a specific sequence of actions is necessary (as when progressively learning multiplication tables), allow children to go directly to their favourite parts of the system.

An interface should use language and concepts that the user is familiar with. *Familiarity* [12] has a different meaning for children than for adults. Children have limited world experience and what may seem to adults like fantasy can be very real to children. Adults are not always good at judging what children will find familiar or what not and designers should consult the users in this regard. Uden and Dix [35] tested several full colour representational icons for email and movie watching functions on a web interface for young children. Many of the icons were meaningless to five-year-olds. For example, a printing press, a fountain pen, an old-fashioned typewriter, and email icons that depend largely on text or recognising the '@' sign. Some are culture specific and only has meaning in an American context, for example, images of specific mailboxes found only in the United States. The cameras used in the 'watch a movie' icons are mostly old-fashioned movie cameras and the television set has legs and an aerial on top. Uden and Dix speculate that the reason for the wide use of old fashioned images rather than pictures of modern versions, is that the new equipment is not easily distinguishable in a small image [35]. Unfortunately the features that make the old-fashioned images more recognisable for adults are what may make them meaningless to a young child. It is possible that children could have seen these objects in picture books and cartoons, but they may not make the right connections in the context of a computer application. Animated icons are more successful as the action sequence may reinforce the icon's meaning in the interface. In Uden and Dix's experiment the children surprisingly found a cameraman with an old-fashioned camera as the best indication of the movie watching function [35]. The researchers give the frequent occurrence of cameramen in children's media as a possible explanation. This unexpected outcome emphasises the importance of extensive user testing when choosing icons for a graphical interface.

A third principle that has special meaning for children's products is s*ubstitutivity* [12]. Children have varying skill levels and preferences that will influence the type of input or output that is suitable for a specific user. Designers should therefore allow equivalent values of input and output to be arbitrarily substituted for each other, or provide multi-modal input and output. Different modalities (channels of communication) can be combined to improve articulation of input or output or to make the system accessible to more users.

5. CONCLUSION

Designers should not rely on their intuition or memories of their own childhood when designing for children. They cannot merely interview some children, ask them about their preferences, give them questionnaires to fill out and come up with a profile of the intended user. Cooperative design with young children, like research with young children, is a specialised skill that requires training and experience.

This paper showed that designers can gain knowledge about children from experts such as developmental psychologists and researchers in the field of interaction design for children. Immersing ourselves in the work of authorities, we discovered various guidelines for the design of technology for young children and here shared some of the valuable lessons

6. REFERENCES

[1] BigBrainz: *TimezAttack*. 2005, BigBrainz, Inc.

[2] Borgh, K.,Dickson, W.: The Effects on Children's Writing of Adding Speech Synthesis to a Word Processor. *Journal of Research on Computing in Education*. 24(4) (1992), 533-554.

[3] Buckleitner, W.: The State of Children's Software Evaluation – Yesterday, Today, and in the 21st Century. *Information Technology in Childhood Education*. (1999) (1999), 211-220.

[4] Carr, M.,Hettinger, H., *Perspectives on Mathematics Strategy Development*. In: J.M. Royer, (eds.): Mathematical Cognition. Information Age Publishing, 2003, 33-68.

[5] Case, R.: *The Mind's Staircase*. Lawrence Earlbaum Associated Publishers, (1992).

[6] Case, R., *A Neo-Piagetian Approach to the Issue of Cognitive Generality and Specificity*. In: R. Case, (eds.): The Mind's Staircase. Lawrence Erlbaum Associates, Publishers, 1992, 17-36.

[7] Chimbo, B.,Gelderblom, H.: The Suitability of American Storytelling Software for South African Diversity. In *To appear in Proceedings of MCCSIS 2008*. 2008. Amsterdam.

[8] Clements, D.H.: Computers and Young Children: A Review of Research. *Young Children*. 43 (1987), 34-44.

[9] Clements, D.H.: Computers in Early Childhood Mathematics. *Contemporary Issues in Early Childhood*. 3(2) (2002), 160-181.

[10] Cole, M.,Scribner, S., *Introduction*. In: L.S. Vygotsky, (eds.): Mind in Society. Harvard\University Press, 1978, 1-14.

[11] Crook, C., *Cultural Artefacts in Social Development: The Case of Computers*. In: H. McGurk, (eds.): Childhood Social Development: Contemporary Perspectives. Lawrence Earlbaum Associates, 1992, 207-231.

[12] Dix, A., Finlay, J., Abowd, G., Beal, R.: *Human-Computer Interaction*. Third edition ed. Pearson Education Ltd, (2004).

[13] Druin, A., *CHIKids: A Common Ground for Kids and Adults*. In: M.J. Tauber, (eds.): Common Ground: CHI 96 Conference Proceedings. ACM, 1996, 506-507.

[14] Durkin, K., *The Representation of Number in Infancy and Early Childhood*. In: C. Pratt and A.F. Garton, (eds.): Systems of Representation in Children: Development and Use. John Wiley and Sons, Ltd., 1993, 235-250.

[15] Durrheim, K., *Research Design*. In: M. TerreBlanche and K. Durrheim, (eds.): Research in Practice: Applied Methods for the Social Sciences. UCT Press, Cape Town, 1999, 29 - 53.

[16] Fischer, K.,Bidell, T.R., *Dynamic Development of Action and Thought*. In: R.M. Lerner, (eds.): Theoretical Models of Human Development. John Wiley & Sons, Inc, 2006, 313-399.

[17] Fischer, K.W.,Immordino-Yang, M.H.: *Cognitive Development and Education: From Dynamic General Structure to Specifc Learning and Teaching*. 2002, Harvard Graduate School of Education, 1-55.

[18] Gathercole, S.E., *Memory Development During the Childhood Years*. In: A.D. Baddaley, M.D. Kopelman, and B.A. Wilson, (eds.): The Handbook of Memory Disorders. John Wiley & Sons, Ltd, 2002, 475-500.

[19] Gelderblom, J.H.: *Designing Technology for Young Children: Guidelines Grounded in a Literature Investigation on Child Development and Children's Technology*. PhD Thesis, School of Computing, University of South Africa, 2008.

[20] Griffin, S., *The Development of Math Competence in the Preschool and Early School Years*. In: J.M. Royer, (eds.): Mathematical Cognition. Information Age Publishing, 2003, 1-32.

[21] Haugland, S.W.,Wright, J.: *Young Children and Technology: a World of Discovery*, New York, (1997).

[22] 22. Hughes, M., *What is Difficult About Arithmetic*. In: P. Light, S. Sheldon, and M. Woodhead, (eds.): Learning to Think. Routledge, London, 1991, 183-204.

[23] Inkpen, K.M.: Drag-and-Drop versus Point-and-Click Mouse Interaction Styles for Children. *ACM Transactions on Computer-Human Interaction*. 8(1) (2001), 1-38.

[24] John-Steiner, V.,Souberman, E., *Afterword*. In: L.S. Vygotsky, (eds.): Mind in Society. Harvard University Press, 1978, 121-133.

[25] Klein, P.,Nir-Gal, O.: Effects of Computerized Mediation of Analogical Thinking in Kindergartens. *Journal of Computer Assisted Learning*. 8 (1992), 244-254.

[26] Klein, P.S., Nir-Gal, O., Darom, E.: The Use of Computers in Kindergarten, With or Without Adult Mediation: Effects of Children's Cognitive Performance and Behavior. *Computers in Human Behavior*. 16(6) (2000), 591-608.

[27] Meece, J.L.: *Child and Adolescent development for educators*. 2 ed. McGraw-Hill, Boston, (2002).

[28] Miles, M.B.,Huberman, A.M.: *Qualitative Data Analysis*. 2 ed. SAGE Publications, (1994).

[29] NAEYC: *NAEYC Position Statement: Technology and Young Children - Ages 3 through 8*. [Web document] (1996) [cited 2004 17 February]; Available from: http://www.naeyc.org/resources/position_statements/pstech98.htm.

[30] Plowman, L.,Luckin, R.: Interactivity, Interface, and Smart Toys. *IEEE Computer*. February(February) (2004), 98-100.

[31] Roschelle, J.M., Pea, R.D., Hoadley, C.M., Gordin, D.N.: Changing How and What Children Learn in School with Computer-Based Technology. *The Future of Children*. 10(2) (2000), 76-100.

[32] Shilling, W.: Young Children Using Computers to Make Discoveries About Written Language. *Early Childhood Education Journal*. 24(4) (1997), 253-259.

[33] Soloway, E.,Norris, C.: Using Technology to Address Old Problems in New Ways. *Communications of the ACM*. 41(8) (1998), 11-18.

[34] TerreBlanche, M.,Durrheim, K., *Histories of the present: Social science research in context*. In: M. TerreBlanche and K. Durrheim, (eds.): Research in Practice: Applied Methods for the Social Sciences. UCT Press, Cape Town, 1999, 1 - 16.

[35] Uden, L.,Dix, A., *Iconic Interfaces for Kids on the Internet*. In: (eds.): Proceedings of IFIP WorldComputer Congress, August 2000. Beijing, 2000, 279-286.