DESIGNING TECHNOLOGY FOR YOUNG CHILDREN: GUIDELINES GROUNDED IN A LITERATURE INVESTIGATION ON CHILD DEVELOPMENT AND CHILDREN'S TECHNOLOGY

by

JOHANNA HELÉNE GELDERBLOM

Submitted in accordance with the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in the subject

COMPUTER SCIENCE

at the

UNIVERSITY OF SOUTH AFRICA

PROMOTOR: PROF P KOTZÉ

JUNE 2008

Dedicated to the memory of my father,

David Rosenblatt.

Abstract

This thesis is about the design of technology for children from five to eight years of age. The majority of available guidelines and principles for design or evaluation of technology support the design of products aimed at adults. The limited guidelines available for design of young children's technology do not focus sufficiently on age-related requirements or they offer high-level advice that is only useful in the planning stages of design. Working from the assumption that knowledge available in the literature provides sufficient information to support this process, my aim with this study was to demonstrate how a dependable and useful set of guidelines for the design of technology for children aged five to eight years could be derived from an existing body of knowledge.

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 the researcher gathered design-relevant factors that could potentially become design guidelines. Using qualitative data analysis techniques, more than five hundred such data elements were systematically coded, processed, analysed and categorised. The result is three hundred and fifty guidelines organised into a framework of six categories and twenty-six subcategories that integrates the relevant theoretical fields and provides practical support for designers. To demonstrate the credibility and usefulness of the emerging guidelines they were used to do an evaluation and re-design of an existing product aimed at the target group.

The thesis reports in detail on the different stages of the research, and systematically takes the reader through the process of deriving guidelines from existing theory and research findings, and integrating them into a useful framework.

Keywords: Child-computer interaction; young children's technology; interaction design; design guidelines; developmental psychology; cognitive development

Acknowledgements

I wish to thank the following people for the parts they played in the completion of this thesis:

- Professor Paula Kotzé, my supervisor, for providing me with the necessary guidance and academic support when needed.
- My husband, Derik Gelderblom, for always being a patient sounding board, for continuously encouraging me to continue, and for his valuable academic advice.
- My daughters, Magdalene and Emma (the dancers in Figure 9.4) who were my constantly available and willing research subjects and whose existence was always a reminder that there is more to life than work.
- My mother, Babs Rosenblatt, for her ongoing interest and for helping with the children during school breaks.
- My colleagues and friends in the School of Computing at Unisa: Mac van der Merwe, Estelle de Kock, Ruth de Villiers, Tertia Horne, Biffie Viljoen, Lucas Venter, Linda Miller, Marthie Schoeman, Judy van Biljon, Corné Reyneke, Petra le Roux, Alta van der Merwe and Mariki Eloff (for many kinds of support).

Table of Contents

Preface (A Note to the Reader)

This study involves a broad literature investigation with the purpose of extracting data in support of the formulation of design guidelines, and the subsequent formulation of a comprehensive set of guidelines for the design of technology for children from five to eight-years-old.

I report on the literature study in Chapters 4 to 7. During the process of writing down what I learn from the literature, I collect potential guideline-generating data. To make the research process transparent I present these data elements in data boxes. Below is an extract from one of the data boxes in Chapter 4. Each labeled element is a design-related factor or potential guideline-generating idea that comes from the preceding discussion. It is thus easy to trace back the statement to the literature on which it was based. In most data boxes I, therefore, do not cite the relevant source again. Some of these statements already sound like guidelines, while others are merely statements that can potentially be transformed into a guideline. During analysis of the data (reported in Chapter 8), all these statements will go through a process that may involve merging, splitting, grouping, questioning or discarding. Finally, they will all be reformulated as proper guidelines.

P14	Designers must acknowledge their own context and how that may consciously or subconsciously influence their design practice.
P15	They must consider the specific learning or entertainment goals of the product and how these goals fit the context of different kinds of users.
P16	Computer-based tasks for children should always be embedded in scenarios that children can relate to. These scenarios are important elements of the context of use.
P17	If a product is aimed at children from a variety of cultures designers may settle on one generic scenario, but it may be difficult to find one that all children can relate to.

It may seem like a lot of unnecessary repetition, but when I finally present the resulting set of guidelines that emerged from the literature, it is important that each individual guideline can be traced back to its origin in the literature. I could, instead, include the labels in my discussions of the literature, but this would be detrimental to the presentation and would hamper the reading process. Note that the labels are associated with the class of literature (and the corresponding chapter) through the letter it begins with:

P – Chapter 4 (Psychological theories)
T – Chapter 5 (Technology for children)
I – Chapter 6 (Interaction environments)
E – Chapter 7 (Existing guidelines).

The reader can thus skip the data boxes without missing any information, and only come back to them later, to check the origin of a specific proposed guideline presented in Chapter 8.

CHAPTER 1

Introduction

Table of Contents

1.1 Introduction

This thesis is about the design of technology for children aged five to eight years. The majority of available guidelines and principles for design of technology support the design of products aimed at adults. The limited guidelines available for design of young children's technology do not focus sufficiently on age-related requirements or they offer high-level advice that is mostly useful in the planning stages of design. I believe that there is a vast knowledge base, covering a range of theoretical and design-related fields that could inform the design of children's technology. My objective is to show that a systematic study of psychological theories of children's development, existing research results on children's cognitive development, existing results on children's use of technology, and existing design guidelines and usability principles will yield a useful, integrated framework to guide the design and evaluation of technology aimed at children aged five to eight.

In this introductory chapter I give the background and motivation for the study (section 1.2). I then discuss the specific research objectives (section 1.3) and delineate the study in section 1.4. In section 1.5 I explain how the research was conducted and section 1.6 gives a chapter-by-chapter overview of the dissertation.

1.2 Background and Rationale

1.2.1 Assumptions about Children and Technology

My study was conducted against the background of the following general assumptions:

1. Young children are increasingly spending time in front of computers or with other types of technology, and will continue to do so in the future. In the USA in 2003, 42% of children aged five to nine used the Internet [Hutchinson, Druin and Bederson, 2007]. Children are attracted to technology and, in general, want to 'play' with it. Attractive displays and computer novelty naturally captures children's interest [Bracken and Lombard, 2004]. My concern is to establish how designers of young children's software can maximize its benefits for cognitive development while making it fun to use.

2. Technology can potentially support the development of young children [Clements, 1987; Liang and Johnson, 1999; Papert, 1980]. Research on the effects of computer use on children's cognitive, social and physical development is ambiguous – both positive and negative effects have been reported [Subrahmanyam, Kraut, Greenfield and Gross, 2000]. I accept that there may be disadvantages to children's computer use, but I will not enter into the debate about whether computer use facilitates or constrains children's development. I acknowledge the problems with children's use of technology: that its compelling quality may keep children from playing outside, skipping and building sandcastles; that it fails to teach them the fine motor skills that knitting and carpentry do and that it may interfere with healthy communication between family members. On the other hand, it offers numerous possibilities for

improving thinking and other skills in a fun-filled way. With this research I embarked on a mission to find out how the general quality of technology aimed at young children can be improved, so that the time that they do spend with it is meaningful and to their benefit.

3. For some years to come children will still use software running on a desktop PC or a laptop computer. The majority of young children in the world has had no exposure to computers and there is little chance that they will ever experience more sophisticated technologies. There are a privileged few who will always have access to the latest robotic pet [Fujita, Kitano and Doi, 2000] or who live close to science centres where fantastic technological facilities offer them the best educational opportunities. Although these advanced technologies for children are very real and exciting, researchers should not forget that computer-based software are still being developed and used. My research is conducted in the context of a developing country, where, for the majority of young children, robotic pets and handheld computers are in the realm of science fiction. Attending to the design of computer-based software products is therefore just as important as the design of more advanced or modern technologies.

4. Cognitive and developmental psychology provides us with an extensive knowledge-base on children's development that can help designers to understand the young user. Although some designers of children's software (see for example, Carlson and White [1998], Masterman and Rogers [Masterman and Rogers, 2002] and Wyeth and Purchase [2003]) do refer to the work of developmental psychologists like Piaget and Vygotsky, I believe that software designers could utilize the results of a century of research on children's development to a much greater extent.

5. There is a huge body of research findings on young children's use of technology that – through thorough digestion, processing and analysis – can contribute significantly towards developing a framework for design.

1.2.2 Design Guidelines vs. Usability Testing

It should be clear from the outset that I do not believe that any product can be designed without user input and usability testing. No set of guidelines alone can guarantee design success and therefore user involvement in any design process is imperative. I do, however, trust that proper guidelines can reduce the required amount of usability testing. The two main reasons for wanting to reduce usability testing are that it is expensive and that it is difficult to perform usability experiments with young children.

1.2.2.1 The Cost of User Testing

A large variety of software products for entertainment and educational purposes aimed at children of all ages, is available for home and school use. Some of these products are excellent and achieve positive results with regard to child development [Druin and Solomon, 1996]. Many of the products, however, are not of a standard that best addresses the specific developmental needs and skill levels of their intended users [Scaife and Rogers, 2001]. Druin and Solomon [1996] have found that the most successful products are developed by large scale software development companies and that large teams were involved in these projects. Developers such as Microsoft and Sony Corporation can afford to support the development of children's technology with teams of professional usability engineers and child psychologists who can do comprehensive laboratory research involving children. Hanna, Risden and Alexander [1997] who are usability engineers at Microsoft, give useful guidelines for usability testing with children, but assume that the necessary usability testing facilities and manpower are available. Many products are developed within rich academic institutions where funding is not a problem. For example, the LEGO Mindstorms Robotics Invention System is researched, developed and maintained by the MIT Media Laboratory [Martin, Mikhak, Resnick, Silverman and Berg, 2000] and PETS (Personal Electronic Teller of Stories) is developed in the Institute for Advanced Computer Studies at the University of Maryland [Montemayor, Druin and Hendler, 2000]. Smaller software developers, academic or research institutions that want to develop technology for children but do not have the infrastructure or funds for extensive usability testing and cooperative enquiry [Druin, 1999], can benefit from a reliable set of guidelines that will aid them in producing products that are captivating, fun to use, age appropriate and supports cognitive development.

1.2.2.2 Young Children as Test Subjects

Another problem with usability testing with five to eight-year-old children, is that they are a complex user population to involve in the design process. At this age they are still inclined to say what they think adults would want to hear. Being observed, they may easily feel judged and their behaviour affected [Höysniemi, Hämäläinen and Turkki, 2004]. Young children find it difficult to verbalise their thoughts and traditional relationships between adults and children (child-parent, pupil-teacher) reinforce this problem [Montemayor et al., 2000]. Special skills and experience is required to communicate effectively with child users. As Kline [1993] says, it is easy to talk to children about their likes and dislikes, but it is difficult to read the real message behind what they tell us. Research with children requires researchers to be aware of their subjects' level of understanding, their knowledge and interests, and how their context may impact on the research.

A major factor when studying children is the authority adults typically have over children, which make it difficult for children be open and honest, especially if they do not agree with the adult view or if they think their opinion might be unacceptable [Greene and Hill, 2005]. Depending on their own context and the way they normally relate to adults, children may not believe that an adult would take their opinions seriously. On the other hand, researchers may be confronted with children who are very adept at evading, resisting or subverting adult authority [Greene and Hill, 2005].

MacFarlane, Sim and Horton [2005] found that when young children were asked to rate an educational software game using a child-friendly evaluation tool that requires them to choose between awful, not very good, good, really good and brilliant, they mostly chose the 'brilliant' option and there was little variation in their evaluations. Young children tend to be over-enthusiastic and cannot give a balanced opinion [MacFarlane et al., 2005].

1.2.3 Shortcomings of Existing Guidelines

1.2.3.1 Insufficient Distinction between Different Age Groups

Guidelines such as those proposed by Druin and Solomon [1996] and Malone [1982] do not focus on specific age groups. When designing software for children, designers should focus their design on a specific age group. During early childhood – due to rapid cognitive development during this stage – children of different ages have vastly different preferences and levels of skills [Grammenos and Stephanidis, 2002; Hoenderdos, Vermeeren, Bekker and Pierik, 2002]. My research will focus on software aimed at children aged five to eight. There are several reasons for focusing on this specific age group:

- This is a period of rapid growth in cognitive abilities.
- It is the age when children start their schooling and appropriate products can enhance their school readiness and support the acquisition of cognitive skills like reading, writing and story construction. Some evidence exist that children who have not reached their cognitive potential, benefit the most from using software that teaches literacy [Boone, Higgins, Notari and Stump, 1996]. In South Africa many preschool children do not attend nursery schools and are cared for by illiterate family members who cannot provide them with sufficient stimulation. They enter the formal schooling system with a handicap which may be overcome to some extent through the use of appropriate educational software.
- At this age their motor development is adequately developed to use input devices with ease.
- More research has been done on software for children of school going age than for preschool children, especially with regard to the influence of technology on cognition.

1.2.3.2 No Low Level Directing Principles Available

The research of Wyeth and Purchase [2003] led to a set of six design criteria that are based on theories of development and learning. Although these criteria can be useful during the initial planning stages of the design process, they do not provide practical guidelines for the actual implementation of computer-based activities. For example, one criterion is that construction activities that involve design, creation and evaluation processes should form the basis of interactions, but it gives no indication of what kind of construction activities would be appropriate. They do not tell us what kind of design or creation processes will engage a preschool child or whether a six-year-old boy will enjoy the same construction activity as a six-year-old girl. Malone's [1982] guidelines have similar shortcomings. Although they may be useful on a high level, they lack specificity that a designer will need when making design decisions. For example, he suggests designers use emotionally appealing fantasy, but gives no indication of what would be and emotionally appealing fantasy for a five-year-old girl?

1.2.3.3 Existing Guidelines are Aimed at Adult Products

Almost all existing guidelines for software design are aimed at products for adults where the emphasis is on improving work performance and productivity. Preece, Roger and Sharp's [2007] usability goals include effectiveness, efficiency and utility which are not necessarily goals of edutainment products for young children. Similarly, the way Dix, Finlay, Abowd and Beale [2004] discuss their usability principles of learnability, flexibility and robustness makes them naturally applicable to productivity enhancing products for adults. Making design principles such as these suitable for products aimed at young children will require adjustment in focus and sometimes reformulation of the principles. In Chapter 7 I discuss a selection of valued design guidelines and usability principles and explain whether and how they can be made applicable to young children's technology.

1.2.3.4 Insufficient Integration of Knowledge

There is a general lack of integration of knowledge from different theoretical fields or research disciplines. Most of the existing design guidelines that exist for young children's products have been derived from practical experience and empirical research. With this research I hope to draw together these guidelines with relevant knowledge contained in psychological theories of development, knowledge on how specific cognitive skills develop and knowledge on children's experiences with different kinds of technology. It would be an impediment to progress in the field of design field to ignore useful knowledge gained elsewhere.

1.3 Thesis Statement, Research Objectives and Research Questions

A thesis statement gives the overarching argument of a thesis and the process of writing a thesis or dissertation is all about supporting or testing this argument. My thesis statement is:
It is possible to develop a credible, dependable and useful set of guidelines for the design and evaluation of technology for children aged five to eight years by studying

- psychological theories of children's development,
- existing research results on children's cognitive development,
- existing results on children's use of technology and
- existing design guidelines and usability principles.

The purpose of my PhD research is to defend the above argument by going through the process of developing such a set of guidelines. The next nine chapters of this thesis provide a detailed report on every aspect of this process. To fulfil this purpose I have identified six objectives. I list them in Table 1.1 and associate each with a research question that will guide my study. After each question I indicate the chapter of the thesis where I will address the question.

Table 1.1 Research objectives and research questions

Objective	Associated research question
1.a To identify and study a representative sample of psychological theories of children's cognitive development to gain the knowledge to describe, in sufficient detail, the cognitive development of five to eight-year-old children with regard to skills relevant to the use of technology. 1.b To find, for each of the cognitive skills identified as relevant to the study, key sources in the literature that will provide information about the development of that skill in five to eight-year-old children and to study these to discover aspects that may be translated into guidelines for the design of technology.	What are the cognitive and developmental characteristics of typical five to eight-year-old children with regard to skills relevant to the use of technology? (Chapter 4)
2. To study the literature and research findings on young children's use of technology for skill development and where applicable translate these into design guidelines.	What can we learn from existing research into role of technology on skill development that can inform designers of technology for children aged five to eight? (Chapter 5)
3. To study the literature and research findings on interaction environments for young children and where applicable translate these into design guidelines.	What does the literature on interaction environments for young children tell us in terms of the design of technology for five to eight-year-old children? (Chapter 6)
4. To investigate the existing guidelines and principles for the design of technology and identify those that can be applied to young children's technology or that can be adapted for application to such technology.	What guidelines exist for the design of technology for children aged five to eight? Which existing guidelines not specifically aimed at the design of young children's technologies apply to technology for children aged five to eight? (Chapter 7)
5. To analyse all the potential guidelines discovered in sub goals 1 to 4 and organise them into a useful framework of guidelines for design.	How can the guidelines emerging from the literature be organised into a framework that is useful for designers? (Chapter 8)
6. To demonstrate the credibility and the practical usefulness of the proposed framework.	Is the proposed set of guidelines credible and useful? (Chapter 9)

1.4 Delineation of the Study

It would be presumptuous to claim that I could possibly, within the scope of a PhD study, come up with a complete, all-inclusive set of guidelines for the design of technology for young children. My study is therefore delimited as follows:

- The focus is only on technology aimed at children aged five to eight years. Since I emphasise the importance of age-appropriateness of technology the results are not intended for generalisation to other age groups. While focussing on a specific age group and trying to establish a user profile of five to eight-year-old computer users, I do acknowledge the variability of children's development. Each child is unique and has an individual growth and development pattern, personality, temperament, learning style and background [NAEYC, 1997]. I regard the child's age as a rough index of developmental status and recognise that children are not merely members of an age group that performs to a fixed norm. I strongly support the idea of adaptation to individual variation.

- My study of psychological theory is limited to the work of four prominent developmental psychologists (I provide a full justification of my choice of theories in Chapter 3).
- The developmental domains that I will focus 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. My definition of 'thought' as a domain of development includes memory, knowledge representation and problem solving. Interacting with technology necessarily involves these skills so it stands to reason that they should be included in my study.
- Every guideline included in the framework resulting from the study will be traceable back to one of the following:
  1. An aspect of one of the psychological theories studied.
  2. The results of a respected empirical study on the cognitive skills addressed in the study.
  3. The results of a respected empirical study on some aspect of children's use of technology.
  4. An existing, accepted guideline or principle for design.
- Although the framework offers a comprehensive set of guidelines I do not claim that it addresses every possible aspect of the design of young children's technology. There may, for example, be theories of development that I exclude from my study that may reveal guidelines that the theories included do not reveal. The framework will be an expandable tool that can be refined in the advent of further investigation, future research results and developments in the relevant fields.
- I do not study psychological theories of children with specific cognitive or physical disabilities, but in my analysis of research on children and technology I do include some discussion of technology that support children with disabilities.
- From a theoretical point of view I focus on *psychological* theories of development rather than *educational* theories of development. As it is impossible to separate these completely, my study does include some literature on the development of specific cognitive skills based on research in the field of Educational psychology. I do, however, not attempt to cover the related work from this discipline.
- This is a broad study covering many aspects of the design of technology rather than an in depth study of one aspect of it. There are many researchers who focus on specific aspects but very few have tried to provide a thorough survey of all the possible technologies and interaction environments that are suitable for children.

1.5 The Research Process

The research design and methodology is discussed in detail in Chapter 3. Here I give a brief overview of my approach. Following TerreBlanche and Durrheim [1999], my research can be viewed as descriptive, applied and qualitative. It is descriptive in the sense that I will give a narrative-like description of phenomena – for example, the developmental characteristics of children aged five to eight that may have some bearing on their relationship with technology. The research is applied, since the results of the research will assist people with problem-solving and decision-making in the context of designing technology for young children. I use qualitative methods because the data is in the form of written language that will be analysed by identifying elements, themes or patterns that may play a part in achieving the research objective.

The research was conducted in four phases:

- Phase 1 – Study and Analysis of Psychological Theories.
  In this phase I investigate four prominent theories of cognitive development and some empirical studies from cognitive and developmental psychology extracting design-relevant factors to guide the formulation of guidelines for designing technology aimed at young children (Chapter 4).
- Phase 2 – Analysis of Existing Literature on Young Children and Technology.
  Here I analyse existing literature on young children and technology with the aim of extracting potential guidelines for the design of technology (Chapters 5 and 6) and critically investigate existing guidelines for the design and evaluation of children's technology (Chapter 7). I also evaluate the applicability of existing usability principles and design guidelines to technology aimed at children (Chapter 7).
- Phase 3 – Organising the Guidelines into a Framework.
  This phase involves construction of a framework that organises and integrates the design-relevant data emerging from phases 1 and 2 (Chapter 8). Since these address a huge range of design aspects (including content, interface elements, interaction processes, educational practice and game modules) designing this framework requires thorough inspection and intense commitment to ensure its credibility and usefulness.
- Phase 4 – Validation of the Proposed Guidelines.
  In this phase I evaluate the credibility, dependability and usefulness of the proposed guidelines by demonstrating how they can be used to evaluate existing software and to design prototypes of a selection of computer-based activities.

1.6 Structure of the Dissertation

The rest of this dissertation is organised as follows:

Chapter 2 contains a description of the theoretical framework of the study and discusses key concepts and theories relating to the following main topics:

- Human-computer interaction and interaction design.
- Guidelines for the design and evaluation of technology.
- Young children and technology.
- Cognitive and developmental psychology.

Chapter 3 discusses the research design and methodology. Chapter 4 reports on an investigation of four theories of cognitive and developmental psychology and explains how this knowledge can be applied to the design of technology for children aged from five to eight. In Chapter 5 I probe the literature on young children and technology for information which can be translated into guidelines for the design of technology. In Chapter 6 I investigate existing interaction environments for young children. Chapter 7 gives a critical examination of the guidelines that exist for the design of technology for children and considers other existing usability and design guidelines to asses their applicability to children's technology. Throughout Chapters 4 to 7 the emerging design-related factors and potential guidelines for design are clearly highlighted and labelled in data boxes.

In Chapter 8 involves the analysis and integration of the emerging data. Here I create a framework for organising the guidelines identified in Chapters 4 to 7 and present the framework in an accessible and functional manner. Chapter 9 is devoted to proof-of-concept activities. I describe how the proposed guidelines can be used to evaluate an existing software application and to re-design aspects of the software that were found lacking.

Chapter 10 concludes the thesis with a summary of the research, a reflection on the practical and scientific value of the results and a summary of contributions. It ends with suggestions for possible refinements of the proposed guidelines and recommendations for related future research.

1.7 Conclusion

Designers should not rely on their intuition or memories of their own childhood when designing for children. When their target audience is eight or younger, 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 thesis reports on a study that shows that designers can learn about children from experts such as developmental psychologists, education specialists and researchers experienced in working with children. Ploughing through thousands of pages written by such authorities, many guidelines for the design of technology for young children have been identified and are presented here. My contribution is twofold: firstly I demonstrate a process of developing guidelines from existing theory and research and, secondly, present a useful, integrated framework of guidelines for the design or evaluation of technology for children aged five to eight.

CHAPTER 2

Conceptual Framework

Table of Contents

2.1 Introduction

To reach the objective of this thesis my task is to formulate a set of guidelines for the design of technology for children aged five to eight years. To do this, I will investigate psychological theories, research results on children's cognitive development, results on children's use of technology and existing design guidelines and usability principles. It is clear then that this is a multidisciplinary study that involves child development, cognitive psychology, child-computer interaction and the design and evaluation of interactive systems. In this chapter I describe the broad theoretical context of my study by first organising the various disciplines, sub-disciplines and knowledge constructs into a graphical conceptual model and then giving a narrative discussion of the components, their theoretical bases and the links between them.

According to Miles and Huberman [1994] the purpose of a conceptual framework is to explain the main things to be studied and the relationships between them. Such a framework can be rudimentary or elaborate, theory-driven or commonsensical, descriptive or causal. The framework I have constructed for my research is made up of the theoretical fields and subfields that form the context of the study, the specific categories of knowledge from those fields that will be studied, and the intended result of the study. It indicates how information flows between these categories of knowledge to lead to the outcome in the form of guidelines for the design of children's technology.

Using my framework as guide, I will introduce the main theories and define the key concepts around which the study was built. The chapter is organised as follows:

- In section 2.2 I present my conceptual framework.
- Section 2.3 introduces the fields of human-computer interaction, including the subfields child-computer interaction, interaction design and interface design.
- In section 2.4 I give a brief introduction of developmental psychology looking in particular at cognitive development and the relationship between cognitive development and technology.
- I conclude the chapter in section 2.5.

2.2 The Conceptual Framework

The two theoretical fields within which my study lies are *human-computer interaction*[1] *(HCI)* and *psychology*. Although the study of psychological theory is pivotal, the intended outcome of the research is to make a contribution to the field of HCI rather than to psychological theory. The subfields that delimit the study are *child-computer interaction* and children's *cognitive and developmental psychology*. Two classes of knowledge from child-computer interaction are relevant to the study, namely *knowledge about technology for children aged five to eight* and *knowledge about existing guidelines for children's technology*. Together

[1] Terms given in italics appear as components in the graphical representation of the conceptual framework (Figure 2.1).

with existing general design guidelines, these two classes of knowledge provide direct input for the formulation of *guidelines for the design of young children's technology*. My interest in developmental psychology relates to *theories of cognitive development* and specifically what these theories say about cognitive skill development of children aged five to eight. To formulate the intended guidelines, *knowledge about cognitive skills* that can influence, or be influenced by, children's use of technology is essential. Figure 2.1 illustrates how these knowledge components fit into my research plan, how they relate to each other and to the intended outcome of my study.

As explained in Chapter 1, the first three phases of the study is devoted to the study of the relevant literature and the consequent formulation of the emerging design guidelines, and the fourth phase to the validation or evaluation of the results. Since the theoretical underpinnings of the evaluation phase relate mainly to research methodology issues, I leave discussion of the evaluation phase and its theoretical grounding to Chapter 3. I now continue in this chapter to define or describe the theories, concepts and classes of knowledge that form the conceptual framework for phases 1, 2 and 3.

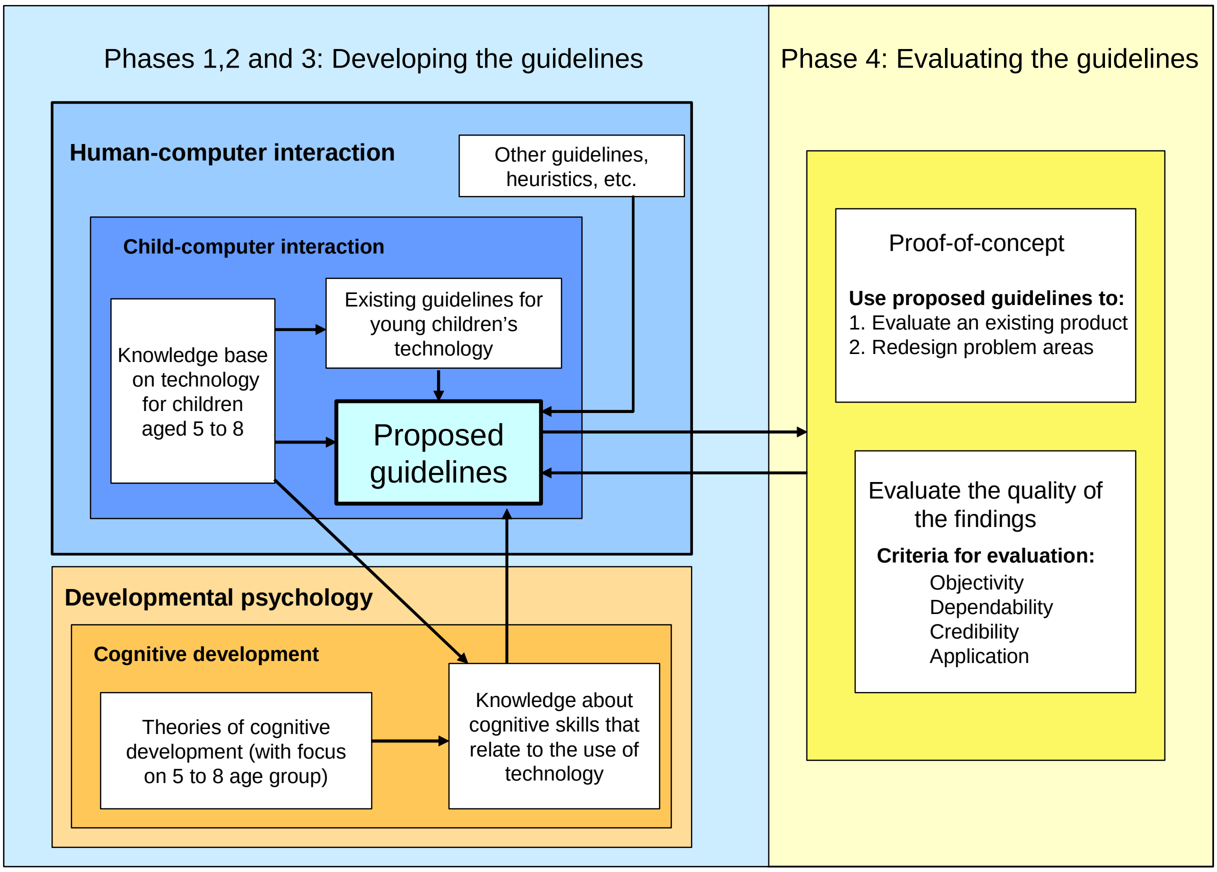


Figure 2.1 Conceptual framework for this study

2.3 Human-Computer Interaction

2.3.1 Introducing Human-Computer Interaction

Computers and computer software are there for people to use. They should therefore be designed in a way that allows the intended user to use them successfully for the intended purpose and with the least amount of effort. To design a successful system the designers must know how to support the tasks that the user will perform with it. They must understand why the users need the system, what tasks they will want to perform with the system, what knowledge they might have (or lack) that may influence their interaction with the system, how the system fits into the user's existing context, and so forth.

The term *human-computer interaction* was adopted in the mid-1980s to denote a new field of study concerned with studying and improving the effectiveness and efficiency of computer use [Kotzé and Johnson, 2004]. Human-computer interaction is concerned with the design, implementation and evaluation of interactive systems, with specific consideration of what a user needs to accomplish when using the system [Dix et al., 2004]. For Dix et al. the *user* is an individual user, a group of users working together, or a sequence of users who respectively deal with different parts of the process. The *computer* refers to any technology ranging from a desktop computer to a process control system or an embedded system. *Interaction* refers to the communication between the user and the computer in the process of using the system to perform a task. For Kotzé and Johnson [2004] the purpose of studying human-computer interaction is to improve the quality of interaction between human and machine by systematically applying knowledge about human capabilities and limitations, and machine capabilities an limitations; also, to improve the productivity, functionality, effectiveness, efficiency, and usability of technology. All this applies directly to my research, the implicit aim of which is to improve the quality of interaction between children and technology.

Human-computer interaction (HCI) is a multi-disciplinary subject with computer science, psychology and cognitive science at its core [Dix et al., 2004]. When HCI became one of the domains of cognitive science research in the 1970s, the idea was to apply cognitive science methods to software development [Carroll, 2003]. General principles of perception, motor activity, problem solving, language and communication were viewed as sources that could guide design. Although HCI has now expanded into a much broader field of study, it is still true that knowledge of cognitive psychology can help designers to understand the capabilities and limitations of the intended users. Human perception, information processing, memory and problem solving are some of the concepts from cognitive psychology that are related to people's use of computers [Dix et al., 2004]. (I return to cognitive psychology and its role in human-computer interaction in section 2.4.1 below.)

The remainder of this section on HCI is devoted to the specific aspects and domains of HCI that relate to my research, namely interaction design, design guidelines, evaluation of technology and child-computer interaction.

2.3.2 Interaction Design

Preece et al. [2007] define interaction design as ‘designing interactive products to support the way people communicate and interact in their everyday and working lives’ (p. 8). The focus is on how to design user experiences using a variety of methods. Preece et al. regard interaction design as a broader discipline than HCI since it deals with the theory, research and practice of design of much more than just computing systems. They identify the following basic activities through which the process of interaction design iterates:

1. Identifying needs and establishing requirements.
2. Developing alternative designs that meet those requirements.
3. Building interactive versions of the designs so that they can be communicated and assessed.
4. Evaluating what is being built throughout the process.

Interaction design places emphasis on user participation in the design process, but according to Preece et al. [2007] it is equally important to understand how people act and interact with one another, with information and with technology; and their abilities, emotions, needs and interests.

One of the main objectives of interaction design is to create usable products. *Usability* is therefore a key concept in interface design. According to Preece et al. [2007] usability generally refers to the ease with which a system can be learnt, how effective it is to use and how enjoyable it is from the user’s perspective. Improving the usability of a system involves optimising people’s interactions with it to help them perform their activities at work, school and in their everyday lives. Preece et al. [2007] break usability down into six different goals: effectiveness, efficiency, safety, utility, learnability, and memorability. (I return to these in Chapter 7 where I investigate the relevance of these goals to children’s products.) In addition, Preece et al. discuss *user experience goals* that address the quality of the user experience. These include the goals of making a product satisfying, enjoyable, engaging, exciting, entertaining, aesthetically pleasing, supportive of creativity, emotionally fulfilling, and so on. All usability goals do not apply to all kinds of interactive product. For example, a word processor should be usable but need not be entertaining or fun. In children’s products the user experience goals are particularly important and more generally applicable. There is, however, always a relationship between usability and user experience and it is the interaction designer’s responsibility to determine which user experience goals will contribute to the usability [Preece et al., 2007].

2.3.3 Interface Design

Interaction design includes interface design. Preece et al. [2007] give an overview fourteen types of interfaces. Following Preece et al, sections 2.3.3.1 to 2.3.3.9 provide a brief overview of those interfaces that are relevant to young children’s products.

2.3.3.1 Advanced Graphical Interfaces

Advanced graphical interfaces involve interactive animations, multimedia, virtual environments, and visualisations. Multimedia includes graphics, text, video, sound and animations that the user can interact with. It supports quick access to multiple representations of information and is well suited for training, education and entertainment. A problem with multimedia is that users tend to favour animations and video clips and easily ignore accompanying text and static diagrams.

Virtual reality and virtual environments are graphical simulations that create the illusion that the user is part of the environment. It gives user the experience of operating in 3D environments in ways that are not possible in the real world. Virtual objects can appear very true to life. Users in a virtual environment have a first-person perspective where they see the environment through their own eyes, or a third-person perspective where they see the environment through the eyes of an avatar[2].

2.3.3.2 Web-Based Interfaces

Web design is restricted by download time. Although in first-world countries high bandwidth is available to most people, large numbers of internet users in developing countries do not have fast internet access. Nowadays web sites can have most of the characteristics of advanced graphical interfaces, but uncluttered design and easy accessibility of the required information are still preferable to web pages filled with flashing advertisements and lots of graphics and animations. Users should always know where they are, what they can find there and where they can go next. Web design relies heavily on the use of text. When designing for young children who cannot yet read this will be a problem.

2.3.3.3 Speech Interfaces

A speech interface allows the user to talk to a system that has the capacity to interpret spoken language. It is commonly used in systems that provide specific information (e.g. flight times) or perform a specific transaction (e.g. buy a movie ticket). Technology such as web readers and speech operated home control systems (e.g. for switching appliances on and off) can be especially helpful to people with disabilities. Current technology allows for much more natural sounding speech than the early synthesized speech. Speech interfaces in applications for children who cannot yet read will expand the possibilities that technology can offer them.

2.3.3.4 Pen, Gesture and Touchscreen Interfaces

Personal digital assistants (PDAs) come with a pen for making on-screen selections, or to write or sketch freehand. Objects can also be manipulated through swiping or stroking gestures. Pen-based interfaces are also suitable for large displays. Through a process called ‘digital ink’ that uses sophisticated handwriting recognition and conversion techniques, text written on a PDA screen or tablet PC, for example, can be converted into text.

---

[2] Artifical representations of real people [Dix et al., 2004].

Gesture-based input involves camera capture and computer vision to detect people's arm and hand gestures. This makes sign language interpreting systems possible. The latest systems use sensor technologies to detect touch, bend and speed of movement.

Touchscreens allow users to manipulate screen objects with their fingers. Two hands can, for example, be used to stretch an object in two different directions at the same time.

2.3.3.5 Multimodal Interfaces

In multimodal interfaces, different ways of interacting – including touch, sight, sound and speech – are combined so that users can experience or control information in multiple different ways. Different input or output methods are used simultaneously, for example speech and gesture, or eye-gaze and gesture. Speech and touch combinations are already being used, but otherwise multimodal interfaces are not commercially available yet.

2.3.3.6 Shareable Interfaces

These interfaces allow more than one user, providing multiple (sometimes simultaneous) inputs. Tabletop environments already exist that detects touch input from multiple users at the same time. They use and array of embedded antennae that each transmits a unique signal. The users each sit on their own chair or mat which has a receiver installed. Through the user's body, a signal goes from the tabletop to the receiver that tells the computer which antenna was touched.

2.3.3.7 Tangible Interfaces

These interfaces use sensor-based interaction. Physical objects that contain sensors (typically RFID tags [3]) react to user input which can be in the form of speech, touch or manipulation of the object. The effect can take place in the physical object (e.g. a toy that reacts to a child's spoken commands) or in some other place (e.g. on a computer screen). Tangible interfaces have been used for urban planning and storytelling technologies, and are generally good for learning, design and collaboration.

2.3.3.8 Augmented and Mixed Reality Interfaces

In an augmented reality interface virtual representations are superimposed on physical devices and objects, while in a mixed reality environment views of the real world are combined with views of a virtual environment. Mixed reality systems have been used for medical applications, where, for example, a scanned image of organs or an unborn baby is projected onto the body of the patient to help doctors to 'see' what goes on inside the body.

[3] Radio Frequency Identification tags can be stickers, cards or disks that can be used to store and retrieve data through a wireless connection with a RFID transceiver [Preece et al., 2007].

2.3.3.9 Robotic Interfaces

These are interfaces that enable users to move and steer a remote robot. Domestic robots can be manipulated to help in the house. This is especially useful for the disabled. Pet-like robots have been developed to host events or act as companion. They contain embedded sensors that detect user behaviours and respond to them.

When I discuss existing research on young children's experiences with technology all of the above interface types will come under discussion again. I conclude the discussion of interface design with an explanation of how it fits into the overall design process.

2.3.3.10 Interface Design in the Software Development Process

The activities included in the classical waterfall model of software design are requirements elicitation, high-level specification, detailed design, coding, testing and maintenance [Kotzé and Johnson, 2004]. It did not regard interface design as a core activity of the design and development process.

Williges and Williges [1984] produced a classic model of software development whereby interface design drives the overall design process. A graphical representation of their model appears in Figure 2.2. Their standpoint is that by identifying user requirements early in the software development process, code generation and modification effort will be reduced [Kotzé and Johnson, 2004].

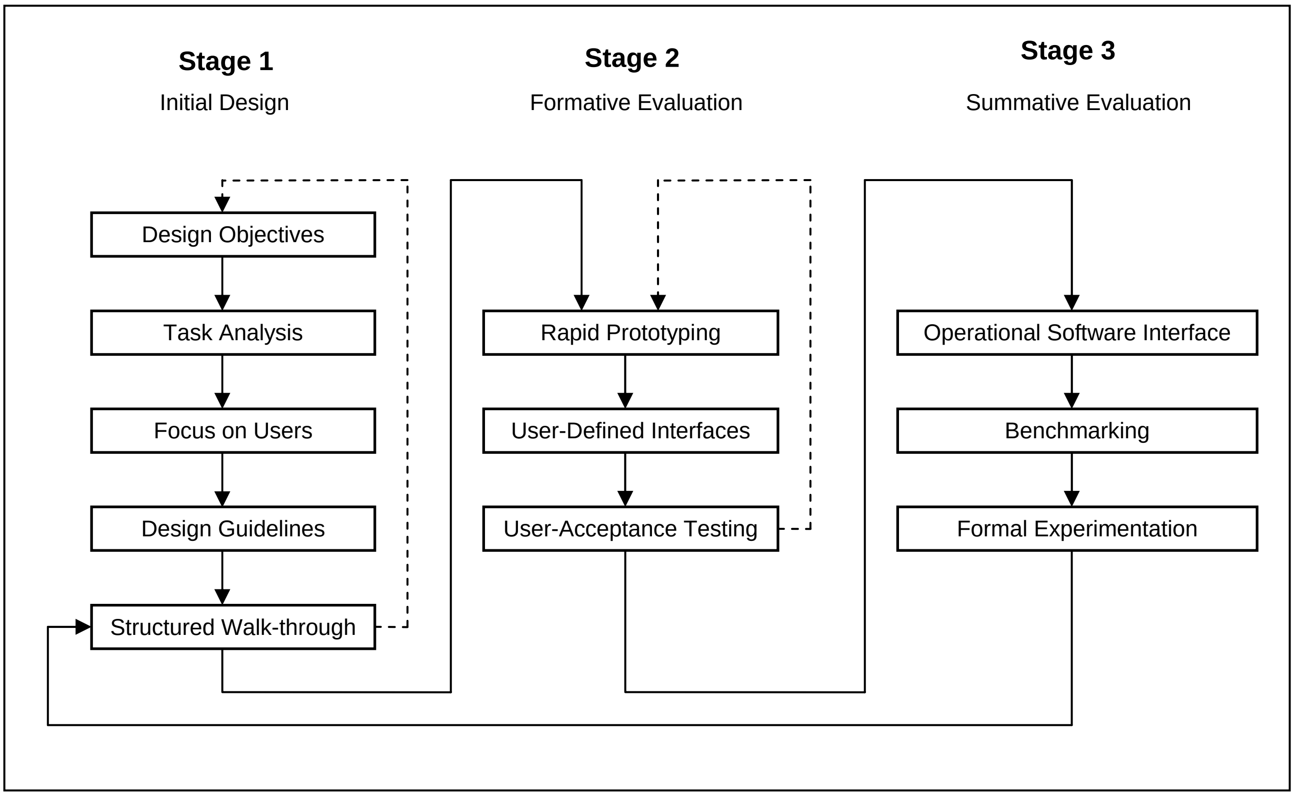


Figure 2.2 HCI Life Cycle [Williges and Williges, 1984]

In this model, design guidelines and evaluation are important elements of the design and development process. I discuss these in the next sections.

2.3.4 Design Guidelines for Interactive Systems

The aim of design guidelines, standards and design principles is to help designers to improve the usability of their product by giving them rules according to which they can make design decisions [Dix et al., 2004]. They restrict the range of design options and prevent the designer from making choices which are likely to harm the usability of the product. Dix et al. classify design rules as standards or guidelines. Standards are usually set by national or international bodies, are high in authority and limited in application, while guidelines are more general in application. Dix et al. believe that designers should understand the theory underlying a design guideline to be able to apply it sensibly.

Kotzé and Johnson [2004] distinguish between two types of design guidelines: low-level detailed rules and high-level directing principles. High-level principles are relatively abstract and applicable to different systems, while design rules are detailed instructions that are application-specific and do not need much interpretation. Examples of design principles are Dix et al.'s principles of observability (e.g. users must be able to observe the effects of their actions) and predictability (e.g. users should be able to predict the effects of planned actions based on the information displayed to them). Examples of design principles are Shneiderman's [1998] eight golden rules, Dix, et al.'s [2004] principles to support usability and Preece, et al.'s [2007] usability and user experience goals. (All of these will be discussed in detail in Chapter 7 when I evaluate existing guidelines for applicability to children's technology.)

It is important to realise that design guidelines do not provide recipes for designing successful systems. They can only provide guidance and do not guarantee maximum usability. Even when armed with very good guidelines, a designer should still make an effort to understand the technology involved, the relevant psychological characteristics of the intended users and what usability means in the context of the particular product [Kotzé and Johnson, 2004].

The difference between design and usability principles are that design principles usually informs the design of a system, while usability principles are mostly used as the basis for evaluating prototypes and complete systems [Preece et al., 2007]. Usability principles can be more prescriptive than design principles. When used in practice, design or usability principles are often referred to as heuristics [Preece et al., 2007]. Nielsen's [2001] ten usability principles are probably the best known heuristics for evaluating interactive systems. (I discuss them in detail in Chapter 7.)

The set of guidelines that will be an outcome of my research will consist of a combination of design guidelines, design principles and usability principles.

2.3.5 Evaluation of Interactive Systems

Evaluation is a key aspect of human-computer interaction (and of interaction design in particular) that refers to the validation of an interactive system against human-computer interaction requirements [Dix et al., 2004]. Any design needs to be assessed and any system needs to be tested to ensure that they meet the users' requirements. Evaluation is not a single phase that comes at the end of the design process, but rather an activity that is used throughout the design process to provide feedback on the design right from the beginning. The model of Williges and Williges [1984] (see Figure 2.2) distinguishes between formative and summative evaluation. Formative evaluation is done early in the design process and continues through the design cycle to support design decisions [Dix et al., 2004]. Low cost techniques such as pen and paper prototypes or the use of prototyping tools are suitable for formative evaluation. Early evaluation helps to predict the usability of a product and assesses the designer's understanding of the user requirements. Summative evaluation is done at the end of the design cycle and tests the end product [Dix et al., 2004]. Its aim is to demonstrate that the completed system fulfils its requirements or to identify problems users have with the system. Usability testing with real users is suitable for summative evaluation.

For Dix et al. [2004] the three main goals of evaluation are to assess the extent of the system's functionality, to assess the effect of the interface on the user and to identify specific problems with the system. Evaluation can be done in laboratories or in the real-life environment where the system will be used. Usability laboratories with sophisticated audio and video recording facilities, specialised hardware and software for recording and analysing users' behaviour when using a system, are often used for usability testing. Such a laboratory setting gives the evaluator control over the conditions of the study, but it removes the natural context (with associated interferences) which may be important in the use of the system. Evaluation done in the real environment of use provides the natural context of use but it may be more difficult to set up equipment required and subjects can still be influenced by the presence of researchers in their working environment. Ultimately, Dix et al. believe that there are circumstances where laboratory testing will be necessary and that the specific system and user population will determine what the balance between the two approaches should be.

Preece et al. [2007] identified three main evaluation approaches which I now discuss in brief.

1. Usability testing: With usability testing typical users perform selected tasks, usually in a controlled laboratory setting where they are observed and their actions recorded. The evaluator analyses the data collected to judge performance identify errors and explain user behaviour. Such experiments are usually supplemented with interviews and satisfaction questionnaires.
2. Field studies: This type of evaluation is done in natural settings. The aim is to understand what users do naturally and how the technology affects them in the real-life environment. The evaluator can be an outsider that observes and records what is happening, or an insider or participant that enters the world of the user to experience the impact of the technology first-hand.
3. Analytical evaluation: This either a heuristic evaluation, that involves experts who use heuristics and their knowledge of typical users to predict usability problems, or walkthroughs where experts 'walk

through' typical tasks. The users need not be present and prototypes can be used in the evaluation. Popular heuristics such as that of Nielsen [2001] were designed for screen-based applications and are inappropriate for technologies such as mobiles and computerised toys.

The last of these, namely analytic evaluation, relates to my study as the resulting guidelines will be suitable for heuristic evaluation of prototypes of systems aimed at young children. Evaluation of children's technology has its own specific challenges and problems and the complexities of working with young children can make methods such as usability testing and field studies less appealing.

2.3.6 Child-Computer Interaction

Historically, computers and computer applications have been designed for use by adults for assisting them in their work. In many accepted definitions of human-computer interaction and interaction design, there is a hidden assumption that the users are adults. There are, for example, references to users' 'everyday working lives' [Preece, Rogers and Sharp, 2002] or the 'organization' they belong to [Dix et al., 2004]. Children, however, make up a substantial part of the larger user population. Whereas products for adult users usually aim to improve productivity and enhance performance, the purpose of children's products is more likely to provide entertainment or engaging educational experiences. Applications designed for use by children in learning environments have completely different goals and contexts of use than applications for adults in a work environment [Inkpen, 1997]. While adults' main reason for using technology is to improve productivity, children do it for enjoyment. Another reason for distinguishing between adult and child products is children's slower information processing skills that affect their motor skills and consequently their use of the mouse and other input devices [Hutchinson et al., 2007].

Child-computer interaction has emerged in recent years as a special research field in human-computer interaction. This is manifested in the annual Interaction Design and Children conference that was held for the first time in 2002. Not long ago it was only developmental psychologists, educationists, and market researchers that were interested in children as users of interactive technology [Bekker, Markopoulos and Kersten-Tsikalkina, 2002]. Today numerous researchers in the field of human-computer interaction are focussing their attention on design of children's technology. Many governments support research on children's technology, based on the general acknowledgement of the need to prepare children for the inevitable presence and increasing sophistication of technology in their lives [Plowman and Stephen, 2003].

The term 'computer' in child-computer interaction, or 'children's technology' refer not only to the ordinary desktop computer, but also to programmable toys, cellular phones, remote controls, programmable musical keyboards, and more [Plowman and Stephen, 2003]. In Chapter 5 and 6 I survey the existing research on old and new technologies for young children and discuss children's actual experiences with these technologies.

2.3.7 Young Children and Technology

Computers will never replace important play and learning material such as paint, blocks, sand, water and books [NAEYC, 1996], but technology does provide new and exciting opportunities for childhood activities. If used appropriately, technology can enhance children's cognitive and social abilities. Unlike many other educational materials, computers are intrinsically compelling for children, with the sound and graphics capabilities helping to keep their attention [NAEYC, 1996]. Research has shown that the extent of interactivity involved when using media, may affect the learning process [Wartella and Jennings, 2000]. Computer activities are potentially highly interactive and can thus provide learning experiences that are rich in participation, responsiveness and engagement. This is promoted by the fact that children have varying degrees of control over the context of the exchange.

Computer technology makes it possible for children to easily apply concepts in a variety of contexts [Roschelle, Pea, Hoadley and Gordin, 2000]. It exposes them to activities and knowledge that would not be possible without computers. For example, a young child who cannot yet play a musical instrument can use software to compose music. Roschelle et al. report on eighteen major studies that investigated the effectiveness of computers as a learning tool. The general finding is that the use of computer aided instruction or, in some cases, ordinary applications such as word processors, improve achievement in the following areas of development or cognition: 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. Several of these studies emphasised that the gains in proficiency depend on the quality of the learning material. Liang and Johnson [1999] also emphasise the importance of the quality of the software for young children, especially with regard to problem-solving orientation, developmental appropriateness, playfulness and incorporating new technologies.

People opposed to the use of computers by young children have warned against some potential dangers. These include 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 [Haugland and Wright, 1997]. 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 [Plowman and Stephen, 2003].

Research has refuted the earlier belief that children can only use computers in an appropriate way when they have reached the stage of concrete operations in Piagetian terms, that is, around the age of seven [Clements, 1987]. The fact that computer use requires symbolic reasoning was also regarded as a problem that Clements [1987] played down with the argument that much of young children's behaviour is symbolic. Another argument against early computer use is that children are being 'rushed'. Clements 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. Clements [2002] define developmental appropriateness as follows: ‘developmentally appropriate 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 [1997] 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.

Computers also do not prevent children from engaging in pretend play. Children have been observed humanising lines constructed with a drawing program (for example, exclaiming that the line is sleeping or has woken up) [Clements, 1987]. Another pretended the cursor was a termite eating wood while he was erasing something. These examples illustrate that technology, and even interface elements, can become the objects of fantasy play. In Chapters 5 and 6 I elaborate on technology’s expanding role in creative and collaborative play that involve more than what happens on a computer screen.

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 [1996] we should use this understanding to improve technology so that it supports children optimally. The development of any technology can only be successfully if the designers truly understand the target user group. Knowledge of children’s developmental and familiarity with the theories of children’s cognitive development is thus essential when designing for them. The way children learn and play, the movies and television programmes they watch, the way they make friends and communicate with others are all influenced by the presence of computer technology, be it visibly or hidden, in their everyday lives. For this reason Druin [1996] believes it is critical that designers of future technology observe and involve children in their work.

2.4 Developmental Psychology

Developmental psychology studies humans as they grow from a fertilized cell into an adult and from there to old age [Louw, Van Ede and Louw, 1998]. It describes developmental changes and aims to explain the different factors that influence these changes. The aspects of development that interest developmental psychologists are changes that are relatively permanent, links and interaction between different kinds of developmental changes and developmental patterns [Louw et al., 1998]. Different areas of development are distinguished, each of which can be regarded as a field of study in its own right, but all connected in some way. Berk [2000] describe the following domains of child development:

- Physical development that involves changes in body size, appearance, functioning of body systems and perceptual and motor skills.
- Cognitive development that includes development of thought processes and intellectual abilities such as attention, memory, knowledge representation, problem solving, imagination, creativity and language.
- Emotional and social development that has to do with emotional communication, self understanding, ability to manage feelings and moral reasoning and behaviour.

Although all of these can be related in some way to the use of technology, the focus in my study will be the relationship between cognitive development and child-computer interaction. Where specific aspects of physical, social or emotional development are relevant, I will consider them.

Theories of child development differ with regard to three basic issues, namely whether development happens in stages or as a continuous process; whether there is one or many courses of development; and to what extent development is determined by biological factors versus external factors [Berk, 2000]. When theorists see development as a smooth and continuous process, they believe that children gradually improve the skills that they already have. Theorists who follow the stage view believe that children change dramatically when they move from one stage to the next but develop very little in a stage [Berk, 2000]. The stage view usually assumes that all children follow the same sequence of development. Most contemporary theorists, however, regard context as an important factor in determining the course of development. To them aspects such as the home environment, school, historical context and culture all contribute to diversity in development [Berk, 2000]. Theories also differ in the extent to which they assign development to inborn biological factors or to the influence of the environment [Berk, 2000]. Some theorists believe in stability – if a child is good or bad at some ability early on he or she will remain so later on – while others feel change is possible if new experiences support it. In Table 2.1 I summarise how the theoretical approaches relevant to this thesis deal with the above issues.

Table 2.1 Stances of four theoretical approaches (adapted from Berk[2000]).

Theory	**Continuous or stage-wise development**	**One course of development or many**	**Nature or nurture**
Piaget's cognitive development theory	Development takes place in stages.	One universal course of development.	Nature and nurture: development depends on brain maturation as well as on a stimulating environment. Early and later experiences are important.
Information processing	Continuous: perception, attention, memory and problem-solving improves gradually.	One course of development.	Nature and nurture: development depends on brain maturation as well as on a stimulating environment. Early and later experiences are important.
Vygotsky's cultural theory	Combination of both: language acquisition and schooling cause stage-wise development while social and cultural experiences support continuous changes.	Many possible courses depending on social and cultural context.	Nature and nurture: heredity, brain maturation and interaction with more knowledgeable or skilled people all contribute to development. Early and later experiences are important.
Dynamic systems perspective	Combination of both: change is always ongoing but children reorganise their behaviour in stages.	Many possible courses: biological factors, environment and social experiences lead to individual differences in development path.	Both nature and nurture: mind, body, physical and social environment form and integrated system that guides development. Early and later experiences are important.

The remainder of this section is devoted to a synopsis of cognitive development as a field of research.

2.4.1 Cognitive Development

Cognitive psychology studies cognitive processes such as perception, attention, language processing, knowledge representation, reasoning, learning, creating, problem solving and memory. Cognitive development is concerned with how these processes and skills are developed and refined [Louw et al., 1998]. A defining characteristic of humans is their ability to use their mental processes to adapt to, or change, their environment [Berk, 2000]. Thus, successful development of cognitive skills will improve their normal functioning and help towards their survival. According to Berk [2000], research on cognitive development are concerned with the typical course of development, individual differences in cognitive development and mechanisms of cognitive development.

When cognitive development of children was first studied in the beginning of the twentieth century it was done through intelligence tests such as the Stanford Binet Intelligence Quotient (IQ) test. Psychologist Lewis Terman adapted this test from a 1905 French test for use in the United States [Gale, 2001]. IQ tests use the concept of 'mental age'. The IQ score of an average child matches the score associated with his or her age, while the score of a gifted child matches the mental age of an older child. IQ tests have been widely criticised for their narrow definition of intelligence and for being biased with regard to social and cultural issues [Gale, 2001].

Whereas IQ tests focussed on people's inborn abilities, the work of behaviourist psychologists such as Watson and Skinner lead to the development of learning theory that emphasised the influence of environmental factors on intelligence [Gale, 2001]. They believed children learn by having certain behaviours rewarded and others discouraged and consequently that children do not have an active role in their own development.

Piaget, on the other hand, saw children as active participants in their own development. Through decades of observation of children in their natural environments he provided the most influential theory of cognitive development [Berk, 2000]. According to his theory, cognitive development occurs in four distinct, universal stages. Piaget regarded the development of general cognitive structures as a necessary prerequisite to learning [Fischer and Immordino-Yang, 2002]. His structural theory represents a monolithic view of development where cognition depends on universal logical structures and learning happens from within the individual. I discuss Piaget's theory in detail in Chapter 4.

According to the Gale Encyclopaedia of Psychology [Gale, 2001] the information processing approach offers, after Piaget, the most significant account of cognitive development. This theory views the mind as a computer-like symbol manipulating system that encodes information from the environment and store it in symbolic form [Berk, 2000]. Through internal processes the representation of this information can be revised or it can be interpreted with consideration of other information in the system [Berk, 2000]. Case's neo-Piagetian theory is one of the most prominent information processing theories. I discuss it fully in Chapter 4.

An important current theoretical approach is the dynamic systems perspective which sees the child's mind, body and physical and social worlds as a dynamic integrated system [Berk, 2000]. Change in any part of the system leads to reorganisation of the whole system into a more complex, but effective one. Fischer is currently a key figure in this theoretical approach and I come back to his theory in Chapter 4.

The reason for the brevity of this introduction to cognitive psychology is to avoid repetition. In Chapter 4, where I investigate the theories of cognitive development selected for my research, I will provide detailed discussions of the aspects introduced here. In Chapter 3, where I explain the research methodology I will justify my choice of theories in detail.

2.4.2 Cognitive Development and Technology

The relationship between human-computer interaction and cognitive psychology is eminent [Dix et al., 2004; Preece et al., 2007; Shneiderman, 1998]. Using technology requires perception, attention, memory, information processing, decision making, and more. This is true for adult and child users. Relevant to my study is the link between child-computer interaction and cognitive development. There are computer applications with the purpose of teaching adults specific skills, such as flying an aeroplane, but usually, when adults use technology, it is to perform a task and not to develop a skill. With children, especially young children, the situation is different. Any interaction young children experience may have an influence on their development. How this happens or whether it is good or bad depends on the characteristics of the interaction. Interaction with computers may improve children's cognitive skills but it can also deprive them of other kinds of interaction that may be more beneficial.

2.5 Conclusion

The aim of this chapter was to describe the conceptual framework that forms the context for my research. In doing this I introduced the theoretical fields of human-computer interaction, child-computer interaction, and children's cognitive development. The scene has now been set for the core of the research that I will present in Chapters 4 to 8 – an extensive literature study the consequent formulation of design guidelines from existing theory and research results. Before continuing with that, I provide a complete description of the research design and methodology in Chapter 3.

CHAPTER 3

Research Design and Methodology

Table of Contents

3.1 Introduction

In this chapter I describe the design and methodology of my research. There are different views on exactly what a research design entails, but its broad purpose is to draw up a plan that will help the researcher to work in a focussed and coherent way towards solving the research problem. I have chosen a framework proposed by Durrheim [1999] to guide me in formulating my research design and start this chapter with a discussion of the components of this framework (section 3.2). In the remainder of the chapter (sections 3.3 to 3.6) I then discuss each of these components in terms of my research.

3.2 The Structure of a Research Design

Durrheim [1999] describes a research design as the framework or plan that guides research in a way that ensures that it is properly done. His view of what a research design entails is summarised in Figure 3.1. Following the diagram is an explanation of each of the four components.

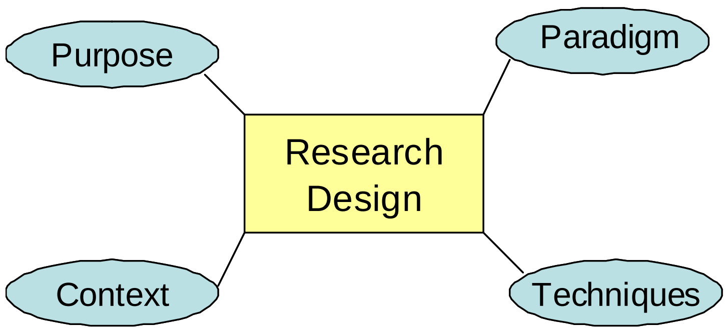


Figure 3.1 Durrheim's [1999] structure of a research design

3.2.1 The Research Paradigm

Rossouw [1980] describes philosopher Thomas Kuhn's understanding of a paradigm as 'a research tradition with its origin in core examples of scientific work that is dependent on implicit conceptual, methodological and ontological premises … not a set of well-articulated rules, principles or procedures, but rather a frame of reference.' (p. 10). The paradigm is thus an abstract framework that can help a researcher define what should be studied, what questions should be asked and how researchers should go about interpreting the answers. When choosing a specific paradigm, the researcher commits him or herself to methods of data collection and analysis that fit logically within the paradigm. The three major research paradigms in social research are the positivist, interpretivist and constructionist paradigms. These are distinguished by their differing ontological, epistemological and methodological assumptions [Terre Blanche and Durrheim, 1999].

Ontological assumptions describe what researchers believe exists and is real. They tell us what can be known about the reality being investigated. Positivists assume a stable external reality, interpretivists believe in internal reality of subjective experience and constructionists support a socially constructed reality [Terre

Blanche and Durrheim, 1999]. Epistemological assumptions describe how the researcher can know and explain things or phenomena. Positivists are detached from what they study and explain things objectively. Interpretivists work from an interactional perspective which acknowledges the subjective relationship between the researcher and the subject, and constructivists, who have a critical or suspicious view, explain reality by deconstructing different socially constructed versions of it [Terre Blanche and Durrheim, 1999]. Methodological assumptions just explain the course of action researchers will take to acquire the kind of knowledge they believe is possible. Positivists favour experimental, quantitative and hypothesis testing methods, interpretivists typically use interactional[4], interpretative and qualitative techniques and constructivists uses deconstruction and textual and discourse analysis. My research is neither positivist nor constructivist, but also not indisputably interpretivist. The specific research methods that I use are often associated with interpretivism and some aspects of the study can be regarded as interpretivist. I therefore discuss aspects of this particular paradigm in more detail in section 3.3.

3.2.2 Purpose

To define the purpose of the study the researcher must firstly specify who or what the objects of investigation will be and, secondly, what approach will be followed in studying it [Durrheim, 1999]. In other words the researcher has to make explicit the units of analysis and the type of study. I discuss the purpose of my research in section 3.4.

3.2.3 Context

Research always takes place in a specific context and how the researcher views the context will depend on the chosen research paradigm. Positivists, whose research is mostly experimental and quantitative, usually try to control and manipulate the context of the research. Interpretivists and constructivists, on the other hand, regard the context (both their own and that of the object of their study) as an important contributing factor in their investigation and will take into account how the research act influences and are influenced by the context. The role of context in my study will be explained in section 3.5.

3.2.4 Techniques

What Durrheim calls techniques, I will refer to as research methodology. It has three elements, namely sampling, data collection and data analysis [Durrheim, 1999] and a description of the research methodology should include detailed information about all of these. Sampling is fundamentally concerned with finding a representative set of the unit of analysis. Data collection methods are influenced by the research paradigm – positivists prefer objective, quantitative and experimental techniques, while interpretivists and constructivists use qualitative methods such as observation and interviews. Data analysis is the process whereby data is transformed into an answer to the research question [Durrheim, 1999] and is to a large extent determined by

[4] This occurs through subjective interaction with the research subjects.

the research paradigm, but also by the nature of the data collected. My research methodology is explained in section 3.6.

3.3 Research Paradigm

Since my research has some characteristics of interpretive research, I will discuss this paradigm and then explain its relevance to my research design. I begin with an introduction to the classical understanding of interpretivism as applied to the social sciences and then discuss some views on interpretivism pertaining to the more scientific field of information systems.

3.3.1 The Basic Principles of Interpretive Social Science: Brian Fay's View

Interpretive research is fundamentally about the relationship between the interpretation and the description of an action – how the description of an action is related to the explanation of that action [Fay, 1975]. If the description of a physical action implicitly refers to the purpose of the action, the description contains an interpretation (or explanation) of the action. Such an explanation typically gives information about the meaning the action had for the person performing it. According to Fay [1975] any action 'may have many descriptions which place it in a wider and wider context of purposes, intentions and rules…' (p.72). The aim of interpretation of action is to discover intentions the actor may have that is not evident from the initial description, and to include these discoveries in subsequent descriptions. So, according to Fay, the interpretive social scientist offers rich explanations of individual actions which show the reasons for the action and place the action in a wider context that includes the aims, cognitions and specific circumstances of the actor. In forming these descriptions the researcher uses public evidence rather than the actor's own account of the reasons for the actions, because people are not necessarily good judges of their own intentions. Furthermore, all descriptions and explanations of actions have an inherent social element – actions are always performed in the context of a particular set of social rules [Fay, 1975]. As an example, Fay explains that use of the action concepts 'buying' and 'selling' can only be described if certain economic rules exist that explain these actions. He calls the set of social rules that action descriptions implicitly refer to, *social practice*.

To correctly understand and describe (interpret) and action, one must understand the intentions of the actors as well as the implicit social practice. Fundamental to social practices are *constitutive meanings* which include shared assumptions, definitions and conceptions that give specific meaning to the world. People are unaware of the constitutive meanings of their behaviour, but these meanings determine the language they use to explain themselves, their beliefs and their attitudes. They would only be able to see these meanings if they could see themselves 'from the outside'. For the researcher, to discover the relevant constitutive meanings and the relationships between different constitutive meanings, means to find out what the point of a certain practice is in a society. He or she must identify 'the basic notions which people share about the world, society, and human nature' (p. 78). The basic notions are things such as conception of masculinity and femininity, the meaning and role of work, views of nature, ideas about authority, and beliefs about God.

Fay's view represents the traditional description of interpretivism that was exclusively associated with the social sciences. In the context of the social sciences, the broad ontological, epistemological and methodological assumptions of the interpretivist paradigm can be summarised as follows [TerreBlanche and Kelly, 1999]:

- Ontology: People's subjective experiences are real and should be taken seriously.
- Epistemology: People's experiences can be understood by interacting with them and listening to them.
- Methodology: Research is done through interactional, interpretative and qualitative techniques.

Today, however interpretive research also has a place in more scientifically oriented fields such as information systems.

3.3.2 Interpretive Research in Information Systems

Klein and Myers [1999] discuss the practice of interpretive research in information systems, and, acknowledging that there are different views of interpretive research they classify their view as interpretive research from a hermeneutic perspective (see section 0 for a discussion of Hermeneutics). HCI is a subfield of information systems, and therefore Klein and Myers' view is relevant to my study. They contrast interpretive research with positivist research and critical research in information systems, where positivist research is characterised by 'evidence from formal propositions, quantifiable measures of variables, hypothesis testing, and the drawing of inferences about a phenomenon from a representative sample to a stated population' (p. 69). Critical research aims to provide social critique by identifying and eliminating causes of unacceptable social or economic conditions [Klein and Myers, 1999]. Critical theorists assume that humans are able to act to improve their conditions, but recognise the social, cultural and political constraints and other debilitating factors. According to Klein and Myers' view of interpretive research, knowledge of reality can be gained through social constructions such as language, documents, tools and shared meanings. In the field of information systems, interpretive research aims to understand the relationship between a system (in other words, some form of technology) and the context within which it is used. More specifically, interpretive research aims to understand the purpose of an information system (that is, the intention of the designers of the system), the intention of the person or organisation using it, the effect of the system on those who use it and on the environment within which it is used.

For my research Klein and Myers' description of interpretive research can be translated to the following: I do this research to understand the relationship between young children's technology and its context of use as determined by the cognitive characteristics of five to eight-year-old users. I want to help designers to understand the purpose of the technology they design and the reasons why children would use it. I am particularly interested in the effect that the technology might have on the children using it and on the way it influences their development and their lives in general.

Since my research does not fall clearly within one of the traditional paradigms it seems sensible to clarify my theoretical point of departure by describing my ontological, epistemological and methodological assumptions. I end this section with a summary of these assumptions in table format.

Table 3.1 Assumptions associated with the literature analysis (phases 1, 2 and 3)

Ontological assumptions	Children from five to eight have specific characteristics that distinguish them from other children and I can learn about them through other people's (i.e. theorists and researchers) writings about their behaviour, ideas and thoughts. I follow a nominalist approach whereby I learn about the group through the individuals [Puttergill, 2000]. I also assume that through understanding children and their experiences with technology I can learn how to design technology that will be good for them.
Epistemological assumptions	I assume that I can learn the above through studying relevant texts with deep attentiveness, and with ongoing acknowledgement of the role of the different contexts that may influence the knowledge gained (that of the authors of the texts under study as well as that of the researcher). I expand my knowledge by re-interpreting earlier interpretations of children's behaviour with questions in mind that were not necessarily of interest to those researchers.
Methodological assumptions	The research methods that will lead me towards the required knowledge are qualitative analysis of psychological theories and research reports following a hermeneutical approach: attentive reading and re-reading of these texts, looking for themes, tensions, contradictions, et cetera. The researcher is the primary data collection and analysis agent.

3.3.3 Hermeneutics

I have said in the previous section that Klein and Myers [1999] describe interpretive research from a hermeneutic perspective. According to Inwood [2005], *hermeneutics* originated as a method of interpreting the Bible and other difficult texts. It was later extended to the interpretation of all human acts and products. The question within hermeneutics is whether the one doing the interpretation should place him or herself in the author's position in order to gain insight into the author's thoughts and intentions, or whether the text should be related to a wider context that influences its meaning? When viewed in the latter way, the parts of a text can only be understood if we understand the whole, and the whole text can only be understood if we understand its parts. This is called the hermeneutical circle.

Klein and Myers [1999] describe the nature of interpretive research from a hermeneutic perspective by means of a set of principles for conducting and evaluating such research. The first of these is the *fundamental principle of the hermeneutic circle* which translates to the view that interpretation is a continuous loop from an understanding of the parts to an interpretation of the whole and then from this understanding of the whole back to a more accurate understanding of the parts. In phase 1 of my research I investigate each of four theories of development in this way, moving between the components that make up the theory and an understanding of the theory as a whole. Then I study each theory as a part of the bigger whole, namely a general understanding of children's development and use this understanding to reinterpret the individual theories, and so on. Similarly, in my study of children and technology, I look at specific instances of children's interaction with computers to understand this interaction in general, while a better general understanding improves my interpretation of the individual cases.

3.4 The Purpose of this Research

According to Durrheim [1999] defining the purpose of a study involves specifying the units of analysis and the approach in studying them. Before I discuss these I briefly remind the reader of the thesis statement for this study.

3.4.1 Thesis Statement

Recapitulating what I have said in section 1.3 of Chapter 1, with this research I want to defend the following assertion: It is possible to develop a credible, dependable and useful set of guidelines for the design and evaluation of technology for children aged five to eight years by studying 1) psychological theories of children's development, 2) existing research results on children's cognitive development, 3) existing results on children's use of technology and 4) existing design guidelines and usability principles.

The purpose of my research is thus to demonstrate that it is possible to formulate a set of guidelines for the design of technology for children aged five to eight years. I will do this by systematically answering the following set of questions:

1. What are the cognitive and developmental characteristics of typical five to eight-year-old children with regard to skills relevant to the use of technology?
2. What can we learn from existing research into role of technology on skill development that can inform designers of technology for children aged five to eight?
3. What does the literature on interaction environments for young children tell us in terms of the design of technology for five to eight-year-old children?
4. What guidelines exist for the design of technology for children aged five to eight and which existing guidelines not specifically aimed at the design of young children's technologies apply to technology for children aged five to eight?
5. How can the guidelines emerging from the literature be organised into a framework that is useful for designers?
6. Is the proposed set of guidelines credible and useful?

3.4.2 Units of Analysis

I believe that the aim of the research can be achieved through an in-depth study of existing literature and have identified the following classes of literature that are relevant to the study:

1. Psychological theories of development, with an emphasis on young children's cognitive development.

2. Results of empirical research on young children's acquisition of cognitive skills which are relevant to computer use.
3. Results of empirical research on young children's experiences with computers and other technology.
4. Existing guidelines for the design of technology.

For phases 1 and 2 these texts are the units of analysis. I shall identify the most important relevant works in these categories and through systematic analysis identify specific information that can contribute to the formulation of guidelines for the design of technology for children aged five to eight (see section 3.6.1.1 for a discussion of how I will select appropriate texts).

The potential guidelines and other design-related factors identified in phases 1 and 2 will be the units of analysis in phase 3. Through careful synthesis, analysis and coding, the data gathered during the literature study will be integrated and reformulated and then organised into a logical framework.

In phase 4 the final framework will be tested with an evaluation and design case study.

3.4.3 The Research Approach (Type of Study)

De Villiers [2005] describes a research approach as the underlying model that operationalises the study. Different classifications of the research approach can be found in the literature on research methods [Terre Blanche and Durrheim, 1999]. Research can be

- exploratory, explanatory or descriptive,
- applied or basic, and
- quantitative or qualitative.

My research can be described as descriptive, applied and qualitative. It is *descriptive* in the sense that I aim to give a narrative-like description of a phenomenon [TerreBlanche and Durrheim, 1999], namely the developmental characteristics of children aged five to eight that may have some bearing on their relationship with technology. I do *applied* research, since the results of my research will assist people with problem-solving and decision-making in the context of designing technology for young children. My research is *qualitative* because the data will be in the form of written language that will be analysed by identifying elements, themes or patterns that may play a part in solving the research problem.

3.5 The Role of Context in my Research

An implication of conducting research from an interpretivist point of view is that context is assumed to have an important impact on the research process. The second of Klein and Myers' [1999] principles of interpretive research is *the principle of contextualisation* which maintains that historical and social distance cause a difference between an interpreter's understanding of a text and that of the author. This indicates the importance of setting subject matter in its social and historical context to explain how the current interpretation emerged.

Kelly [1999b] uses Dilthey's method of verstehen and hermeneutic philosopher Ricoeur's concept of distanciation to explain hermeneutic interpretation of text. When interpreting text, *verstehen* refers to a process of understanding the author's intention and the context in which the text was written. In other words the researcher tries to interpret the text from within the author's context (see section 3.5.1). *Distanciation*, on the other hand, refers to the process of understanding a text's context from the outside, assuming that some aspects of context are only detectable from the outside. The reader's context brings with it new questions and concerns which may call for a new way of reading a text. For example, when I study the theories of development I do that with the specific purpose of finding data that may contribute to the setting up of principles of design of technology. Reading from this perspective may bring to light aspects of the theories that may not have emerged otherwise.

There are different levels of context that play a role in my research. I discuss these in the subsequent sections.

3.5.1 The Historical Context of Developmental Theorists

When I investigate the theories of cognitive development I have to acknowledge the context from within which the theorist formulated their theories. Some of these theories date back to the early and middle 1900's and it is important to be familiar with their historical context when trying to understand and interpret them. I also need to acknowledge the contexts of the particular experiments and observation that the theorists based their findings on. For example, Piaget conducted a limited number of case studies from which he made generalisations. Many researchers have since demonstrated that Piaget failed to recognise the role of context in some of his experiments. Developmental theorists whose work came after that of Piaget and Vygotsky, for example, worked from within a context where they already had access to earlier theories that they could challenge, improve or prove wrong. According to Klein and Myers [1999] our reports should acknowledge that these theories (and whatever else we study) are not only products of history, but that they also play a part in creating history.

3.5.2 The Researcher's Context

Researchers must be aware of their own context and what that brings to the research process. Being a computer scientists and an information technologist, I work from a natural science context, investigating theories that fall within the field of social science. Whereas natural scientists traditionally take a positivist stance, I choose to work from an interpretive perspective.

Applying the concept of *verstehen* I know that when analysing existing results of research done by other researchers on children and technology, I need to recognise the background and assumptions of those researchers. I have to take into account the broader social context within which that specific research was undertaken as well as the specific context of their experiments. For example, was it based on observations in a natural environment or were the subjects placed in an unfamiliar setting such as a laboratory? When analysing the results I need to consider what the effect of these contexts may have been on the results.

Then again, the process of *distanciation* refers to understanding a context from outside that context, suggesting that certain aspects of context can only become evident when viewed from outside [Kelly, 1999b]. This implies that, when seen from the outside, a text can mean more than what the author intended. This extended meaning depends on the researchers' reasons for analysing the text. My investigation of research on children's development and their experiences with technology is driven by the aim to extract design guidelines. So, although the research that I study may not have been concerned with design practices, I may find information pertaining to that because that is what I purposely look for.

3.5.3 The Designer's Context

When it comes to evaluating technology, one has to be informed of the context from within which it was designed. Design can be motivated by the urge to sell a product, in which case the producers would typically not consider whether the product is accessible to minority groups who will not be able to afford it. If, however, the design is motivated by sound educational and developmental practices, the buyers may rather be parents and educators and not the children themselves. Products for children that are developed in the USA but sold all over the world, seldom offer children the choice to select their home language, currency or seasonal information so that the activities would be meaningful to all users.

3.6 Research Methodology

Since my research involves firstly a literature study and the consequent formulation of design guidelines and, secondly, the validation of the proposed guidelines, I divide my discussion of the research methodology into two corresponding parts. As I have said in section 3.2.4, research methods include sampling, data collection and data analysis. I begin by discussing the research methodology for the literature analysis under these three headings.

3.6.1 Literature Analysis (Phases 1, 2 and 3)

3.6.1.1 Sampling

A sample is a selected subset of elements from a defined larger set [Puttergill, 2000]. In this research I want to study theories of cognitive development, reported results about research on children's cognitive development, reported results of research about children and technology and existing guidelines for the design of technology. Since it will be impossible to study every existing case in these categories I need to set boundaries that will limit the number of cases to study. I discuss the sampling of each category of literature separately.

3.6.1.1.1 Theories of Cognitive Development

I will limit the number of theories to be studied to four and will make an effort to identify four that are prominent and useful in the context of my study. According to Kelly [1999a], in a field with a well-established body of existing theory, a researcher should use specific research questions to identify particular cases to verify certain ideas. Since I specifically want to investigate cognitive characteristics that relate to young children's use of technology, I can use this research focus in selecting appropriate theories. I believe that multiple case sampling [Miles and Huberman, 1994], as opposed to studying a single developmental theory, is necessary to establish a trustworthy profile of five to eight year old children.

I will choose theories according to the following criteria:

1. It is a prominent and respected theory in the field of developmental psychology, has been studied thoroughly and is currently still regarded as important, credible and influential. (With this criterion I apply what Miles and Huberman [1994] call *reputational case selection*.)
2. It includes examination of the cognitive development of children aged five to eight.
3. It contains reference to aspects of cognitive development that may possibly relate to children's use of technology.

3.6.1.1.2 Results on Children's Cognitive Development

When studying the theories of development, specific themes will emerge that relate to my research problem. When I identify a need to investigate such themes further (beyond the specific theory it emerged from), I will search for relevant literature in respected academic journals.

3.6.1.1.3 Results of Research on Children and Technology

Here I will follow a strategy of comprehensive sampling [Miles and Huberman, 1994], trying to find every possible case that is relevant to my study and worthy of investigation. One limiting factor will be the quality and credibility of the research, which can be determined by the status of the researcher in his or her field and the status of the journal, book or proceedings where it was published. Examples of credible sources are ACM Interactions, ACM Transactions on Computer-Human Interaction, CHI conference proceedings, the International Journal of Human-Computer Interaction, peer reviewed IEEE journals and the International Journal of Human-Computer Studies. Although there are some classic works that was published as early as 1980, I will give preference to more recent results. It is important that I cover enough material to justify the formulation of generalisations on the basis of the specific cases I look at [Kelly, 1999a].

3.6.1.1.4 Existing Guidelines

Again I will follow a strategy of comprehensive sampling. In this case it will be easier as there are a limited number of respected sets of guidelines for the design of technology, and very few aimed specifically at technology for children. I will use academic texts written by leaders in the field of human-computer interaction as guide in identifying the guidelines that are worth including in the study. As with the theories of cognitive development I will restrict my study to prominent and respected work in the field of human-computer interaction that are referred to often and is currently regarded as important, credible and influential.

In general, to determine whether I have covered an adequate quantity of material I will use the concept of *theoretical saturation* [Kelly, 1999a] that occurs when new information ceases to add to the interpretive description. This happens when no new categories or themes emerge, when the relationships between existing themes are clear and not disputed and when the theoretical discussion is complete.

3.6.1.2 Data Collection

Collecting the material described in the sampling methods above takes care of a considerable part of the data collection process. Studying the chosen texts and identifying design-relevant factors will complete the process. Decisions on what count as potential guidelines or design-related factors require critical reading of the sources. This can also be regarded as part of the data analysis, reflecting the view of Terre Blanche and Kelly [1999] that there is no clear point at which data collection ends and analysis begins. All data collected will be presented in data boxes in the relevant chapters.

3.6.1.3 Data Analysis

Although they admit that interpretive data analysis cannot be done according to fixed steps carried out in a specific order, Terre Blanche and Kelly [1999] describe the course of analysis as a five step process. I discuss my approach to data analysis in phases 1, 2 and 3 with reference to these 'steps' but stress that in practice there are considerable overlaps and that the processes cannot be separated as this discussion may imply.

1. Familiarisation and immersion

Because the data collection process in my research already requires development of suppositions and ideas about the phenomena being studied, I should have a good understanding of the data when analysis begins. At this point I will again immerse myself in the material gathered, reading it repeatedly, making notes, and drawing diagrams. The aim is to know the data as well as possible. The outcome of this will be written descriptions of the specific theories, research results and existing guidelines.

2. Inducing themes

Induction refers to the inference of general rules or classes from specific cases. This is a bottom-up approach to identifying themes in the data. A top-down approach would mean the researcher starts with an existing set of categories and looks for material that support, or belong under, those categories. My approach will be a combination of bottom-up and top-down. The formulation of my thesis statement already provides me with some organising principles (such as the age group I am interested in and the fact that I will focus on selected developmental characteristics), but I will refine these and search for new themes during analysis. The aim is to reach the right level of complexity with not too many themes, but enough to be useful. Analysis isn't just a process of summarising content – it identifies and investigates processes, functions, tensions and contradictions that arise in the literature [Terre Blanche and Kelly, 1999]. The descriptions of the theories, research results and existing guidelines can now be refined and reorganised according to the themes identified.

3. Coding

Now the descriptions are scrutinised to identify instances of specific themes, or relevant to specific themes or categories. These are coded in a way that links them to that theme or category. In my study the themes will relate to aspects of children's use of technology. Instead of annotating the relevant piece of text right there in the document or in the margin, I copy it to a separate part of the document where it is paraphrased in the form of a design guideline and labelled in a way that will clearly link it to its origin in the text. This way I group related guidelines together in highlighted text boxes while still allowing the reader to relate them back to their origin.

Codes, themes or categories need not stay fixed throughout the coding and analysis process. New themes may emerge at any stage, or existing themes may be discarded based on newly acquired knowledge.

4. Elaboration

Elaboration involves exploring the newly organised material to identify similarities and differences in the data that may lead to new insights. The accrued material resulting from analysis and coding is now reorganised and arranged into a coherent discussion of the phenomena under investigation. Analysis continues until no new insights emerge. In my case, the outcome of elaboration will be a well-organised, integrated framework containing the final set of guidelines.

5. Interpretation and checking

Here I give the final account of my study. I must now make sure there are no weak points, contradictions or holes in the proposed framework. Problems such as over-interpretation of trivial matters or parts where I was obviously led by my prejudices should be identified and corrected.

3.6.2 Validation of the Proposed Guidelines (Phase 4)

Validation of the results of qualitative research is not a straightforward process for which clear guidelines exist [Miles and Huberman, 1994]. The question is: how do I show that my findings are good? The first part of the validation process will involve two proof-of-concept exercises: an evaluation of an existing software product using the guidelines and the re-design of parts of the application according to the guidelines. I describe this in section 3.6.4.1.

Miles and Huberman [1994] provide what they call 'practical standards' for assessing the quality of conclusions based on qualitative research. I will use these standards for the validation of my research results and explain what each of them entails in section 3.6.4.2.

3.6.2.1 Proof-of-Concept Exercises

To conclude the research I will demonstrate the usefulness of the proposed guidelines by evaluating an existing software product aimed at young children and re-designing parts of it.

3.6.2.1.1 Using the Proposed Guidelines to Evaluate a Software Application

The first step here will be to choose a suitable product for the evaluation. I will follow an approach of judgemental sampling [Van Rensburg, 2000] that relies on the subjective considerations of the researcher who will make a selection on the basis of his or her knowledge of the population being studied as well as the purpose of the study. I will use the following criteria in choosing the software:

- It must be aimed at children aged five to eight years.
- It must address one or more of the cognitive skills that my research puts emphasis on.

Dix et al. [2004] describe the goals of evaluation as assessing the extent and accessibility of the system's functionality, assessing users' experience with the system and identifying problems with the system. As I have said in Chapter 2 (section 2.3.5) I will conduct a analytical evaluation where an expert uses heuristics and knowledge of typical users to predict usability problems [Preece et al., 2007]. The guidelines resulting from my research will be the heuristics used in this evaluation. Since there will be no user involvement in using the guidelines for evaluation, I will not address the second of Dix et al.'s goals, namely to assess user experience. I do believe that I can achieve the other two goals.

Nielsen [1994] recommends that between three and four experts evaluate a system to find around 75% of the usability problems. My aim with this evaluation is to demonstrate that the guidelines can be used in a heuristic evaluation and not to provide an exhaustive evaluation of the chosen product. For this exercise I will therefore be the only evaluator.

After choosing the product for evaluation I will go through the set of guidelines one-by-one to separate out those guidelines that are suitable for this evaluation. If, for example, the product has no reference to mathematics skills it would be pointless to include the guidelines that specifically address mathematics-related activities as heuristics for this evaluation. Each of the selected guidelines will then be used as a heuristic as described by Preece et al. [2007].

3.6.2.1.2 Designing Computer-Based Activities Using the Proposed Guidelines

The scope of this work does not justify development of a complete product according to the guidelines, although that would have been the ideal way to demonstrate the usefulness of the guidelines for supporting the design process. Instead I will settle for the design of a selection of detached functions that make use of a representative selection of the guidelines. The emphasis will be on showing how the guidelines can be used to make design decisions.

I will present the outcome of this exercise with graphical illustrations of the relevant design aspects, supplemented by a complete narrative description of the associated activities and user actions.

3.6.2.2 Validation According to Miles and Huberman's Practical Standards

In formulating their practical standards Miles and Huberman [1994] address five issues, namely:

1. Objectivity (or confirmability).
2. Reliability (or dependability or auditability).
3. Internal validity (or credibility or authenticity).
4. External validity (or transferability or fittingness).
5. Utilisation (or application or action orientation).

In each case the traditional term used in validation of research findings is given first and then in brackets some alternative terms that fit better with results based on qualitative data analysis. In validating my research I will use the terms confirmability, dependability, credibility, transferability and application. For each standard Miles and Huberman [1994] provide a set of questions that researchers can answer in order to assess their research. I include the complete lists of questions from Miles and Huberman in Appendix 2. In Chapter 10 I will apply them to do a detailed assessment of the validity of my research findings. In the remainder of this section I briefly describe what each standard entails.

3.6.2.2.1 Confirmability

The question here is whether a different researcher would have come up with the same results. A study can only be replicable if the data collection and analysis methods are described in sufficient detail, giving a complete picture [Miles and Huberman, 1994]. Also the results must be explicitly linked to visible evidence in the data and did the researchers' acknowledge her own assumptions and possible biases [Miles and Huberman, 1994].

3.6.2.2.2 Dependability

According to Miles and Huberman a study is dependable if it is consistent over time and across researchers and methods. It reflects how carefully the research is conducted. In my case evidence for dependability will, for example, be conformity of sets of results based on different data sources, respectively.

3.6.2.2.3 Credibility

Here one asks whether the findings are credible to those who read it. Credibility requires context-rich and meaningful descriptions that make the findings convincing [Miles and Huberman, 1994]. Clear links to existing theory and triangulation of research methods and data sources can also improve credibility.

3.6.2.2.4 Transferability

Transferability refers to the extent to which the findings can be applied to other contexts or generalised [Miles and Huberman, 1994]. Generalisability involves linking findings to situations or aspects that did not form part of the specific study.

3.6.2.2.5 Application

This principle evaluates the practical applicability of the findings and their accessibility for potential users [Miles and Huberman, 1994].

3.7 Conclusion

In this chapter I described the design of my research and explained the methodology I will follow in conducting the research. In doing this I explained and justified my sampling methods, my strategies for collecting the required data and the ways in which I will analyse the data through qualitative methods. I explained the rationale for choosing these methods and explained how my research fits into the interpretive research paradigm. I also presented my plan to validate the results of my research in terms of confirmability, dependability, credibility, transferability and application.

I have now completed the introductory part of this thesis by giving an overview in Chapter 1, the conceptual framework in Chapter 2 and describing the research design and methodology in Chapter 3. The real work can now begin. In Chapters 4 to 7 I will report in detail on the literature analyses that make up phases 1 and 2 of this study.

CHAPTER 4

Young Children's Developmental Psychology

Table of Contents

4.1 Introduction

The thesis being investigated in this research is whether it is possible to develop a credible, dependable and useful set of guidelines for the design and evaluation of technology for children aged five to eight years by studying 1) psychological theories of children's development, 2) existing research results on children's cognitive development, 3) existing results on children's use of technology and 4) existing design guidelines and usability principles. In this chapter I address the first and second of these four classes of knowledge in an attempt to answer the following question: What are the cognitive and developmental characteristics of typical five to eight-year-old children with regard to skills relevant to the use of technology and how do they inform the design of technology? I survey relevant theory and empirical studies in the field of developmental psychology, focussing on the cognitive development of children aged five to eight, and limiting the survey to aspects of child development that may have some relation to the use of technology.

I identified four theories of cognitive development to study in detail, namely those of Piaget [1953], Vygotsky [1978], Case [1992c] and Fischer [Fischer and Bidell, 2006]. I discuss them in section 4.2 first explaining why I have chosen these particular theories. Section 4.3 deals with the developmental domains I have identified as relevant to my study. As I have explained in Chapter 1 (section 1.4), the skills I will focus on are literacy, mathematics and thought. My discussion of thought includes the related concepts of memory, knowledge representation and problem solving. I discuss 'play' and its role in development in section 4.4. All of these domains are interrelated [NAEYC, 1996] – in practice, children's cognition is not neatly divided into separate areas like 'mathematics' or 'literacy' and the development of one competence can be woven into experience of another area [NAEYC, 2002]. For example, language skills influence social relationships and hence socio-cognitive development. So, although I have organised part of this chapter using headings such as 'Literacy' and 'Mathematics', these sections may include discussion of aspects of other domains that are somehow linked to the one under discussion.

The overarching purpose of this literature study is to provide input for the formulation of guidelines for the design of young children's technology. Therefore, after discussion of each theory and each domain of development, I include shaded data boxes that list potential guidelines and other design-related factors that emerge from the discussion. The data elements are labelled so that when I organise them into a framework in Chapter 8, they can be traced back to their original sources.

4.2 Prominent Theories of Cognitive Development

4.2.1 Introduction

Many theories of development view cognitive development from a constructivist perspective, whereby children contribute to their own development through interaction with their environment [Gardner, 1991; Piaget, 1953; Vygotsky, 1978]. Children learn actively by observing and interacting with other children and adults and construct their own mental models of different aspects of the world [NAEYC, 1997]). Objects, events or other people may prove such a model wrong, forcing the child to alter it according to the newly acquired knowledge. Development progresses through continual adjustment, expanding and reorganising of mental structures [Case and Okamoto, 1996; Piaget, 1953; Vygotsky, 1978]. Different theories of development explain this process in different ways. Piaget regarded the development of general cognitive structures as a necessary prerequisite to learning. His structural theory represents a monolithic view of development where cognition depends on universal logical structures and learning happens from within the individual [Fischer and Immordino-Yang, 2002]. Vygotsky demonstrated that the learning of concepts and strategies lead to the formation of more complex cognitive structures and viewed development as contextually determined and driven by external factors [Berk, 2000]. Case [1992c] believed that both these approaches explain aspects of cognitive development and his neo-Piagetian theory finds a balance between them. Building on Case's theory, Fischer and co-workers [Fischer and Bidell, 2006; Fischer and Yan, 2002; Fischer and Immordino-Yang, 2002] describe development as a constructive web, where skills develop in a specific order but with great variation due to contextual, biological and emotional factors.

These four theories are representative of the historical progression in developmental psychology. There are other important theories such as Chomsky's [1986] neo-nativist theory, Siegler's [1976] model of strategy choice and Gardner's [Gardner, 1983]theory of multiple intelligences that are also relevant to my study, but that I do not review in detail. This is not because I regard them as less significant in the history of development psychology, but rather because the scope of my study does not justify inclusion of them all. I will refer to important aspects of some of these theories when discussing the different skill domains in section 4.3.

The remainder of this section is devoted to an overview of the four chosen theories.

4.2.2 Piaget

4.2.2.1 The Basic Idea

Jean Piaget was born in 1896 in Switzerland. He received a PhD in biology from the University of Neuchâtel at the age of 21 before deciding to explore psychology [Ginsburg and Opper, 1969]. Piaget's research was motivated by the search for an underlying logic of the mind [Fischer and Immordino-Yang, 2002] and his assumption was that the mind consists of universal, domain-independent structures. According to Piaget's theory, cognitive development progresses as changes in knowledge structures. Knowledge is organised into schemes that are sets of physical actions, mental operations, concepts or theories [Berk, 2000]. In the process of cognitive development new schemes are created and existing schemes are reorganised through *organisation* (integrating schemes into more complex structures) and *adaptation* (changing schemes to fit environmental demands or moulding the new information into existing schemes). The developing child actively takes information from the environment and processes and reorganises it in order to maintain a balanced and coherent state of *equilibrium* [Pine, 1999]. The mind adapts to the environment in two ways:

through assimilation or accommodation. *Assimilation* is the process of applying one's current knowledge to interpret the external world [Berk, 2000] (e.g. a young child calls a cat a dog). *Accommodation* happens when current knowledge does not fit the environment and new knowledge schemes must be created or old ones changed to accommodate feedback from the environment (e.g. the child who called a cat a dog sees a cat and a dog and realises they are different things). According to Piaget these two processes form the basis of a child's ability to reverse mental actions and move back and forth through a set of elements, such as numbers or actions [Fischer and Immordino-Yang, 2002].

Piaget identified four factors that contribute to cognitive development, namely

- maturation of inherited physical structures,
- physical experiences with the environment,
- social transmission of information and knowledge, and
- equilibration, which is the tendency described above to maintain balance of the cognitive structures.

4.2.2.2 The Stages of Development

Piaget believed that every child goes through four stages of development and that these stages are universal. I summarise the stages in Table 4.1.

Table 4.1 Piaget's stages of cognitive development (from Pine [1999]) .

Age	Stage	Description
0-2 years	Sensorimotor	Understand the world through the senses. Acquires object concept .
2-6 years	Pre-operational	Beginning of thought and language from a egocentric perspective.
6-12 years	Concrete operational	Mental manipulation possible but tied to the concrete.
12 years +	Formal operational	Manipulation of abstract ideas, logical systematic thinking, reflective thought.

Although, on the surface, different patterns of behaviour occur in a particular stage, Piaget concluded that there is one common structure that underlies each stage respectively [Donaldson, 1978]. To move from one stage to the next therefore requires fundamental re-organisation. Still, he believed that development is continuous and that there are no clear breaks between the stages. Although not all children move from one stage to the next at exactly the same age, the order in which they move through the stages is universal.

I am particularly interested in the later part of the pre-operational stage and the transition to the concrete operational stage. According to Piaget, real reasoning begins in the pre-operational stage. Children develop the ability to think about objects, events and people that are not present and they begin to use symbols such as words, numbers and images to represent real objects [Berk, 2000]. They can use words to communicate, engage in pretend-play and, later in the stage, use numbers to count objects and drawings to express ideas.

The concept of *reversibility,* which Piaget regarded as fundamental for development, depends on the development of operational structures [Donaldson, 1978]. In Piagetian terms an operation is an action carried out in the mind, such as the combining, ordering, separating and recombining of elements. It always forms part of an organised system of operations which Piaget calls a *group* or *grouping*. When a child's thought has become operational, it means that he or she is now able to reverse any action (operation) mentally. So, where a child of three (pre-operational) can put objects in a row, move them around and then move them back, a child of seven or eight can perform these actions mentally. The mental operations are still concrete in the sense that they involve thinking about actions that can actually be performed physically. In the operational stage children can understand the relationship between different states of the world, they are interested to explain and understand the observations and because they operate on a mental level, are less bound by physical space and time.

Another Piagetian concept that develops during the late pre-operational stage and the concrete operational stage is *decentration* [Donaldson, 1978]. At first children are unable to identify a point of view different from their own. In the physical sense this means if a three year old looks at an object from one side and another child looks at it from a different angle, the three year old will assume that the other child sees exactly what he or she sees. They cannot 'decentre' in their imagination. According to Piaget they only acquire this skill from the age of eight or nine.

Based on his experiments Piaget concluded that children's thinking become more logical from the age of seven or eight (see detailed discussion of a selection of experiments in section 4.2.2.3 below). In the concrete operational stage they can attend to more than one aspect of the situation (for example, the number of items and the length of the rows in the conservation task) and they can perform the reversibility operation to connect one aspect of a situation to another. They can solve conservation, transitivity and class inclusion tasks and explain their reasoning. Younger children rely on perceptual evidence, while for older children reversible mental operations form the basis for understanding logical relationships [Thornton, 2002]. However, the reversible operations are still strongly connected to concrete things and situations and concrete operational children cannot represent abstract relationships. The relationships have to be between specific, concrete things in a specific context. They cannot reason abstractly or hypothetically [Thornton, 2002].

4.2.2.3 Piaget's Experiments

Piaget performed countless experiments to refine his theory and illustrate the working of the concepts described above. Below I consider some that have received much attention in the literature.

4.2.2.3.1 Conservation, Transitivity and Class Inclusion

Piaget's experiments about children’s conservation of number and volume led him to believe that pre-operational children have not yet grasped conservation, while concrete operational children have [Pine, 1999]. In other words, younger children cannot hold constant a number of items if the same items are organised in different ways, while children of seven and older can. One experiment works as follows: a row of objects is placed in front of the child. A second row of the same number of items is then lined up with the first row. The young child understands that the rows contain the same number of objects. Next, the adult changes the one row so that it still contains all the objects, but is shorter than the other. Now the pre-operational child will think the one row contains fewer items, while the concrete operational child will realise that the number has not changed.

Similarly, according to Piaget, young children do not understand that the volume of fluid remains the same if it is poured from one container into a container with a different shape [Donaldson, 1978]. They only use the height of the fluid level to judge the amount. Since they do not have the ability to mentally reverse operations, they do not understand the causal connection between the present situation and the original one. (I discuss criticism of these findings in section 4.2.2.4).

In a length conservation experiment, Piaget placed two sticks in parallel so that the child could see that they are equally long. He then moved one stick so that their ends aren’t aligned and found that pre-operational children then said the lengths differ. In response, Rose and Marion [1974; as cited by Donaldson, 1978] did experiments which showed that children regard the adult’s repetition of the question (after the change) as a hint to revise their answer. McGarrigle did the experiments in a way that it appeared as if the change happened by accident (for example, naughty teddy messed up the game). Children between four and six years of age faired better now but some still had difficulty [Donaldson, 1978]. Experiments by Gelman [1969; as cited by Donaldson, 1978] showed that with training young children become better at conservation.

In his transitivity experiment, Piaget showed children a green and a yellow stick. The yellow one is longer. The yellow stick is then removed and replaced with a red stick that is shorter than the green one. The child is asked which of the red and the yellow stick is the longest. Pre-operational children could not give the answer and Piaget explained this in terms of their inability to mentally reverse the actions to connect the initial situation with the final one. They cannot mentally represent the two relevant relationships (yellow-green and green-red) and therefore cannot compare the yellow and red sticks [Thornton, 2002].

Piaget’s class inclusion experiment further illustrates young children’s inability to mentally reverse actions and to think about two aspects of a situation at the same time. If children at the age of six or seven are shown a picture with two cats and four dogs and they are asked how many dogs there are, they will answer correctly. They also know that dogs and cats are animals and they are able to count the number of animals in the picture. However, if you ask them ‘are there more dogs or more animals?' they will say there are more dogs. They cannot include the dogs in two classes (‘dogs’ and ‘animals’) at the same time [Thornton, 2002].

In other words, they cannot compare a subclass with the class it belongs to. Piaget believes that if the child 'centres on' the whole class he cannot think of the parts at the same time. Researchers such as Donaldson [1978] have responded to Piaget's results by pointing out other reasons for children's incorrect responses in this experiment. Since it makes more sense, the children may think that they are supposed to compare the two subclasses. Donaldson [1978] describes experiments conducted by McGarrigle to test the same skills with different ways of questioning. He found significant improvement in children's performance. For example, McGarrigle showed children with an average age of six years a collection of three black and two brown cows and then put them on their sides saying they were asleep. When asked 'Are there more black cows or more cows?' twenty-five percent of the children answered correctly. When asked 'Are there more black cows or more sleeping cows?' forty-eight percent answered correctly. The two things that made the difference here is perceptual contrast (placing all the cows on their sides) and change of wording in the question (adding one adjective). McGarrigle found that, although together they made a significant difference, neither of these alone made a difference.

4.2.2.3.2 Decentration

As I have explained before, Piaget believed that young children are unable to decentre – that is, see a situation from another's point of view [Donaldson, 1978]. He used the following experiment to illustrate this: A child is shown a model of three mountains which differ in colour and other distinguishing features such as having a house on top or being covered in snow. The child sits at a table in front of the model. A doll is then placed on a different side of the model and the child must describe the doll's view. In one version of the experiment the child is shown pictures of ten possible views and has to choose that of the doll. Piaget found that children up to eight or nine could not perform this task successfully and below seven, children pick their own point of view.

Donaldson [1978] describes how Hughes devised a different experiment using a model of two walls that intersect at a ninety degree angle, forming a cross with four quadrants (see Figure 4.1). A doll representing a policeman is placed so that he can see two of the quadrants and two are hidden. For example, if placed as shown in Figure 4.1, the policeman can only see quadrants B and D. Another doll is then placed in different quadrants and the child must say whether the policeman can see the doll or not. A second policeman is then added so that only one of the four quadrants can be seen by neither. Figure 4.1b shows a scenario where only quadrant C is invisible to both policemen. The child has to hide the doll from the policemen. Hughes found that children as young as three-and-a-half could do this successfully.

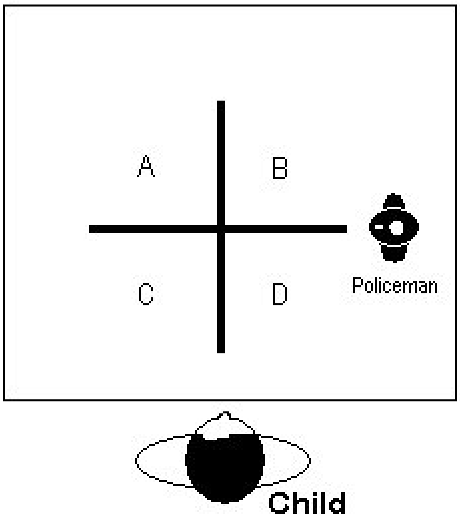


Figure 4.1a Hughes' experiment with one policeman doll.

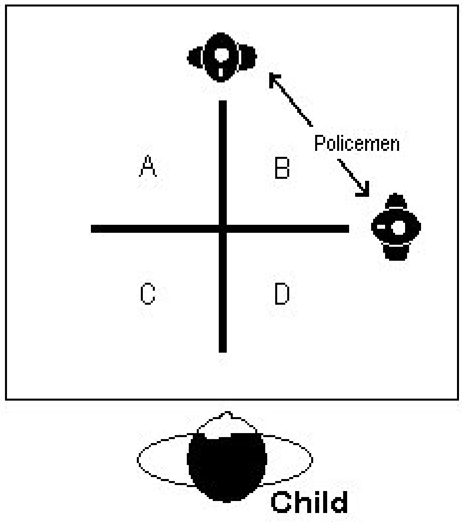


Figure 4.1b Hughes' experiment with two policemen dolls.

One difference between the two experiments is that Hughes' does not deal with left-right reversals. The child only has to say whether something is visible or not, but need not describe the appearance of the situation from a doll's point of view. Describing how something will appear is a much harder task than saying whether something is visible or not. Furthermore, Hughes' characters had intentions and motives that the subject could understand, while the mountains task is abstract in the sense that there is no specific reason for looking at the mountain from a different angle.

4.2.2.4 Comments on Piaget's Theory

Piaget based his research on the assumption that the mind consists of universal knowledge structures, but neither Piaget nor any other supporter of this idea could sufficiently demonstrate the consistency of mind in terms of this assumption [Fischer and Immordino-Yang, 2002]. Their theories do not explain the variation in real learning and problem solving. This does not mean that Piaget's work is of no value. On the contrary, his search for an underlying logic of mind produced many important insights about cognitive development. Piaget did acknowledge variation in children's skills and referred to this unevenness as *décalage*, but he never attempted to explain the variability in skills acquisition [Berk, 2000].

Piaget's rigid stages of development also evoked criticism. Research has shown that children do not automatically apply their knowledge in one domain to other domains. Even within a domain a particular child may understand some things but not others. For example, a child can understand the conservation of one material but fail to understand the conservation of another [Fischer and Immordino-Yang, 2002]. What Piaget did get right was the sequencing of development – in other words, the order in which skills develop within a particular domain.

Referring to the work of various researchers, Thornton [2002] discusses the problems associated with Piaget's conclusions. Some of these studies have shown that children as young as three can solve conservation, transitivity and class inclusion tasks in certain contexts. They blame the discrepancies in the findings on the fact that Piaget's experiments were too abstract and unfamiliar to the children. Defenders of Piaget say that the modified versions of the experiments do not test the use of logical competence as Piaget intended. As discussed above, experiments carried out by Hughes contradicted Piaget's findings about decentration [Donaldson, 1978]. In Hughes' experiments children between the ages of three-and-a-half and five years are able to imagine what another person can see. The difference between Hughes and Piaget's experiments in this regard is that where in Piaget's experiments the children had to determine *what* can be seen and *how* it will appear, Hughes only asked *what* the other person will see. Apparently the children who did Piaget's task did not understand exactly what they had to do. Piaget presented his experimental scenario in an abstract way without explaining to the child why the different characters are looking at the mountain from different perspectives. The scenario had no meaning that the child could relate to, while Hughes chose characters with motives and intentions that were understandable to a child. Based on these results Donaldson [1978] concludes that children are able to decentre at a much younger age than Piaget claimed.

The differences in children's performance on related tasks indicate that the nature of the task and the individual differences between children have an impact on results. The details of the way the task is presented and the ways the questions are asked influence the outcomes. In these experiments it is therefore difficult to test only logical competence, because situational factors cannot be ignored. Rather than asking whether a child succeeded in a task, one should ask why or how he or she succeeded or failed [Thornton, 2002].

P01 Designers must be well informed of all the knowledge schemes that underlie every activity they present to the child.

To support cognitive development technology must produce changes in the child's knowledge schemes. In Piagetian terms this means:

P02 A product must make it possible for the child to
- fit the information presented into existing schemes (assimilation),
- adapt existing schemes so that the new information can find a place (accommodation), or
- combine existing schemes to form more complex schemes (organisation).

P03 Children from five to eight can distinguish software-based characters and objects from real-world characters and objects.

P04 Technology aimed at five to eight year olds may use symbols and images to represent real-life situations.

In support of development of reversibility skills designers can:

P05 Include activities that require children to mentally reverse actions such as combining, ordering, separating and recombining of elements.

Where:

P06 Software aimed at younger pre-operational children should allow users to physically move objects, e.g. by dragging them with the mouse.

P07 Children older than six should be allowed or even expected to perform operations that involve combining, ordering or separating objects, mentally.

With regard to decentration, computer software lends itself perfectly to teaching children to see physical spaces from different points of view. Designers can:

P08 Present children with three-dimensional images that they can manipulate with the mouse and virtual physical spaces through which they can navigate using the mouse, keyboard or other input devices.

For decentration on a more abstract level – for example, imagining what someone else feels or thinks – software designers can:

P09 Employ narrative-based activities where children must help on-screen characters to solve problems and make decisions that may be influenced by or have consequences for the actions and thoughts of one or more other characters.

According to Piaget, operational children can compare different states of the world and are interested in explaining things. It would therefore be age-appropriate to:

P10 Present children with activities where they can experiment with changes of state in a way that explains the differences.

To support development of conservation:

P11 Present computer-based activities that involve number conservation or length conservation. The familiar experiments can just be presented in graphical form on a screen. (Fluid conservation will be more difficult as nothing beats experimenting with real containers and real fluids.)

P12 Class inclusion activities such as McGarrigle's black, brown and sleeping cows experiment (see section 4.2.2.3.1) will be easy to represent with computer graphics.

4.2.3 Vygotsky

4.2.3.1 The Basic Idea

Lev Vygotsky was born in 1896 in Bylorussia and received a degree in literature from Moscow University in 1917. He began his career in psychology in 1924 at the Moscow Institute of Psychology. He was probably the first modern psychologist who regarded culture and society as defining factors in human development [Cole and Scribner, 1978]. 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 [Meece, 2002]. 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 [John-Steiner and Souberman, 1978]. 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 [Crook, 2000]. His theory is appropriately also referred to as *cultural psychology* [Crook, 2000].

Vygotsky's theory supports a view of development as more domain-specific and less general-process than generally accepted in his time. He regarded development as highly 'situated' and believed that to understand cognitive development one must consider the time and place where skills are acquired [Crook, 2000].

John-Steiner and Souberman [1978] describe Vygotsky's view that the process of development involves the emergence of psychological systems that combine separate functions (elementary, biologically determined or higher order cultural functions) into new higher order functional structures. Higher order functions do not exist on top of the elementary functions. The two levels are integrated to form one new system. The way in which these functions are combined, and the new relations that result, depend on the social experiences of the individual child.

Vygotsky [1978] regarded language, and speech in particular, as an important tool in the development process. In his experiments he found that children's activities are very often supported by speech. They do not just say what they are doing, but their speech is an integral part of the action aimed at achieving a specific goal. Speech and action are parts of the same function. Sometimes children are unable to do something if they are not allowed to speak at the same time. In this respect he agrees with Piaget who found that the amount of egocentric speech increases with the complexity of a task. According to Vygotsky [1978], egocentric speech develops into inner speech. Early in development egocentric speech occurs together with a child's actions, while later inner speech will precede action. Thus, language serves as a tool that will eventually allow children to plan a solution to a problem before acting out the solution.

His view that learning happens though interaction with adults led to the development of two concepts that are central to his theory: the 'zone of proximal development' and 'scaffolding'. I discuss these in the next two sections respectively.

4.2.3.2 The Zone of Proximal Development

Vygotsky believed that one of the most important ways in which children can learn about the world and acquire new skills, is by actively working together with or talking to a more experienced person [Thornton, 2002]. In this regard, Vygotsky [1978] introduced the concept of the zone of proximal development (ZPD) which he defined as 'the distance between the actual developmental level as determined by independent problem solving and the level of potential development as determined through problem solving under adult guidance or in collaboration with more able peers' (p. 86). This means that children's cognitive development can benefit from assistance or instruction and that the limits of their competencies can be expanded with the right kind of assistance. For Vygotsky the zone of proximal development represents the next developmental level that a child will reach. He believes that learning should be in advance of development rather than being oriented toward the current developmental level of the child [Vygotsky, 1978]. He also believed that children are highly motivated to master what they can almost do, or to understand what they almost grasp.

Problem solving in the ZPD happens through interaction between a novice and a skilled helper and relies on the language that passes between them. The ZPD is not a measurement of a child's skills – it is a characteristic of a specific relationship between a learner and a tutor. In different situations and across different domains a child will have many different ZPDs that are dependent on the specific learning events and the unique contexts of those events [Mercer and Fisher, 1997].

Using the concept of the ZPD, Mercer and Fisher [1997] propose the following classification of tasks:

- Tasks that lie beyond the child's ZPD and which the child is unable to accomplish, even with help.
- Tasks within the ZPD which the child can sometimes accomplish with the help of a skilled person. Sometimes the child will complete the task without help and sometimes with some help but making some independent decisions.
- Tasks which fall below the ZPD and which the child can accomplish without assistance most of the time.

A fundamental implication of instruction in the ZPD is that it focuses on skills that the child must still acquire and not on tasks that involve things the child can already do.

4.2.3.3 Scaffolding

Scaffolding is a type of assistance to help children accomplish a task or goal that they would not have been able to do on their own. Children learn the most when the skilled partner can find the balance between helping and demanding progress [Berk, 2000]. Its aim is to bring children closer to a state of competence so that they will eventually be able to complete the task by themselves. Scaffolding is always directed at the development of a new skill or understanding and can only be regarded as true scaffolding if it leads to the child's successful accomplishment of the goal. Ideally the child should thereafter be able to accomplish a similar task independently [Mercer and Fisher, 1997].

Vygotsky saw the child as a learner and the adult as the instructor. Adults naturally know how to provide scaffolding and are better at providing scaffolding than children who are more competent at the relevant task than the learner [Thornton, 2002]. Today it is generally accepted that peer tutoring can also be a successful way to teach children and that more competent peers can guide learners through their zones of proximal development [Meece, 2002].

Designers of technology are thus creators of artefacts that may become elements of children's cognition. Context applies at different levels:

P13 Designers must know the cultural context of their intended users.

P14 Designers must acknowledge their own context and how that may consciously or subconsciously influence their design practice.

P15 They must consider the specific learning or entertainment goals of the product and how these goals fit the context of different kinds of users.

P16 Computer-based tasks for children should always be embedded in scenarios that children can relate to. These scenarios are important elements of the context of use.

P17 If a product is aimed at children from a variety of cultures designers may settle on one generic scenario, but it may be difficult to find one that all children can relate to.

Alternatively:

P18 In the same way as some applications allow users to pick a language of choice, children may be given a choice of scenarios.

Technology can play two different roles in the ZPD: it can act as skilled tutor that helps the novice towards skill development or better understanding, or it can be a tool that a human expert can use to help the novice. The first view offers quite a challenge to designers.

P19 The software should assess children's level of understanding of a concept or their competence in a skill and so determine their ZPD for this specific concept or skill. (This can be done by giving them random tasks of different levels of difficulty. Then present them with examples that are below their ZPD in order to build their confidence. This may be tricky as children may become bored if not challenged. Then move to problems that are just beyond their capacity and provide scaffolding where they need it. When a child succeeds with the help of scaffolding, give a similar task without scaffolding at first to determine if the relevant skill has been acquired. When this has been achieved the application may provide a task that falls beyond their ZPD.)

Groups of learners or children of the same age do not necessarily have the same ZPD, so technology must be designed in a way to:

P20 Determine each user's individual ZPD and then use that knowledge to direct further action.

P21 Make sure the application and its user share the required common knowledge. This shared knowledge will determine the tutor's choice of scaffolding.

For children to maintain motivation and persistence they must have the opportunity to successfully negotiate learning tasks – if they repeatedly fail at a task they will lose interest:

P22 Present children with tasks that they are capable of performing and that give them the opportunity to practice newly acquired skills. Then give children challenging tasks that are just beyond their reach, providing supportive scaffolding. This will enable them to move to a next level of understanding or skill.

4.2.4 Case's Neo-Piagetian Theory

4.2.4.1 The Basic Idea

Robbie Case was born in Ontario, Canada in 1944 and died at the age of 55 in 2000. His theory developed out of the broader category of 'Information processing' theories whereby the mind is regarded as a complex symbol manipulation system [Meece, 2002]. These theorists see the mind, metaphorically, as an information processing device that encodes information into symbols in working memory (using information extracted from long-term memory) and transforms it into knowledge [Case, 1992b]. Different internal processes manipulate this information through *recoding* (revising its symbolic structure to make it more effective) and *decoding* (interpreting its meaning using existing knowledge) [Meece, 2002]. These operations prepare the information for use in problem solving and making sense of the world and we can compare them to Piaget's concepts of accommodation and assimilation.

While Piaget regarded logical structures as the basis for cognition, Information processing theorists and consequently the neo-Piagetians, analysed cognition through a set of information-processing structures. They attributed development to increases in information processing capacity and believed that there is an upper bound to the level of structure that children can construct at any age. This upper bound depends on the size of working memory and the speed with which they can execute basic operations in working memory [Case, 1992b]. Case identified knowledge and control structures that are more specific than Piaget's system-wide logical structures [Case, 1992a]. These structures transpire in the child's mind as categories, event scripts, strategies, rules and plans [Fischer and Immordino-Yang, 2002]. 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 [Case, 1992b]. Case specifically attempted to explain in detail how individual structures are modified and provided a sequence of structural development. (I discuss this sequence in the next section.)

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. He recognised domain-specific developmental changes that are influenced by the nature of the tasks and children's varying experience [Meece, 2002]. 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 [Case, 1992a]. 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.

Although the neo-Piagetians place strong emphasis on the variability in the way children learn and develop, they still support the idea of stages of development.

4.2.4.2 Case's Stages of Development

In Case's reformulation of Piaget's stages each stage involves a distinct type of cognitive structure as described below [Case, 1992c]:

- Sensorimotor stage (0 to 18 months): In infancy the distinctive cognitive structures involve sensory input and physical action. By the end of this stage children should be able to solve sensorimotor problems that require them to identify several different subgoals that will lead them to obtaining a larger goal.
- Interrelational stage (1 to 5 years): Early childhood is characterised by internal representations of events and actions. At the end of this stage a child will typically be able to solve problems that involve reversibility or relations between aspects of a problem.
- Dimensional stage (5 to 11 years): In middle childhood the defining structures are simple transformations of representations. Children now start to coordinate structures for dealing with more than one dimension of a problem. They learn to solve problems that require evaluation of two different aspects of a problem situation in order to reach a solution.
- Vectoral stage (11 to 19 years): In adolescence the transformations of representation is complex.

According to Case's theory, children younger than six years have conceptual structures that focus on one dimension of a task or situation [Berk, 2000]. When told a story they can only follow a single story line. From around six years of age the central conceptual structures can coordinate two dimensions and they will be able to combine two storylines into one plot. Only from around age nine can they handle multiple dimensions.

In Case's theory transition within a stage depends on the growth of working memory and on the learning of more complex executive processes. Each level requires different executive control structures to deal with different kinds of tasks (for example, telling time or reading a music score). These executive control structures are different from the central conceptual structure that, when acquired, supports performance in a whole range of tasks [Case, 1992a].

Each of the four stages of development has three substages, namely [Case, 1992a; Case, 1992b]:

- The unifocal stage: In this substage children use two existing operations to form a new class of operations for dealing with familiar problems, but they apply the two operations independently.
- The bifocal stage: As working memory grows and children practice the new operations, they become capable of two such operations in sequence.
- The elaborated stage: Working memory grows further and with more practice children can integrate two operations into one coherent system. These now become the basic operations of the first substage of the next stage of development. Transition from one higher-level stage to the next depends on the integration of two different structures into one more abstract central conceptual structure [Case and Okamoto, 1996].

4.2.4.3 The Role of Working Memory

It is clear from the discussion so far that working memory plays an important role in Case's theory. He identified three aspects that influence the development of working memory capacity [Meece, 2002]:

- Brain development: Children's biological development places a ceiling on their level of cognitive development, meaning that they cannot develop beyond the upper limit of their processing capacity.
- Practice with schemes and automatisation: Repeated use of mental schemes or mental strategies makes their application automatic and this frees up working memory for the formulation of new schemes.
- Formation of central conceptual structures: When the schemes of one stage are sufficiently automatic, children use the newly available working memory to combine them into central conceptual structures. These are thus networks of concepts and relations that children can apply to a wide range of situations.

To summarise: as children develop, their working memory improves and consequently their capacity to mentally represent and manipulate pieces of information, or aspects of a problem, increases. They develop because their mental capacity allows them to practice with strategies and structures and constructively reorganise their thinking by progressively integrating existing knowledge structures into more advanced ones [Berk, 2000].

When the goal of a computer-based activity is to help a child acquire a new skill, Case's theory suggests that:

P23 Designers should:

- identify all the underlying operations that a child will use when learning the new skill,
- determine whether the child can perform these operations,
- find out if the child has the mental capacity for the new skill,
- present the child with problems that require the use of two operations independently, and then in succession, and
- present the child with activities that facilitate merging of the two operations into one that forms part of the new skill.

When the child acquires an operation or a skill, designers should:

P24 Present opportunities to practice a skill until it becomes automatic.

Once an operation becomes automatic some working memory becomes available for other operations. Since working memory is so important in development, designers should:

P25 Strive to relieve a child's working memory of extra processing that may prevent them from moving forward with the coordination of knowledge structures. For example, interpreting and navigating the user interface must require as little working memory capacity as possible.

P26 Children in the same age group may have different upper bounds of memory capacity and support should be adaptable to this variation.

P27 Children of five to eight are in the dimensional stage where they can start to coordinate structures for dealing with more than one aspect of a situation[5].

P28 Designers should acknowledge the culture or sub-culture of the intended user and identify particular problems that are important in that culture and the tools typically used to solve that kind of problem. (For example, 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 of information processing structures for 'eating in a restaurant'.)

Whether a child is capable of a particular activity depends on the information processing structures that they have available that relate to that activity:

P29 Designers should not assume that if a child can solve a specific kind of problem in one domain that they can transfer that skill to a different domain. (Some children may, for example, make the connection between music timing and mathematical fractions, but for others will need explicit instruction about the link before they grasp it.)

4.2.5 Fischer's Dynamic Skills Theory

4.2.5.1 The Basic Idea

The main proponent of the contemporary dynamic skills theory is Kurt W. Fischer, currently a Charles Bigelow Professor of Human Development & Psychology and Director of the Mind, Brain, and Education program at the Harvard Graduate School of Education. Over the past thirty years Fischer collaborated with many researchers and academics to refine and formulate this theory [Fischer and Bidell, 2006; Fischer and Yan, 2002; Fischer and Corrigan, 1981; Fischer and Granott, 1995; Fischer and Immordino-Yang, 2002; Fischer and Silvern, 1996]. Fischer describes his theory as a toolkit of concepts and methods that can be used to investigate changes related to human development, learning, context, and emotion [Fischer and Yan, 2002].

Dynamic skills development, also referred to as dynamic constructivism, integrates the results of many different theoretical viewpoints into a theory of nonlinear development [Fischer and Immordino-Yang, 2002]. It combines three ideas that were previously regarded as irreconcilable or contradictory, namely:

---

[5] In I Spy Spooky Mansion [Scholastic, 2002], for example, they need this competence: The basic task is to find items listed in a riddle and click on them, but the child also knows that by solving a number of riddles he works towards a different goal, namely to get a puzzle piece. Another example in the same application is when a child has to find three bats in a room. To be successful the child should realise that the word 'bat' has different meanings and should have both meanings in mind when searching for the images.

- independent skills development in different domains,
- similarity of development across domains, and
- a universal framework or scale for cognitive development.

The basic units of analysis in this theory are skills. *Skill* is a concept that includes both person and environment [Fischer and Bidell, 2006]. It is task-specific, context-specific and dependent on factors such as emotion, memory, culture, experience and biological maturation. Still, Fischer acknowledges that many of the classic principles of cognitive science and hence, cognitive skill development, are applicable across a range of knowledge domains. Fischer and Immordino-Yang [2002] do, however, not support the idea of a single, overarching mental structure, saying that 'development and learning occur along many parallel, independent strands that have similar properties even though they are from separate cognitive structures' (p. 2). Dynamic skills theory combines independence across domains and the accompanying variability in development with consistent order of development within a domain.

In a typical task children use different skills and to succeed they need to differentiate and coordinate these co-occurring competences [Fischer and Immordino-Yang, 2002]. The process of coordination and differentiation, which is influenced by the social and emotional context, may lead to the formation of new skills so that next time the task will be easier to perform. It may take several attempts at the task before a child acquires the new skill and sometimes this will only be possible with assistance. Coordination and differentiation can also occur between skills from different task domains. For example, storytelling involves skills of pretending or imagining, verbalising and ordering of ideas. Through constructive generalisation and repeated rebuilding, a skill that begins as task or context-specific can gradually be extended to other contexts [Fischer and Immordino-Yang, 2002]. Fischer uses a web metaphor to explain the skill development process.

4.2.5.2 Constructive Webs

In contrast to earlier views of development as a ladder of stages that go across domains, dynamic skills theory describes development as a web of strands which represent skills in different domains [Fischer and Bidell, 2006; Fischer and Yan, 2002; Fischer and Immordino-Yang, 2002]. A strand forks when one skill splits into two independent skills and two strands come together when two skills combine to form a single more general skill. In the web there will be separate but parallel strands for the same skill in two different domains, and these can at some point in development merge to form one skill across both domains. A skill does not develop synchronously in all the domains where it is pertinent, but the sequence of milestones for acquiring a specific skill is fixed within a domain. Every person's web of development is unique.

The web metaphor also reflects the fact the people act on multiple dimensions at the same time and different aspects of development may occur concurrently along different trails [Fischer and Immordino-Yang, 2002]. Two or more children can also collaborate in the construction of their webs [Fischer and Bidell, 2006].

The representation of a developmental web in Figure 4.2 shows two domains, each starting with two strands of development. The highlighted strand shows how development of a skill progresses in a specific order, with the skill splitting at milestones 1, 2 and 4. Strands from the two domains or different strands from one domain can come together to continue as a single strand.

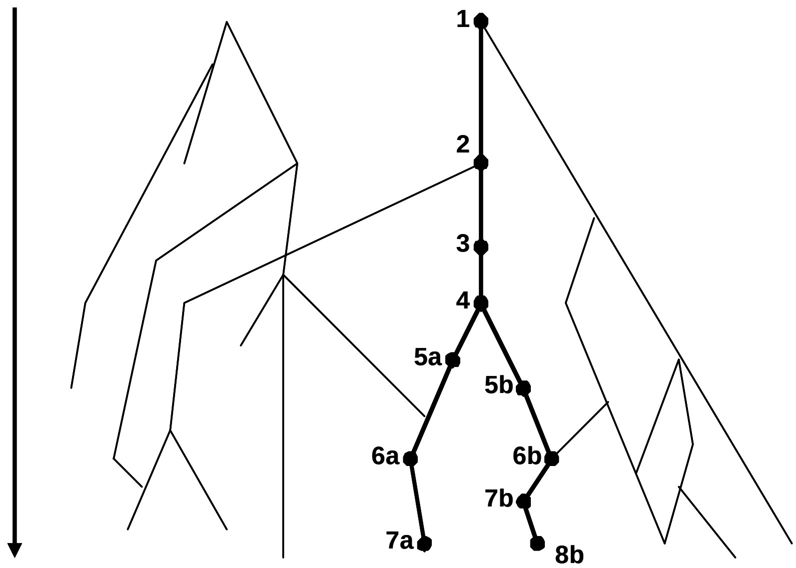


Figure 4.2 A developmental web (from Fischer and Immordino-Yang [2002])

4.2.5.3 Variation in Development

Dynamic skill theory supports the idea that development depends on context – the culture within which learning takes place, the child's emotional state, the level of biological maturation and the social aspects of the situation. It emphasises the role of support, calling the variation between supported an unsupported conditions the *developmental range*. Fischer describes two upper limits of development, namely the functional level and the optimal level [Fischer and Bidell, 2006; Fischer and Immordino-Yang, 2002]. A child can reach the functional level without any support, but to achieve the optimal level requires explicit support.

According to dynamic skill theory, development occurs in definite stages only when there is considerable support [Fischer and Bidell, 2006]. For example, schools cover specific learning domains such as literacy and mathematics and provide structured support in these domains. Clear stages of development are noticeable in these domains and there is a strong association with age. Optimal skills thus develop in a consistent series of stages, while functional skills (which children acquire without support) do not show a stage-like pattern.

Fischer and Immordino-Yang [2002] attribute stages to the 'dynamics of growth within and across strands in the developmental web' (p. 14) and not to universal cognitive structures. They found evidence of at least ten levels of development that can be linked to specific ages. Fischer later added three levels in early infancy, where it is more difficult to find accurate data on development and organised the resulting thirteen levels of development into the *dynamic skills framework*.

4.2.5.4 The Dynamic Skills Framework

The dynamic skills framework consists of four tiers of development, namely [Fischer and Bidell, 2006; Fischer and Immordino-Yang, 2002]:

- reflex actions (0 to 3 months),
- sensorimotor actions (3 months to 2 years),
- representations (2 years to approximately 11 years) and
- abstractions (11 to 25 years).

Within each tier there are four cycles or levels of reorganisation of skills. The fourth level is also the first level of the next tier. Reorganisation involves coordination and differentiation of skills to construct a more complex skill on a next level. Skills on the first level of a tier involve single units (i.e. single actions, single representations or single abstractions). On the second level, pairs of units are coordinated into mappings and on the third level several aspects of two units are related to form a system. At the fourth level the system of units are integrated to form a new skill which will be a single unit of the next tier.

The ages associated with the skill levels apply to optimal levels that a child can only perform with support [Fischer and Bidell, 2006]. Fischer and Bidell admit that these ages may be different for other social or cultural groups as they only considered middle-class American or European children in their experiments.

The levels on the scale are not associated with one psychological structure across domains. People use different skill structures in different situations but the resulting skills fit a common scale across tasks and domains. The framework reflects the theory's two-sided view of development as, on the one hand, 'an overarching set of large-scale changes' and on the other, characterised by the 'incremental, daily, minute-to-minute dynamics' [Fischer and Immordino-Yang, 2002]. There is thus order and variation at the same time.

Following dynamic skills theory, designers would:

P30 Support independent development of skills in different domains, while at the same time, considering how a skill is applicable across domains. 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 then make the link between the two contexts explicit.

P31 to P35 refer to the view of dynamic skills theory that skills are task specific, context-specific, dependent on emotion, culture, experience and biological maturation.

P31 A task or activity chosen to develop a skill must be one that can be naturally associated with that skill.

P32 If a child has mastered a skill in a specific context a designer cannot assume that the child will be able to apply the skill in a different context. Designers should not assume that children will transfer a skill to the real-life or school context and need to make the connection explicit

P33 A child's emotional state influences his or her skill acquisition. A designer does not have control over factors outside the game environment that may influence the child's emotional state, but can use game elements to evoke emotions that may enhance skill development.

P34 If a product is aimed at children from different cultural groups the designer should investigate how these cultures use or teach the skills that the product will support.

P35 Know what the minimum requirements with regard to biological maturation are when designing for young children. Do not expect them to perform actions that they are physically not yet capable of.

P36 A typical task may require a variety of skills which must be differentiated and coordinated. Designers should identify all the skills involved and support differentiation and integration of these skills in their design.

Designers can use the web metaphor of Fischer's theory to facilitate their design process.

P37 Knowing the specific sequences through which the skills that they want to address develop, will be of great value for designers. They can model all the domains, skills and tasks that are relevant to their product in one or more constructive webs and use this as a starting point for the design.

P38 Technology should provide the kind of support that will allow a child to develop skills optimally rather than functionally.

4.3 Domains of Development

4.3.1 Introduction

In this section I move my discussion of children's cognitive development from the overarching views of developmental theories to the examination of specific skill domains. From the age of five to eight, development in some domains is drastic. It is during these years that most children learn to read and write and attain mathematical skills such as basic arithmetic. Literacy and mathematics rely on more general skills such as memory, thought, knowledge representation and problem-solving, which all show remarkable growth in this period. I limit my discussion to these fundamental skills, all of which lend themselves to support through technology.

As with the theories of development it is impossible, within the scope of this chapter, to report on all research that was ever done on the development of skills in the chosen domains. In each domain I focus on the work of a selection of key researchers.

4.3.2 Literacy

4.3.2.1 Reading Skills

Experiences throughout the early childhood years influence a child's literacy development, which means that their literacy development will be variable, depending on their background and home experience [IRA and NAEYC, 1998]. Through their initial experiences and interactions with adults, children begin to read words by learning letter-sound relations and the alphabetic system. They gradually consolidate this knowledge into patterns that will eventually help with fluency in reading and writing.

Reading requires basic skills such as learning to look at a page from the top to the bottom, viewing a page from left to right and opening a book from the right side. More obviously, also learning to say the alphabet, recognizing letters and the phonetic sounds associated with each letter and combination of letters, identifying beginning and ending sounds, and recognizing rhyming words. Listening skills are vital for developing reading skills and include understanding the meaning of a story, recognising the sequence in a story, the main theme in a story, the cause and effect of events in a story, and learning to anticipate the possible outcome of a story [IRA and NAEYC, 1998].

To create developmentally appropriate programmes for literacy development, we must understand the continuum of reading and writing development and recognize when variation is within the typical range. Below I give a summary of the expected literacy skills of a typical grade R (Kindergarten) child and a typical grade 1 child as listed by IRA and NAEYC [1998].

A grade R[6] child (5 to 6 years of age) should:

- Enjoy being read to and retell simple narrative stories or informational texts.
- Use descriptive language to explain and explore.
- Recognise letters and letter-sound matches.
- Show familiarity with rhyming and beginning sounds.
- Understand left-to-right and top-to-bottom orientation and familiar concepts of print.
- Match spoken words with written ones.
- Begin to write letters of the alphabet and some high-frequency words.

A grade 1 child (6 to 7 years of age) should:

- Read and retell familiar stories.
- Use strategies (rereading, predicting, questioning, contextualising) when comprehension breaks down.

[6] In South Africa the school year before grade 1 is called grade R. This is similar to Kindergarten in other countries. I use both terms in this thesis.

- Use reading and writing for various purposes on their own initiative.
- Orally read with reasonable fluency.
- Use letter-sound associations, word parts, and context to identify new words.
- Identify an increasing number of words by sight.
- Sound out and represent all substantial sounds in spelling a word.
- Write about topics that are personally meaningful.
- Attempt to use some punctuation and capitalization.

4.3.2.2 Learning to Read

The classic view of learning to read accepts one basic developmental pathway that requires integration of skills in different domains [Knight and Fischer, 1992]. These are:

- Visual graphic skills such as perception and analysis of letters and words and discrimination of graphic forms.
- Phonological skills such as perception and analysis of sounds related to language such as rhymes, syllables and phonemes. Byrne and Fielding-Barnsley [1991] believe that from around the age of five, children can begin to understand the relationship between letters and sounds and develop a conscious awareness that speech is composed of identifiable units, namely words, syllables and sounds.
- Semantics skills, which include basic language competence and understanding the meaning of words.

Knight and Fischer [1992] contest the classical view, saying that the existence of major reading problems suggests that children may follow different routes when beginning to read. They present a sequence of skills for reading single words that correspond to the classical sequence, but found evidence of two alternative sequences used often by children with reading problems, and sometimes by some normal and good readers. Figure 4.3 shows the three pathways.

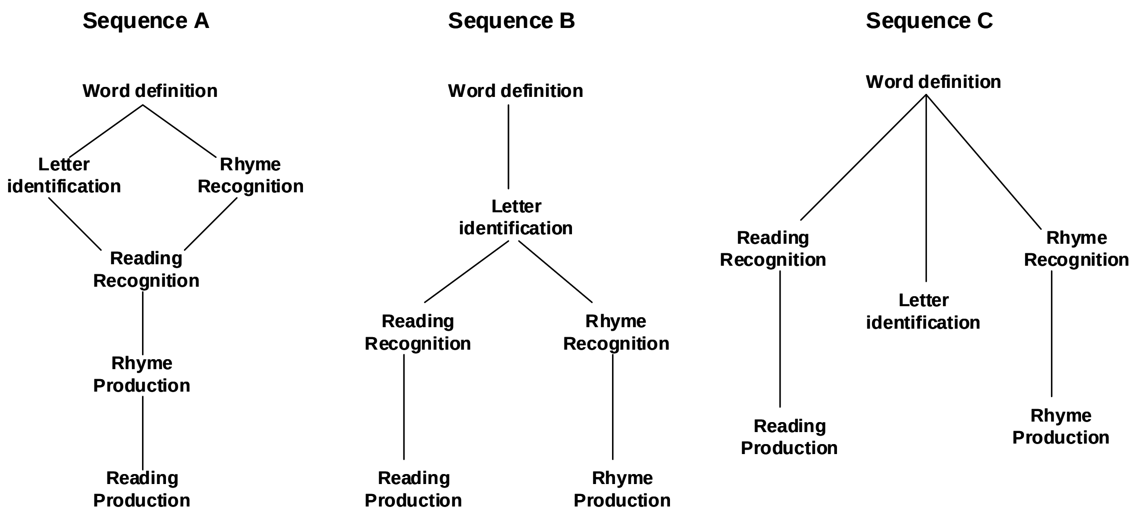


Figure 4.3 Three pathways for learning to read single words (from Fischer and Immordino-Yang [2002])

Pathway A involves integration of visual graphic and phonological skills. In sequence B phonological skills are used differently – children do not use analysis of word sounds as a supporting strategy. Children who follow pathway B typically display poor rhyming skills. Some normal readers show this sequence on selected words. In sequence C, there is no integration of letter identification, rhyming and reading skills and children who use it display general deficiencies in letter identification and rhyming. The general conclusion from Knight and Fischer's study is that children normally follow the classical pathway, but when they lack the ability to integrate skills from different domains, reading development may develop differently.

Before children reach the point where they apply the skills in these sequences they need to understand how written words and letters relate to spoken language. There are two forms of representation in written language: firstly, written words and sentences represent spoken meaningful words and sentences and secondly, alphabetical letters represent meaningless sounds. Bryant [1993] reasons that children need different kinds of knowledge about language to understand these two levels of representation. When learning to read, children must learn that words on paper have meaning and that they represent specific utterances. Before they can do this they need to understand that spoken language is made up of words and sentences. On a second level, children must learn that individual letters represent sounds and that these can be connected to form sequences of letters that make up words [Bryant, 1993].

So, firstly children have to understand how the language system works, that is, what the forms of the units of language are. In this regard Bryant [1993] reports on research by Ferreiro and colleagues who based their work on Piaget's theory of cognitive development. Their core idea is that instead of just teaching children to read, we should allow them to construct reading for themselves. Their research shows that adults generally underestimate how hard it is for children to learn that groups of letters represent words, that any word you speak can be written down and that alphabet letters represent different sounds. They divide children's learning to read into three stages:

1. The pre-syllabic stage: In this stage they do not realise that letters represent sounds, that groups of letters represent words or that words must be written down in a specific order to represent a meaningful spoken sentence.
2. The syllabic stage: Here they begin to understand that a written word represents a spoken word, but they do not yet understand how individual letters relate to different sounds. They now get the idea that groups of letters form words but think that each syllable is denoted by a letter. They therefore realise that longer words have more letters, but they initially have no idea which letters are associated with which sounds. Later in this stage children begin to see a phonetic relationship between letters and sounds, but they still regard the syllable as the smallest phonological unit.
3. The alphabetic stage: In this stage children finally learn that letters represent phonemes.

Ferreiro, et al. [1982; as cited by Bryant 1993] focus on the pre-reading period and do not say much about the actual reading process. In the next section I discuss how pre-reading experiences support the development of reading skills.

4.3.2.3 The Influence of Pre-Reading Experiences on Learning to Read

Bryant [1993] investigates how children learn how to use the units of language when starting to read. There are different opinions in this regard. One school of thought sees phonological skills as the most important requirement for learning to read. Within this group there is disagreement about the effect of children's pre-reading experiences on their learning to read – some follow a discontinuity view and some a continuity view. The discontinuity supporters believe that the phonological skills acquired when learning to read have no connection with their early phonological and reading experiences (in other words, reading depends on artificial skills which children can only acquire by being taught to read). According to the continuity view early experience has a big impact on the ease with which children learn to read and reading skills develop out of skills acquired earlier. Bryant calls this the continuity/discontinuity dispute. He discusses several research projects that support the continuity view, but then also gives good evidence for the discontinuity argument.

The discontinuity argument promotes a manner of teaching that does not rely in any way on natural development. It only focuses on the process of teaching children to read and regards children's pre-school phonological skills as irrelevant to reading. Despite the convincing evidence in favour of the discontinuity view, Bryant [1993] reviews the form of children's early phonological skills to come to the conclusion that their ability to detect rhyming words and alliteration will later help them to learn to read and spell. The ability to detect rhymes is not as refined as the ability to detect phonemes (units of sound), but there is evidence for a developmental relation between detecting rhyme and alliteration and acquiring reading skills.

Bryant discusses an experiment to test children's rhyme and phoneme detection skills. Subjects are verbally given four words at a time and are told to point out the one that does not rhyme or that does not start (or end) with the same letter. Children can detect the non-rhyming words long before they learn to read. Five-year-olds who cannot yet read also do well in detecting the word beginning with a different phoneme, but they find it difficult to find the word that ends with a different letter. Bryant suggests that to determine whether early rhyming skills relate to later reading skills, we need to know whether only the grapheme-phoneme correspondence is important for learning to read or whether children also need to learn about the relations between sequences of letters and strings of phonemes that are not whole words. Goswami [1986, 1988a, 1988b; as cited by Bryant, 1993] has shown that children's ability to detect rhyming words has a direct influence on their learning to pronounce and spell new words. When a child is confronted with a difficult word and is then told how the word is pronounced and what it means (for example, 'peak'), they will later be able to read rhyming words with similar spelling patterns (for example, 'leak' and 'weak'). It turns out that children's preschool experience with rhyme and alliteration are predictors of their later reading success. Training in rhyme helps children to learn to read.

So far my discussion of pre-reading experience focussed on experience with letters and words. Early experience with extended written or printed text and books can also be valuable. The importance of being read to aloud has been emphasised by numerous researchers [IRA and NAEYC, 1998; Van Ijzendoorn and Pellegrini, 1995; Van Kleeck, 1990]. Children should be active participants, talking about the pictures, retelling the story, and answering questions about the story. According to Pappas [1995; as cited in IRA and NAEYC, 1998] children who have not been read to will later have problems understanding the structure of narrative and other texts. Repeated readings of the same text reinforce the language and vocabulary of the text and make children familiar with the structure of different literary genres and the conventions of written language.

4.3.2.4 Cues Children Use When Learning to Read

The group of people who support the continuity argument discussed above are divided into two groups who have different views on how children learn to read [Bryant, 1993]. One group believes there is a link between early phonological skills and reading and the other regard linguistic knowledge (such as the rules of grammar) as the most important factor. According to the latter group children do not decipher text word-by-word on the basis of letter-sound relationships. Instead they use the meaning of the sentence to help them guess the difficult words. While doing this they use three kinds of cues [Goodman 1967, 1982; as cited by Bryant, 1993]:

1. Grapho-phonic cues – the word they choose will typically start with the same letter as the one written down.
2. Syntactic cues – they will guess a verb where a verb is syntactically expected.
3. Semantic cues – they guess a word that would make sense in the context of the sentence.

Tunmer [1989; as cited by Bryant, 1993] supports this by showing how children use their syntactic and semantic knowledge to help them read words that they would not be able to read in isolation. This indicates that sensitivity to context is very important in learning to read. Rego [1991; as cited by Bryant, 1993] has shown that children's early semantic and syntactic skills have a positive effect on learning to read.

Recent research by Nunes and Bryant [2000] investigated the effect of teaching children about morphemes (meaningful units of language that cannot be further divided, for example, 'in', 'come' and 'ing' in 'incoming') on their reading ability. They found that even a modest amount of instruction can have a positive effect on children's progress in learning to read.

4.3.2.5 Learning to Spell

When children start to write they use phonetic spelling, writing for example, 'brd' instead of 'bird'. In early writing experiences children should be allowed to use non-conventional forms of writing. Over time they can move to the conventional forms of writing [IRA and NAEYC, 1998].

According to Rittle-Johnson and Siegler [1999], the general view of how children learn to spell is that they start by matching the initial sound of the word to a letter or, sometimes, the initial and final sounds. Next they match every sound in a word to an appropriate letter. Later, usually from the second or third grade, they begin to use spelling rules, likeness to other words, knowledge about root words and sound patterns. This view focuses on solving spelling problems and therefore does not recognise simple retrieval from memory as an important strategy in learning to spell [Rittle-Johnson and Siegler, 1999].

Rittle-Johnson and Siegler [1999] believe that a complete model of spelling development should describe how retrieval can improve spelling ability. They define retrieval as 'the rapid, automatic or close to automatic, activation of spelling. This activation can rely on both word and sub word (e.g., syllable, letter) information, including rapid phonological and morphological processing. It is not under conscious control and does not involve explicit application of rules.' (p. 334). Strategies other than retrieval are backup strategies and they require conscious application of specific methods. Children can combine retrieval with one or more backup strategy, even when spelling a single word.

Rittle-Johnson and Siegler [1999] conducted an experiment to test whether Siegler's strategy choice model for arithmetic is applicable to strategy choices in spelling. Their subjects were children in the first or second grade, with mean age six years and ten months. They found that children use various strategies in learning to spell right from the beginning. First and second graders prefer the same strategies, namely retrieval first and then sounding out. These two strategies were used 80% of the time, but in general all the children combined retrieval and sounding out with rule use and visual checking. They found that strategy choice is not dependent on level of competence and that there is considerable variability in individual children's choice of strategy.

Based on the results of their experiment, Rittle-Johnson and Siegler identified specific factors that improve speed and accuracy in spelling from the first to the second grade:

- Increased use of retrieval as the fastest and most accurate strategy.
- Reduced use of the slowest and least accurate backup strategy, namely sounding out.
- Increased use of faster and more accurate backup strategies such as drawing analogies and applying rules.
- Faster and more accurate execution of all strategies.

4.3.2.6 Cultural Issues

One of the first skills in learning to read is recognising the first letter of a word. The content of many books and other media aimed at five year old children support development of this skill. Design of material to support letter recognition should make it easy for children to recognise objects associated with letters. Carlson [1996] gives examples from alphabet books that are clearly problematic: N is associated with a 'nightingale' (which young children will just recognise as a 'bird'), Y with 'youngsters' and A with 'armchair'. Objects should be chosen so that a child can easily identify it and that they will most probably name with the intended word. The beginning letter should stand out clearly so that the association between the particular letter of the alphabet and the object is obvious. Cultural differences are important in this regard, especially when children are confronted with material that is not in their home language.

According to the IRA and NAEYC [1998] report on learning to read and write, children can function in more than one cultural context simultaneously, and it is not necessary to base expectations for children only on their home culture and language if they have to develop optimally. Children can learn in English as a second language without giving up their home language [NAEYC, 1995]. This will help children to function well in the broader society and enhance their social interactions with people of different backgrounds. Bilingualism improves cognitive development rather than interferes with it. Full proficiency in the first language do however support the learning of a second language, so children and their parents and teachers should promote the use of the child’s first language even if he or she is taught in a second language [NAEYC, 1995].

Following the continuity view discussed above I accept that pre-reading experience plays an important role in learning to read:

P39 Learning to read requires integration of visual graphic skills, phonological skills and semantic skills. Technology aimed at teaching children to read should therefore address all of these skills.

P40 Tasks for developing visual graphic skills should include activities that teach children to distinguish between different letter forms. For example, show a big ‘d’ and a mixed bunch of letters including several b’s and d’s and ask the child to click or drag the letters that are identical to the given one.

P41 Phonological skills will require audio-enabled applications so that the child can listen to different words and, for example, identify similar beginning or ending sounds. Children often confuse the p-sound and the b-sound, the t and the d, and the m and the n.

P42 Activities involving rhyme are also suitable for developing phonological skills.

P43 To support development of semantic skills activities should aim to improve children’s vocabulary.

P44 When children become competent in the separate skills, they can be presented with activities that require integration of all three.

P45 For a typical child beginning to read, Fischer’s main development pathway suggests the following sequence of activities for learning to read words (most of these require audio output and speech recognition technology):

- Improve the children’s vocabulary so that they understand the meaning of words presented to them.
- Present the child with activities that will develop the skill to *recognise* rhyme words amongst words presented to them in audio format.
- Provide them with examples of sounding out words letter-by-letter using audio combined with visual cues.
- Help them recognise the same word in different contexts.
- Present children with activities to develop the skill to *produce* rhyme words. This will require speech recognition technology.
- Present them with reading tasks. When they struggle to read the word, help them by providing semantic cues, giving words that rhyme with the particular word or, if they are still unsuccessful,

sounding the word out for them.

Computer based activities for learning to spell will at some point require children to type in the letters of a word on the keyboard or to pick letters from a screen display with the mouse.

P46 Initially children can be presented with different spelling options for a word given in audio format and they have to select the correct one. This will not be sufficient to teach them to write the words correctly by themselves.

P47 A spellchecker-like application can be used that, instead of just correcting incorrect spelling, takes children through a process of recognising and correcting their mistakes.

P48 Using analogies and spelling rules are more effective spelling strategies than sounding out, therefore activities can be structured in a way that will draw the child's attention to relevant spelling rules. Repeated demonstration of a rule will help the child to apply it.

P49 Examples of activities suitable for five to six-year-olds are:

- Pick the correct letter that is given in audio format.
- Pick words that rhyme with a word given in audio format (for pre-readers, display pictures with the words).
- Pick words that begin or end with the same letter as a given word.
- Match simple words with words given in audio format.
- Young children enjoy listening to stories – software that simulates being read to from a storybook can be appropriate.

P50 Examples of activities suitable for six to seven-year-olds:

- Create simple stories with characters and either a recorded voice-over for the story line or space to type the story.
- Tell a story and ask children questions about what was told (make sure the response expected from the child is age appropriate).
- Activities that teach them about punctuation.

4.3.3 Mathematics

4.3.3.1 Early Skills in Mathematics

The mathematics skills that are usually associated with five to six-year-old children include writing numbers, sequencing numbers, associating the number symbol with the number of objects, telling time, recognising and counting money, measuring height, weight and length and basic addition and subtraction [NAEYC, 2002]. The processes involved in mathematics are problem solving, reasoning, communication, making connections, and representation [NAEYC, 2002]. Children's use of these processes is crucial in learning mathematics skills.

The USA's National Council of Teachers of Mathematics has identified five major content areas in mathematics teaching, namely number and operations, geometry, measurement, algebra (including patterns), and data analysis [NAEYC, 2002]. These are interrelated – when children connect number to geometry by counting the sides of shapes or measuring the length of their classroom, they strengthen concepts from number and geometry and build knowledge about the applicability of mathematics to other subjects. Number and operations, geometry and measurement are particularly important for three to six-year-old children and should receive special attention. Activities related to patterns (the foundation for algebra) are also appropriate for this age group and will support later algebraic thinking as well as other concepts such as number and space.

NAEYC [2002] provides examples of typical mathematical achievements of three to six-year-old children in the different content areas. I give some of the upper bounds (i.e. for age six) in Table 4.2:

Table 4.2 Typical mathematical achievements of 3 to 6-year-olds

Content area	**Examples of typical knowledge or skills around age six**	**Example activities to support the development of these skills**
Number and operation	Count and produce collections up to 100 using groups of 10. Add or subtract using counting-based strategies.	Demonstrate counting in 10s, while making groups of 10 objects. Tell real-life stories involving numbers and problems, asking 'how many?' questions.
Geometry and spatial skills	Recognise and name a variety of 2-D and 3-D shapes in any orientation. Describes basic features of shapes.	Spot real life objects that are of the different shapes and name the shape, e.g. pyramids, dust bin.
	Make a picture by combining geometric shapes. Build, draw or follow simple maps of familiar places.	Encourage children to make pictures or models of familiar objects using only geometric shapes (paper, wood, etc.) Draw a map of the path from the bathroom, to the kitchen, to the bedroom, adding pictures of objects that appear along the path.
Measurement	Try out various processes and units for measurement and notice the different results.	Create situations to draw attention to the problem of measuring something with two different units (e.g. putting objects 'four shoes' apart, first using daddy's shoe and then a child's shoe).

Patterns (algebra)	Notice and discuss patterns in arithmetic (e.g. adding 1 to any number results in the next counting number).	Encourage children to look for patterns in the environment (e.g. number patterns on calendars).
Data analysis	Organise and display data through simple numerical representations such as bar graphs and counting the number in each group.	Demonstrate how to make simple bar graphs and using it to compare data.

4.3.3.2 Why Children Struggle With Arithmetic

Piaget suggested that if children cannot conserve number, any apparent ability (such as counting) is likely based on parrot-style learning, and that teachers should mistrust such abilities [Donaldson, 1978]. Only when children understand the idea of number conservation, Piaget believed, will they be ready to learn addition and subtraction. New evidence, however, shows that children's understanding of number concepts is much better than what Piaget thought [Hughes, 1991]. Hughes [1991] argues that the problem with Piaget's theory lies in the nature of his conservation experiment (see paragraph 4.2.2.3 for a discussion of this experiment). When the adult spreads out one of the rows of objects, children aged five will say that the row now contains more objects, while from six or seven they will understand that the number does not change. Hughes reports on several studies that have shown that if the procedure is modified to make the change in the one row accidental or incidental, many five-year-old children also give the right answer. Gelman [1969; as cited by Hughes, 1991] used an alternative experiment that she calls a 'magic game' to show that children from three years understand invariance on small arrays of numbers, and that many three and four-year-old children understand the idea of addition and subtraction [Hughes, 1991]. Whereas Gelman's evidence is indirect in the sense that the children were not directly asked to perform addition and subtraction, Hughes [1991] carried out experiments where children aged three to five were given simple addition and subtraction problems. He found that children performed very well on problems that involved numbers below four and where the problems involved physical objects that they could observe being added or taken away. They also performed well when the problems were presented in hypothetical form. Over a quarter of the children could solve problems that involved numbers up to eight. Together with Gelman's findings, these provide evidence that children may have a coherent set of principles for reasoning about small numbers.

Given that preschool children can already reason coherently about number, Hughes [1991] now asks 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?'. Hughes [1991] proposes the explanation that there is no connection between our number words and the numbers that they represent. He refers to an early Indian system where the word for 'one' meant moon, that for 'two' meant 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.

A Queensland Aboriginal language has only two number words from which other numbers are constructed. 'Ganar' is one, 'burla' is two, 'burla-gana' is three and 'burla-burla' is four [Durkin, 1993]. Hindis, on the other hand, have to rote-learn number words for up to 100, or a complex set of rules to generate them. Other languages have different sequences of number words for counting different kinds of things and some languages have no number words. 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 and [Durkin, 1993]. Durkin believes that the interactions (in the context of social activity) between child and adult who have different understandings of number representation lead to children's achievement of the representation.

4.3.3.3 Three Worlds of Mathematics

In accordance with Hughes' theory, Griffin [2003] 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. She believes that competence in mathematics depends fundamentally on the development of relationships among these worlds. This corresponds to Hughes' idea that learners have to acquire skills in using translation procedures for moving between concrete situations and formal code. Together with Case, Griffin identified a set of conceptual structures that forms the basis of for learning arithmetic [Griffin, 2003]. Different subsets of these apply to four different age levels from four to ten. At four, children have two detached structures, namely a schema for comparing quantities and a schema for counting small sets of objects. They can apply only one of these structures at a time. At six, these structures merge to form a structure where numbers are associated with quantities and are used to compare two quantities. For them a small number indicates a little, while a large number means a lot. They also know that counting is like adding one continuously. They can answer questions about addition and subtraction, and can understand hypothetical statements. They can now do simple arithmetic in their heads without having to manipulate real objects. They are still limited to work with only one quantitative variable at a time. At eight the structure splits in two again, so that children can work with quantities along two variables (for example, tens and ones, hours and minutes, rands and cents). Only now can they mentally solve double-digit addition problems.

4.3.3.4 Learning to Write Arithmetic Code

In most countries children start formal schooling at the age of five or six. This is when they are first introduced to written arithmetical symbolism [Hughes, 1991]. Generally they use work books to learn arithmetic where they have to complete additions and subtractions 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 [Hughes, 1991]. When asked to write down the number of bricks displayed, only 38% of children from five to seven used conventional numerals. 45% drew the required number of bricks, others used vertical strokes or blob-like shapes and some drew the appropriate number of some object (for example, houses). When asked to write down a simple subtraction or addition sum, 69% of the children represented only the final number of bricks. 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 mathematics symbols they use in their workbooks can be used to represent quantities of real objects or the operations on these quantities.

The implications of the findings discussed above are that children need to learn that formal arithmetical code can be translated to concrete situations and they must learn the procedures for doing this translation. Hughes [1991] concludes his discussion with a description of a game that can be used to introduce arithmetical symbols to children in an appropriate way. He uses a number of identical containers with different numbers of objects (such as sweets) inside and shows the child how many sweets each contains. The containers are then closed and shuffled around and the child has to guess which has one object, two objects, and so on. When the child realises that it is difficult to keep track of the containers, the adult suggests they label the containers with numbers. Here the children follow different strategies. Some label the container with one object using the label '1', the container with two objects with '2', and so on. Others stick one label (any numeral) on the container with one object, any two labels on the container with two objects, and so on. Both methods help them to distinguish between the tins. The game can be extended to introduce addition and subtraction symbols. The child closes his eyes while the adult changes the number of objects in a container, leaving a 'message' to say what he has done. For example if he adds an object he places a label '+1' on the tin. With this game, Hughes has successfully introduced arithmetical symbolism to children as young as four [Hughes, 1991].

Mismatches between adults' and children's understandings of mathematical activities often cause adults to regard outcomes of children's behaviour on mathematical tasks as incorrect [Durkin, 1993]. Children easily confuse a representation of a hypothetical arithmetical problem and the actual content of the problem. Durkin illustrates this with the example where an adult gives a child a bag with ten buttons and then asks them to imagine that Jane has eight buttons and removes two to sew on a dress. The child must then work out how many buttons are in the bag. They cannot separate Jane's hypothetical bag from the bag that they have in hand and will count the buttons to get the answer ten.

Many mathematical terms have other meanings as well. I have already mentioned the possible confusion between words like 'to' and 'two'. Other examples of words that may be confusing are 'table', 'odd', 'even' and 'volume'. When using such terms when working with children, adults must make sure that the children are attaching the correct meaning to the word [Durkin, 1993]. In Durkin's words: 'Development in the representation of number, then, is not at its core a matter of increasingly sophisticated reasoning about abstract problems and rules by an isolated discoverer. It is a social process in which the learner has to discover what other people mean. Other people convey their meanings in complex and sometimes confusing ways'. (p.162).

4.3.3.5 Developing Strategies to Solve Mathematics Problems

Carr and Hettinger [2003] define strategies broadly as 'any method used to solve a mathematics problem'. Strategies are thus flexible and goal-oriented. In this section, I give an overview of Carr and Hettinger's discussion of mathematics strategy development, concentrating on strategies that can be used by children aged five to eight.

The initial addition and subtraction strategies that preschool and grade R children use are external representations of number by using fingers or counters. 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 is lined up next to each other 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, some children begin solving subtraction problems by turning the problem around to use addition to get to the answer.

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.

The variability in children's strategy use depends on factors such as brain maturation, different levels of understanding of mathematical concepts and procedures and experience. Carr and Hettinger [2003] also discusses the following factors that influence strategy development and use:

- Conceptual knowledge: Knowledge of mathematical concepts sometimes builds on knowledge of concepts learned before. For example, children use their knowledge of addition and subtraction as a basis for learning multiplication strategies.
- The semantic structure of the problem: The semantic structure of word problems particularly influences strategy choice.

- Working memory limitations: When adding numbers, the counting-on strategy requires children to keep two number lines in memory, whereas counting the combined set from 1 requires a single number line. If a child's working memory cannot accommodate the counting-on strategy he or she is forced to use a single line strategy. Word problems require children to read the problem, represent the problem mentally, find a strategy that suits the problem and apply the strategy to solve the problem. All of this use working memory. Since young children's working memory is not yet well-developed they find word problems difficult. The representation task consumes memory capacity so that they have difficulty in selecting a suitable strategy.
- The effects of context: Different schools and countries emphasise different strategies for solving mathematics problems. Flemish schools, for example, teach young children to memorise basic mathematics facts in favour of mental counting strategies. Teachers who prefer that children use a specific strategy should create opportunities where this strategy is used repeatedly in clusters. Stern [1992; as cited by Carr and Hettinger] has demonstrated that this is more effective than when the use of the desired strategy appears among problems that require other strategies.
- Fluency: If children practice using a strategy they become faster and more accurate in their application of the strategy. Improved fluency in strategy use makes more working memory available for other aspects of problem solving.
- Procedural knowledge: Children can only use a strategy successfully if they understand the procedures that make up the strategy.

Carr and Hettinger [2003] discuss three theoretical perspectives on strategy development, namely cognitive psychology, Piagetian constructivism and social constructivism. The basic idea that underlies the cognitive psychology perspective is that children's strategy use varies not only between children, but also within individual children. According to Carr and Hettinger, Siegler believes that strategy selection depends on children's confidence in their ability to retrieve an answer from memory, the characteristics of the problem, individual styles and more. Strategy use changes over time and better strategies are selected more often as the child sees his or her own successes and failures. New strategies develop gradually and existing strategies remain available as back-up.

According to the constructivist view, strategy development is a function of the development of mathematical knowledge – strategies emerge as abstract schemes about number. This happens through a process of 'perturbation' that occurs when the child notices discrepancies between the expected and the actual outcome. Constructivists also believe that children need to develop an internal representation of number before they can discard of manipulatives such as fingers and counters.

Social constructivists believe that the development of mathematics strategies cannot be detached from the context in which they come forth [Carr and Hettinger, 2003]. From this perspective, Geoffrey Saxe developed a framework for studying the development of mathematics. It includes four parameters, namely:

- Activity structure (is it the purchase of an item or a word problem, for example?).

- Social interaction (is a more expert learner available to provide scaffolding?).
- Conventions and artefacts (tools used to solve mathematical problems, such as fingers or money).
- Prior understanding.

Carr and Hettinger [2003] end their discussion of strategy development with suggestions on how to support healthy strategy development. To support the variability in children's strategy preferences, teachers 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 have to emphasise 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 when 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. Finally, children should be encouraged to apply their mathematical knowledge in real-world situations. Teachers and parents should look for opportunities where this can be done.

P51 Examples of skills of a six-year-old with computer-based activities to support them:

- Add or subtract using counting-based strategies. Let them decorate a screen-based cake according to instructions that require addition and subtraction to get the right number of cherries, et cetera.
- Recognise and name 2D and 3D shapes in any rotation. Show a picture of a combination of real-life objects and ask them to click on all the triangles or all the spheres, et cetera.
- Build, draw or follow simple maps of familiar places. Create a 3D on-screen environment through which they can travel using the mouse and keyboard. Ask them to move through it following a route shown on a 2D map.
- Organise and display data through simple numerical representations such as bar graphs and counting the number in each group. Let them play a searching game where they have to find, for example, ten of each of five objects. Display their progress in a bar chart. Each time they find an object the bar for that object moves up until all bars have reached maximum height.

From the discussion of children's problems with arithmetic, I derive the following guidelines:

P52 Acknowledge the fact that young children find it difficult to translate between the formal symbol system of mathematics and the quantities, operations and concepts they represent. That is, do not make assumptions about children's understanding of number and operation symbols.

P53 Design activities to gradually help children associate number symbols with the correct number of objects.

P54 Only introduce operator symbols such as + and – and their associated operations when children can use number symbols confidently.

P55 Take care to present activities in a way that will ensure that children interpret them correctly. For example, when children rely on audio instructions, make sure that they understand numbers and concepts that sound like other words in the language (e.g. table, odd, even, volume) correctly.

P56 Ask questions in a way that helps children interpret them correctly, keeping in mind how the context may influence their understanding and response .

P57 From six years of age children begin to understand hypothetical statements if they are presented carefully, but they may misinterpret hypothetical statements that contradict the real state of affairs. Only use hypothetical statements in activities aimed at children older than five and make sure the statements do not contradict aspects of the context.

P58 Tasks aimed at six year olds can combine counting and comparison of quantities and can even require that they do simple arithmetic mentally.

P59 Activities aimed at eight-year-olds can include operations on quantities that involve two variables, e.g. tens and ones, hours and minutes, rands and cents.

P60 From around eight years children can mentally solve double-digit addition problems.

P61 Do not assume that children younger than eight realise that the mathematics symbols they use in their workbooks at school represent quantities of real objects or the operations on these quantities.

P62 Children need to learn that formal arithmetical code can be translated to concrete situations and they must attain the procedures for doing this translation.

P63 By showing children a strategy in action and then repeatedly getting them to apply that strategy successfully, they will learn to use it automatically. For example, when adding two numbers beginning 'adders' will make two groups of objects and then start counting them all from one to get the sum. This is because they naturally prefer to use only one number line at a time as it requires less working memory capacity. Counting on from one of the numbers is a more effective strategy, but requires them to keep two number lines. Showing them to count on from one of the numbers will only work if their working memory capacity is sufficient. When they understand that strategy it should be easy to convince them that counting on from the bigger of the two numbers is an even better strategy.

P64 To support the variability in children's strategy preferences a program should be flexible in the strategies they require and provide opportunities for using different strategies to solve different kinds of problems.

P65 Provide children with as many views as possible on a specific problem, so that they become aware of connections between different strategies and operations.

P66 Successful strategy use requires conceptual understanding of the strategy. Teaching a child to use a strategy includes teaching them the underlying concepts. Children will only use strategies if they become aware of them and understand how and why they work.

P67 Practice and play with different kinds of mathematical problems lead to the development of more sophisticated strategy use.

P68 Encourage children to apply their mathematical knowledge in real-world situations.

4.3.4 Thought

4.3.4.1 Views on Thought

There are three well-known metaphors of children's thought [Wellman and Gelman, 1998]. According to the *child-as-adult* metaphor children have the same core conceptions and organise knowledge in the same way as adults. The *child-as-novice* metaphor proposes that children start off completely ignorant and develop by adding to their knowledge base. The *child-as-alien* metaphor describes children as having their own way of understanding that is completely different from that of adults.

Following the child-as-novice view, Piaget [1953] believed that children acquire more complex knowledge structures as they go through certain stages of development. He also believed that these knowledge structures are domain general and that competency in logic determines how well children apply the knowledge structures across domains. Donaldson [1978] showed that Piaget underestimated children's cognitive abilities across a number of domains. A major problem with Piaget's results is that when his experiments are conducted in contexts that are meaningful and familiar to the children, they perform better than what he found [Graue and Walsh, 1998].

Post-Piagetian theorists support the domain specific view of learning. They regard knowledge relevant to the task at hand as much more important for learning than logico-mathematical structures and also see the socio-cultural context as an important aspect of cognition. In Case's neo-Piagetian theory a central idea is the development of central conceptual structures that can be defined as internal networks of concepts and conceptual relations which permit children to think in more advanced ways [Case, 1992c]. With this theory he tries to bring the domain-specific and the domain-general theories together. According to Case, brain maturation and exercise of previously mastered schemes improve the efficiency of thought. This makes more working memory resources available so that children consolidate schemes into central conceptual structures [Berk, 2000].

I have identified three cognitive processes or systems closely linked to thought that will be the focus of the rest of this discussion. These are memory, knowledge representation and problem solving.

4.3.4.2 Memory

4.3.4.2.1 The Structure of Memory

Memory consists of a number of systems that can be distinguished in terms of their neural and cognitive structure as well as on their respective roles in the cognitive process [Gathercole, 2002]. Different authors have different views on how memory is structured [Gathercole, 2002; Parkin, 1997], but most distinguish between long term and short term memory. Short term memory store information or events from the immediate past and retrieval is measured in seconds or sometimes minutes [Gathercole, 2002]. Long term memory holds information about events that happened hours, days, months or years ago and the information is usually incomplete.

Gathercole [2002] distinguishes between short term or working memory and long term memory. The latter is divided into episodic and autobiographical memory. Where she regards short term or working memory as the same system, other authors, for example Della Sala and Logie [2002] believe that working memory incorporates short term memory. To them short term memory is passive storage of small amounts of temporary information, while working memory also involves active mental manipulation of information.

Following the model of Baddeley [2000; as cited by Gathercole, 2002], Gathercole describes working memory as consisting of four components:

1. The *central executive* that performs high-level functions such as coordinating and controlling of actions and determining retrieval strategies.
2. The *phonological loop* that stores information in terms of its phonological qualities. It includes a phonological store and a subvocal rehearsal function.
3. The *visuospatial sketchpad* that stores the spatial and visual properties of information.
4. The episodic buffer that is part of the central executive and acts as mediator responsible for integrating information from different sources.

Gathercole [2002] distinguishes between verbal short term memory (supported by the phonological loop) and visuospatial short term memory (supported by the visuospatial sketchpad).

4.3.4.2.2 Verbal Short Term Memory

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. Around the age of three or four the phonological store is fully functional, but spontaneous subvocal rehearsal is only fully developed at the age of seven [Gathercole, 2002].

Information lasts for only two seconds in the phonological store. If rehearsed within two seconds it may last longer. This store is sensitive to phonological similarity, which means that items that are phonologically distinct (e.g. X, R, W) can be recalled better than similar sounding items (e.g. B, C, V). Rehearsal is less successful when items are articulated repeatedly while rehearsing, and completely unsuccessful when irrelevant words are continuously articulated (e.g. someone saying 'the, the, the' while rehearsal is in progress). Because of the two second decay, memory sequences that take longer to articulate are recalled less successfully than short sequences.

According to Gathercole [2002] the improvement of verbal short term memory with age can partly be attributed to children's improved long-term lexical knowledge. Immediate memory is much better for familiar words than for unfamiliar words or non-words. Research has shown that from four years of age children make use of long term lexical memory to reconstruct incomplete memory traces. Here the speed of memory search influences recall.

Gathercole also reports the large differences between the memory spans of individual children, relating this to variations in vocabulary and speech production skills. As the primary function of the phonological loop is to support language learning and use, there is a definite connection between language impairments and problems with the phonological store function.

4.3.4.2.3 Visuospatial Short Term Memory

Visuospatial short term memory supports recall and manipulation of physical features of events including shape, colour and movement [Gathercole, 2002]. The visuospatial sketchpad has two components, namely the visual store that keeps physical characteristics of objects and events, and the spatial mechanism which supports planning of movements as well as rehearsal through reactivation of the contents of the visual store. According to Della Sala and Logie [2002] the sketchpad's capacity increases during childhood.

Visuospatial working memory holds a representation of the environment that we can act on mentally or physically [Della Sala and Logie, 2002]. Della Sala and Logie illustrates this with the following example: If we see a slice of gorgonzola cheese the phonological representation of its name and the articulatory codes associated with it can form a basis of a trace in the phonological loop. The word can now be rehearsed or spoken out. On the other hand, when we hear the word 'gorgonzola', its visual and spatial properties become available so that the visual form that we have in memory can be written down. Semantic information about the cheese, such as its colour, shape and taste may also become available when we hear the word.

4.3.4.2.4 The Central Executive

The central executive supports tasks with significant processing and storage requirements, such as backward recall of a sequence of digits, which requires processing of material as well as storage [Gathercole, 2002]. Its capacity develops throughout childhood. When this capacity does not fulfil processing and storage requirements of a task, there is a tradeoff between allocation for processing and storage. If more is used for processing, less is available for storing the products of processing activities. As children grow older and become better at processing and manipulating information, they require less processing resources, so that more become available for storage.
Age-related improvement of attention also contributes to improved memory capacity [Gathercole, 2002]. It seems that age has a greater influence on changes in memory access and storage capacity than on processing skills.

The central executive function plays an important role in comprehension of written and spoken language (including vocabulary), and in mathematical abilities [Gathercole, 2002] and there is evidence of a causal link between complex memory capacity and reading disabilities. Working memory is usually associated with components of mathematical processing such as mental arithmetic, while the central executive supports the more conceptual aspects of mathematics.

4.3.4.2.5 Episodic Buffer

The episodic buffer is part of the central executive and acts as interface between verbal and visuospatial short term memory and long term memory [Gathercole, 2002]. It integrates and stores information from different cognitive systems. To acquire complex cognitive skills children need to integrate information from short term or working memory and long term memory. For example, in mental arithmetic they have to store partial results temporarily while accessing specific learned rules to complete the calculation.

4.3.4.2.6 Memory Strategies

From five years of age children start to use rehearsal as a memory strategy, but their rehearsal shows production deficiency. In other words, they fail to apply the strategy in situations where it would be helpful [Schneider and Bjorklund, 1998]. With regard to organisation as a memory strategy, young children need training before they can apply it, and even then they show production deficiency. Spontaneous use is only displayed at the end of elementary school, while elaboration is only used by adolescence [Schneider and Bjorklund, 1998]. Memory retrieval strategies are also only used by preschool children if specifically instructed to do so – in which case it usually helps [Schneider and Bjorklund, 1998].

According to Siegler [1976], training in memory strategies does not improve performance. Some studies have shown that memory strategies can actually hamper performance by consuming so much of a child's mental capacity that he or she cannot perform the required task [Schneider and Bjorklund, 1998].

Children younger than six often claim that they have always known information that they have just learned. They believe that all events must be observed directly to be known and do not understand that mental inferences can be a source of knowledge [Meece, 2002]. Children aged five to six are convinced that they always remember well and that they are better at it than their friends [Schneider and Bjorklund, 1998]. Hence, they are always very optimistic about their readiness to take a memory test. This is probably because they are not yet able to do self-testing while they are studying something.

Schneider and Bjorklund [1998] reports on research by Fivush and Hudson, Fivush and Hammond, and Hudson and Nelson, that involve children's use of scripts in event and biographical memory. According to these studies, children learn what usually happens and store this information in knowledge structures called 'scripts'. Scripts then play an important role in the retrieval of memory. Young children tend to remember only script-confirming information and when an event deviates from their corresponding script for it, they sometimes, unknowingly, 'repair' the facts. From five years of age they can distinguish script-deviant and script-confirming information.

Remembering events is connected to storytelling ability. Schneider and Bjorklund [1998] suggest that children should be assisted in this by asking them questions about past events, and then helping them to distinguish the important aspects of the story from the less important ones. Supplying answers when they are unable to will demonstrate how narratives or conversations are organised. This implies that the way in which adults interact with children influences their memory skills. For example, children who are asked more questions do better in memory tasks [Schneider and Bjorklund, 1998].

Studies concerning eyewitness memory have shown that six-year-old children keep to the facts to a much larger extent than three-year-old children when asked misleading questions [Schneider and Bjorklund, 1998]. Children aged six can remember lists of items that are meaningful and familiar [Rogoff, Gauvain and Ellis, 1991]. For example, they will remember a list of toys to fetch from the toy storeroom, but not a list of arbitrary, meaningless objects.

Vygostky [1978] suggests that it is not memory (as a single function) that changes through development, but those functions that help a child with the task of remembering. It is thus the relations between memory and the functions that support it that change. According to Vygotsky, 'for the very young child, to think means to recall; but for the adolescent, to recall means to think' (p. 51). How young children describe a concept demonstrates that their definitions are determined by their personal, concrete recollections of situations rather than the abstract, logical structure of the concept.

4.3.4.3 Knowledge Representation

Piaget [1953] said that thought has structure, function and content, and that knowledge forms the content. Wellman and Gelman [1998] investigated how and when children's thinking is organised into three core domains of knowledge namely naive physics, naive psychology and naive biology.

With regard to physics, children know by the age of three that different things have different 'stuff' on the inside and by four they know that members of the same category have the same insides. For example, they know that stuffed toys have cotton wool or foam inside, while people have bones, blood and organs. From four children can predict that, for example, a wooden pillow is hard and a glass frying pan will break if it falls. Between the ages of three and eight, they understand that if material such as wood or play dough is transformed into chunks or powder, or dissolved in water, it still remains the same material [Wellman and Gelman, 1998].

With regard to naive psychology, three to five-year-old children can distinguish between mental and physical phenomena [Wellman and Gelman, 1998]. For example, they know that when someone is thinking of a dog it is mental, but if someone says his dog ran away, this dog is physically real although it cannot be seen. They can also understand psychological causality depicted in stories about human characters. They understand that these characters have goals and beliefs which determine their behaviour and that they will be disappointed if they cannot reach their goals. They can predict actions, emotions and statements if they know a character's beliefs and they know that perception influences beliefs.

By four to five years of age a child can reason about false beliefs in simple situations. For example, if someone puts something in a box and someone else removes it without the first person knowing, they can predict that the first person will still think it is in the box.

From age three children understand biological transformation such as growth. They know that animals get bigger with age and that caterpillars turn into butterflies, but that toys do not grow. They realise that a person's shaven hair will grow back, but not a doll's. Children younger than seven may, however, think that with non-biological processes such as surgery, a dog can be changed into a cat [Wellman and Gelman, 1998]. Thornton [1995] believes that young children do not really understand what a biological entity is or how it functions. For example, they will tell you a brain is for thinking, but they will deny that it controls wiggling your toes. She claims that six-year-olds can only organise biological entities into categories based on how similar or dissimilar they look.

According to Gardner [1991] the conceptions of the world, stereotypes and scripts that young children have formed by the age of five, are very difficult to change during later schooling. He suggests that these conceptions have 'surprising power and persistence'. The opposite is also true – children who fail to develop minimal competence in several cognitive domains, may experience severe learning and social problems later in life [NAEYC, 1997].

Children can acquire symbolic knowledge through representing their experiences (i.e. behavioural knowledge) in media such as drawing, painting, dramatic play and verbal descriptions [NAEYC, 1997]. Even very young children are able to represent their understanding of concepts using these tools, and through this representation of their knowledge, the knowledge itself is enhanced [NAEYC, 1997].

4.3.4.4 Problem Solving

According to Piaget's theory of development, children's problem-solving skills depend on their proficiency in logic which develops through specific stages, and that at a certain stage of development children have specific logic skills that they apply across different kinds of tasks. However, Thornton [1995] explains that, in some contexts, young children can indeed draw very sophisticated inferences. It has become clear that the tasks that Piaget used in his experiments did not accurately measure children's ability to draw references. He used unfamiliar situations so that children relied on their logic competency and not their experience to solve the problems. But even for adults it is difficult to solve problems in an unfamiliar context. The following example from Thornton [1995] illustrates this. Very few young children will be able to solve the problem:

If A is true, then B is true. A is true. What follows?

But they will have no trouble solving the next one that involves exactly the same logical rule:

If you are good on the shopping trip you can have an ice cream. You were good on the shopping trip. What follows?

Clearly the content and the meaning of the problem determine how children will process the information to come up with a solution. From the fact that children's performance is not consistent for similar tasks across contexts, Thornton concludes that logical skill is not the main factor that underlies children's problem solving.

An important element of children's problem-solving behaviour is their concrete knowledge of the task at hand [Thornton, 1995]. The concrete knowledge determines the answers they come up with, as well as the mental tools they apply in order to get to the answer. For example, in a problem where a scale must be balanced, five-year-olds know that putting an equal amount of weight on each side will balance the scale. They do not, however, know that the distance of the weight from the centre point also plays a role in balancing the scale. A nine-year-old will probably have this knowledge. Siegler [1976] found that the age is not always an accurate indication of the child's skill, since a five-year-old can, for example, be taught to take the distance from the centre point into consideration. So, the knowledge a child has does not depend only on his or her age.

An important part of problem solving is drawing analogies [Thornton, 1995]. Children as old as ten or eleven find it very difficult to solve problems through analogies. Thornton believes that the ability to draw analogies does not depend on logical skills (as Piaget suggested), but on having a good enough understanding of the potentially analogical situations. Thornton reports on a study by Goswami and Brown who found that even three-year-olds could draw analogies if they have enough information about the two areas involved. For example, a three-year-old can solve an analogy such as: *Chocolate is to melted chocolate as snow is to ...?*

Planning a solution to a problem also depends more on the child's knowledge of the task domain than on the child's age [Thornton, 1995]. From infancy children use trial-and-error for problem-solving and from the age of two they can solve very simple problems by analysing the sub goals [Thornton, 1995]. The success is dependent on their ability to recognise what is relevant and what not and this in turn depends on their knowledge. One cannot decide between several alternative solutions if you do not realise that different options exist and what the advantages or disadvantages of each option are. So, successful problem solving requires metacognitive knowledge that young children may not yet have. The limitations of children's memory capacity also hamper their planning ability [Case, 1992c].

No child thinks in just one way – even on the same task. This is the core idea of Siegler's theory of strategy choice [Berk, 2000]. When a five to seven-year-old child is given the problem '5 + 3' to solve, they will first try to retrieve the answer from memory. If this doesn't work they will count the numbers out on their fingers and then count all of them to get the answer. If they have some experience with this kind of problem, they will probably know that to count from the larger number will be quicker.

It is easy to teach children concrete problem-solving skills such as how to play a video on the video recorder [Thornton, 1995]. On the other hand, teaching them to apply abstract thinking skills or to analyse the problem, or to come up with a strategy is hard. Bringing meaning and purpose into the problem-solving situation will motivate the child. Seeing the point of solving the problem helps to get them engaged. Children's problem-solving ability can also be improved by boosting their confidence in their own abilities [Hartley, 1986]. This can be done by providing as much positive feedback as possible and, if really necessary, criticising sensitively and constructively [Thornton, 1995].

Vygotsky [1978] believed that children's level of skill depends on the support they receive from other people and the environment. By tackling a problem with a more skilled person, the child is given the opportunity to attempt things that may be beyond his or her capacity. Through scaffolded support children can learn many skills from adults. It is most effective if the adult provides just the right amount of support to stretch the child's thinking to a point where he or she can learn something new. (See the discussion of the zone of proximal development in section 4.2.3.2.)

In summary, growth in problem-solving is associated with increased levels of knowledge and experience, rather than with the development of logical reasoning [Thornton, 1995]. If children are exposed to the right opportunities, experiences and support they will develop appropriate problem-solving skills.

P69 It is possible to help children to capture information or knowledge in long term memory, but this requires some drill-and-practice or repetitive activities that are generally discouraged.

P70 The phonological store of five to eight year old children is fully functional, but not necessarily their spontaneous subvocal rehearsal capacity that is usually only developed at seven years of age.

P71 The phonological store keeps information only for two seconds. Since this does not allow for rehearsal, the information it will decay after two seconds. If the child is expected to act on audio input but fails to do so within two seconds, the information needs to be repeated. Sequences that the child should keep in the phonological store should be short.

P72 The phonological store is sensitive to similarity – children can recall a sequence of different sounding items better than sounds that are similar.

P73 Visuospatial and verbal short term memory work together. A visual image can trigger activity in the phonological loop and hearing a word can activate visual images in the sketchpad. The specific image that a child associates with an audio stimulus may influence how that child processes the input. For designers this means that children will process whatever is presented to them in different ways depending on the associations triggered by the elements of the interaction.

P74 With regard to the trade-off between processing capacity and storage capacity, designers should be aware that a high processing load will reduce the amount of storage capacity in working memory. The interaction with technology should not interfere with processing as this will place more strain on working memory. Ideally designers should identify ways to free up processing or storage capacity.

P75 Preschool children do not use memory strategies spontaneously and should be taught explicitly how to use them if this is required. Choosing between, and using, memory strategies may use up processing capacity that could have been used more productively.

P76 Children have scripts for familiar situations and they use these when retrieving information relating to such a situation from memory. Designers can help children to construct accurate scripts for everyday situations as well as for task-specific circumstances. Children can distinguish between script-deviant and script-confirming information from five years of age.

P77 Children who are familiar with narrative structures are better at recalling events in the correct order and at distinguishing between more and less important information. Support them in this by asking them questions that will help them to order information Children who are generally asked more questions do better at memory tasks.

P78 By five years of age children have a good idea of the material different things are made of. They can, for example, predict whether something will break if it falls.

P79 They can distinguish between physical things and things that exist only in their or someone else's mind.

P80 They understand psychological causality in stories and realise that characters have goals and beliefs that may influence their behaviour.

P81 They can, to a limited extent, reason about false beliefs.

P82 Although they do not fully understand how a biological system functions, they do understand natural biological transformation such as growth. They know that living things grow but toys do not.

P83 Designers should not expect young children to solve logic problems presented abstractly or symbolically. If the problem has meaning to them they are capable of logical inference. Content and meaning determine how children process information to reach a solution and concrete knowledge about elements of a problem has a huge effect on their ability.

P84 Do not use age as the only indicator of knowledge. It is not necessarily dependent on age – experience plays an important role.

P85 Children's problem solving skills are limited by a lack of metacognitive knowledge required to choose between possible solutions and designers should therefore compensate for the lack of metacognitive knowledge.

P86 Young children can only use analogies in problem solving if they have ample knowledge about the two areas involved. Designers should use analogies only when they are certain that this will be the case.

P87 Designers should bring meaning and purpose into the problem-solving situation. If children can see the point in solving a problem they will remain engaged.

4.4 Theory of Play

4.4.1 Play and Development

Play activities can be powerful learning experiences for the child. This does not mean that through having fun they will automatically learn everything they need to know, but children who are enjoying themselves will learn more effectively [Anderson, 2000]. Play gives children opportunities to understand the world, express and control emotions, and develop symbolic capabilities [NAEYC, 1997]. It also helps to improve imagination and creativity.

According to Vygostky [1978] play is not necessarily pleasurable, since often the outcome of a game can make a child unhappy. This leads him to believe that play cannot only be motivated by the feelings of pleasure – it clearly has to fulfil children's needs in some other way. To understand the role of play in development we therefore have to know the nature of these needs. Vygostky argues that play behaviour emerges at the point when children realise that their needs cannot always be fulfilled immediately. They overcome their frustration by creating imaginary worlds where they can have everything they want, the way they want it. This does not happen consciously, and it does not imply that the child will quickly create an imaginary world for every unsatisfied desire.

Vygotsky [1978] does not believe that play is imagination in action, but rather that imagination develops out of earlier play action. To him the distinguishing characteristic of play activity in general is that the child creates an imaginary situation. What other theorists regard as games with rules (where the rules are regarded as the primary characteristic of the game), Vygotsky sees as games with imagination. This is because in any imaginary world there must be rules of behaviour that comes with that world (for example, if the child plays the role of a mother she has to follow the rules of maternal behaviour). Even games with rules involve an imaginary situation (for example, Monopoly which is played in the imaginary world of the property market).

To play, children must be able to act independently of what they see [Vygotsky, 1978]. A very young child does not have the ability to separate the meaning of a situation from what is perceived and is therefore not able to create an imaginary world.

Vygostky [1978] sums up his view on play as follows: ' ... play gives a child a new form of desires. It teaches her to desire by relating her desires to a fictitious "I", to her role in the game and its rules. In this way a child's greatest achievements are possible in play, achievements that tomorrow will become her basic level of real action and morality.' (p.100).

4.4.2 Different Kinds of Play

Johnson [1998] distinguishes between the following kinds of play:

- Active play: By the age of four to six years, children's gross motor skills allow for activities such as skipping, bicycle riding and acrobatics, while their fine muscle development supports activities such as cutting with scissors, stringing beads an also typing on a computer keyboard. The fine motor development progresses rapidly from six to eight, when they can snap their fingers and build model aeroplanes.

- Constructive play: By the age of four children's ability to order objects in time and space has reached a reasonable level of complexity and they become skilled in building structures and recognisable objects through drawing, painting and using construction toys. This becomes more elaborate between five and six and the construction activities often take place in the context of social collaboration. Their constructions regularly serve as objects for socio-dramatic play. Children of seven to eight prefer construction toys with complex interlocking pieces and they like to build realistic models. While younger children (three to five) prefer matching and sorting objects, older children can classify objects using combinations of different criteria.
- Dramatic play: Pretend play is a sign of the development of symbolic functioning and starts before the age of two. It forces children to take on the role of someone or something else, thereby enhancing their perspective taking abilities. It also promotes social development and helps them cope with fear and other emotions. Older children's pretend play has richer texts, more complicated and organised scripts and involves more stage managing and directing. It serves the same purpose as before, but also plays an important role in forming friendships.
- Creative play: This involves the production of original and useful expressions or artefacts. Whatever is original for a specific child can be regarded as creative. Although a four-year-old can make representational products, those of children aged five or older are more elaborate and realistic. From five, children also construct and play with miniature worlds, sometimes creating stories about the play scene. Older children prefer very realistic and detailed models and there are strong gender differences in the choice of material.
- Cognitive play: This kind of play involves games with rules, books and educational or skill-development toys. Young preschool children like matching games with pictures and colours. They find it easier to use spinners and colour cards than dice. They can also play race games where the child moves a piece forward along a specific path. For children of six or seven, games must still be simple with few rules and requiring little skill or strategy.

4.4.3 Individual Differences in Play

Children's individual preferences regarding play activities depend on factors such as socio-economic status, gender, personality and special needs [Johnson, 1998; Kline, 1993]. For example, girls prefer smaller playgroups and enjoy a greater variety of play materials. Depending on their personalities, some children prefer realistic play, while others enjoy fantasy play. Play themes are dependent on culture, individual experience and personal taste.

Four to five-year-old children enjoy games that involve naming and classifying their world. Naming, matching and sorting are appropriate for them, as well as material that teach colours, shapes and simple letter and number concepts. Children aged six to eight are interested in their own anatomy, other countries and times past. They enjoy microscopes, binoculars and computer games [Johnson, 1998]. When it comes to books, five and six-year-olds prefer realistic and credible stories, holiday and seasonal stories and comics. Six-year-olds also enjoy stories about fears, magic, nature and the elements. Only by seven or eight can they use an index or table of contents [Johnson, 1998].

P88 Play behaviour is not only motivated by feelings of pleasure. On an unconscious level, children's play can be inspired by the need to be something that they are not or to have what they do not have.

P89 Competitive games have some winning and some losing, but it is not always the winners that want to continue playing.

P90 Every game involves an imaginary situation and it requires children to act independently from who and where they really are.

P91 Software can provide opportunities for dramatic and creative play – although children do not physically carry out the actions, they can create the characters and the story line and through recording facilities give the characters the power of speech.

P92 Active play can be facilitated through the use of cameras that detect movement or mats that function as input devices, making it possible to interact with technology through active movement.

P93 Factors that influence children's variable play preferences are age, gender, socio-economic status, personality, taste, special needs, culture and experience. For example, five year olds like naming, classifying and sorting games; six to eight year olds are interested in their own anatomy and in other times and countries; and girls prefer a greater variety of play materials than boys.

4.5 Conclusion

My aim with this chapter was to describe of the cognitive makeup of typical five to eight-year-old children. Having such a psychological picture puts one in a better position to formulate useful guidelines for the design of technology for this age group. To reach the aim I studied four prominent theories of children's cognitive development and then examined four specific domains of development, namely literacy, mathematics, thought (including memory, knowledge representation and problem solving) and play.

To remain focussed on the broader purpose of my study, I concluded the inspection of each of the four theories and each of the four skill domains with a list of emerging potential guidelines or design-related factors for the design of technology. Together, the information included in the data boxes provides the required description of the cognitive characteristics of five to eight-year-olds that may have bearing on the use and design of technology.

At this stage of my research I have already contributed on various levels by way of new knowledge in the field of child-computer interaction:

- By immersing myself in established knowledge on children's cognitive development, and then interpreting it from the perspective of child-computer interaction, I transformed this knowledge into new knowledge about the design of technology young for children. Human-computer interaction has always been linked to cognitive psychology (see section 2.4 of Chapter 2), but the scope and depth of this literature investigation surpass the extent to which child-computer interaction has drawn on psychological knowledge up to now.
- With regard to the intended product of my study (that is, a framework of guidelines for the design of young children's technology) I identified, from the literature on child development, ninety-three design-related factors that can potentially be reformulated as design guidelines. With this I have partially confirmed my thesis statement – it is possible to develop design guidelines for the design of technology for children aged five to eight years by studying psychological theories of children's development and existing research results on children's cognitive development.
- Regarding the research method, I followed a disciplined process of selecting suitable theories and research reports, studying these to the point of complete understanding, systematically searching for design-related aspects and collecting these for detailed analysis at a later stage. This is an unusual approach for the interaction design field that generally involves research of a more practical or empirical nature. By giving a full account of the range of literature studied and the subsequent presentation of the design-related data, I have made my research process transparent.

In the next chapter I continue my investigation of the thesis and explore the literature on young children and technology. I will show that existing results on children's use of technology also provide us with valuable knowledge in support of the design of technology.

CHAPTER 5

Skills Development with Technology

Table of Contents

5.1 Introduction

In his classical Mindstorms, Papert [1980] said: ‘I believe that the computer presence will enable us to so modify the learning environment outside the classrooms that much if not all the knowledge schools presently try to teach with such pain and expense and such limited success will be learned, as the child learns to talk, painlessly, successfully, and without organized instruction’ (p. 9). He believed that, if used correctly, computer technology could provide children with new possibilities for learning, thinking, and growing emotionally and cognitively. Almost thirty years have passed since those words were written and during this time many aspects of children’s computer use have been studied by researchers. In this chapter I review the literature on young children’s use of computer technology to gather knowledge that can help designers of children’s technology. Here I focus on research about technology that supports skill development and that can contribute to the formulation of guidelines for the design of technology for five to eight-year-old children. The question that drives this survey is: What can we learn from existing research into role of technology on skill development that can inform designers of technology for children aged five to eight?

The chapter progresses as follows: following up on Chapter 4’s discussion of the development of specific skills, I start by examining the role of technology in the development of mathematics and thinking skills (section 5.2) and then look at how technology has been applied to literacy and storytelling development (section 5.3). I next investigate the different ways in which children have been supported in their use of technology, looking also at the role of peer and adult collaboration in this regard (section 5.4). I end the chapter with a brief discussion of the possible disadvantages of technology use for children’s development (section 5.5) and conclude the chapter in section 5.6.

As in Chapter 4, I gather possible guidelines for technology as the discussion progresses and present them in shaded data boxes, where appropriate, at the end of sections. I want to point out again that these lists of potential guidelines are presented as they emerge in the discussion of the literature. At this point I do not analyse or organise them in any way. Only when I have completed the literature study (phases 1, 2 and 3 of my study) will I embark on the venture to construct an organised framework for design out of the unorganised assortment of guidelines I have collected.

5.2 Mathematics and Thinking Skills

My discussion of computers and mathematics is largely based on the work of Douglas Clements who has been doing research on the role of computers in education and especially mathematics education since 1983.

Clements’ [2002] response to the argument that children should not be exposed to computers too early is that children can be pushed to learn to read or write too early as well, but nobody is saying that we should keep books, pencils and paper away from them for that reason. Some studies have shown that when children are just beginning to recognise numerals, instruction by a teacher is more effective than computer aided instruction [Clements, 1987]. Also, that children will benefit more if they only start working with mathematics support programs when they already understand the concepts [Ager and Kendall, 2003; Clements, 1987]. I do not disregard the importance of instruction by teachers or parents (see section 5.4.2), but there is ample evidence that technology can provide young children with a very positive introduction to mathematics.

Mathematics skills with which computers have successfully helped children as young as three are counting, sorting and numeral recognition [Clements, 1987]. Graphics programs potentially provide good opportunities for children to explore geometric concepts such as shapes and angles. They can, for example, stretch a square to make a rectangle and use fill functions to colour shapes to learn about closure. Graphical cloning can teach them about duplication and similarity, while through moving objects around, they can learn about spatial dimensions. A graphics program that has the option of replaying a sequence of user actions can encourage them to plan their actions to make the replay interesting or meaningful. This can be the beginning of creating animated stories, although developing the ability to create stories is just a side effect of learning about cause and effect, logic relations and spatial relations.

The approach Clements and Samara [2002] followed when designing their *Building Blocks* software was to help children discover the mathematics in their everyday activities such as playing with blocks, doing art, singing, listening to stories and building puzzles. For example, in an activity to decorate cookies they have to perform counting, addition and comparison activities to determine the correct number of decorations to use.

Mathematics for young children includes the following content areas: number and operations, geometry, measurement, algebra, patterns and data analysis. In his latest review of research on the role of computers in mathematics, Clements [2002] distinguish between four subtopics, namely computer-mediated practice of arithmetic processes, computer-based manipulatives, turtle geometry and higher order thinking skills. Since much has been written about turtle geometry and Logo in particular, I devote a separate section to a review thereof, including a discussion of its role in geometry, measurement and patterns. For now, I reflect on Clements' discussion of the other three topics, supplementing his views with the work of other researchers where applicable.

5.2.1 Practicing Arithmetic with Computers

In the 1980s, when the quality of graphical interfaces was limited, many computer-assisted instruction (CAI) products followed a drill-and-practice approach. Research during that time showed that using such software for ten minutes per day improved primary grade children's skills such as counting and sorting significantly, and twenty minutes per day even more so [Clements, 2002]. Second graders who used a CAI game for an average of one hour over two weeks answered twice as many addition sums correctly as students in a control group [Kraus, 1981; as cited by Clements, 2002].

More recent products 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, 1998]. Success leads to an award of appealing feedback or the chance to play some delightful game (that is invariably not linked to what they have just learned). Soloway and Norris [1998] 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 it sends children a message that mathematics can be mastered through memorisation [Soloway and Norris, 1998]. It is also a problem that the reward for success is often unrelated to the mathematics just mastered. This tells children that fun comes after mathematics and thus that mathematics itself cannot be fun [Soloway and Norris, 1998]. *TimezAttack* [BigBrainz, 2005] 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.

My daughter recently provided a good example of how strongly a child can link the mathematical content of a game to the game environment. I suggested that she use a thirty minute break between school and a music lesson to study the 'times eight' table for a test the following day. Later, when I asked her whether she had time to do this she said that she could only fit three sums into the thirty minutes. She is a dreamer I so did not find it too strange, but later I 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 a 'times eight' sum as the program would present it. Figures 5.1 and 5.2 show her drawing and a scene from the program respectively.

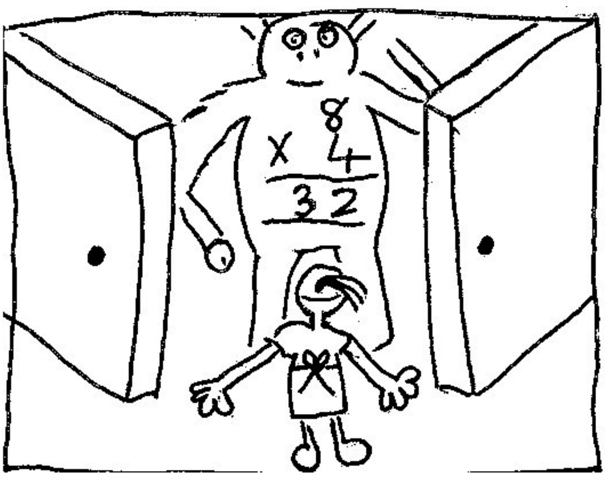


Figure 5.1 Child's drawing of a scene from TimezAttack

Figure 5.2 A screen from TimezAttack

Dettori and Lemut [1995] emphasise the importance of giving mathematics a real-life context. They say that learning material should present children with problems that 'are meaningful within the chosen experience fields; [they] are as rich as possible from the point of view of involved mathematical structures; [they] can be gradually recognised by pupils as related to their extra-scholastic experience; [they] allow a non-fake relationship between the child and the problematic situation and between the child and the required answer' (pp. 21-22).

There are computer-based products available that put the learning material in a real-life context. Soloway and Norris [1998] discuss the *Tenth Planet* educational software's fractions module that simulates the actions of cutting up pieces of paper and placing them on top of one another. Children get immediate feedback in the form of a measuring table about which part of the whole a newly cut piece makes. This illustrates how designers can use the computational capacity of technology to support learning, making the computer-based activity a cut above the paper version. *Tenth Planet* further allows children to link different related representations in an electronic journal, thereby supporting an important element of mathematics learning, namely connecting abstract or symbolic representations with concrete manifestations. In addition, this product uses video and animation to place the fractions activities in real-world contexts and everyday activities such as gardening. Soloway and Norris [1998] concluded that at that time of writing there was no academic evidence that material such as the fractions module of *Tenth Planet* was effective, but that children kept returning to it. They also linked the quality of *Tenth Planet* to its cost – in 1998 it sold for $80 whereas one could buy products that were just 'glorified worksheets' [Soloway and Norris, 1998] for $20.

In summary, computers provide unique possibilities for practice: visual displays, animated graphics and speech, immediate feedback, keeping track of progress, opportunities for exploration and adaptability to individuals [Clements, 2002].

T01 Do not use screen-based versions of paper-based drill-and-practice sheets. Present practice activities in a fun and engaging environment and link the concepts practiced to their uses in real-life.

T02 Do not convey a message that mathematics is all about memorization.

T03 If the product rewards children with a game, use the learned mathematics skill or concept in the game so that they link the fun with the mathematics.

But at the same time:

T04 Ensure that the child can detach the skill from the game to apply it in real-life situations. This can be achieved by presenting the mathematics in scenarios that resemble real life situations or by following more abstract activities with video or animation that place those concepts in real-world contexts.

T05 Use the computational capacity of the computer to enhance learning and engagement, that is, visual displays, animated graphics, speech, recording progress, detection of and adaptation to individuals.

T06 Link the abstract to the symbolic. When dealing with fractions, for example, always show the mathematical notation associated with graphical representations of fractions so that the child gradually grasps the connection.

T07 Use the computer's computational power to give immediate feedback during an activity.

5.2.2 Higher-Order Thinking Skills

According to Sedighian [1997] a computer game can only teach mathematics successfully if it involves conscious reflection upon performance and problem solving strategy. Sedighian proposes a model for designing computer-based activities so that they promote reflective thought while at the same time keeping children engaged. To keep children engaged there must be a balance between the challenge provided and the child's ability to perform the required activity [Sedighian, 1997]. Sedighian's model includes a game module to engage children in the mathematical activity and an instructional module that provides them with mathematical knowledge. To accomplish the goals of the game children must meet a sequence of challenges. To meet the challenges they need to master the underlying mathematical concepts, and to do this they need the knowledge provided by the instructional module. The presentation of knowledge in the instructional module should therefore directly support the goals of the game. The instructional module is not forced on the child at specific points in the game, but can be called upon at any time by the child. So, the child is motivated to get the mathematical knowledge by the desire to accomplish a goal of the game. As the game becomes more challenging and knowledge intensive the instructional module provides more sophisticated conceptual knowledge and increased need for reflective thought [Sedighian, 1997].

In their research on interactive multimedia environments and how they can support children's understanding of historical chronology, Masterman and Rogers [2002] found that it is possible for such a system to help young children to learn abstract concepts. To be successful, a multimedia system should provide a range of activities that allows the child to build up their reasoning skills through interactive exploration and manipulation of different kinds of representations. This is easier to do using CD-ROMs than on Web based applications. Voice instructions and narrative are easy to include in a CD-ROM and the fixed boundaries of the content prevent children from going astray in the program [Plowman and Stephen, 2003]. This is especially true for children who cannot yet read.

A study by Fletcher-Flinn and Suddendorf [1996; as cited by Clements, 2002] showed that computer use improves preschoolers' metacognition in the sense that they are better at keeping a number of mental states in mind at the same time. They also had more sophisticated theories of mind. Computer activities that involve problem solving encourage children to make choices and decisions, to try out different strategies and to think critically about their actions [Clements, 2002]. Computers offer unique opportunities to support higher-order thinking, by

- allowing children to create, change, save and retrieve ideas,
- promoting reflection and engagement,
- providing links between different knowledge or cognitive domains (e.g. mathematics and art), and
- providing constraints and feedback that children need to interpret correctly [Clements, 2002].

Clements [2002] cites several authors who found that using turtle graphics applications such as Logo has a significant effect on young children's ability to reason about their understanding and their problem-solving processes. I discuss this in the following section.

T08 To keep children engaged there must be a balance between the challenge provided and the child's ability to perform the required activity.

T09 Include a game module and an instructional module in an activity. To accomplish the goals of the game, present children with a sequence of challenges that they can only meet if they master the underlying mathematical concepts. The knowledge they need for this is provided by the instructional module.

T10 The presentation of knowledge in the instructional module of an activity should directly support the goals of the game module.

T11 The instructional module is not forced on the child at specific points in the game, but can be called upon at any time by the child. The child is motivated to get the mathematical knowledge by the desire to accomplish a goal of the game.

T12 As the game becomes more challenging and knowledge intensive the instructional module provides more sophisticated conceptual knowledge and increased need for reflective thought.

T13 Provide a range of activities that allows the child to build up their reasoning skills through interactive exploration and manipulation of different kinds of representations.

We learn from research on turtle graphics that designers can take advantage of the unique opportunities that computers offer to support higher-order thinking in many ways. For example:

T14 Allow children to create, change, save and retrieve ideas.

T15 Promote reflection on actions.

T16 Provide links between different kinds of knowledge or different cognitive domains (e.g. mathematics and art).

T17 Provide constraints and feedback that children need to interpret correctly.

5.2.3 Problem-Solving With Logo

Logo is a computer programming language designed as a learning environment for young children [Papert, 1980]. Children use it to construct programs that create graphics by moving a drawing turtle around the screen, or that guides a turtle through mazes. Logo often requires children to rethink their actions when the instructions they gave do not have the desired result. Papert [1980] thought that Logo would make it possible for young children to master abstract concepts that are normally considered beyond their capabilities. Clements [1986] investigated this with regard to six to eight-year-old children and came to the following conclusions:

- Logo training improves specific problem-solving tasks such as rule learning, but has no clear effect on general or routine problem solving. Reading and mathematics achievement are not significantly improved.
- Logo programming has a positive effect on the metacognitive task of monitoring comprehension as well as on creativity.

- The cognitive development of eight-year-old children benefits significantly more than that of six-year old-children.
- Eight-year-olds improved in classification, whether they used Logo or a CAI, but the Logo users showed more improvement.
- With regard to seriation, Logo use led to greater improvement in six-year-olds than in eight-year-old.

A study by Strand et al. [1986; cited by Clements 1987] found that preschool children displayed enthusiasm and confidence in managing the Logo environment. They were engaged by it for 'substantial time periods' and preferred to continue with the Logo tasks when alternative activities were available. For primary grade children, Logo promotes behaviour such as motivation to actively control the environment, engaging in self-directed explorations and showing pleasure at discovery [Clements and Nastasi, 1985; cited by Clements, 1987].

Later research by Clements and Nastasi [1993], Clements and Battista [2000] and Clements and Samara [2002] revealed that a later version of Logo called Turtle Math™ offer better support in terms of mathematics skills. In particular it:

- Improves young children's awareness of the properties of shapes and the meaning of measurements (of both length and angle).
- Promotes the connection of formal representations with dynamic visual representations and thereby supports construction of mathematical ideas from intuition and visual approaches.
- Encourages manipulation of shapes in a way that helps children to see them as representing a class of shapes.
- Provides an environment where children can test their ideas and get feedback about them – this encourages them to formulate their own problems.
- Helps children understand mathematical notions such as inverse operation.
- Facilitates reflection on and modification of Logo code.
- Provides them with knowledge that they can transfer to other domains such as map reading and right and left rotation of objects.

Logo's most prominent contribution is in the development of spatial concepts [Clements, 2002]. In general, children develop intuitive spatial knowledge from real and imagined actions and reflections on those actions. Logo helps to turn these intuitions into real knowledge. When drawing a shape with Logo, children have to think about the visual aspects of the shape and what movements will draw it. By writing the sequence of commands to draw the figure, children externalise their intuitive expectations. Watching their procedure in action gives them the opportunity to evaluate their plans and reflect on them when they do not produce the expected result.

Logo helps children to synthesise and integrate turn as body movement and turn as number [Clements, 2002]. Children first use their bodies to experiment with movements when planning the turtle's movement. Gradually they use only an arm, a hand or a finger, and eventually only need to do the planning mentally.

In 2001 Clements and co-workers conducted and comprehensive evaluation of a Logo-based geometry curriculum with 1624 students, that included pen and paper based pre- and post-tests, interviews, classroom observations and case studies. They found that from grade R to grade 6, Logo students fared significantly better on a general geometry achievement test and that Logo is particularly suitable for learning mathematics, reasoning and problem solving [Clements, 2002].

It is true however that children will not appreciate the mathematics in Logo if teachers do not draw their attention to the mathematics involved [Clements, 2002]. Teachers should provide tasks that make mathematical ideas explicit.

Logo programming has taught first graders about inverse operation. For example, to undo a RT 45 (right turn of 45 degrees) command one can give the LEFT 45 command. In all the studies with Logo the children's success was to some degree dependent on the teacher's scaffolding, but the nature of Logo makes it easy for adults to naturally provide this scaffolding [Clements, 1987].

T18	Children should be given opportunities to make choices and to try out different strategies.
T19	Help children make the connection between formal representations and dynamic visual representations and thereby support construction of mathematical ideas from intuition and visual approaches.
T20	Provide an environment where children can test their ideas and get feedback about them, encouraging them to formulate their own problems.
T21	Children from six are able to write simple programming commands to direct movement of an on-screen object, but this kind of activity is more suitable for children aged eight and older.
T22	Giving the child control over what happens on screen may promote engagement and self-directed exploration.
T23	By writing simple programs to move an object along a specified path a child can turn intuitive spatial knowledge into real knowledge.
T24	To write down commands to move an on-screen object a child has to plan a solution. This means they have to imagine the movements and so develop their thinking and problem-solving skills.
T25	Performing simple programming tasks to draw shapes on screen supports awareness of the properties of shapes, the meaning of measurement, understanding of inverse operations and geometry, but the knowledge will only be transferred to school mathematics if the child's attention is explicitly drawn to the connection.

5.2.4 Computer Manipulatives and Screen Images for Mathematics

I define computer manipulatives as any screen-based object that the child can manipulate in order to solve a mathematical problem or puzzle or to learn about mathematical concepts. This includes shapes that can be used to build up more complex shapes and perform mathematical transformations, or objects that can be used as counters when solving simple arithmetic problems. Manipulating shapes can help children to understand concepts such as symmetry, patterns, spatial order and fractions [Clements, 2002]. There are advantages to using computer-based manipulatives instead of physical ones. For example, when learning about fractions, on-screen blocks can be designed in a way that allows children to divide them in equal sized, smaller blocks, whereas with physical blocks this operation must be done mentally or with a separate set of blocks. With computer-based blocks, number symbols can be linked dynamically to the blocks, so that the relationships between the symbols and the quantities or sizes of the blocks are always visible.

Dreyfus [1995] writes about the importance of visual imagery for reasoning in the sense that it allows people to mentally combine familiar images to create new ones, or transform existing images into new ones. It is not always easy to do these transformations mentally and therefore external visual support can be of great help in the reasoning process [Dreyfus, 1995]. This is especially true for mathematics. Drawing a picture or diagram to represent a mathematical problem may make an element or relationship of the problem that was previously unnoticed, explicit. According to Dreyfus [1995] children do not use pictures or diagrams instinctively and need instruction to do so. Dettori and Lemut [1995] agree, saying that children should learn right from the start to make useful representations of arithmetic problems and they must be taught how to do this. The ability to associate a mental image with an abstract concept will be especially useful later when confronted with higher level abstractions. Computer-based learning material is an ideal medium to teach children to associate pictures with abstract mathematical ideas at an early stage. A visual representation can provide the bigger mathematical picture while the symbolic representations refer to specific aspects of it [Moreno and Sacristán, 1995].

An important factor in the use of any manipulative and is previous experience. Kaput [1995] believes that although visualisation depends on visual perception, it is actually controlled by semantically defined mental structures or schemata. These mental schemata are determined by existing knowledge and prior experience. Thus, when designing manipulatives and the activities on them, it is important to consider how children with different backgrounds and levels of experience will react to them. Children with mental schemata that support the interpretation of visual representation will have the advantage of more cognitive resources to apply to the underlying mathematical content.

Computer-based use of manipulatives has the advantage that a sequence of actions that children perform on the manipulatives can be recorded. With physical manipulatives a change of state ‘overwrites’ the previous state [Kaput, 1995]. If the sequence of steps they follow is visible it is easier to provide feedback about the strategies they used or mistakes they made along the line. Kaput suggests that the sequence of prior states can be hidden and replayed in animated form on request using a standard replay metaphor (including play, rewind, pause, and so on). Children should have the opportunity to change or overwrite previous actions in the sequence if they feel they can improve on them. The use of such records of activity can teach children the higher-order skills of reflection and debugging. Children can compare their recordings to see how different paths can lead to the same result and how their reasoning strategies differ [Kaput, 1995]. Creating recordings of a number of similar problem-solving activities and playing them back one after the other can reveal general patterns in problem solving, teaching the child the underlying principles of the particular mathematical problem.

The graphics capabilities of computers can also provide specialised support for children with learning disabilities. Martens [1997] discusses the design of a computer-based mathematics program called *Spatial Math Tutor*, explaining how it may help children with dyslexia. Dyslexic children's reading problems directly influence their mathematics ability. Martens argues that replacing or augmenting written representations of mathematical operations such as multiplication, division and fractions with computer-based manipulatives or other graphical representations, will help dyslexic children acquire the skills more quickly. *Spatial Math Tutor* works on the assumption that direct manipulation of spatial representations of mathematical concepts can help a child to hold the necessary information in working memory [Martens, 1997].

Unique features that can be associated with computer manipulatives are: children can save a project in its current state to continue with it later, manipulatives can be designed to snap into position when this may support interaction and learning, manipulatives can be resized, audio or visual feedback can link the concrete and the symbolic aspects of an activity, and actions can be recorded and replayed [Clements, 2002].

I now move on to language development and literacy, and how technology can help to support young children's development of the associated skills.

T26	Manipulating shapes can help children to understand concepts such as symmetry, patterns, spatial order and fractions.
T27	In activities about fractions, on-screen blocks can be designed in a way that allows children to divide them in equal sized, smaller blocks, whereas with physical blocks this operation must be done mentally or with a separate set of blocks.
T28	Link number symbols dynamically to the blocks, so that the relationship between the symbols and the quantities or sizes of the blocks are always visible.
T29	Computer-based learning material is an ideal medium to teach children to associate pictures with abstract mathematical ideas at an early stage. A visual representation can provide the bigger mathematical picture while the symbolic representations refer to specific aspects of it.
T30	Designers must appreciate the importance of the visual representations they choose for mathematical content. Children need to be taught how they can use visual representations in

	solving mathematical problems.
T31	Include activities that help children to build a visual representation of a mathematical problem.
T32	Consider how children with different backgrounds and levels of experience will react to manipulatives and the activities on them. Children with mental schemata that support the interpretation of visual representation will have the advantage of more cognitive resources to apply to the underlying mathematical content.
T33	Make use of the possibility to record a sequence of actions that a child performs on a computer manipulative.
T34	The sequence of prior states of a solution can be hidden and replayed in animated form on request using a standard replay metaphor (including play, rewind, pause, and so on).
T35	Give children the opportunity to change or overwrite previous actions in a sequence if they feel they can improve on them.

5.3 Language Development and Literacy

Like mathematics, literacy is often the subject of research on computers and early development [Plowman and Stephen, 2003]. Clements [1987] reports on several research studies on the influence of computer use on children's language and literacy development. It was found, for example, that the number of words preschoolers speaks almost doubles during computer activities, and that computer use encourages humour, emotion and imagination in the use of language. Also, that children's reading can be improved significantly when spending ten minutes a day with computer assisted instruction (CAI)[7]. CAI also provides good opportunities for learning visual discrimination and letter recognition [Clements, 1987]. A study involving the *Writing to Read* software showed that kindergartners using the software learned to read and write better than children in control groups. In this experiment, computer use did not have a detrimental effect on their spelling ability [Clements, 1987].

My aim in this section is to find, in published research reports, guidelines that will support the design of activities through which children aged five to eight years can effectively learn to read, write and create stories. Studies on the effect of computers on children's literacy usually focus on improving traditional literacy [Plowman and Stephen, 2003], forgetting that children need to attain computer literacy before they can use computers. Before I discuss reading, writing and storytelling support through technology, I briefly look at computer literacy.

---

[7] This, however, does not mean that ten minutes CAI is more successful than ten minutes of other kinds of reading instruction.

5.3.1 Computer Literacy

In line with the UK Curriculum Guidance for the Foundation Stage (CCGFS) and the British National Curriculum, Siraj-Blatchford and Siraj-Blatchford [2001] state that an Information and Communication Technology (ICT) Curriculum for young children must address two things:

1. Emergent technological literacy and children's understanding of how ICT can be used.
2. Developing children's skills to use ICT tools.

Siraj-Blatchford and Siraj-Blatchford [2001] maintain that children should, before they turn six, be able to identify the roles of technology in their everyday lives and use technology to support their learning. To achieve this, children should be given opportunities to explore and play with computers before they start using it for structured learning.

Plowman and Stephen [2003] provide a list of competencies that could be included in definition of literacy so as to incorporate new, technology-related aspects of literacy. These include:

- interpreting the computer's interface and being familiar with conventions in this regard,
- navigating electronic text,
- locating and retrieving appropriate information,
- having the ability to produce multimedia texts, and
- having the sensorimotor skills necessary for computer interaction.

Some of these competencies assume that the user can already read. With child users even more basic competencies are concerned, such as:

- moving the cursor with the mouse,
- left, right and double clicking the mouse buttons,
- dragging and dropping screen-based objects, and
- locating specific keys on the keyboard.

Plowman and Stephen [2003] consider the possibility that the cognitive load of interaction with the computer may hinder the learning of reading and writing because there may not be sufficient cognitive resources left for this purpose. They warn designers to be very aware of the problem of cognitive load when designing software for young children. The computer literacy requirements of a product should therefore be very clear and designers should not assume outright that the users have these skills.

T36 Help children to identify the roles of technology in their everyday lives and to use technology to support their learning.

T37 Make the level of computer literacy expected of a child that will use their product explicit. This level is determined by basic competencies such as:

- moving the cursor with the mouse,
- left, right and double clicking the mouse buttons,

- dragging and dropping screen-based objects,
- locating specific keys on the keyboard,

T38 Or by more advanced skills, such as:

- interpreting the computer's interface and being familiar with conventions in this regard,
- navigating electronic text,
- locating and retrieving appropriate information, and
- having the sensorimotor skills necessary for computer interaction.

T39 Designers should not assume that the users have these skills. For example, if a product requires left, right and double-clicking the mouse it may be necessary to coach young users in this before they can start using the product.

T40 Designers should be very aware of the problem of cognitive load when designing software for young children. The cognitive load of interaction with the computer may not leave sufficient cognitive resources left for learning to take place.

5.3.2 Learning to Read and Write with Computers

In this section I discuss several studies that show how computers have been used successfully to support development of early reading skills.

Mioduser, Tur-Kaspa and Leitner [2000] conducted a study with special education kindergarten children (aged five to six) in the central region of Israel to discover:

1. Whether computer-based materials can add 'learning value' (p. 55) when early reading skills to children who are at-risk for reading development.
2. Features of computer technology that are associated with acquisition of early reading skills.

The software that forms part of the *I have a secret – I can read* program [CET, 1996; cited by Mioduser et al., 2000] was used with touchscreen input and speakers or headphones for audio feedback. The instruction first focused on reading readiness that includes letter recognition, auditory perception and visual discrimination, and then moved to training in reading activities. After completing a series of blocks of activities (each comprising four tutorials and a practice session in the form of a game) children were tested on the following skills:

- Phonological awareness, including phoneme recognition at the beginning or end of a word, new word composition out of known-word phonemes, and rhyme recognition and formation.
- Word recognition that includes general-content words as well as kindergarten-topic words. (The children's vocabulary was tested before the training using a formal existing vocabulary test.)
- Letter naming where children have to name twenty-two uppercase Hebrew letters.

Mioduser et al. [2000] found that, compared to children who received reading instruction using printed materials, children who received computer-based training showed significantly greater improvement in phonological awareness, word recognition and letter naming skills. The children's motivational and self-confidence levels were also positively affected.

Mioduser et al. [2000] attribute a lot of the success of the computer-based material to the use of sound for giving instructions and providing feedback. Children can request audio information at character, syllable, word or sentence level using the touchscreen. Information is presented on screen in the form of text and rich still and animated images. Activities progress from reading readiness tasks to learning the Hebrew alphabet to actual reading of text [Mioduser et al., 2000].

The study by Mioduser et al. [2000] revealed different aspects of technology that can contribute to children's acquisition of reading skills. Table 5.1 summarises their findings.

Table 5.1 Summary of findings by Mioduser et al. [2000]

Reading related skills/characteristics	**Supporting aspect of technology**
Understanding the relationship between letters and sounds (necessary for the development of phonological awareness, letter identification and spelling).	Through tactile, visual and auditory communication with the technology children can actively learn to identify letter-sound relationships. Immediate audio feedback when a child touches a letter, word or sentence and the fact that different senses are applied simultaneously, are particularly supportive of developing these skills. This is also true for less able readers.
Awareness of phonemes – this includes manipulation of structural phonological units such as phoneme segmentation, replacement and elimination that is particularly difficulty for poor readers.	Concrete manipulation of letters and word components in activities (through touch, hearing, seeing, constructing, playing and replaying auditory constructs) that let children experience decomposition, recomposition and creation of words.
Acquisition of reading skills is highly individualised.	The software includes a management module for keeping record of children's progress, the tasks performed and their competency level at the different skills. For a teacher or facilitator, this supports close observation of performance and making decisions about how to adapt the sequence of activities to individual needs. This ensures that learner needs and learning activities fit, which is important for supporting reading-skills acquisition.
Children need motivation to learn to read, especially if they have reading problems.	Features such as multiple modes of representation, interactivity, immediate and individualised feedback, and the sense of control are strongly motivational.

Van Daal and Reitsma [2000] conducted a similar study in which they found that both advanced and struggling six-year-old readers can benefit from exposure to computer-based reading support. Their emphasis was on the variation in ability of children of the same age to acquire reading skills and how computer applications can deal with this variability. Advanced readers have good phonological skills, they understand what reading and writing is about and they progress rapidly [Van Daal and Reitsma, 2000]. Children with reading difficulty display poor word recognition that is possibly the result of problems with phonological recoding (transforming letter patterns into phonological codes) which, in turn, is caused by lack of phonological awareness or phonological sensitivity [Van Daal and Reitsma, 2000]. Addressing these problems is a costly and time-consuming task. Van Daal and Reitsma [2000] hypothesised that computer-based products can provide a way to adapt literacy instruction to the needs of advanced as well as disabled readers.

As with the Mioduser et al. [2000] study, the primary supporting tool of the *Leescircus* software used by Van Daal and Reitsma is a text-to-speech unit that allows the child to use the mouse to request the pronunciation of an on-screen word. In general, *Leesircus* uses high quality digitised speech for supportive feedback [Van Daal and Reitsma, 2000]. It also includes some drill-and-practice features. Children can use the software independently as all instructions and feedback are given in audio format.

Van Daal and Reitsma [2000] point out the following general shortcomings of existing reading support systems:

- Lack of proper systems for the administration of individual users in support of individualised training.
- Insufficient training for the early stages of phonological skills.
- Activities focus on practice rather than on acquisition of news skills.
- Diagnostic systems are unsatisfactory.

They consequently promote the use of an integrated learning system (ILS) that takes responsibility for continuous assessment of accomplishments and reducing facilitator duties [Van Daal and Reitsma, 2000].

The *Leescircus* software consists of sections that respectively deal with different skills that are critical for reading development [Van Daal and Reitsma, 2000]. It begins with pre-reading skills such as teaching children that there is more to words than their meaning, expanding their vocabulary and making them aware of the phonological structure of words. The latter is done through activities that require users to create words by putting together specific phonemes the system provides, or to find a particular phoneme in a given word. It also teaches children the correspondence between letters and sounds [Van Daal and Reitsma, 2000]. The aim is to make reading an automatic skill, so there is some focus on increasing the speed at which children decode words. Van Daal and Reitsma lists the following exercises included in *Leescircus*:

- Matching pictures with spoken words.
- Indicating where a sound is heard in a spoken word.
- Indicating which letter sound is heard.
- Pointing out a requested letter in a word.

- Filling in a missing letter.
- Matching pictures with written words.
- Selecting a word by its sound.
- Spelling a word that is already written on the screen.
- Spelling a word by its sound.

In each exercise there are varying levels of difficulty. *Leescircus* is therefore suitable for use by children with normal reading progression who can use it independently, or it can be tailored to suit children with specific difficulties. It was designed in a way that allows new exercises to be added and it can easily be adapted for different languages. It is currently available in Dutch, Swedish and English [Van Daal and Reitsma, 2000].

Van Daal and Reitsma [2000] conducted two experiments with *Leescircus* – one with six-year-olds that display normal reading abilities and the other with eight to twelve-year-olds with reading difficulties. In the first study, after spending between 1.5 and 6 hours with *Leescircus*, the experimental group's level of performance in letter knowledge, word reading and non-word reading, was equal to what an average grade one child will attain after three months of formal teaching. Their performance was significantly better than that of the control group.

The second study showed similar success, but the ages of the subjects fall outside the age group of interest here.

Other studies using different reading support software, e.g. *Reader's Interactive Teaching Assistant* (RITA) [Nicolson, Fawcett and Nicolson, 2000] and *Alphie's Alley* [Chambers, Abrami, Slavin, Cheung and Gifford, 2005], confirm the findings discussed above. Since these studies did not reveal any design-related factors not already identified in Van Daal and Reitsma [2000] and Mioduser et al. [2000] I do not include a review of them.

The software used by Van Daal and Reitsma [2000] and Mioduser et al. [2000] are aimed at teaching specific reading skills through appropriate activities. Another class of software that have been found to support reading is so-called talking books. These are electronic versions of storybooks that the child views online while listening to a narrator reading the story [Matthew, 1997]. A study that examined the effect of talking books software on children aged five and six, found that the basic software version improved sight recognition of key words of children with reading problems [Plowman and Stephen, 2003]. Talking books can also improve reading comprehension [Matthew, 1997]. I will return to this topic when I discuss storytelling technologies in section 5.3.3.

Learning to write cannot be studied separately from learning to read as the two skills are interrelated [Ryokai, Vaucelle and Cassell, 2003]. Although the discussion so far in this section refers mostly to reading, many of the aspects discussed also apply to writing skills. Computers are a natural source of scaffolding for beginning writers [Clements, 1987]. It allows them to experiment with words and letters using a word processor, for example, without their lack of fine motor development getting in the way. They can easily correct mistakes and will therefore be more willing to take risks. Children who are reluctant to write by hand can be motivated to write using a word processor [Borgh and Dickson, 1992]. Combining a text-to-speech facility with a word processor makes it even more attractive.

A word processor with speech feedback increases the amount of editing children do, but does not significantly affect the length or quality of what has been written [Borgh and Dickson, 1992; Clements, 1987]. It does make children aware of the need to edit if they can hear what they have written. The audio feedback is especially effective in allowing younger children to experiment with how their writing sounds. They can for example scramble the letters of their names and hear what that does to its pronunciation or create meaningful words by accident. When the software does not recognise the word it spells out the letters. Speech feedback gives children an idea of how their writing would sound to others (that is, it gives them an audience perspective) [Borgh and Dickson, 1992].

Shilling [1997] questions the value of speech-synthesised feedback when beginning writers 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 [Shilling, 1997]. Shilling believes that speech-synthesised feedback is most effective when children have already acquired metalinguistic awareness – in other words, when they already have some reading ability.

Different speech-software handles speech feedback differently [Borgh and Dickson, 1992]. In some applications the user can select feedback on letter, phoneme, word or sentence level (for example, *Writing to Read*, *Talking Text Writer*, and *LeesCircus*). The word-processing software Borgh and Dickson [1992] 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 [Borgh and Dickson, 1992]. Children responded positively to the speech-synthesiser, although some indicated that they would prefer to have more control over when feedback was given.

It is clear from this discussion that technology can successfully support the development of reading and writing abilities. In Chapter 6, when I discuss input and output mechanisms and tangible interfaces, the role of technology in learning to read and write will be discussed further with reference to tabletop environments [Sluis, Weevers, Van Schijndel, Kolos-Mazuryk, Fitrianie and Martens, 2004] and handwriting based input [Read, MacFarlane and Casey, 2002b; Read, MacFarlane and Gregory, 2004].

I end this section with a list of derived design-related factors and then continue to investigate how computers can support children's storytelling skills.

T41	Technology for reading support should include activities that develop pre-reading skills such as phonological awareness and letter-sound association.
T42	Poor word recognition is often related to inability to do phonological recoding, that requires phonological awareness. So only after children have acquired some phonological sensitivity they can be presented with activities where they transform letter patterns into phonological codes and vice versa.
T43	Tactile, visual and auditory communication with the technology can be combined so that children can actively learn to identify letter-sound relationships.
T44	Immediate audio feedback when a child touches a letter, word or sentence and the fact that different senses are applied simultaneously, are particularly supportive of developing reading skills.
T45	Text-to-speech facilities that allows the child to use the mouse to request the pronunciation of an on-screen word are essential for reading support.
T46	Concrete manipulation of letters and word components in activities (through touch, hearing, seeing, constructing, playing and replaying auditory constructs) can provide children with experience in decomposition, recomposition and creation of words.
T47	Any reading support system should include a management module for keeping record of children's progress, the tasks performed and their competency level at the different skills. This can then be used (either by the system or a human facilitator) to make decisions about how to adapt activities to individual needs.
T48	The following features can be used to increase user motivation: multiple modes of representation, interactivity, immediate and individualised feedback, and the sense of control.
T49	Design to support the needs of advanced as well as disabled readers – this requires adaptability. Activities must have varying levels of difficulty.
T50	Competent children should be able to use the software independently, provided all instructions and feedback are given in audio format (it makes no sense to include written instructions or feedback in a product that teaches beginners to read).
T51	Reading skills do require practice but the focus should be on skill acquisition rather than practice.
T52	Reading support software requires good diagnostic systems as well as proper systems for the administration of individual users to support individualised training.
T53	The aim is to make reading an automatic skill, so there should be some focus on increasing the speed at which children decode words.
T54	Examples of activities that support acquisition of reading skills: • Matching pictures with spoken words. • Indicating where a sound is heard in a spoken word. • Indicating which letter sound is heard.

- Pointing out a specific letter in a word.
- Filling in a missing letter.
- Matching pictures with written words.
- Selecting a word by its sound.
- Spelling a word that is already written on the screen.
- Spelling a word by its sound.

T55 The software must be designed in a way that allows easy adaptability to different languages.

T56 Speech feedback is essential for writing support.

T57 Children should be able to control when speech feedback is given and should have the option to turn it off. This applies more to activities where they write extended text than to activities aimed at teaching letter-sound and other pre-reading skills. Children should be given the option to listen to the last sentence, the last paragraph or the whole text.

T58 Children should have some control over whether feedback is given on letter, phoneme, word or sentence level.

T59 Designers should consider the value added by speech-feedback before including it in writing support activities. Misspelled words may yield feedback that confuses rather than helps a beginning writer.

5.3.3 Storytelling Technologies

I begin this section with a brief overview of the history of storytelling media and then discuss the findings of a number of prominent studies on the role of computers in the development of storytelling skills.

5.3.3.1 From Oral Tradition to Print to Digital Media

Computers provide children with all of the storytelling media of the past (the oral tradition, print, film, animation) combined into one form of technology [Madej, 2003]. Madej believes that the immersive and compelling qualities of computers can provide positive opportunities for reshaping children’s narrative expression. When the print tradition began in the fourteenth century and printed material became available to many, not everybody accepted this as a positive development [Madej, 2003]. Critics believed, for example, that the density of the text would distract the reader from understanding the text. The benefits outweighed the concerns and print medium was generally accepted as a replacement of the oral tradition in learning. At first, children’s books were no different from adult books, with no effort at narratives that would appeal to their level of understanding. Educational books were written to teach children about morality. Only after the philosopher John Locke introduced the idea of childhood in the late 1600s, were books for children written with the purpose to encourage reading. The literature for children became more suitable for their age. The work of philosopher Jean Jacques Rosseau who believed children should be accompanied in learning rather than led, inspired children’s writers to provide books that made learning active, engaging and self-directed [Madej, 2003]. From the late 1800 children’s books were written to delight and entertain.

Just as oral tradition was the basis for print books when they first appeared, book narratives were the basis for the new media like television, video and computer software [Madej, 2003]. The development of hypertext made a new kind of storytelling possible that could include interactive exploration and discovery. However, early CD-ROMs were mostly digital books that looked very much like the printed versions, with music and sound effects and modest interactivity. Much development has taken place since then and the current software products for children are abundant with interactivity. The World Wide Web provides a platform for children to write and 'publish' their own stories and many websites are available for this purpose.

According to Casell and Ryokai [2001] there is still a need for products that support children's storytelling and fantasy play in a more open-ended way that encourages their creativity. Children's spontaneous storytelling is child-driven and if technology is to support this aspect of their development it should allow child-driven storytelling. Other aspects of storytelling that Cassel and Ryokai [2001] point out are that children need to learn to understand the listener's point of view, that collaboration with peers can help them to create rich narrative worlds, and that children use more mature language in stories than in everyday life.

Children engage in storytelling for various reasons [Casell and Ryokai, 2001]: it lets them practice different roles in the social world, it gives them a way to express their knowledge and experiment with their experience and it helps them develop their language skills. Around age four children start giving their toys a voice and they can give different characters different voices or linguistic styles [Casell and Ryokai, 2001]. By seven years of age they can add the voice of a narrator and shift between characters and narrators, moving closer to the ability to take the perspective of a listener.

Storytelling is clearly an important aspect of development and involves many aspects that technology can potentially support. I consequently discuss a selection of systems that provide such support.

5.3.3.2 Story Construction with Computers

I have chosen three prominent studies as the focus of this discussion, namely Scaife and Rogers [2001], Cooper and Brna, [2001] and Ryoaki, Vaucelle and Cassel [2003]. Later, in Chapter 6, I return to storytelling technologies but with the focus on tangible interfaces (for example, *StoryMat* [Casell and Ryokai, 2001], *PETS* [Montemayor et al., 2000] and *oTTomer* [Valinho and Correia, 2004]).

Scaife and Rogers [2001] developed a virtual theatre for young children that supports learning through play. It provides them with a set of tools that they can use to create, direct, edit and act out plays in an imaginary setting. Scaife and Rogers felt that existing computer applications for children does not compare well with children's everyday play activities and that they provide insufficient support for improvisation and imagination. They claim that a virtual setting can provide a more extensive range of tools to support creativity than the real physical world. According to this research, a successful virtual story construction environment will provide enough support to enable children to focus on the structure of the story rather than the details. Their research on this project led Scaife and Rogers [2001] to the following conclusions:

- Children between five and six years of age have no problem switching between character viewpoints.
- They need to be able to implement their own ideas to remain interested and motivated, but do not want to create the whole story from scratch.
- External mechanisms should be included to constrain children's management of sequences and to help them plan themes and events. For example, the system can be programmed not to allow animals to walk across a river.
- Limiting the degree of realism promotes their use of imagination when choosing characters.
- Children can manage the switch between editing mode and acting mode if the change in mode is very clear.
- The interface must be kept very simple and the child needs training to build up a good enough mental model to use the system.
- With small children crude forms of movement and exaggerated facial expressions are acceptable. High-fidelity realism is not important and may even hamper the use of imagination.

Story writing depends largely on creativity that includes being open to new ideas, to make connections between different ideas and to try new forms of self-expression [Bryson, 1999; as cited by Cooper and Brna, 2001]. Creative behaviour and lateral thinking often occur when the rational part of the brain is resting and the unconscious, intuitive parts can work. Children can only engage in creative activity if they feel secure and confident about what they are doing and, like a teacher, software must offer just enough structure and intervention, enough stimulation, and variety of approach to provide encouragement in a secure environment [Cooper and Brna, 2001]. What is needed for creativity is a combination of security, stimulation and receptivity to different ideas. Receptivity requires a relaxed brain and humour is one mechanism to achieve this state. As a creative connection, humour is often caused by juxtapositioning diverse elements, by the contrast between image and reality or the voiced and the unvoiced [Cooper and Brna, 2001].

According to Cooper and Brna, [2001] young children who have the skills to tell a story verbally, may still have difficulty translating their ideas into text. They need help with words, ideas, structure and visual representation. Together with step-by-step assistance in content, form and structure, they require emotional support that should be adaptive to their individual needs. Teachers cannot always provide one-to-one attention exactly when required. Cooper and Brna believe that a combination of peer and agent help with speech synthesis can provide some of the required support.

The writing support software that Cooper and Brna [2001] used for their research, *T'rrific Tales*, was developed for use on tablet PCs or with a large touchscreen and is aimed at users aged five to six. This age group has unique problems with writing – for example, they need affirmation and they are not very proficient in talking about stories in structured way. The software provides spelling support, helps children to decide what to do next and supports collaborative work over a network. It includes text to speech technology, structured word banks that the children can easily access, and a software agent that provides proactive assistance. The text-to-speech synthesiser has some inadequacies but fascinated the users and helped them to associate sounds with unfamiliar text. Text-to-speech synthesis has been shown to help children write longer stories, revise more and enjoy writing more [Cooper and Brna, 2001].

A *T'rrific Tales* story typically consists of six scenes. Each scene can include a picture (drawn by the child or taken from a picture library), text and speech or thought bubbles. To get a balance between secure and more imaginative elements, children can select props, scenes and characters from home or school environments or from fairy tale or space scenes. They can add their own photographs to the picture database, mix real world elements with fantasy world elements, and elements from the past, present and future. This encourages the use of imagination to challenge stereotypes, change power relations and have fun. The text bank can be accessed through speech synthesis and gives the children ideas and suggestions that support unusual combinations of scenes and characters. A prince may for example sleep with a teddy bear and wish he worked in a supermarket. A princess may dream of riding off on a motor bike, or a knight may secretly wish he was a dancer. The word bank provides story titles, story starters, story stirrers, examples of events and endings. There is also a word bank of feelings to add excitement or emotion. The software can best be used with an adult facilitator. The idea is that children should gather around the large touch screen to create, present and discuss their writing.

A software agent called Louisa supports the children in a positive and caring way by showing interest, affirming them and making suggestions for further action. She is modelled on the emphatic and encouraging behaviour of a teacher in a one-to-one situation and never criticises or corrects [Cooper and Brna, 2001]. She seems busy with her own work and looks up and interacts at appropriate moments. Initially she gives broad suggestions and if the child does not show progress would give more detailed advice. She might say 'Oh great – you chose the witch ... what is she going to do?'

Experiments with children using a prototype of the software over several months yielded the following results [Cooper and Brna, 2001]: the children enjoyed using the software and created many stories, individually, in pairs and in pairs over the network. They remained enthusiastic about the software over time. They enjoyed mixing up scenes and characters. The stories differ from stories written on paper in the sense that they are shorter, have more scenes, fewer occurrences of 'and then...and then' type progression, more mixing of character and scene types. They frequently redraft the stories and because there is more discussion, thinking, reading, listening, vocabulary learning and story building, the stories take longer to complete. Despite different levels of reading ability, all the children listened to the words and phrases in the word bank and used them. They are keen to help one another and exchange ideas. In general, Cooper and Brna [2001] have found 'that the software is fun to use, stimulate creative writing and allows children to write humorous, thought-provoking stories in a safe environment' (p. 39). New activities that were planned for *T'rrific Tales* at the time of Cooper and Brna's research are sequencing jumbled stories, finishing partly written stories, changing existing stories and helping the agent to write a story.

Ryokai, Vaucelle and Cassell [2003] looked specifically at the role of peer collaboration in the use of storytelling technologies. Young children's language development, and hence their storytelling ability, is influenced by interaction with more able conversational partners [Ryokai et al., 2003]. Ryokai et al. investigated the role of peer partners in storytelling and chose *Sam* – an innovative storytelling system – as the object of their study. The *Sam* system involves an embodied, six-year-old, conversational agent called Sam and a toy castle with a figurine [Ryokai et al., 2003]. A life-size image of a child called Sam is projected on a screen behind the castle that acts as prop for the stories Sam and the users create (see Figure 5.3). The child sits in front of the castle and takes turns with Sam to tell a story. When Sam has completed a story she invites the child to use the figurine to tell a story. While the child tells a story Sam acknowledges this by nodding and uttering prompts such as 'And then what happens?'. When it is Sam's turn again she begins her story in the room of the castle where the child's story ended. Sam has good storytelling skills and can therefore provide scaffolding to move the child's skills to a next level [Ryokai et al., 2003]. To achieve this it important that Sam provides exactly the right amount of support.

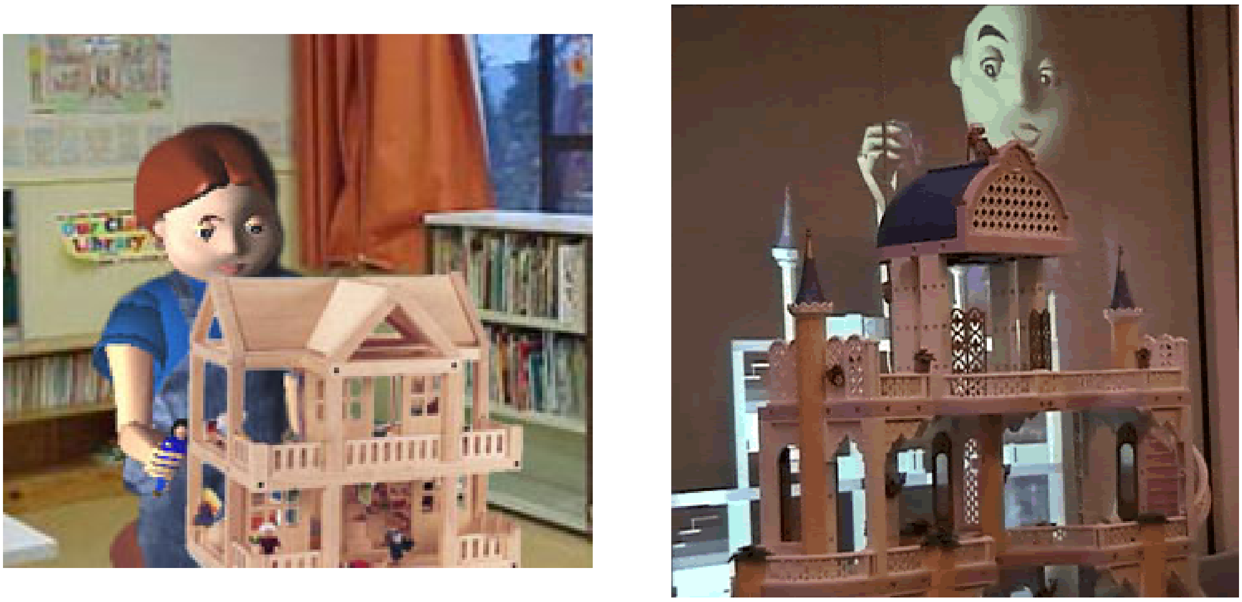


Figure 5.3 The Sam System [MIT, 2008b]

A microphone and motion detector act as input devices and audio threshold detection is used to determine when some feedback from Sam is appropriate [Ryokai et al., 2003]. A tag is attached to the figurine and radio frequency tag readers in every room of the castle tell the system where the figurine is at any moment. Sam speaks with a real child's voice.

Ryokai et al. [2003] conducted an experiment with twenty-eight five-year-old girls, some of whom played alone and without Sam, some in pairs without Sam, some alone with Sam and some in pairs with Sam. For the purpose of the experiment, Sam's responses were controlled by a researcher in a Wizard of Oz[8] setting [Ryokai et al., 2003]. They specifically looked for occurrences of spatial (Where in the castle?) and temporal expressions (When did it happen?) in the children's stories as an indication of how complicated the stories were. Further indication was instances of quoted speech, where the characters in the story said something. The children using the system with Sam (alone or in pairs) displayed dramatically more spatial and temporal expressions and quoted speech than children playing only with the castle. They found that interaction with Sam has a much greater effect on a child's stories than interaction with another child. There wasn't a significant difference between individuals playing with Sam and pairs playing with Sam. Sam also succeeded in improving the children's use of language over time. The children acknowledged Sam as a real partner, making eye contact while talking to her and checking if Sam was 'OK' or offering her their help [Ryokai et al., 2003]. The obvious shortcoming of Sam is that she does not understand the content of the children's speech and can only use silences to determine when to speak. As part of their future plans, Ryokai et al. [2003] wants to investigate the possibility of using keyword spotting to control Sam's responses and to adapt Sam's responses so that children can have a longer-term relationship with her. The main contributions of Sam are that she provides a linguistic model for children and can act as a peer who promotes constructive criticism and perspective taking [Ryokai et al., 2003].

In *Sam*, the role of peer help is central to the storytelling environment. In the next section I continue the discussion of providing children with support when using technology, looking at different ways in which scaffolding can be provided. One of the scaffolding applications I will discuss is concerned with supporting children's development of time concepts, that is closely linked to developing storytelling skills.

I now end the current section with a list of emerging guidelines for the design of technology.

T60	Children aged five to six years have no problem switching between character viewpoints.
T61	Children aged five to six must be allowed to implement their own ideas to remain engaged, but they do not want to create a whole story from scratch.
T62	Designers should include external mechanisms to constrain children's organisation of sequences and to help them plan themes and events. For example, if they initially include a character that never plays a role in the story, the system can ask whether that character is still needed.
T63	A high degree of realism is not essential in storytelling technologies aimed at young children. Less realistic presentation can promote children's use of imagination when choosing characters. With young children crude forms of movement and exaggerated facial expressions are acceptable.
T64	Children can switch between editing mode and acting (or play) mode but the change in mode should be very clear.
T65	Storytelling interfaces must be kept very simple and should include a training module to help

[8] A person sits behind a screen and provides Sam's voice for the duration of the experiment.

children build up a good enough mental model to use the system.

T66 For children to engage in creative activity they should feel secure and confident about what they are doing (this will help relax their rational mind so that the subconscious can do its work). To provide encouragement in a secure environment, technology should

- offer some structure and intervention,
- enough stimulation, and
- allow a variety of approaches in performing activities.

T67 Creativity requires receptivity to ideas and hence a relaxed brain. Humour is one mechanism to achieve this state and can, for example, be achieved by juxtapositioning diverse elements.

T68 In addition to step-by-step assistance in content, form and structure, designers should provide emotional support that is adaptive to children's individual needs. A combination of peer and agent help with speech synthesis can provide some of the required support.

T69 Young children need affirmation and they are not very proficient in talking about stories in structured way. Technology should help children to decide what to do next.

T70 Text-to-speech technology, spelling support, structured word banks that children can access easily, and a software agent that provides proactive assistance are examples of supportive elements in storytelling products.

T71 Create a balance between secure and imaginative elements by allowing, children to select props, scenes and characters from real life, familiar environments or from fairy tale or space scenes (that is, mix real world elements such as family photos with fantasy world elements, and elements from the past, present and future).

T72 Provide a text bank that gives the children ideas and suggestions and support unusual combinations of scenes and characters to encourage the use of imagination, to challenge stereotypes, to change power relations and to have fun. The following can be included in a word bank: story titles, story starters, story stirrers, story events, endings and feelings.

T73 Software agents should support children in a positive and caring way by showing interest, affirming them and making suggestions for further action. They should not criticise or correct.

T74 Initially an agent can give broad suggestions and if the child does not show progress, provide more detailed advice.

T75 Example activities include: reading existing stories, sequencing jumbled stories, finishing partly written stories; changing existing stories and helping an agent to write a story.

T76 Peer collaboration can be included in storytelling technologies through an embodied conversational agent that resembles a young child and speaks with a real child's voice, but has better storytelling skills than the potential users.

T77 Ideally an agent should be able to respond to the content of the child's story, prompting the child to continue or just nodding to show interest. This could be done through audio threshold detection (silences and pauses in the child's story) or possibly keyword recognition. To provide effective scaffolding, such an agent must provide exactly the right amount of support.

T78	If the child should build up a long term relationship with an agent, the agent's interaction should acknowledge previous encounters. For example, it cannot greet the child with exactly the same words every time the child starts up the system.
T79	An agent that supports storytelling skills should provide a linguistic model for children and should act as a peer who promotes constructive criticism and perspective taking.
T80	Children of seven years or older can be presented with activities that require them to shift between character voices and the voice of the narrator.

5.4 Scaffolding in the Context of Technology Use

Scaffolding is a concept that came out of Vygotsky's theory. As discussed in Chapter 4 (section 4.2.3.3) it refers to assistance that helps children to achieve goals that they would not have been able to master on their own. In this section I discuss three ways in which scaffolding (and other kinds of support) has been associated with children's technology, namely

- through peer collaboration (section 5.4.1),
- through adult support (section 5.4.2), and
- through mechanisms built into the system (section 5.4.3).

5.4.1 Peer Support and Collaboration

In the *Sam* system discussed in the previous section, the role of peer collaboration was a central factor in supporting the development of storytelling and language skills. I continue the discussion of peer collaboration in this section, looking at peer support in the context of technologies that support other aspects of development.

5.4.1.1 Collaborative Computer Use and Social Development

The use of technology provides good opportunities to support and encourage collaborative learning [Benford, Bederson, Akesson, Bayon, Druin, Hansson, Houcarde, Ingram, Neale, O'Malley, Simsarian, Stanton, Sundblad and Taxén, 2000]. The NAEYC [1996] reports on research indicating that children prefer using a computer with one or two partners, and that children constantly talk and cooperate at the computer. Social relationships are also promoted by children's preference for advice from peers rather than adults when it comes to computer use [Wartella and Jennings, 2000]. It can be beneficial for a child's self-esteem and status in a social group if he or she can demonstrate computer expertise and can help others improve their computer skills [Wartella and Jennings, 2000]. This is especially true for children who have problems socialising in other situations. Clements [1987] tells of a boy called Darius who was introverted, never talked and worked very slowly. A computer was brought into the classroom and Darius immediately showed interest, spending ninety minutes at it on the first day. While other children used Logo on the computer he watched and started helping them when they struggled. His general work pace doubled, he moved to the high reading group and he started participating in class discussions. His success at the computer gave him self-confidence that was not there before.

Children are far more active in their display of emotions – vocally and through facial expressions – in front of a computer than when they watch television [Clements, 1986; Clements, 1987]. Clements [1987] reviews several studies that found computer play to be superior to other play areas in facilitating social interaction and cooperation, friendship formation and group play. Some researchers believe that computer use facilitates only some forms of interactive behaviour, for example, problem solving, but not interactive play [Clements, 1987]. The type of software also plays a role in how behaviour is influenced. For example, a drawing program facilitates concentration, planning and social engagement more than a face construction and a counting program. Kull [1986; cited by Clements, 1987] found that first graders often display peer tutoring modelled on their teachers' teaching strategies and that this was particularly effective when the assistance was requested. Collaboration at a computer require that children interact with each other to come to an agreement on what should be done during an activity, thereby promoting co-construction of solutions to problems [Siraj-Blatchford and Siraj-Blatchford, 2001].

5.4.1.2 Encouraging Collaborative Use of Technology

Desktop and notebook computers are designed for use by a single user, which is not conducive to successful collaboration of more than one user at the same computer. Users can collaborate from different computers, but this kind of collaboration is not really relevant for the age group under discussion. Benford et al. [2000] believe that enabling young children to collaborate is not enough. They should subtly be encouraged to collaborate, but never forced. Benford et al.'s idea is that children should discover for themselves the benefits of working together and therefore, if collaboration is desired, the activities should be designed so that children will gain something by choosing to work together.

Benford et al.'s [2000] research was done in the context of developing a collaborative storytelling system, *KidStory*, that uses Single Display Groupware (SDG) as basis. SDG allows several users to interact with a single display and multiple input devices. *KidStory* was developed from two existing products, namely *KidPad* [University of Maryland HCI Lab, 1997] and *The Klump*. *KidPad* is a shared 2D drawing tool with extensive zooming capabilities. Story construction relies on zooming and spatial structure. More than one user can manipulate the interface simultaneously using more than one mouse, meaning that two users can grab two different tools at the same time. *The Klump* uses a 3D modelling approach whereby an amorphous object can be stretched, textured and coloured. It makes sounds while changing form. Figure 5.4 shows screen shots from The Klump and KidPad.

From their research using *KidPad* and the *Klump*, Benford et al. [2000] learnt that it is difficult to encourage children to collaborate if their natural tendency is to work alone or to compete. Children with these characteristics do not share their ideas and disregard other children's efforts. Benford et al.'s aim was to design interfaces that encourage even these children to work together. To them encouraging collaboration without enforcing it means that a child could achieve the outcome of a task on his or her own, but that doing it with someone would make it easier and more fun. Or, by working together, children could achieve subtle extensions and variations on the actions. They re-designed *KidPad* and *Klump* to encourage collaboration.

To *KidPad*, for example, they added the following functionality: when two children use different colour crayons close together, it creates a filled area in a colour obtained by mixing the two selected crayon colours. The idea is that this interesting and not completely predictable effect will be intrinsically encouraging. For most of the tools they added special behaviour with multiple users as a natural extension of the behaviour with a single user. Two or more children can simultaneously shrink and expand an object in different directions with very interesting effect. When two children use the eraser tool together it immediately erases the whole object, whereas a single user has to move across the whole object to erase it.

To add collaboration encouraging elements to the *Klump*, the designers took away a single user's ability to pull out a group of vertices and allow an individual to pull only a single vertex. When more than one user pull at vertices that are close together, the effect is the same as pulling a group of vertices. Also, by pressing buttons for facial expression and texture change simultaneously, children can achieve combined textures and new facial expressions. *Klump* also reacts with sounds that provide feedback on collaborative efforts [Benford et al., 2000].

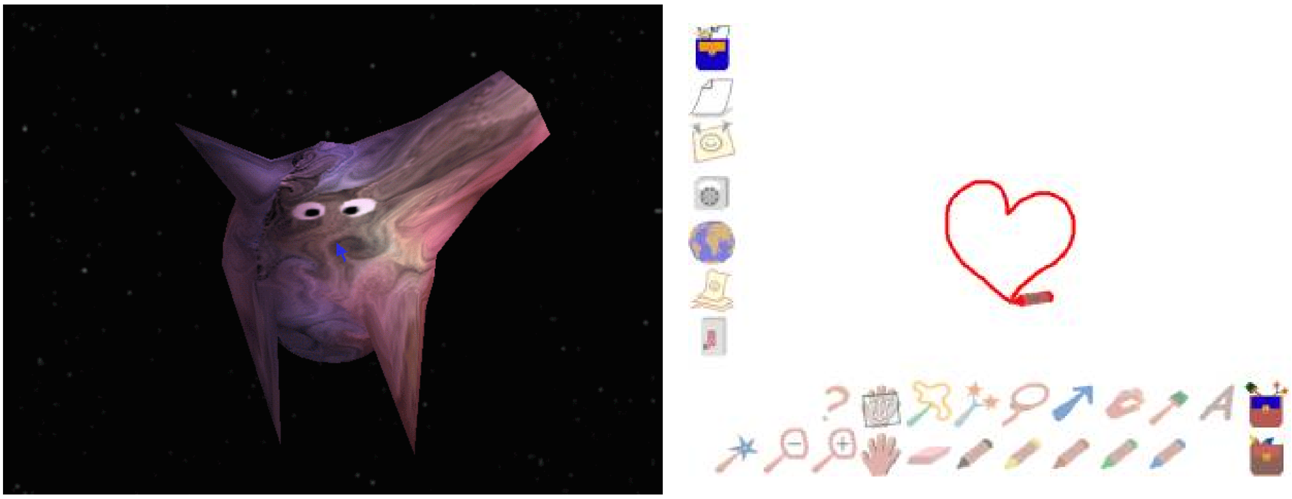


Figure 5.4 Screen shots from The Klump and KidPad respectively [HCILab, 2008]

A main lesson learnt is that the enhanced effect of collaboration versus lesser effect of individual use should be obvious and noticeable in advance [Benford et al., 2000]. The informal evaluations done on the new versions of *KidPad* and *Klump* showed that it took long before children saw the benefit of collaboration. It took considerable experience for them to collaborate using the software [Benford et al., 2000].

Peer collaboration and support and built-in support will again come up in Chapter 6 when I discuss interaction environments.

T81	Children prefer advice from peers rather than adults with regard to computer use. Technology that encourages collaboration between peers will promote social relationships.
T82	The type of software also plays a role in how social behaviour at a computer is influenced. For

example, drawing programs have been found to facilitate social engagement more than face construction programs.

T83 Design should encourage co-construction of solutions to problems so that children are required to interact with each other to come to an agreement on what should be done.

T84 Children should subtly be encouraged to collaborate, but never forced. Ideally they should discover for themselves the benefits of working together and therefore, if collaboration is desired, the activities should be designed so that they will gain something by choosing to work together – that is, doing it with someone would make it easier and more fun. (Be aware that it is difficult to encourage children to collaborate if their natural tendency is to work alone or to compete. Children with these characteristics do not share their ideas and disregard other children's efforts.)

T85 The enhanced effect of collaboration versus lesser effect of individual use should be clearly noticeable in advance.

T86 Some examples of how collaboration is encouraged in KidStory:

- Interesting and not completely predictable effects will be intrinsically encouraging. For example, when two children use different colour crayons close together, it creates a filled area in a colour obtained by mixing the two selected crayon colours.
- Most of the drawing tools' special behaviour with multiple users is a natural extension of the behaviour with a single user. For example, two or more children can simultaneously shrink and expand an object in different directions with very interesting effect, and when two children use the eraser tool together it immediately erases the whole object, whereas a single user has to move across the whole object to erase it.
- By pressing buttons for facial expression and texture change simultaneously, children can achieve combined textures and new facial expressions.
- Sound is used as feedback only on some collaborative efforts.

T87 Collaboration does not come naturally with child users – it is something they have to learn to do.

5.4.2 Adult Support and Collaboration

Klein, Nir-Gal and Darom [2000] found that integrating adult mediation in preschool computer learning environments has positive effects on children's performance. According to Clements and Samara [2002], children benefit significantly from support by teachers who closely guide children's interaction and continually encourage, question, prompt and demonstrate. Clements and Samara [2002] report on two studies by Yelland (conducted in 1994 and 1998 respectively) that demonstrate the importance of scaffolding provided by adults. In the first study children were given instructions for specific tasks and received no further support. It was found that these children did not plan their actions, they rarely cooperated, became frustrated, lacked confidence and did not finish the given tasks [Clements and Samara, 2002]. In the second study, the teacher provided scaffolding in the form of structured, open-ended tasks, arranging group discussions about problem-solving strategies, encouraging collaboration between the children and suggesting that they think about and discuss their plans before doing an activity on the computer. These children planned, collaborated, could explain their strategies and worked efficiently [Clements and Samara, 2002].

Klein et al. [2000] examined the effects of three types of adult interaction on the problem solving and cognitive performance of preschool children. They used Feuerstein and Feuerstein's [1991; cited by Klein et al. 2000] theory of 'structural cognitive modifiability' and 'mediated learning' as starting point. This theory identifies characteristic *Mediated Learning Experiences* (MLEs) and their effects on children's cognitive abilities. In mediated learning an adult modifies and changes the environment according to the child's needs, interests and abilities. These changes can involve changing the intensity, frequency, order, form or context of stimuli and arousing the child's curiosity, vigilance and perceptual acuity [Klein et al., 2000]. They can also try to improve the child's cognitive ability in terms of temporal, spatial and cause-effect relationships. Based on different studies by Feuerstein, Feuerstein and Klein, the following five basic criteria of MLEs were defined [Klein and Nir-Gal, 1992; Klein et al., 2000]:

1. Focussing (intentionality and reciprocity): This involves adult behaviour that attempts to change the child's perception or response and includes actions such as selecting, exaggerating, accentuating, scheduling, grouping, sequencing or pacing.
2. Affecting (exciting): Verbal or non-verbal appreciation or affect.
3. Expanding (transcendence): Adult behaviour that may improve cognitive awareness.
4. Encouraging (mediated feelings of competence): Verbal or non-verbal expression of satisfaction with specific components of a child's behaviour.
5. Regulating: Mediated regulation of behaviour.

Through mediation, children gradually internalise the MLE processes, so that they can start using them independently and modify their own cognitive systems through self-mediation [Klein et al., 2000]. Klein et al. 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:

1. Mediation – focussing, affecting, expansion, encouragement and regulation of behaviour.
2. Accompaniment – availability of an adult to answer children's questions.
3. No assistance – only technical or basic instructions provided at the beginning of a new activity.

They tested the types of guidance on 150 randomly selected children aged five and six from thirty different kindergartens in the south of Israel. The adult mediators used in the study were all kindergarten teachers and the study was conducted as part of the regular school activities. Participating teachers were trained extensively for the experiment. Two types of software was used, namely Logo and computer game software.

The researchers 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. The effect of mediation is greater when the children use Logo than when they use games software. The reason for this may be that Logo is an open program that does not dictate user actions and invites spatial mediation. Gender, ethnic origin and parents' level of education had no effect on the results and there were no differences between children who had access to computers at home and those who did not [Klein et al., 2000].

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. One 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. They have not learnt to focus their attention and plan their actions [Klein et al., 2000]. The implication of Klein et al.'s findings is that children using computers without adult 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.

The question is now whether adult or peer support can be replaced by support built into the system. I explore this in the next section.

T88 Children benefit significantly from support by teachers who closely guide interaction and continually encourage, question, prompt and demonstrate.

T89 When an adult provides scaffolding in the form of

- structured, open-ended tasks,
- arranging group discussions about problem-solving strategies,
- encouraging collaboration between the children and
- suggesting that they think about and discuss their plans before doing an activity on the computer,

children plan better, collaborate more, and can explain their strategies and work at their tasks efficiently.

T90 In mediated learning an adult modifies and changes the environment according to the child's needs, interests and abilities. The changes can involve

- changing the intensity, frequency, order, form or context of stimuli,
- arousing the child's curiosity, vigilance and perceptual acuity, and
- improving the child's cognitive ability in terms of temporal, spatial and cause-effect relationships.

T91 Five basic criteria of mediated learning experiences (MLEs) defined by Klein et al. are:

- Focussing (intentionality and reciprocity): this involves adult behaviour that attempts to change the child's perception or response and includes actions such as selecting, exaggerating, accentuating, scheduling, grouping, sequencing or pacing.

- Affecting (exciting): verbal or non-verbal appreciation or affect.
- Expanding (transcendence): adult behaviour that may improve cognitive awareness.
- Encouraging (mediated feelings of competence): verbal or non-verbal expression of satisfaction with specific components of a child's behaviour.
- Regulating: mediated regulation of behaviour.

T92 Through mediation, children gradually internalise the MLE processes, so that they can start using them independently and modify their own cognitive systems through self-mediation.

T93 Children who use computers at home without adult support may internalise a trial-and-error way of solving problems without any conceptualisation, as they do not learn to focus their attention and plan their actions.

T94 Through mediation children should learn to focus on a problem, to seek and receive precise information, to compare different perceptions and to plan their actions.

5.4.3 Built-in Support Mechanisms

Klein and Nir Gal [1992] suggest different ways in which mediation can be incorporated into the construction of software. They say that programs should provide children with immediate feedback about their performance and give specific reasons for success or failure. Providing encouragement is not sufficient – children should develop awareness about the processes underlying success or failure. Telling the child what specific actions or choices led to the correct or incorrect result will help the child to generalise from an experience to future ones. The content of the program should progress from simple to complex and from concrete to abstract. Specific mediation variables that can potentially be incorporated into software are:

- focussing (ensure that the child focuses on the right interface element),
- affecting (focus the child's attention on the concepts he or she used to solve the problem), and
- expanding and encouraging (through immediate vocal, musical and/or visual feedback).

To these, Klein et al. [2000] added regulation of behaviour.

Klein and Nir Gal [1992] compared adult mediation in children's use of software that develops analogical thinking skills with built-in adult-like support. In line with the theoretical and empirical knowledge about adult-child mediation, the software included individualised, flexible and responsive elements with mediational feedback. The subjects were between four and a half and six years of age and were divided into three groups. One group used the software with adult mediation, one used the software without adult mediation and the control group did not use the software. They found that the two groups using the software fared equally well on the post test and both significantly better than the control group. The group who received adult assistance more often stopped to think about their actions than the group using the software without assistance. The latter group displayed more trial-and-error actions. Klein and Nir Gal [1992] demonstrated that the basic characteristics of adult-child mediation can successfully be incorporated into computer activities in order to help children perform these without adult assistance.

According to Jackson, Krajcik and Soloway [1998], scaffolding that is built into software makes it possible to cater for users with diverse skills, backgrounds and learning styles in one product. It can also provide individualised support that adapts to a single user's development. To adapt to a user's improving skills, scaffolding should gradually decrease to allow the user to work more independently. Jackson et al. [1998] refer to this reduction in scaffolding as 'fading'. Based on the research on adaptable and adaptive interfaces, they distinguish between two ways in which scaffolding can be implemented: as support that changes automatically based on a model of the user's understanding (adaptive), or as support that is faded by the user (adaptable). Both approaches have problems. It is difficult to construct an accurate model of the user, but it may also be difficult for a user to decide when to fade scaffolding. In order to help the user to make fading decisions, Jackson et al. suggest that the software should allow self-evaluation that helps users to judge their own progress and understanding. Users should know exactly what the different scaffolding options are. Although Jackson et al. [1998] used older children in their experiments, the results can be transferred to software for younger children.

Jackson et al. [1998] believe that designers should use a combination of adaptable and adaptive fading and developed a design approach called *Guided Learner-Adaptable Scaffolding* (GLAS), that entails fine-grained scaffolding of different types. The user controls the fading, but with guidance from the software. Jackson et al. [1998] describe their implementation of fading with three types of scaffolding in a product called *TheoryBuilder*. The three types are:

- Supportive scaffolding that supports a task without changing the task itself and includes guiding (through messages that appear when the software detects that the user needs advice), coaching and modelling (by providing examples that explain concepts). Guiding scaffolding allows fading by displaying a button that the user can click to switch off the support. Coaching and modelling examples only appear on the user's request, so they fade by not being used.
- Reflective scaffolding that encourages users to think about a task before doing it. It doesn't change the task, but asks the user to provide plans, predictions or evaluations. Fading involves reducing the requests for reflection.
- Intrinsic scaffolding that is built into tasks by, for example, starting at an easy level and gradually increasing the complexity of the tasks. Fading is implemented as changes in the task.

The Sesame Workshop products provide examples of how built-in supportive and intrinsic scaffolding can be provided to younger children. The well-known Muppet characters act as mentors and guides in an attempt to create an environment of 'extended engagement and dialogue between the child and the Muppet character' [Revelle, 2003]. One feature of their software is the use of scaffolding to support learning. This is particularly useful when users have different levels of ability [Revelle, 2003]. They incorporate scaffolding into their software in two ways. The first is to include multiple levels of difficulty for each activity. The beginning levels are designed to address the most fundamental skills involved. From there difficulty is increased by adding one or more higher-level skills per level. In the more advanced levels they provide added challenge by increasing the level of cognitive skill and not only the physical demands of, for example, manual dexterity and eye-hand coordination. The second way in which they include scaffolding is by providing hints when a child makes an error. A series of hints guides the child to the correct answer. The challenge for designers is to structure the hints to give the child just the right amount of support and not more. As the child becomes more proficient, the supports must gradually be removed until the child can succeed on his or her own. No fading options are available in these products.

5.4.3.1 Scaffolding Children's Understanding of Temporal Systems

Masterman and Rogers [2002] investigated the possibility of supporting children's understanding of historical chronology by integrating the children's own experience of chronology with knowledge of historical information. Based on a thorough analysis of the cognitive issues involved, they developed a prototype of an interactive multimedia program that would scaffold young children's understanding of historical chronology. Scaffolding is built-in and falls in Jackson et al.'s [1998] class of intrinsic scaffolding.

To develop temporal awareness, between the ages of six and eight, children need to develop the following skills:

- Give the correct sequence in which events occur.
- Incorporate cyclical patterns (for example, the days of the week) into the bigger time system, so that they can understand that two Tuesdays are similar but also different.
- Co-ordinate different temporal systems (for example, days with weeks).
- Link temporal systems to number concepts.

Understanding the chronological concepts and temporal systems of their everyday lives does not necessarily mean they can apply these concepts to historical events [Masterman and Rogers, 2002]. Therefore, the aim of the software was to help children recognise the conceptual errors in their reasoning and assist them to overcome these. Factors that make it difficult for children to reason about temporal relationships are:

- The difficulty to link temporal systems to numbers. Young children can sequence events from the past if these are presented as strongly contrasted artifacts or pictures rather than dates.
- The abstract nature of conventional time systems that are represented as lists of seemingly arbitrary names.
- Inability to enact the passage of time or to reflect on the course of events in order to become aware of patterns.
- They have not developed the ability to use time charts and only use verbal and possibly image representations to reason about time.

What children need are explicit representations through which they can better understand the relationships between the different temporal systems and the ability to use these representations to reason about temporal concepts [Masterman and Rogers, 2002]. With their interactive multimedia program, Masterman and Rogers wanted to provide children with computer-based representations of chronological events with which they can interact in ways not available in the traditional classroom. With this they hoped to bridge the gap between children's everyday experience and abstract history. They used 'cognitive interactivity' as the theoretical basis for their design. This approach emphasises how new information is integrated with existing knowledge and attends to the cognitive benefits and costs of different forms of representation. They specifically used the concept of computational offloading.

Computational offloading refers to the extent to which external representations reduce or increase the cognitive effort required to understand or reason about what is represented. High computational offloading involves extensive scaffolding and requires minimal effort from the learner, while low computational offloading lacks scaffolding and requires more cognitive effort [Masterman and Rogers, 2002]. The main forms of cognitive offloading are re-representation, graphical constraining and temporal and spatial constraining. Using re-representation, a different external representation of the same abstract structure can simplify or complicate its interpretation. For example, relative proportions of daily activities can be identified more easily from a diagram than from a textual description. In graphical constraining, graphical elements can constrain reasoning about the concept being represented. For example, with a circular time-chart it would be easier to make inferences about recurrence (midnight is also the start of a new day) than with a time-line. Temporal and spatial constraining involves representations that make relevant aspects more prominent when distributed over time and space.

Masterman and Rogers [2002] identified external representations that could provide the correct amount of computational offloading to include in their software. They selected a metaphor of time-travel in which chronological sequence is represented as a route along which an individual travels. A road map represents the dynamic and temporal dimensions and a winding road makes it possible to display a whole 'route' on a single screen. The learner is a traveller in a time machine represented as an icon on the map. Events or people (also represented by icons on the map) appear on the road in the correct chronological order and children can interact with them by clicking on the icons. When the child clicks on an icon, the time machine moves back or forth along the road to the selected person or event. As it passes other icons, the colour of these icons changes to indicate whether they are from the past (full colour) or the future (half tone). The distances between the icons reflect the number of years that passed between the events and the speed at which the time machine travels is adjusted to reinforce the illusion of time.

To counter the specific difficulties that young children have when dealing with historical time, Masterman and Rogers included the following features in their program:

- Linear structures to represent a unique sequence of events and circular structures to represent recurring properties of a temporal system (for example, the seasons of the year).
- Signposts on the roadmap show 'number of years ago' rather than the dates of the events, since children are more comfortable with the concept 'how long ago'.
- Animated travels of the time machine along the road make up for children's lack of understanding of the notion that events happened in succession.

- Visibility and accessibility of elements to assist the learner in drawing inferences that may not be obvious but are necessary for the learning task.
- Cognitive tracing that allows the child to develop his or her own understanding of the content by modifying or annotating the representation.
- Games that prompt inference about similarities and differences between past and present events or people.
- 'Learning through doing' activities to support cognitive tracing (for example, rearranging jumbled elements).
- A 'log book' in which children paste images and narrative passages.

They assumed the software would be used in the classroom context and that teacher assistance would be available while the children are using it. In their evaluation of the software prototype, Masterman and Rogers found that children were comfortable with the time-travel metaphor and could accomplish activities with an abstract structure that they were familiar with. The interactive activities (sequencing task and spotting the anachronisms) proved successful in stimulating reasoning. Teachers felt that the program could add value to their teaching and likes the interactive learning activities.

A possible problem was that children may understand the concepts only at the level at which it was represented and would not be able to reason about them in a general way. Masterman and Rogers [2002] addressed this by letting children move between an integrated set of representations that help them to understand the relationships between the representations.

5.4.3.2 On-Screen Support vs. Digital Toy Support

Working from the assumption that intellectual development requires activities that include tasks that are beyond a child's independent ability, Luckin, Connolly, Plowman and Airey [2003] studied the potential of digitally enhanced soft toys to offer collaborative support to children using technology. As peer discussion is regarded as a powerful way to implement scaffolding, they also investigated whether such toys can encourage collaboration between peers.

A survey of studies on screen-based scaffolding showed that children do not always use the available help effectively [Luckin et al., 2003]. Effective collaborative assistance does not only depend on the content of the help provided, but also on the way the help is made available. Luckin et al. [2003] hypothesised that placing the support outside the computer in a helpful toy may enhance scaffolding and encourage peer collaboration.

Luckin et al.'s [2003] study formed part of the CACHET (Computers and Children's Electronic Toys) project that involves the use of free-standing soft toys that can move, speak and respond to touch. Interaction occurs through sensors in the toy's body that control its different functions. The toys can be linked to a desktop computer via a wireless connection. With a digitised vocabulary of over four thousand words, such a toy can play simple games. Children can play directly with the toy or they can play an on-screen computer game with the toy providing feedback and support. They can also play the computer game without the toy, in which case a screen-based icon representing the toy provides the support. To request help, the child squeezes the toy's ear or clicks on the icon.

Using the Arthur and DW toys[9], Luckin et al. [2003] conducted three studies: one in children's homes, one in a preschool classroom and one at an independent computer club. The average ages of the children were six years and two months, four years and seven months and five years and five months respectively. They followed the same procedure in each case. To begin with they gave the children brief instructions, made them aware of the help available and how to get help.

Luckin et al. [2003] found that children rarely ask for assistance. When they do, it is initially from a parent, teacher or researcher and often concerns clarification of what the character said. Only when they have become comfortable with the toy do they ask for its help and this usually happens when prompted by an adult to do so. If children take notice of the help offered by the toy and they succeed at the task concerned, it causes much pleasure (even if they were prompted by an adult to ask for help). If the toy gives incorrect help, they ignore any further assistance. Incorrect advice from the on-screen character evoked abusive behaviour towards the character. In general, less advice was taken from the on-screen toy than from the real toy. Ineffective or irritating feedback caused irritation and distraction and the children did not appreciate phony praise or flattery. The type of task also played a role in determining whether children requested help – discrete tasks with clear goals seem to encourage children to ask for help.

The study showed that children aged four-and-a-half to six years are able to use the integrated interfaces successfully, that tactile toys encourage social interactions and collaboration between peers, but that the toys did not succeed as collaborative learning partners. The help provided by the toys was inadequate and not always appropriate [Luckin et al., 2003].

T95 Providing encouragement is not sufficient – children should develop awareness about the processes underlying success or failure. Telling the child what specific actions or choices led to the correct or incorrect result will help the child to generalise from an experience to future ones.

T96 The content of the program should progress from simple to complex and from concrete to abstract.

T97 Specific mediation variables that can potentially be incorporated into software are

- focussing (ensure that the child focuses on the right interface element),
- affecting (focus the child's attention on the concepts he or she used to solve the problem),

[9] I discuss these specific toys again in Chapter 6, section 6.4.1.1 where I will explain the technical interaction features. Here I focus on the results with regard to collaboration and support.

- expanding and encouraging (through immediate vocal, musical and/or visual feedback).

T98 Scaffolding can be provided by including multiple levels of difficulty for each activity. The beginning levels address the most fundamental cognitive skills involved and from there difficulty is increased by adding one or more higher-level skills per level.

T99 Scaffolding can also be provided through hints when a child makes an error. A series of hints guides the child to the correct answer. The hints should give the child just the right amount of support.

T100 As the child becomes more proficient, the support must gradually be removed until the child can succeed on his or her own.

T101 Two ways in which scaffolding can be implemented:

- as support that changes automatically based on a model of the user's understanding (adaptive) or
- as support that is faded by the user (adaptable).

Both approaches have problems. It is difficult to construct an accurate model of the user, but it may also be difficult for a user to decide when to fade scaffolding.

T102 To help the user to make fading decisions the software should allow self-evaluation that helps users to judge their own progress and understanding. Too many fading options will confuse users.

T103 Users should know exactly what the different scaffolding options are.

T104 Use a combination of adaptable and adaptive fading.

T105 Three types of scaffolding are:

- Supportive scaffolding that supports a task without changing the task itself and includes guiding, coaching and modelling. Guiding scaffolding allows fading by displaying a button that the user can click to switch off the support. Coaching and modelling examples only appear on the user's request, so they fade by not being used.
- Reflective scaffolding that encourages users to think about a task before doing it. It doesn't change the task, but asks the user to provide plans, predictions or evaluations. Fading involves reducing the requests for reflection.
- Intrinsic scaffolding that is built into tasks by, for example, starting at an easy level and gradually increasing the complexity of the tasks. Fading is implemented as changes in the task.

The following set of guidelines (T106 to T112) relate specifically to helping children understand temporal systems and is based on Masterman and Rogers [2002].

T106 Technology that support development of temporal awareness in children between the ages of six and eight, can address the following skills:

- giving the correct sequence in which events occur,
- incorporating cyclical patterns (like the days of the week) into the bigger time system, so that they can, for example, understand that two Tuesdays are similar but also different,
- co-ordinating different temporal systems (for example, days with weeks), and

- linking temporal systems to number concepts.

T107 Children need explicit representations through which they can better understand the relationships between the different temporal systems, and the ability to use these representations to reason about temporal concepts.

T108 Using re-representation, a different external representation of the same abstract structure can simplify or complicate its interpretation. For example, relative proportions of daily activities can be identified more easily from a diagram than from a textual description.

T109 Using graphical constraining, graphical elements can constrain reasoning about the concept being represented. For example, with a circular time-chart it would be easier to make inferences about recurrence (midnight is also the start of a new day) than with a time-line.

T110 Using temporal and spatial constraining, representations can make relevant aspects more prominent when distributed over time and space.

T111 Difficulties that young children have when dealing with historical time can be addressed through:

- Linear structures to represent a unique sequence of events and circular structures to represent recurring properties of a temporal system (for example, the seasons of the year).
- Signposts on a roadmap show ‘number of years ago’ rather than the dates of the events, since children are more comfortable with the concept ‘how long ago’.
- Visibility and accessibility of elements to assist the learner in drawing inferences that may not be obvious but are necessary for the learning task.
- Games that prompt inference about similarities and differences between past and present events or people (for example, children have to decide whether a statement such as ‘When Pepys woke up he looked at his watch’ is anachronistic).
- ‘Learning through doing’ activities to support cognitive tracing (for example, rearranging jumbled elements).

T112 To ensure that children are able to reason about temporal concepts in a general way, provide an integrated set of representations for the child to move between to develop an understanding of how they relate to each other.

The next set of guidelines refers to collaboration with an electronic toy that offers support during a computer game:

T113 Effective collaborative assistance does not only depend on the content of the help provided, but also on the way the help is made available.

T114 Even when children are aware that they could ask a toy or screen-based agent for help, they rarely do. Initially they prefer to ask help from a parent, teacher or researcher. Only when they have become comfortable with the toy or agent do they ask for its help (usually when prompted by an adult to do so).

T115 If children take notice of the help offered by the toy and they succeed at the task concerned, it causes much pleasure (even if they were prompted by an adult to ask for help).

T116 If the toy gives incorrect help, they ignore any further assistance. Incorrect advice from the on-screen character may evoke abusive behaviour towards the character.

T117 In general, children take less advice from the on-screen agent than from a real toy.

T118 Ineffective or annoying feedback causes irritation and distraction and children do not appreciate phoney praise or flattery.

T119 The type of task plays a role in determining whether children request help – discrete tasks with clear goals seem to encourage children to ask for help.

T120 Children aged four-and-a-half to six years are able to use integrated interfaces successfully.

T121 Tactile toys encourage social interactions and collaboration between peers, but that the toys will only succeed as collaborative learning partners if the help they provide is adequate and appropriate.

5.5 Negative Effects of Technology on Children's Development

In this section I briefly discuss some of the concerns that have been raised in the literature about young children's use of computers. Very few of these concerns have been verified by research – especially with regard to children aged eight and younger. Possible problems raised are physical harm, addiction to computer games, social isolation and impeding on activities that may benefit them more.

5.5.1 Common Complaints

Brouwer-Janse, Suri, Yawitz, Vries, Fozard and Coleman [1997] cites Fulton Suri who provided the following list of unfavourable aspects of children's interaction with computers:

- Children play alone and when they do use a computer together, only one child is in control.
- Computer interactions are limited with respect to all senses except the visual and auditory, repetitive button pushing are 'impoverished forms of tactile interaction' and the restricted body position may affect their postures.
- Computer interaction proceeds sequentially from one application to another with no fluidity between the contexts.
- The need for adult help when a problem occurs during a computer game may interfere with the child's play activity, whereas a broken truck or doll can stimulate a new direction in a game.

The literature surveyed in this chapter display a very clear awareness of all these potential problems and most, if not all, researchers cited are committed to providing children with technology that address or overcome the problems. In section 5.4.1 above I provided lots of evidence that contradict Fulton Suri's concern that computer games foster solitary play. New tactile and movement interfaces (that I will discuss in Chapter 6) should set those who worry about a lack of sensory input and physical activity at ease. Broken trucks and dolls very often lead to tears and tantrums that also require adult attention, so Fulton Suri's comparison in this regard is unconvincing. In the next sections I elaborate on these issues and a few other potential problems with children's use of technology.

5.5.2 Physical Effects

Physical effects of prolonged computer use include repetitive strain injuries, addiction and sedentary lifestyles [Plowman and Stephen, 2003]. Wrist and neck injury, eyestrain, obesity and toxic emissions and radiation are problems raised in the infamous *Fool's Gold* report [Alliance for Childhood, 2000]. Obviously, the position and size of the display screen, keyboard and other devices are important in this regard. Children should be taught from the start not to type with the index fingers only and should be discouraged to sit in the same position for extended periods of time.

Concerns have also been raised that computer use may contribute to obesity in children, but no direct evidence for this have been found [Attewell, Suazo-Garcia and Battle, 2003]. Attewell et al. found some correlation between heavy use of computers (more than eight hours a week) and higher weight, but it is unclear whether overweight children just prefer computer use to other activities, or whether the time spent at the computer leads to weight increase. No longitudinal study has been done to determine the causal direction.

Addiction to computer games is mostly a problem with older children and therefore not relevant to my study.

5.5.3 Effects on Cognitive Development

Some researchers claim that computer use requires cognitive resources that could have been applied to other types of learning that are more beneficial, especially with young children [Plowman and Stephen, 2003]. The Alliance for Childhood [2000] includes lack of creativity, stunted imagination and poor language and literacy skills and attention deficit as intellectual hazards of computer use. It also maintains that programs such as Logo expects too much (in terms of cognitive processing) too soon from children.

Blakemore and Frith [2000], however, says that there is no evidence that preschool children's brains undergo more changes than those of adolescents, so young children is not more at risk in this regard. They also maintain that there is no evidence that early introduction to computers are more detrimental or more beneficial for brain development than early introduction of language, mathematics or music.

Attewell et al. [2003] conducted a comprehensive study using data provided by the US Panel Study of Income Dynamics (PSID) of daily activities of 1680 children aged four to thirteen years. Their aim was to determine 'whether computing enhances or undermines young children's cognitive and educational development and well-being' (p. 292). They found that children who used computers moderately at home (less than eight hours a week) were better at letter-word recognition, reading comprehension and mathematical calculations, although not dramatically so [Attewell et al., 2003]. They also found that children who spend less than eight hours with a computer, read significantly more than children who do not have computers at home (family background and other possible influences have been controlled for). Attewell et al.'s study shows that computer use does not keep children away from other activities that support development.

5.5.4 Effects on Social Development

The main concern with regard to social development is that too much time spent in front of a computer can interfere with the development of social relationships [Shields and Behrman, 2000]. Also, children may be exposed to violent, sexual or commercial content that they are not emotionally ready to deal with. The Alliance for Childhood [2000] also mentions lack of self-discipline and self-motivation as possible outcomes of computer use.

Fogg, Cypher, Druin, Friedman and Strommen [1999] discussed ethical concerns with regard to tangible toys such as Actimates Barney (see section 6.4.2 of Chapter 6). In his position statement in Fogg et al. [1999], Allen Cypher points out that users ascribe 'high degrees of emotional connection to a system that exhibits a slight degree of empathy' (p. 91) and therefore finds it problematic that Actimates Barney says things such as 'I really like you' or 'you're my special friend'. He worries that this may lead children to form emotional connections with electronic toys instead of real emotional connections with other people. Strommen, in his position statement [Fogg et al., 1999] dismisses these kinds of fears as groundless, saying that children are much more sophisticated in their use and perceptions of technology than people think.

The research findings that I have discussed in section 5.4.1.1 [Benford et al., 2000; Clements, 1987; NAEYC, 1996; Wartella and Jennings, 2000] give sufficient reason to believe that computers are good for young children's social development rather than detrimental. Children prefer to use computers with others, it increases their talking to one another and helping others with computer activities can improve a child's self-esteem and status in social group. The Alliance for Childhood [2000] report is based on the false assumption that computer use always involves a single child sitting in front of a badly positioned computer, isolated from other people, searching the Web for answers to teachers' questions [Harvey, Accessed 23 Oct 2007].

Subrahmanyam et al. [2000] report that moderate use of computer games has no significant impact on friendships and family relationships. They do, however, warn that the amount of violence in computer games increases from year to year and that parents are often unaware of the extent of the violence in these games. In general, I have not found evidence in the literature that violent content is a problem in games aimed at children of eight years and younger.

5.5.5 The Digital Divide

The digital divide refers to unequal access to technology that separates people into those who have it and those who do not [Attewell et al., 2003]. Children of those who do not have access to computers are in danger of being excluded from the new technological society. If society, and educators in particular, increasingly assume that children have access to computers, those who do not will suffer educational as well as social disadvantages [Attewell et al., 2003]. This problem can only be solved if computer technology is somehow made available to all, and many initiatives are under way in attempt to make this happen, for example the *Digital Doorway* project in South Africa [Meraka Institute, 2007], MIT's *One Laptop Per Child* project [MIT, 2007] and the ongoing *Hole-in-the-Wall* project in India [Mitra, 2003].

In developing countries where technology and internet access are relatively widely available, there are still problems with fast internet access. Specific problems are associated with online interaction, especially where bandwidth is limited. Since they lack reading skills, children's interaction with software depends largely on sound and graphic elements [Fisch, 2004]. Sound files and some graphics can take a long time to download, making the use of such software unusable by a large part of the potential user population.

Literacy levels also contribute to the digital divide. Many computer based-products aimed at young children are designed for use with the help of an adult [Fisch, 2004]. Children using these products without the necessary guidance cannot reap the intended benefits and it may even teach them incorrect strategies for problem solving. In a developing country such as South Africa, many children's parents are completely illiterate and the idea of a parent helping with computer-based activities is far-fetched.

5.5.6 Guidelines that Address these Concerns

The condemnation of introducing technology to children at a young age seems largely unfounded, but the critics of early use of technology do raise issues that designers should take note of. The design-related factors that have emerged so far in this thesis already address many of the problems mentioned here, and more will be addressed in the next two chapters. Problems that I have not dealt with in this section are those associated with the Internet and the World Wide Web (WWW). I discuss the WWW as an interaction environment for children in the next chapter and will then also look at the possible problems and how to address them.

T122 Design activities in a way that makes it possible for children to complete them in twenty to thirty minutes. The beginning and end of an activity should be clearly defined so that restricting their time at the computer does not interfere with an ongoing activity.

5.6 Conclusion

Brouwer-Janse et al. [1997] said: 'A major goal of our society is to have people who discover, who are creative and inventive. To achieve this goal, we need children who are active, who learn early to find out things by themselves; partly by their own spontaneous activity and partly through the materials we design for them' (p. 36). Children's most successful learning experiences happen when they are engaged in designing and creating things that are meaningful to themselves or to others around them [Resnick, 2000]. As was clearly illustrated in this chapter, computers and other technology provide a range of possibilities for creating and learning things that traditional materials cannot. Designers should therefore provide children with technology that does more than provide them with information. It should allow them to develop by giving them opportunities to create, experiment and explore.

In answer to the question ‘What can we learn from existing research into role of technology on skill development that can inform designers of technology for children aged five to eight?’, the research surveyed in this chapter yielded numerous potential guidelines for the design of technology.

The main scientific contributions of this chapter are:

- It provides a broad, methodical and functional review of the literature on the role of technology in children’s development of mathematical, thinking, problem-solving, reading, writing and storytelling skills.
- Knowledge contained in the detailed literature review has, where applicable, been translated into knowledge about the design of young children’s technology and presented in the form of design-related data elements. Together, this data provide expert knowledge on how to design technology that supports the development of the abovementioned skills. In Chapter 8, this knowledge will become part of the product of my study when the individual factors are analysed, reformulated as design guidelines and assimilated into the final framework of guidelines.
- Continuing with the research approach followed in Chapter 4, I demonstrated how a careful process of investigating existing research results (as opposed to empirical studies designed for this purpose) provides insight into, and practical guidance for, the design of children’s technology.

I continue my study of the literature on young children and technology in the next chapter, looking at research on available and envisaged interaction devices and environments for young children.

CHAPTER 6

Interaction Environments for Young Children

Table of Contents

6.1 Introduction

In Chapter 5 I began my investigation of the literature on young children and technology. There the focus was on the way technology use supports skill development. In this chapter I continue that investigation, now looking at the ways young children interact with technology. A substantial amount of research has been done with regard to input and output mechanisms and interaction environments specifically aimed at young children. My aim in this chapter is to provide a comprehensive survey of the results and, consequently, to identify guidelines for designing interaction environments for young children. The questions that guide the survey are:

- Which input and output mechanisms exits that are suitable for children aged five to eight?
- How should interaction environments for young children be designed to maximise the benefits of their interaction with technology?

The chapter is organised as follows: In section 6.2 I survey the research on the range of standard interaction devices and environments available, including the mouse, joystick keyboard and touchscreen. I also discuss the cooperative uses of these devices. Section 6.3 deals with the less conventional interaction mechanisms such as speech input and output, handwriting input, and movement input. Section 6.4 is devoted to technologies that use tangible interfaces, such as robotic pets, mixed reality environments and tabletop environments as well as affective aspects of interfaces for children. In section 6.5 I discuss a selection of environments that support children with disabilities and then briefly consider Web-based applications for young children in section 6.6. I conclude the chapter in section 6.7.

As before, I end each main section of this chapter with a summary of emerging factors that may influence design in shaded data boxes.

6.2 Conventional Input and Output Devices for Children

The standard input and output devices are the mouse, keyboard, joystick and touchscreen. Although much of the research with regard to young children's use of these devices is somewhat dated, most of the results are still relevant today. I review the research on each of these devices and then look at some comparative studies. I end this section with a discussion of how these devices can potentially support or hamper collaborative use of technology.

6.2.1 The Mouse

The mouse is still the most popular input device used by children. Although much has been written about young children's use of the mouse very little has been said about the influence of the size or shape on children's experience with it. According to Plowman and Stephen [2003] mice for children should be larger than the normal mice used by adults. This is contradictory to what Ager and Kendall [2003] found. They conducted a study with preschool children using a 'tiny mouse' half the size of a standard mouse. All the teachers who acted as observers during the study reported that children found it easier to use the small mouse [Ager and Kendall, 2003]. I could not find other reports on comparative studies in the literature and Plowman and Stephen did not support their opinion with relevant research.

In her review of the research, Inkpen [2001] picked up several problems relating to children's use of the mouse. For example, children have difficulty using a marquee-type selection because they struggle to choose the initial corner to begin the selection. They also find it difficult to maintain pressure on the mouse button. Inkpen [2001] further reports that boys and girls think differently about computers, have different motivations for using a computer, and different usage styles. An earlier study by Inkpen, McGrenere, Booth and Klawe [1997] revealed specific differences between the effect of mouse-use on boys and girls. I discuss their results in detail in section 6.2.6.

With regard to the problems with marquee-type selection, Berkovitz [1994] proposed a solution that involves changing the way the rectangle is formed when the user drags the mouse. In this method the user encircles the objects that need to be included in the selection with the mouse cursor. The marquee is formed around the outside edges of the encirclement [Berkovitz, 1994]. Clearly Berkovitz's solution was not widely accepted as, seven years later, Inkpen [2001] still mentioned the original marquee selection method as a problem.

Hourcade, Bederson and Druin [2004] did research on children's use of the mouse buttons and found that even five-year-olds may find it difficult to click the left button consistently. The reason for this is that most children achieve orientation with respect to themselves by the age of six and can only apply the concepts of left and right relative to other objects by the age of six. Based on this knowledge and the results of their research, Hourcade et al. [2004] suggest that software designed for preschoolers should not give the buttons different functionality. The best way to prevent frustration or confusion would be to give both mouse buttons the same functionality.

Younger children find it difficult to stop hand movement at a precise moment, hence they have problems to stop the movement of a mouse accurately enough to position the cursor on the target object [Revelle, 2003]. They may also find it difficult to focus their visual attention on the screen while their hand is performing the action of moving the mouse. In normal play behaviour children look at the part of the toy that they are manipulating with their hands. Research at Sesame Workshop suggests that mouse use can be made easier for young children by adapting the software [Revelle, 2003]. For example, hotspots must be large, widely spaced and with enough 'dead-space' between them; frequently used hotspots should be placed in a corner where it will be easier to stop the mouse on target; and click-and-carry is preferable to drag-and-drop (see section 6.2.2 below).

Strommen [1994] compared children's use of three mouse-based movement interfaces to determine which method is most appropriate for grade one and two children. He used a prototype of *Woods Visit*, developed by IBM and Children's Television Workshop. This software uses video footage filmed while moving along forest trails to provide children with a point-of-view interface to a virtual walk in the woods [Strommen, 1994]. Children choose between six connected trails and then have to search for animals hidden in the forest. The three interfaces designed for the study used the following implementations of movement:

1. Click go/click stop: To start moving forward children click once on an up-arrow icon. Movement along the trail continues until the mouse is clicked again and the navigation icons appear.
2. Hold and go: Here the child clicks on the up-arrow icon but holds the mouse button down to move forward. Movement stops when the user releases the button.
3. Slide and go: The child clicks on the up-arrow and releases the button. Movement starts when the mouse is moved forward on the table and ends when the mouse movement stops.

Testing a prototype of *Woods Visit* with ninety-four first and second graders working in pairs, Strommen [1994] found that the 'hold and go' interface was problematic because of the physical demand of keeping the mouse button down. The choice between 'click go/click stop' (CGCS) and 'slide and go' (SG) was not clear cut. SG worked well for searching for the animals as it created an impression of 'creeping'. Problems with SG are that children sometimes click the mouse button while still moving it with no effect, and there was more disruption of movement on the trails than with CGCS. CGCS had the least accidental disruption, but, because the children can take their hands off the mouse while movement continues, there were some conflict for mouse control between co-users. It seems then that the choice between SG and CGCS depends on the task and how the interaction style supports the objective of the interaction.

6.2.2 Drag-and-Drop vs. Point-and-Click

Inkpen [2001] conducted extensive research on the effect of mouse interaction style on children's performance and motivation when playing a computer game. Studies with adults have found that dragging tasks were slower and more error prone than tasks that use point-and-click [Inkpen, 2001]. None of these studies included children and they did not investigate gender influences.

Inkpen's [2001] experiments involved a puzzle-solving computer game called *The Incredible Machine* developed by Sierra On-Line Inc. Some subjects used the IBM-version and some the Macintosh version. The IBM-version uses a point-and-click interaction style. This means that 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 the Macintosh version, if the object is dropped in a position that is already occupied it causes and error and the object is returned to the original position and the user must pick it up there 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. Again, in the case of an incorrect placement, the IBM version will keep the object selected and allow the user a second attempt, while the Macintosh version will require the whole operation to be redone.

Inkpen [2001] performed two experiments. In the first she used girls only, some of which used the IBM version and the others the Macintosh version. The results of this experiment showed that girls were more successful at playing the IBM version than the Macintosh 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 and everybody did tasks on both versions of the game. The tasks were simplified so that all elements that might have influenced the results were removed. 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 kinesthetic connectivity of holding the mouse down to hold on to the object can help to reinforce the conceptual connectivity [Inkpen, 2001].

The final conclusions are that point-and-click interaction is faster than drag-and-drop and leads to fewer errors. Inkpen [2001] 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.

I01	Conflicting findings with regard to mouse size lead to the conclusion that mouse size is not really important for young children. They are able to use both the standard mouse and a smaller version.
I02	Marquee-type selection is difficult for young children as they find it difficult to select the initial corner. A more suitable way to select a group of objects would be to circle the objects with the mouse.
I03	Children younger than six find it difficult to distinguish between left and right and will therefore find it difficult to click the left mouse button consistently. It is advisable to give both mouse button the same functionality.
I04	Designers can help children to stop mouse movement on target by: • using large hotspots (clickable areas) that are widely spaced, and • placing frequently used hotspots in a corner where it is easy to stop the mouse.

I05	Point-and-click (or click-and-carry) is a quicker and more accurate way for children to move objects on the screen than drag-and-drop (with the mouse button held down). Most children prefer point-and-click.
I06	Drag-and-drop may be better for tasks where the kinaesthetic connection between holding the mouse button down and 'holding on' to the object involved contributes to successful performance of the task.
I07	In an interface where an object or character (or the user him or herself) moves along a trail and the user has to start and stop the movement, a click-and-go (click to start and click again to stop) or slide-and-go (move the mouse as long as the object has to move) interface is preferable to a hold-and-go (hold the mouse button down until movement must stop) interface.
I08	The choice between click-and-go and slide-and-go will depend on how well they respectively support the goals of the interaction.

6.2.3 Home Entertainment Controllers

The most common controllers for home entertainments systems are the joystick (figure 6.1a) and the button device used with Sony Playstation (see Figure 6.1b). These two types of controller share some features that differentiate them from the mouse and trackball, namely [Revelle, 2003; Revelle and Medoff, 2002]:

- The relationship between hand movement and cursor movement is indirect.
- Cursor movement starts when the user presses a key or pushes the joystick forward, and continues until a finger is lifted or the joystick repositioned. The cursor continues to move when the hand stops. To stop cursor movement another hand action is required. There is thus not a one-to-one correspondence between hand and cursor movement as there is with a mouse.
- The psychomotor task of stopping on a target is difficult for preschoolers, so that they often overshoot the target.

Sesame Workshop research showed that joystick and arrow key control devices are difficult for preschoolers to use [Revelle, 2003]. It is possible to reduce the problem with arrow key devices by adapting the software, but for the joystick no useful adaptations could be identified [Revelle, 2003; Revelle and Medoff, 2002].

Figure 6.1Joystick [Control Centre, 1999]

Figure 6.1b Sony PS 2 controller [Sony, 2008]

Revelle and Medoff [2002] hypothesised that adapting the software so that continuous cursor movement is broken up into discrete steps may reduce these problems for an arrow key device (such adaptation is not possible with the joystick). With each key press the cursor moves one step towards the target. They created three adaptations of the interface for the Sony Playstation 2 controller and tested children's performance with these interfaces. The three options were:

1. The 'arrow key' condition: Children used the up and down arrow keys to move a selection highlight between four objects arranged in a vertical column. Cursor movement occurred in discrete steps rather than in a continuous path to curb overshooting. To move down three squares, the child had to press the down arrow three times.
2. The 'colour/shape key, controller pattern' condition: Here he objects on screen were labelled with coloured shapes corresponding to the shapes and colours on the controller buttons. Children used a shape key to select an on-screen object. Furthermore, the objects on screen were arranged in the same visual cross-hair pattern than the keys on the controller.
3. The 'colour/shape keys, vertical column' condition: This was similar to the previous one, except that the on-screen objects were arranged in a vertical column.

Revelle and Medoff [2002] found that it is possible to create an interface for a home entertainment system controller that preschoolers can use by limiting the kind of input choices available. Very young children are quicker and more accurate with the colour/shape interfaces than with the arrow keys, but from age four, accuracy is not influenced by the interface. With the arrow keys they tend to just press the X key instead of moving the cursor with the arrow keys first. The arrow keys involve a two-step process while the colour/shape interface requires one step at a time.

The colour/shape interface is thus recommended, especially for very young children [Revelle and Medoff, 2002]. The arrow keys that move an on-screen highlight to select an option give designers more flexibility as more than four options can be presented. From the fact that four-year-olds could use the arrow keys, we deduce that children aged five and older will be able to use this interface with ease.

I will discuss these controllers further when I look at studies comparing the standard input devices in section 6.2.5.

I09 Young children find the joystick difficult to control and the often overshoot the target.

I10 The joystick is not suitable for tasks that require fine motor control such as tracing.

I11 With regard to arrow key devices such as the Sony Playstation controller, it is possible to design the interface to improve young children's interaction using the device by:

- limiting the input choices available, and
- matching the choices with the colour/shape icons on the keys of the controller. This means the choices must be limited to the number of keys available. If the options can be arranged on screen to match the arrangement of the keys on the controller, even better.

The child selects an option with one keypress.

I12 Another way to adapt the interface is to allow children to use the up and down arrow keys to move a selection highlight between four the choices. Cursor movement occurs in discrete steps rather than in a continuous path to prevent overshooting. (To move down three squares, the child had to press the down arrow three times.) To select and option the child then presses the X-key, for example. This allows for more options, but it requires more keypresses from the user.

6.2.4 Touchscreens

The direct relationship between the child's action and the on-screen effect when using a touchscreen makes it a suitable input device for young children [Scaife and Bond, 1991], but research shows that it is not necessarily more effective than the mouse. In a comparison between children and adults Romeo, Edwards, McNamara, Walker and Ziguras [2003] found that whereas adults learn to use a keyboard and a mouse faster than children, all ages learn to use a touchscreen at similar rates. For adults the touchscreen is the fastest and preferred input method, but it leads to the most inaccurate interaction. Romeo et al's [2003] aim was to examine how children perform when using a touchscreen to accomplish tasks such as selecting objects on the screen and dragging and dropping icons. Five groups of children with ages ranging from three to seven were observed by researchers and teachers over a seven week period.

With regard to developmental issues they found the following: children's competence with the touchscreen improved over time; all the age groups had difficulty in selecting, dragging and moving objects around the screen using a finger; and they generally fared better using a mouse. This is attributed to the children's level of perceptual-motor development and the fact that they are more familiar with a mouse as an input device. Performance was also influenced by the position of the touchscreen in the classroom – children had difficulty reaching the screen because of its height and some teachers placed the touchscreen workstation in a part of the classroom where it was not readily accessible.

According to Romeo et al. [2003], the two main factors that influenced input device preferences were the effectiveness of the relationship between the software and the device, and children's prior experience with the device. The software used in the study was designed for mouse and keyboard input and the small icons could not be selected and moved around effectively on the touchscreen. Some children realised that they needed a finer selection device and used the rubber tip of a pencil. The software with large icons and simple input requirements were the most suitable for touchscreen interaction. All of the children had some prior school experience with the mouse and keyboard and many had used these devices at home. There was a general tendency for the children to go back to the mouse after they have explored the touchscreen, although some observers thought that introducing the touchscreen to the children before they learn to use a mouse and keyboard may have caused them to favour the touchscreen. Ager and Kendall [2003] also found that preschool children progress faster if they are allowed to learn to control the mouse instead of using a touchscreen. In the short term the touchscreen seemed advantageous, but once children are proficient with the mouse, they seem to do better with it [Ager and Kendall, 2003].

Romeo et al. [2003] found that the touchscreen did not promote positive collaboration between children. I discuss this further in section 6.2.6 below.

I13	Children usually fare better moving objects around with the mouse than with a touchscreen. This may be due to familiarity with the mouse as touchscreens are relatively rare and children do not use them at home.
I14	Children's competence with the touchscreen improves over time, so designers can let them start off with some practice activities.
I15	Screen objects on a touchscreen need to be bigger than when a mouse is used as it is more difficult to make fine selections with a finger.
I16	Use simple input requirements when designing for a touchscreen.
I17	The effectiveness of the relationship between the task and the device will determine which device is best for the task. For tracing tasks the touchscreen is better than the mouse as there is a direct relationship between the required action and the on-screen effect.
I18	The touchscreen is not ideal for tasks that require collaboration between users.

6.2.5 Comparing Different Input Devices

Scaife and Bond [1991] compared children's use of the mouse, touchscreen and joystick, focussing on how the use of these devices is influenced by children's development from age five to age ten. In an experiment where children had to track a moving target at two different speeds, they found the following: all ages fared significantly better with the touchscreen than with any of the other devices; with increasing age children improved with the mouse and joystick, but not much with the touchscreen; improvement with age was more pronounced with the mouse than with the joystick; younger children did almost as well as the older children using the touchscreen; by age ten children did equally well with the mouse and the touchscreen, and better with these than with the joystick; with the touchscreen there was little individual variation, while with the other two devices there were more variation within each age group; with the mouse it was easier for children to maintain tracking and to recover from error than with the joystick [Scaife and Bond, 1991].

The fact that all children could track movement relatively easily with the touchscreen means the problems they encountered with the other two devices were not because they were not able to track. They had the perceptual ability but not the required motor ability to control the cursor position with the mouse or joystick.

Scaife and Bond [1991] refer to the relationship between the distance the mouse is moved and the corresponding distance the cursor moves as ‘gain’. In their experiment discussed above the gain was 1:1. They conducted a further experiment to determine if changes in gain influenced the effectiveness of children’s tracking. They found there was not significant difference between a 1:1 and a 2:1 gain (mouse movement is double that of cursor movement), but that performance using a 1:2 gain was worse for all ages. At eight children can control the mouse at 1:1 and 2:1 gain rate, but at ten they still struggle with 1:2 gain. Joysticks necessarily have a higher gain rate than the mouse and Scaife and Bond attribute children’s difficulties with the joystick to this.

Scaife and Bond [1991] also compared discrete and continuous modes of control. They provided children with a keypad with left and right buttons that moved the cursor stepwise to the left or right respectively. Pressing a button made the cursor move a predefined distance and the child had to press the button again to move the cursor again. They found that children aged five and six could not keep the cursor on target – they fell behind, then pressed quickly to catch up and consequently overshot the target. Seven-year-old children fared better and ten-year-olds could do it from the start. In general, young children do better with the mouse than with a keypad.

A further experiment to compare the mouse and joystick involved tracing the outlines of letters [Scaife and Bond, 1991]. From age six children could do this with the mouse and performance did not increase much with age. Younger children struggled to do this with the joystick and there was a clear improvement with age. By age ten children did equally well with the mouse and joystick.

In general, Scaife and Bond [1991] believe that children must learn to integrate the properties of input devices such as the mouse and joystick with their cognitive schema for tracking or tracing. Young children tend to move the joystick around more energetically as the device allows this. There is thus a greater need to keep a rein on actions which is difficult for young children. Children older than six use their free hand as a brake, but five and six-year-olds do not use this strategy. Aspects that influence interaction are thus device properties, task demands and the availability of cognitive schemata for controlling behaviour.

Revelle and Strommen [1990] compared young children's use of the mouse, joystick and trackball to determine which is easiest for children to use and what properties of each contribute to effective use. In their experiment, the three input devices were used over five days with the same software by sixty-four three-year-olds. Although their subjects were younger than the age group I focus on, the results can be regarded as relevant if we take into account that it agrees with results of similar studies done with adult users [Card et al. 1983; cited by Revelle and Strommen, 1990]. Revelle and Strommen specifically looked at response time and incorrect placement of the cursor. Response time increased significantly for all three devices over the five day period. Boys were faster than girls using the joystick and girls were faster with the trackball. There was no difference between boys and girls in speed of mouse use. Although children were initially quick with the mouse, they made more mistakes than with the other two devices. By day five, however the number of mistakes with the mouse had dropped to equal that with the other devices. The mistakes mostly resulted from moving the mouse while pressing down the keys. Initially children found the trackball the easiest to use, but at the end the subjects fared as well with the mouse. The joystick was the most difficult to control as children found it hard to change direction while moving. They also often released the shaft too late, missing their target. With no experience the trackball device seemed the easiest to use. The trackball and mouse's success can be attributed to the fact that the cursor movements map the user's movements in terms of direction, speed and distance travelled. The joystick is held still while movement occurs and the user controls the direction only [Revelle and Strommen, 1990]. The cognitive demands of using the joystick are possibly too high for a young child.

One study that contradicts the above findings is that of Jones [1991; cited by Alloway, 1994] who found that six, eight and ten-year-old children did better with the joystick than with the mouse or trackball. Alloway [1994] attributes this to the fact that movements in the Jones experiment was restricted to between 90 and 180 degrees, while in other studies there was no restriction on the movements children could make with the input devices.

Alloway [1994] explains children's inefficiency with the keyboard by the abstract connection between the arrow buttons and movement of a screen object. Understanding and using these abstract connections take time and attention which slows down the child's actions.

King and Alloway [1992] compared young children's use of the keyboard, mouse and joystick to pick up, move and drop objects on the screen. They found the mouse by far the most efficient and the keyboard the least. Children could speed up movement with the keyboard by using a double-key fast move option, but declined to do so even when reminded of this option. They seemed to feel more comfortable using just the arrow key.

In another experiment, King and Alloway [1992] allowed children to freely choose and play with one of three setups using the keyboard, mouse and joystick respectively. The children did not show any preference for a specific input device and could not articulate their reasons for choosing a specific device.

Yet another study by Alloway and King [1993; cited by Alloway, 1994] revealed that although preschoolers did as well with the mouse as grade one to grade three children and notably worse with the keyboard, they preferred the keyboard. They found that, as children get older, efficiency plays and increasing role in their preference: first graders were equally divided in their choice and most second and third graders preferred the most efficient device. Preschoolers lack awareness of temporal constraints, they focus on processes rather than the end product and they like to explore. From grade two children start to see input devices as tools rather than objects of enquiry and they start taking note of the time it takes to do something and their achievement in a task [Alloway, 1994].

I19	Initially children fare better with the touchscreen, but with practice the mouse is more effective. If the system is used once-off or only once in a while the touchscreen is good, but for applications for extended use rather let children use the mouse. Touchscreens are particularly suitable for tasks where children have to track movement of an object on the screen.
I20	In mouse use, 'gain' refers to the relationship between the distance the mouse moves and the distance of the movement on screen. (A 1:1 gain means the distances are equal while a 2:1 gain means the mouse moves twice the distance of the screen object.) When the child has to track movement with the mouse a 1:1 gain is best, a 2:1 gain is acceptable, but a 1:2 gain is unacceptable for children under eight years of age.
I21	Continuous control is better than discrete control (involving repeated keypresses to move the cursor along) for tracking tasks. When children fall behind in the tracking task using the discrete option, they will quickly press the key a number of times to catch up and then overshoot.
I22	Five and six-year-olds can trace large on-screen letters with the mouse but not with the joystick. They tend move around energetically so that it is difficult to control finer movements.
I23	Boys have been found to be faster with the joystick than girls and girls faster with a trackball device. Speed of mouse use is equal for boys and girls.
I24	When children are required to move the mouse and the mouse buttons should not be pressed during movement, designers should deactivate the buttons if possible. Children often press the mouse buttons accidentally while moving the mouse.
I25	When using the joystick, children older than six use their free hand as a brake to prevent overshooting.
I26	In general, children's response time with the joystick is worse than with the other devices. This can be because they find it difficult to change direction when using the joystick and they often release the shaft too late. (Restricting direction changes to between 90 and 180 degrees may help.)

I27	The mouse, trackball and touchscreen map the user's movement in terms of direction, speed and distance of the movement. This is not the case with the joystick. Keyboard keys also have a more abstract connection with screen objects.
I28	For tasks that require the user to pick up, move and drop objects, the mouse is most efficient and the joystick is more efficient than a keyboard.
I29	Although preschoolers do as well with the mouse as grade one to grade three children (ages six to nine), they prefer the keyboard. Efficiency only plays a role in preference from grade one onwards. In grade one they are equally divided between the mouse and the keyboard. In grade three they clearly prefer the most efficient device.
I30	Preschoolers lack awareness of temporal constraints and focus on the process rather than the end product.
I31	From grade 2 (age seven) children begin to see input devices as tools rather than objects of enquiry.

6.2.6 Cooperative Use of Input Devices

With regard to social interaction and collaboration between children using a computer, both Ager and Kendall [2003] and Romeo et al. [2003] found the touchscreen to promote negative rather than positive collaborations. The touchscreen is more supportive of individual goals and did not encourage cooperation. In Romeo et al.'s study only two instances of effective collaboration were observed, while many instances of negative collaboration were recorded. The negative situations included children interfering with the 'operator' assigned to the task by taking over control using the touchscreen. This caused concern especially about conflicts between boys and girls and confident and less confident children. The older children were slightly more aware of their 'partners' than the preschool children who did not attend to the wishes of other children at all. Some of the disagreements could have been caused by the fact that more than one input device were available at the same time. A child could easily use the touchscreen to interfere with the actions of a child using the mouse. The children may not have reached a level of social development to respect other children's aims if they differ from their own. Ager and Kendall [2003] also observed that interference with interaction was a problem when using a touchscreen.

A possible problem with Romeo et al's [2003] study was the fact that children always had the option to use the mouse. Collaboration between children might have been very different if the touchscreen was the only input device available. The attitude of the teachers toward the touchscreen and computers in general also influenced the results. Different teachers set different rules and the equipment was more accessible in some classes than in others. Ager and Kendall [2003] found some evidence that touchscreens can enhance learning for children struggling to control the mouse because of poor hand-eye coordination.

One problem with collaborative computer use is the competition for control over the input device. Inkpen et al. [1997] compared three turn-taking protocols with the mouse as input device. These were:

1. Two children sharing a single mouse.
2. Two children, each with their own mouse, using a ‘give’ protocol to transfer control between two mice. The current user with control decides when to give control over and uses the right mouse button to indicate this.
3. Two children, each with their own mouse, using a ‘take’ protocol to transfer control between two mice. The user who does not have control decides when to take control and uses the right mouse button to do this.

Inkpen et al. [1997] measured the time each child had control and they counted the number of transfers of control. They also measured the children’s achievement and learning in order to determine the effect of the different protocols on children’s learning. I summarise their findings in table 6.1.

Table 6.1 Results of Inkpen et al.’s [1997] study

Results for girls	**Results for boys**
Exchanged the mouse significantly more times in the ‘give’ and ‘take’ protocols than in the one mouse shared protocol.	Exchanged control significantly more in the ‘take’ protocol than in any of the other two.
There was no correlation between mouse control and increased performance.	Boys who had control of the mouse longer in the ‘shared’ and ‘give’ collaborative session, showed significantly larger improvement on their score in the game than the partner who had the mouse for a shorter period of time. This was not the case for the ‘take’ protocol.
The number of exchanges in the ‘give’ and ‘take’ protocols was almost the same.	There was a marked difference between the number of exchanges in the ‘give’ and ‘take’ protocols, with more in the latter.
In terms of performance girls fared best with the two-mouse ‘give’ protocol.	The ‘take’ protocol yielded a more equal distribution of mouse time.
In the ‘give’ set-up girls sometimes gave away control when the partner did not request it.	

For both genders, the mouse was exchanged the least number of times in the shared protocol and the most number of times in the ‘take’ protocol – this is probably because it was easy to obtain control in the ‘take’ protocol [Inkpen et al., 1997].

Inkpen et al.’s [1997] study shows that there are gender differences with regard to turn-taking protocols. The two-mouse ‘take’ protocol worked best for boys while the girls performed best with the two-mouse ‘give’ protocol. This cannot be generalised to all boys and girls, but it tells us that the turn-taking protocol should be chosen with care. To make an interaction environment equally suitable for all users, they should be able to select between different turn-taking protocols.

This concludes my discussion of the conventional input and output devices as used by young children. Next, I review the literature on less common, technologically more sophisticated input and output mechanisms.

I32	Touchscreens are more supportive of individual goals than collaboration.
I33	If both the touchscreen and the mouse are available a child can easily interfere with the actions of the mouse user with touchscreen.
I34	The best way to get children to take turns when sharing a mouse interface is to use a two mouse 'take' protocol: the child who does not have control clicks his or her mouse's right button to indicate that he or she wants control.
I35	In a two mouse 'give' protocol the child who has control decides when to give control to the partner.
I36	Boys have been found to perform better with the 'take' protocol, while girls do better with the 'give' protocol. The conclusion is that designers should include different turn-taking protocols so that users can choose one that suits them best.

6.3 Alternative Input and Output Mechanisms for Children

The interaction methods I will discuss in this section are speech input and output, handwriting input and movement input.

6.3.1 Speech Input and Output

Nicol, Casey and MacFarlane [2002] investigate the role of speech technology as a computer interface component for children. Speech recognition can serve a useful purpose in programs that aim to teach pre-reading children things such as colours, shapes and the alphabet. For older children it can help with practising pronunciation, reading, foreign language tuition or anything that would normally involve the assistance of an adult listener. It is important for a child's learning to read out loud to an adult listener. Nicol, Read and MacFarlane [2005] believe that speech recognition systems can potentially replace a human listener in this regard, but there are some problems that need to be addressed first. They found that where adults tend to feel self-conscious when talking to a computer, children have no problem with this.

At the time of writing, Nicol et al. [2002] believed that speech recognition technology, and specifically training of the recognition engine, was still too complicated to be usable in commercial educational software aimed at young children, as the one who trains the recognition engine must be able to read. The accuracy of the current technology is unacceptably low and the error rates too high to make its use viable. In a more recent article Nicol et al. [2005] discuss two ways to improve recognition performance of speech recognition systems for young children, namely through engine optimisation and through interaction design. If a speech recognition system fails it either rejects the word or phrase, or it gives an incorrect response. If used by children learning to read, incorrect responses are obviously undesirable and it is therefore important to have a high effectiveness factor (EF). The EF of a speech recognition system incorporates correct, incorrect and failed recognitions in one figure up to 100. A high EF is good and for children it needs to be at least 75.

Nicol et al. [2005] developed a prototype of an interface for an off-the-shelf speech recognition system to see if they could improve its effectiveness by addressing common problems through thoughtful interaction design. The problems and their proposed solutions are:

1. Out-of-turn speech: Children sometimes sound out a word before speaking it and, because it is difficult to train a recogniser to ignore these utterances, the system will probably pick them up. If a teacher or other child speaks to the user, this can also be detected by the system. Nicol et al's [2005] solution is to let the user press the space bar while speaking. Visual feedback tells the child when the system is in 'listening' mode and when not. For this they use a cat that sits up (in listening) or lies down (not listing).
2. Deadlock: This occurs when the system and the user are waiting for each other to say something. Nicol et al. [2005] use a technique called 'backchannelling' to detect this and prompt the child. Subtle prompts make the interaction realistic [2005]. They use a cat that is lying down but lifts its head. If there is still no response the system will prompt the user more directly.
3. Incorrect recognition: Nicol et al. [2005] prevent incorrect recognition by letting the system ask the user for confirmation that it recognised the word correctly. When the child utters a word the system responds with 'Did you say …?' If the child says 'no' the system asks for the word again.

Nicol et al. [2005] conducted an experiment to test their prototype and learnt the following: children found the system easy to use and had no problems using the space bar to indicate when they spoke; the verbal confirmation of each utterance worked well and children did not find it frustrating. Backchannelling was not very successful as it often prompted the child too soon – for example, when the child fiddled with the microphone – and when this was corrected the waiting time was sometimes too long. Overall, the interface improved the EF of the system from 57 to 73 [Nicol et al., 2005].

Oviatt, Darves and Coulston [2004] discuss a new class of so-called conversational interfaces which 'aim to support large-vocabulary spontaneous spoken language that is exchanged as part of a fluid dialogue between user and computer' (p. 301). Animated software personas are used to facilitate the conversational interaction, but research has not provided results on users' actual experience with this kind of interface element [Oviatt et al., 2004]. Up to now research has highlighted the negative aspects of animated characters, but Oviatt et al. believe that this is changing. High fidelity moving lips can, for example, improve the intelligibility of speech output; realistic gazing and gestures can improve dialogue efficiency; and the dialogue style and personality of the character can influence the user's fluency. Oviatt et al. [2004] also report that correctly designed characters can stimulate learning-oriented behaviour, encourage self-disclosure and influence purchasing behaviour.

According to Oviatt et al. [2004], state-of-the-art text-to-speech (TTS) systems have moved beyond basic intelligibility and are successfully applied in telephony and mobile environments. In a study that involved testing a TTS system with seven to nine-year-old children, the computer was asked to repeat less than 1% of the time. Although the evaluation of TTS technology now focusses on how 'natural' it sounds in terms of duration, pitch, intonation and so on, more research is required on the human perception and behavioural reaction to TTS and on the influence of TTS parameters on users' spoken input [Oviatt et al., 2004]. They point out that research usually looks only at TTS or only at speech recognition, but that these should ideally be studied together, since speech output from animated characters can be coordinated with users' spoken input in certain task domains. They report on a study done by Moreno et al. [2001; cited by Oviatt et al., 2004] which showed that animated characters that produced speech rather than text improved children's learning, transfer and retention on a plant biology task. The same study showed that the visual image of the character had no impact. Another study by Darvis and Oviatt [2004; cited by Oviatt et al., 2004] showed that an extrovert TTS voice encouraged more task-related questions from the user than an introverted voice. In their study, the type of voice had no impact on the children's social questions.

The general conclusion is that a TTS can be designed to influence users' reaction and behaviour in certain conversational applications. Recent research has, however, shown that speech recognition is less successful in children's applications and that error rates can be up to five times higher than for adults [Oviatt et al., 2004]. Oviatt et al. agree with Nicol et al. [2002] that children's speech production is very unpredictable and that speech recognition systems for children's applications require special strategies to deal with high levels of variability.

Oviatt et al. [2004] investigated whether users' speech characteristics are influenced by the TTS voice they hear during interaction. The TTS voice was either extrovert (higher in amplitude, higher in pitch with expanded pitch range, shorter in duration and faster in delivery) or introvert. They found that the children adapted the amplitude and duration of their speech by ten to fifty percent when interacting with animated characters that represent different speech styles. An extrovert character voice led to increased amplitude and shorter utterances, while an introvert voice caused amplitude reduction and shorter utterances. So, the children's speech converged significantly towards that of the computer partner, reflecting the partner's level of energy and patterns of silence. There was no significant difference in the reactions of younger and older children and gender also had no influence.

Speech recognition can benefit from the fact that users adjust their speech to match that of the software character. The software can subtly guide users to speak within a range that can easily be recognised by the software [Oviatt et al., 2004].

Oviatt et al. [2004] could not confirm the results of earlier work that found that users are more responsive to characters whose style and personality match their own. An example of such a study is that of Nass and Lee [2000] who investigated whether the vocal characteristics of speech can convey personality and how this influences user' perception of the system. They found that users apply vocal stereotypes to computer-synthesised voices – they regard a voice that is manipulated to sound extrovert as more extrovert and introvert voice as more introvert. The personality of the voice influences the user's perception in a similarity-attraction way: extrovert users find the extrovert voice more credible and attractive and they prefer the content of that voice's speech. Introverts also display similarity-attraction [Nass and Lee, 2000]. The implication of this is that designers can use the social and personality aspects of speech in a TTS system to manipulate users' perception of the system. According to Nass and Lee [2000] most TTS systems include social parameters that can easily be manipulated to give the system voice vocal qualities associated with specific personalities.

Speech recognition software involves training of the recognition software for use by individual users. Users are required to provide training utterances so that the software can be adjusted to recognise their individual voices. This is difficult with very young users as they may not be able to read the training text. An adult typically has to whisper the words into their ears so that they can repeat it out loud for training purposes. Another problem with conversational interfaces for children is the time delay between user and computer responses. In Oviatt et al's [2004] study, children's response latencies for the younger children were 6.2 seconds on average, while the older children averaged at 4.2.

In section 6.4.1.1 the topic of speech interaction surfaces again when I review a study by Strommen and Alexander [1999] about the emotional effect of speech on children's experience with technology.

Another innovative interaction tool is handwriting recognition which Read and co-workers [Read, 2005; Read, Gregory, MacFarlane, McManus, Gray and Patel, 2002a; Read, MacFarlane and Casey, 2001; Read et al., 2002b; Read et al., 2004] regard as a good alternative to speech input. I review their research next.

I37 Speech recognition can be useful in teaching pre-reading children concepts such as colours, shapes and the alphabet.

I38 Speech recognition applications can help children who can read with practising pronunciation, reading, foreign language tuition or anything that would normally involve the assistance of an adult listener.

I39 Children do not feel self-conscious when talking to a computer.

I40 Training a speech recognition engine usually involves reading of training text, but with pre- reading children this is not possible. An adult can whisper the training text into their ears or it can be played to them through headphones.

I41 Problems that can occur when using a speech recognition system are out-of-turn speech, deadlock (user and system wait for each other) and incorrect recognition. Nicol et al. [2005] propose the following solutions:

- Out-of-turn speech: The user presses the space bar while speaking. Visual feedback tells the child when the system is in 'listening' mode and when not.

- Deadlock: 'Backchannelling' is used to detect this and prompt the child subtly.
- Incorrect recognition: When the child utters a word the system responds with 'Did you say …?' If the child says 'no' the system asks for the word again.

The first and last of these solutions have been used successfully, but backchannelling did not work as well.

I42 Animated software personas can facilitate conversational interaction as follows:

- High fidelity moving lips can improve intelligibility of speech output.
- Realistic gazing and gestures can improve the efficiency of dialogue.
- The persona's dialogue style and personality can influence the user's fluency.

I43 Users adapt the amplitude and duration of their speech to the speech style of synthesised speech. This can be used to subtly guide users to speak within a range that the software can recognise. An extrovert voice leads to increased amplitude and shorter utterances while an introvert voice leads to reduced amplitude and longer utterances.

I44 Extrovert users prefer extrovert voices and they prefer the content of the utterances of an extrovert voice. This similarity attraction also applies to introvert users. Designers can exploit this by determining, at setup time, whether the user is an extrovert or introvert. By adapting the synthesised voice accordingly they can make the interaction more pleasurable for the user. In most TTS systems the parameters to achieve this can be set easily.

I45 An application that involves a conversational interface should be adaptable in terms of response time. Children's response times are slower than that of adults and their response times increase with age.

6.3.2 Text Input Methods

Using the standard QWERTY keyboard for text entry has some problems for young children [Read et al., 2002b]. Typing involves five phases, namely character recognition, storage, motor activity, keystroke and feedback [Read, 2005]. For children, these processes, the layout of the keys on the keyboard and their limited short-term memory capacity slows down text entry with a keyboard. The fact that they type very slowly may cause them to lose their train of thought. An alternative text input method is available through technologies that allow users to write as they would using a pen and paper.

Research by Read et al. [2002b] found handwriting interfaces a suitable option for young children and showed that children write more fluently with a handwriting tablet than with a keyboard. Various handwriting recognition (also called 'digital ink') technologies are available such as the Tablet PC and a Digital Pen used with a digital paper notebook. The writing can be captured in graphic format exactly as it appears on the graphics tablet or digital paper, or it can be converted to ASCII (or similar) text. The latter option requires recognition software [Read, 2005].

Like speech and gesture recognition, handwriting interfaces are 'disobedient' in the sense that they may cause interaction errors [Read et al., 2002b]. Empirical research with adults on handwriting recognition rates, have found a rate of 87% on average in one study and between 87% and 93% in another [Read et al., 2002b]. In a study with children, Read and colleagues found an average recognition rate of 86%. They categorised errors in handwriting recognition interfaces for children as spelling errors, construction errors (incorrectly formed letters), execution errors (failing to touch the tablet with the pen or adding spurious characters) and recognition errors (software induced errors).

In a study to test handwriting recognition software with seven and eight-year-old children, Read et al. [2002b] looked specifically at error discovery, error recovery and error avoidance. With regard to error discovery, they found that almost 50% of the errors that children missed involved capitalisation that was incorrectly changed by the recogniser. 32% were spelling errors and 22% recognition errors. The keyboard was always available to correct errors and children used it often to insert punctuation or delete unwanted spaces created by the software. When a child paused too long between two letters of a word, the software would insert a space. Some children realised this and learnt to prepare a word in their heads before writing it. When correcting errors some children would erase and rewrite a whole word when there was just one letter wrong, while others would erase the word just up to the error. To avoid errors the children used different strategies. They chose words that they could spell or asked the researcher how to spell a word they were unsure about. They noted which of their letters caused problems for the recogniser and learnt to form those more carefully.

Correcting errors can sometimes aggravate the problem. This can be prevented if users have a clear and correct mental model of how the system works. Achieving this is more challenging with young children. It would be less demanding to give users clear advice on more effective error correction strategies and help them understand that recognition errors are not their mistakes [Read et al., 2002b].

Read [2005] compared the usability of three digital ink technologies for children, namely a stylus used with a Tablet PC, a graphics tablet and pen, and a digital pen on digital paper. Children aged seven, eight, twelve and thirteen were given a sheet of text phrases that they had to copy using the different technologies. With digital pens the recognition rate for older children was significantly better than for younger children. With the Tablet PC the recognition rate was the same for younger and older children, but the older children had fewer recognition errors here than with the digital pen. There was not much difference between the recognition rates of the Tablet PC and the graphics tablet for the older children. (The younger children did not use the graphics tablet.) The children generally preferred the Tablet PC.

In general, Read [2005] found that children learned to use the technologies quickly and only some children needed assistance when starting to write on the Tablet PC. The reasons for recognition errors in Read's [2005] study are (in the order of most instances to the least): spelling errors, use of 'text speak', missed words and substituted words. With the digital pen children sometimes started to close to the top of the page where the writing was not detected by the system and one child used the digital paper upside down so that no word was recognised. Older girls sometimes use a decorative handwriting style that caused recognition problems.

Read, MacFarlane and Casey [2001] compared handwriting input with speech, keyboard and mouse input. When using the mouse for text entry the children wrote by selecting from letters displayed on screen. With regard to effectiveness (correctness of the input) they found the mouse and keyboard to be equally good with percentage correctness measures (PCM) of 90% and higher. The PCMs for speech was between 36% and 44% and for handwriting input between 73% and 86%. The efficiency (input speed) of speech input was by far the best. Handwriting input was slightly faster than the keyboard with the mouse the least efficient. Children were sometimes distracted while writing on the tablet by unexpected feedback. This affected the efficiency measures. Read et al.'s [2001] general conclusion is that there is a trade-off between effectiveness and efficiency and that children prefer an interface that is easier to use and need correcting to one that is accurate but difficult to use.

When interviewing children about their requirements, Read et al. [2004] noted that although children were comfortable using computer jargon, they sometimes replace these words with their own preferred words. For example: 'rubbing out' or 'get rid' for 'delete'; 'tell' for 'show'; and 'fix' for 'repair'. This is something designers may want to consider when deciding on audio or text captions for buttons.

Read et al. [2001] made the following observations regarding spaces between words: younger children often fail to include spaces between words when they copy text using the keyboard or mouse; when they do include spaces they sometimes put more than one space; and the software inserts a space whenever the child lifts the pen for long enough (young children do this often while writing a word).

Read et al. [2004] formulated a set of requirements for systems aimed at young children that uses handwriting input. They derived these requirements from observations of children using prototypes of such systems [Read et al., 2001; Read et al., 2002b] and through questionnaires and interviews with children [Read et al., 2004]. I end this section with a summary of these requirements. They organised the requirements following Read et al.'s organisation according to Preece et al's [2002] requirements framework:

1. Functional requirements: The system should support the planning, translation (writing) and review phases of the writing process, providing ideas for planning, allowing fast and accurate transcription and allowing for easy movement, alteration and deletion of text. It should include spelling support and file handling facilities. When children state their requirements, spelling support is high on their list. They even asked that the system spell out words that they had misspelled. They also want to be able to use different fonts and typefaces.

2. Data requirements: The system should be able to cope with multiple users that use many different documents. Each document may have many associated files (i.e. text files and 'ink' files) that can go through editing processes to create new versions.
3. Environmental requirements: The system must be robust, easy to learn and include online help. It should work on a standard PC with tablet and pen. Large font sizes will support collaborative use.
4. User requirements: The system should not assume that users can read well, spell well or write well. Help should be provided through speech. Users can be expected to hold and manipulate a pen and to write even sized, legible upper and lower case letters. Designers should not assume that there will be expert adult help available.
5. Usability requirements: It should not take longer than ten minutes for a child to learn to use the system. They should be able to use it without asking for help and the help facility should be easily accessible.

I46 Handwriting interfaces can be used by young children.

I47 Children have been found to write more fluently with a handwriting tablet that with a keyboard.

I48 Errors that occur with handwriting recognition interfaces for children are spelling errors, construction errors (incorrectly formed letters), execution errors (fails to touch the tablet with the pen or adds spurious characters) and software induced errors (recognition errors and incorrectly changed capitalisation). A keyboard should always be available to correct errors.

I49 Children do not always discover the errors. Those that they miss most often are incorrect capitalisation caused by the software, spelling errors and recognition errors.

I50 Unwanted spaces appear between letters because children pause too long. Designers should make them aware of this so that they can learn to prepare a word in their heads before typing.

I51 Designers should provide clear advice on efficient error correction strategies. Children will, for example, delete a whole word if only one letter is incorrect.

I52 It should also be made clear that recognition errors are not the user's mistakes.

I53 Children prefer the tablet PC to the digital pen and the Tablet PC shows a better recognition rate.

I54 The top and bottom of the tablet or digital paper should be clearly distinguishable. If used upside down no words will be recognised.

I55 Children tend to start writing near the top edge that is outside the recognition area, so, if possible, it should be indicated where they have to write.

I56 When choosing between the keyboard, mouse, handwriting input or speech recognition for text input, there is a trade-off between efficiency (speed) and effectiveness (correctness):

- Correctness of input is equally good with the keyboard and mouse (percentage correctness measure is 90% and higher).
- Percentage correctness measure for speech input is between 36% and 44% and for handwriting input it is between 73% and 86%.
- Speech input is by far the fastest, with handwriting slightly faster than the keyboard and the mouse

the least efficient.

I57 Children prefer an interface that is easier to use and needs correcting to an accurate, but difficult one.

I58 Read et al. [2004] formulated the following set of requirements for systems aimed at young children that uses handwriting input.

1. Functional requirements: The system should support the planning, translation (writing) and review phases of the writing process, providing ideas for planning, allowing fast and accurate transcription and allowing for easy movement, alteration and deletion of text. It should include spelling support and file handling facilities.
2. Data requirements: The system should cope with multiple users that use many different documents.
3. Environmental requirements: The system must be robust, easy to learn and include online help. It should work on a standard PC with tablet and pen. Large font sizes will support collaborative use.
4. User requirements: The system should not assume that users can read well, spell well or write well. Help should be provided through speech. Users can be expected to hold and manipulate a pen and to write even sized, legible upper and lower case letters. Designers should not assume that there will be expert adult help available.
5. Usability requirements: It should not take longer than ten minutes for a child to learn to use the system. They should be able to use it without asking for help and the help facility should be easily accessible.

6.3.3 Movement Input

One way to address the concerns about the physical harm in spending too much time passively in front of a computer screen is to develop technology that require children to move around. Dance mats that use sensory devices to detect movement are widely available. Computer vision and hearing technology can also be used to create games that use movement as input [Höysniemi, Hämäläinen, Turkki and Rouvi, 2005]. Li, Moraveji, Kimura and Ofek [2006] describe the two ways in which camera-based input is currently used to map users' movements to onscreen characters. These are:

1. Through a predetermined set of physical gestures that initiate character action. The user has to learn these gestures.
2. Through direct manipulation of the on-screen character by user movements. The character mimics all the movements of the user in real time.

The first of these is indirect or disconnected, while the second is fully connected [Li et al., 2006].

Höysniemi et al. [2004] investigated the type of actions most suitable for vision based computer games that are indirectly controlled with body movements and voice. In their *QuiQui* game a dragon character mimics the user's running, jumping or swimming movements and breathes fire when the user shouts (see Figure 6.2). An ordinary low end USB camera and a microphone are used to capture the movement and sound and no sensors are attached to the user. The aim of Höysniemi et al.'s [2004] research was to improve the usability of vision based computer games for children aged four to nine, so that they would be able to play without adult guidance. In the game, movements are context based so that the child will know when a particular kind of movement is required. For example, when the avatar is in the air, flying movements are required and when in the sea, the child must make swimming movements. A web cam image is displayed in a bottom corner of the screen so that the children can see when they move out of the camera's view.

Figure 6.2 Children playing QuiQui [ARS Electronica, 2006; LECO Research Group, 2005]

In their experiments Höysniemi et al. [2004] evaluated four movement games, namely the spider game, running, swimming and jumping games and tried to determine which movements are the most likely to support successful interaction. For the swimming game they compared doggy stroke, breast stroke, freestyle and mole stroke (similar to doggy but the hands move together). Children clearly preferred to use doggy stroke and breast stroke although the designers anticipated that they would only use the freestyle. They also tried a variety of ways to dive.

In their first prototype, a child made the dragon fly upward by flapping both hands. To let him make a left or right turn they had to flap only one hand [Höysniemi et al., 2005]. Children found this steering movement difficult and intuitively leaned their bodies to the side to which the dragon should turn. Höysniemi et al. [2005] used this as the steering movement in their subsequent prototype. They learnt that to analyse children's game control gestures, it is necessary to identify children's preferred movements in a specific game context, and to study individual properties and differences in children's movements (including range of motion, symmetry, pace, space used and transition between movements). Finding patterns in movements can help to improve computer vision algorithms and avatar animations [Höysniemi et al., 2005]. In this experiment a lot of variation was found in swimming movements, while jumping movements were similar for all children.

Subdued movements are more difficult to detect than dynamic movements and some training may help to get users to move in a way that supports successful detection [Höysniemi et al., 2005]. Often movements occur below the waistline when the camera focuses on the user's upper body. Young children find it difficult to go directly from one movement into another and could be given the chance to stop in between, but ideally they should learn to perform smooth transitions.

Höysniemi et al. [2005] identified four requirements for computer vision-based games: robustness, responsiveness, intuitiveness and physical appropriateness. Intuitiveness and physical appropriateness are especially important to ensure an enjoyable experience and to reduce learning time. They concluded from their experiment that, because vision-based action games require gross motor skills and continuous movement, children become tired and cannot play for long periods. When they are tired their movements become less pronounced and more difficult to detect. Such games should be designed so that there are resting times and Höysniemi et al. suggest that rest periods should occur every four or five minutes for children as young as five or six.

Höysniemi et al.'s [2005] overall conclusion was that there is a need for more research to establish what gestures children find intuitive.

Whereas gesture-based, indirect movement input requires the user to learn the correct way to move, direct manipulation movement interfaces also have problems [Li et al., 2006]. It can be frustrating to control a system with limb movements as users sometimes want their avatars to move in ways that they do not naturally move themselves. Li et al. propose a middle ground between the two options where the user moves freely in front of the camera and the system recognises and extracts only meaningful gestures to apply to the avatar. So the avatar can perform a limited number of movements and any other movement by the user will have no effect (there is spatial disconnection). Also, the system can take complete control of the avatar for predetermined durations (there is temporal disconnection).

Li et al. [2006] combined spatial and temporal disconnection. Their example involves the user swinging a hammer to get an avatar to hit nails into a wooden log using a hammer. The system measures the vertical position of the user's hammer and the velocity of the hammer movement. If these values exceed a predefined threshold (that they called the 'trigger'), the systems takes control of the avatar. Visual cues inform the user of the triggers. For example, a trigger line is displayed through which the hammer must swing to give control to the system. When they tested their ideas, Li et al. found that different users have different control preferences. Some preferred the trigger and some preferred direct control. They observed that using the trigger caused less frustration and less physical effort than direct manipulation, but that users sometimes adopted awkward physical positions. Some found the visual design of the trigger line disturbing.

Human-robot interaction often involves some form of motion detection. Michalowski, Sabanovic and Kozima [2007] report on research that investigated the role of interactional synchrony on users' experience when they interact with a robotic toy. In interaction between people, interactional synchrony involves posture mirroring and correspondence between the gestures made by two interactors [Michalowski et al., 2007]. This coordination of behaviour contributes to the success of their communication. Michalowski et al.'s research involves a simple creature-like robot named Keepon (Figure 6.3) that is programmed to detect rhythm through auditory and visual stimuli detection. It reacts to such input by moving as if dancing to the rhythm. When a child stands in front of Keepon making dancing movements, it detects the movement through cameras set in its eyes and picks up the rhythm of the movements. The software uses this rhythm to guide movements of robot's body and head. Since these movements are synchronised with the rhythm at which the child dances, it seems as if Keepon is dancing with the child [Michalowski et al., 2007].

Figure 6.3 The Keepon Dancing Robot [Ozoux.com, 2007]

Michalowski et al.'s [2007] results show that children are more inclined to dance with the robot if it was already dancing with music and keeping to the rhythm, than when it was still or when its movements were not in synchrony with the music. This means that properly synchronised movement is important to engage users in the dancing game. They also found that girls interacted with the robot more than boys and performed more rhythmic dancing actions with the robot than boys.

A widely used commercial application that uses movement input is Sony's EyeToy™ . The EyeToy is a motion recognition USB camera used with Sony's Playstation 2. It can detect movement of any part of the body, but most EyeToy games involve arm movements [Loke, Larssen, Robertson and Edwards, 2007]. An image of the player is projected on the screen to form part of the gamespace (see Figure 6.4). Depending on the game context, certain areas of the screen are active during the game. Players must move so that their hands on the projected image interact with screen objects that are active in the game. For example, they have to hit or catch a moving ball. In other words, the user manipulates screen elements through his or her projected image.

Loke et al.'s [2007] research with the EyeToy™ involved only adult players and their results do not provide specific information with regard to children's use of the technology. Demming [2004] investigated the usability and appeal of the EyeToy™ Play game and included five to fourteen-year-olds in her study. According to Demming, intuitive interaction is hindered by lack of tangible feedback. Younger children sometimes find it difficult to associate the feedback on the screen with their own movements. The timing between the movement and the effect is not always predictable. Movements of onlookers that are detected by the camera and lack of contrast between the player and the background may influence the results of the interaction [Demming, 2004].

Figure 6.4 Projected images of children playing Sony EyeToy games [Game Vortex, 2008]

Demming [2004] found that EyeToy™ promotes social game play although it usually involves one person playing. She found no gender differences in the interaction with EyeToy™ games. In her study the children (aged five to fourteen) confused the interface with a touchscreen interface and negative feedback caused them to move closer and closer to the screen. Demming [2004] infers that they want to fall back on more familiar tactile interaction that they are familiar with. Children also found it difficult to map their movements in three-dimensional space to the two-dimensional image on screen. Although Demming does not articulate this, it seems that EyeToy™ games are more suitable for older children who can create an accurate mental model of how the whole system works. To play effectively a user has to understand the relationship between the two-dimensional game world and the three-dimensional interaction space.

Another obvious application of movement input is sign language recognition. I discuss sign language recognition technology as used in young hearing impaired children's products in section 6.5.4 below. Next, I address the topic of tangible interfaces.

I59 Two ways in which camera-based input can be used to map users' movements to onscreen characters are:

- Through a predetermined set of physical gestures that initiate character action. The user has to learn these gestures (indirect or disconnected mapping).
- Through direct manipulation of the on-screen character by user movements. The character mimics all the movements of the user in real time.

I60 Displaying a web image of the user on screen will help the user to stay in the camera's view.

I61 Designers can use the Wizard of Oz method to test whether the movements they require the children to perform are appropriate.

I62	Try to identify movements that do not vary too much from child to child. (For example, children perform swimming movements in a variety of ways, but their jumping movements are very similar.)
I63	Dynamic movements are recognised more successfully than subdued ones.
I64	If the camera focuses on the upper body, movement of the feet cannot be recognised.
I65	Children should not be expected to move vigorously for long periods. When they tire their movements become less pronounced. Five to six-year-olds should rest every four or five minutes.
I66	In direct control interfaces users may find it difficult to move in exactly the way they want the character to move. To solve this, the system can detect partial movement and then take over control to complete the movement in a realistic way.
I67	Interactional synchrony (coordination of behaviour) improves interaction between people, so it would also improve interaction between a child and a robotic toy if the toy's behaviour is coordinated with the child's behaviour.
I68	In a dancing game, properly synchronised movement is particularly important.
I69	When movement input involves manipulating objects in a two-dimensional game world by making movements in a three-dimensional space, the interface must help young children to bridge the conceptual discordance between the movements and their effects.

6.4 Tangible Interfaces

Hornecker and Buur [2006] describe tangible interaction as encompassing 'a broad range of systems and interfaces relying on embodied interaction, tangible manipulation and physical representation (of data), embeddedness in real space and digitally augmenting physical spaces' (p. 437).

Existing technologies with tangible interfaces that are specifically aimed at children include programmable objects that can be made to move around on the floor [Frei, Su, Mikhak and Ishii, 2000], building blocks that can be used for programming [Wyeth and Purchase, 2003], conversational agents like Sam [Ryokai et al., 2003] and Pets [Montemayor et al., 2000], robotic pets [Bartlett, Estivill-Castro and Seymon, 2004; Kahn(Jr.), Friedman, Perez-Granados and Freier, 2004] or intelligent soft toys [Alexander and Strommen, 1998]. According to Plowman and Stephen [2003], many of these toys are marketed as being supportive of 'interactive learning' and 'nurturing play', but these claims are not substantiated by research. Plowman and Stephen do, however, believe that some of these toys may support children's understanding of social relationships. A clear advantage of non-screen-based interfaces for preschool children is the absence of text or symbols that need to be read and understood [Plowman and Stephen, 2003]. Researchers are trying to establish whether children attribute feelings or emotions to these toys, what the effect is of the way the toys structure play activities, and whether they enhance or limit the child's imagination [Plowman and Stephen, 2003].

Children's body movements and their ability to touch, feel and manipulate things are important for developing sensory awareness and therefore also for their general cognitive development [Healy, 1998; cited by Antle, 2007]. In this sense, tangible interfaces are ideal to support children's cognitive development. Tangibles can also help children develop understanding of abstract concepts, as these are often based on their understanding of spatial concepts and how they use their bodies in space [Antle, 2007].

In the rest of this section we discuss research with regard to some recognised examples of children's technology that use tangible interfaces.

6.4.1 Tangible Interfaces and Emotional Interaction

Interfaces that evoke emotional responses with users are more engaging. In the same way that positive emotions can play a role in learning and development, it will affect how users interact with and perceive technology [Strommen and Alexander, 1999]. This is particularly true for children and Strommen and Alexander believe that emotional interfaces in educational technology can, in addition to making the interface better, improve achievement of a product's learning goals.

When technology is merely seen as a tool, task motivation, satisfaction with the task performed and the intrinsic pleasure of feeling in control, are the important emotions [Strommen and Alexander, 1999]. A spreadsheet's interface need not convey warmth or playfulness to be successful in the tool model of interaction. It is true, however, that the emotional responses computers elicit in users are similar those displayed in social interaction between people. A more social approach to interaction is not concerned with a user-tool relationship, but rather sees the computer as the user's partner or collaborator.

Research has shown that adult users respond to some emotions displayed by interface agents as if they were produced by humans [Strommen and Alexander, 1999]. Children's perceptions about technology have been found to be strongly influenced by the emotional tone of speech output.

6.4.1.1 Audio Interfaces for Emotional Interaction

Focussing on the use of humour, praise and affection, Strommen and Alexander [1999] report on research with regard to emotional interactions in the audio interfaces of two different toys. The toys that they used for their research were Actimates DW and Actimates Arthur – anthropomorphised aardvark siblings based on the Marc Brown stories and cartoon series. Each of them has very specific character traits and their individual personalities are the foundation of the social interface. Consistent personalities make the interface consistent and predictable, which is good. Their facial expressions are fixed, so that their interaction relies on speech and gesture alone. Children interact with them by activating sensors in different parts of their bodies. By squeezing their ears they will share their 'thoughts' with the children. This includes asking questions, offering opinions, giving complements and telling jokes. Squeezing their feet activates games. They do not react to speech input.

Speech is a rich medium for displaying emotions. Speech patterns give critical information about personality and feelings, which is necessary to create the impression of realism [Strommen and Alexander, 1999]. Familiarity of voice is also important and, since the toys are based on existing television characters, their speech interfaces use recordings of voice actors. Existing interface design guidelines promote consistency and brevity of audio interface elements to improve efficiency of navigation. Strommen and Alexander do not agree – human speech includes natural variations which a character-based interface requires to sound realistic. In the two toys involved variability was accomplished by varying the syntactic construction of specific interface instructions, randomly varying the order in which phrases are presented to make the response unpredictable and more natural sounding. Their speech also includes interjections such as ‘Hey!’ or ‘I know!’ which is consistent with the Arthur and DW fictional characters as well as with the speech of four to eight-year-olds. It is also important for fictional characters to have signature phrases to reinforce the character’s identity and the authenticity of the interface.

The three emotional interactions included in the Actimates Arthur and Actimates DW interfaces are praise and encouragement, laughter and humour, and warmth and affection. All three of these fulfill the following criteria [Strommen and Alexander, 1999]:

- It is consistent with the personality of the character.
- There is definite empirical evidence that the emotion contributes positively to children’s learning and development.
- It could be integrated into the interface successfully.

There is ample evidence that task-sensitive praise affects task performance and motivation and that it helps with task persistence when children are learning new material [Strommen and Alexander, 1999]. The praise offered by Arthur and DW is performance-specific with a tone to match the challenge. Testing showed children’s appreciation for the praise and some children responded verbally to the praise. The tests also showed that praise reduces disappointment with failure. If the toy says ‘That was hard’ or ‘That was a tough one’ before starting a new round after the child has failed, the child reacts more positively than when it only says ‘Let's try again’. When the child has stopped playing for a while, the toy utters a phrase such as ‘You rule!’, ‘You're so cool!’, ‘Don't stop now, you're doing great!’. This both motivates the child to continue playing (and it has proved successful in this regard) and makes the character seem less task-driven.

The benefits of humour for social interaction are that it reduces social distance and humorous peers are more popular [Strommen and Alexander, 1999]. Laughter and smiling are contagious. When two children experience a humorous situation together, their response is more pronounced. The Actimates toys use humour in a Silly Sentence game and in the phrases uttered when an ear is squeezed. In Strommen and Alexander’s tests, children smiled when the character laughed. When DW said ‘Does my hair look stringy to you? (giggle)’, most girls smiled, giggled or laughed and spontaneously groomed the character’s hair. When it said ‘I'd squeeze your foot if I could get it! (giggle)’, most children laughed and few of the younger children held their feet up to the character.

Warmth and affection of peers and authority figures influence mental growth, increases motivation, improve self-esteem, and so on [Strommen and Alexander, 1999]. The Actimates characters are explicit about how much they value friendship and how they enjoy the company of their mates. Their phrases that reflects warmth and affection are, for example, 'You're the best friend a guy could have!', 'I'm lucky to have a friend like you!', 'I wish you were in class with me!'. They also build affection by sharing secrets and playful teasing. Testing showed that these were successful with boys and girls and they responded by commenting and interacting with the toy. Some girls responded to the secret sharing by telling the researcher the secret as well.

Social interfaces are a form of pretend – users behave 'as if' the computer has a social presence, while they know that they are just machines [Strommen and Alexander, 1999]. Obviously, interfaces that display humour, warmth and spontaneity are not appropriate for all kinds of applications, but they are particularly suitable when the users are children.

In their study involving children aged four to six playing with these same toys, Plowman and Luckin [2004] found that the younger children are more likely to think the toys have feelings and could really think and talk. When asked to suggest improvements, the children said they would like to toys to be able to walk [Plowman and Luckin, 2004]. Some children preferred to use the toy switched off, so that it resembled a normal inanimate toy – they tended to take the toys' abilities for granted.

Contradicting the results of Strommen and Alexander [1999], Plowman and Luckin [2004] found that children regarded the toys' talking as monotonous and irritating and that they switched it off after playing for a while. The latter study was done five years after that of Strommen and Alexander. It may be that children have become more sophisticated users of technology in that time. Plowman and Luckin [2004] could not see any difference in children's imaginative play that resulted from the interactivity provided by Arthur of DW, but they did find that these toys were beneficial in terms of providing scaffolding. The help provided through the tangible interface worked better than the same kind of help being offered on screen. The toys also increased collaboration between children, and between children and adults.

My next example involves a more sophisticated interface – not to be found in toy shops for some time.

6.4.1.2 Advanced Emotional Expression in Robotic Toys

Tanaka, Noda, Sawada and Fujita [2004] pose three requirements for personalised interaction between humans and robotic toys:

- The robot should be able to recognise the human user as an individual and it should be able to alter its behaviour according to characteristics and behaviour of that particular user.
- It should be able to accumulate experiences in its memory so that future behaviour can take earlier experiences into account.

- It should express its emotions in a way that is clearly identifiable to the user.

Tanaka et al. [2004] describe how they have accomplished this in QRIO (Figure 6.5), a small humanoid, biped robot which developed out of their work on the dog-like quadruped AIBO (see section 6.4.5 below). It can walk, run, dance, sing, play soccer, throw a ball, participate in a conversation, and more [Tanaka et al., 2004]. QRIO receives input through several sophisticated hardware and software components. It can recognise a human through vision technology and audio detection; it can associate variation in values of its internal variables with a specific situation and store this in memory for future use; and its rich motion control system allows accurate portrayal of emotional reactions. Its internal software consists of a perception part, a recognition part, a memory part, a behaviour and motion control part and an internal model part [Tanaka et al., 2004]. The recognition part, for example, includes a face recognition engine and a general object recognition engine. QRIO has separate long-term and short-term memories.

QRIO's emotional expression and internal states are handled by the internal model part [Tanaka et al., 2004]. It maintains variables that represent hunger, fullness, pain, comfort, fatigue and sleepiness and these variables affect the robot's reaction to external stimuli (e.g. facial recognition) and its internal state (e.g. low battery power). An emotion generator supports the following emotions that are represented by variables in the system: joy, sadness, anger, surprise, disgust, fear and 'neutral'. If a user twists QRIO's hand, the robot will respond by, for example, increasing the values of its pain and fear variables. It will at the same time recognise and record the user's face. When the robot recognises the user's face at a later point in time, it will respond with and increased fear value. When the pain or fear values reach a predefined threshold, QRIO displays appropriate facial expressions, makes a sound associated with this emotion and displays appropriate gestures.

Figure 6.5a Children interacting with QRIO [Swaminathan, 2007]

Figure 6.5b QRIO as pictured on Web Japan [Web Japan, 2004]

Although these functions have been successfully implemented in QRIO, this kind of technology is still too expensive to become commercially available. Tanaka et al. [2004] show, however, what is possible and in future it may become possible to incorporate these features in more widely available technologies.

I70 Make interfaces more engaging by letting them evoke emotional responses with users.

I71 Children's perceptions about technology are strongly influenced by the emotional tone of speech output.

I72 Task-sensitive praise affects task performance and motivation and that it helps with task persistence when children are learning new material.

I73 Speech patterns give critical information about personality and feelings that is necessary to create an impression of realism.

I74 If characters are based on familiar characters the voices must be consistent with the known voices.

I75 Speech output should include natural variation.

I76 Emotional interactions should be consistent with the personality of the character.

I77 Praise can be positive but children will not be convinced if they hear the same praise words every time they do well. On the contrary, they may become irritated.

I78 If a product will be used for an extended period, children must have the option to switch off speech feedback that may become irritating.

I79 Humour, warmth and spontaneity in interfaces are appropriate for children's products and may increase motivation.

I80 Requirements for personalised interaction between humans and robotic toys Tanaka et al. [2004]:

- The robot should be able to recognise the human user as an individual and it should be able to alter its behaviour according to characteristics and behaviour of that particular user.
- It should be able to accumulate experiences in its memory so that future behaviour can take earlier experiences into account.
- It should express its emotions in a way that is clearly identifiable to the user.

6.4.2 Soft Toy Interfaces

The most familiar interactive toy that also acts as interface to a software application is ActiMates Barney. It is a freestanding animated doll that can move his arms and head and speak [Alexander and Strommen, 1998]. It can be used with a wireless radio link, PC-based software or linear video tapes (see Figure 6.6). In freestanding mode the child activates songs and games by squeezing his feet and hands respectively. Each touch interrupts the current song or game and starts the next one. A light sensor in Barney's eye allows the child to play peek-a-boo with him. When used with a PC the child plays a game on the PC and Barney provides praise and encouragement as a team mate. When the child watches the video tapes, Barney watches along and asks questions, directs attention and encourages participation. The interface is modelled on children's pretend play behaviour with dolls.

Figure 6.6 Barney in action [LECO Research Group, 2005]

Originally, once a song or game was activated, the child had to wait until it finished before starting a new one. In other words, the child played along with Barney. Research indicated, however, that children want to have some control and want Barney to play along with them, hence the current interruptive nature of the interface [Alexander and Strommen, 1998]. Barney's instructional phrases are intermingled with friendship phrases such as 'This is fun!' and 'I like playing with you', as research showed that instructions alone let Barney seem robot-like which made children to lose interest. In the version of ActiMates Barney that is used along with video tapes Barney fulfils the role of an older peer or adult watching the video with the child. He asks questions, participates with activities and queries the child about program content. When the child now squeezes a hand or foot he reacts with a phrase like 'I like watching TV with you'.

Johnson, Wilson, Blumberg, Kline and Bobick [1999] introduced a different kind of soft toy interface that they refer to as a 'sympathetic interface'. A soft toy – embedded with an array of thirteen wireless sensors – is used to manipulate a similar on-screen character. The interface is sympathetic in the sense that it uses context to interpret the user's input correctly. In Johnson et al's [1999] example the character is a hen protecting her eggs from a racoon. The user can flap the soft toy's wings to make the hen fly. If the racoon is nearby, the hen will fly around its head and try to scratch it. If the racoon is not a threat, the same flying actions by the user will make the hen fly around. When the user makes the soft toy walk to get the hen to walk in the direction of the henhouse, the software picks this up and helps to get the chicken there without requiring the user to control the steering. The interface tries to understand the users' intentions based on the context and helps them to achieve their goals. The user has to control the character's behaviour rather than its motor actions.

Following this ‘intentional control’ approach allows the designers to have artistic control over the animations as the character need not move exactly the way the user moves the soft toy [Johnson et al., 1999]. It also makes it easier to keep virtual character ‘in character’.

To recognise the user’s actions, data from the doll (provided through the wireless sensors) is processed in real-time. Each action such as walk, run, fly, hop, and so on has at least one gesture recognition model, but when user styles tend to vary, multiple models are used. Machine learning is implemented through hidden Markov models which were originally developed for speech recognition systems.

When testing the interface, Johnson et al. [1999] found three types of users: those that used the interface without a problem, immediately understanding what to do (usually children); those who could be taught to use the interface; and those who could not learn to use the interface at all. While adults tired of manipulating the doll, children could do this continually and thought it was great fun [Johnson et al., 1999].

Some problems that Johnson et al. [1999] identified with the interface was that the character’s turning radius was too large, making it difficult for the user to make sharp turns; the chicken’s walking and running speeds could not be adjusted; there was sometimes so much variation in the way users performed an action and the system could not recognise the whole range; when the interface misunderstood users’ gestures they often responded by repeating the action more boldly and this may took further away from their intended goal.

I81	Children want to have control. The toy must play with them – they do not want to play with the toy.
I82	A soft toy can be used to manipulate an on-screen character by using a recognition engine that recognises the toy’s movements through wireless sensors embedded in the toy. The software includes one or more models for each possible action to aid the recognition process.
I83	When using a soft toy to manipulate an on-screen character a sympathetic interface can improve interaction. This means it must interpret the user’s intentions based on the context and takes control to let the character complete the action.

6.4.3 Storytelling Technologies

The *Sam* system [Ryokai et al., 2003] that I discussed in section 5.3.3.2, is an example of a tangible storytelling system. In this section I discuss four more recognised examples that use different kinds of interaction mechanisms.

6.4.3.1 Storymat

Storymat [Ryokai and Cassell, 1999] is a computer-based system for creating and replaying stories created by children. It consists of a cotton carpet with pictures of possible story elements on it, and a stuffed toy operating as a wireless mouse and a voice recording device. The child sits on the carpet and moves the toy around while making up a story (Figure 6.7). While speaking, the child squeezes the toy to activate the voice recorder. Simultaneously, the movements of the toy across the carpet are recorded. If, during a new storytelling session, a child moves the toy to a location where it has been in a previous story, that recorded story is played back from that point onwards. An overhead projector projects an image of the toy moving across the carpet while the story is played back. If the current user squeezes the toy, playback stops and a new ending is recorded for the partly played back story.

Figure 6.7 Storymat [MIT, 2008a]

Ryokai and Cassel [1999] found that children managed the system with ease and that they could use all the functionality available. The researchers claim that this system supports collaborative storytelling, although children use it without a playmate present. They only collaborate in the sense that one child can listen to and change a story created earlier by a different child.

6.4.3.2 oTTomer

oTTomer [Valinho and Correia, 2004] is an interactive story about a faraway planet aimed at children from six to twelve. Interaction takes place in a large interactive playground divided into six sections. The children are transported to the planet where they have to navigate physically and in real time through the story, helping small creatures with special powers. In each room, they collect story elements and interact with physical objects and devices. The interactive devices include video, sensors, colour and sound detectors, conversational agents and 3D graphics projections. At the time the Valinho and Correia [2004] paper was written oTTomer was still in the design phase. The questions they planned to address as the project progresses are: Is the interaction intuitive enough? Will children immediately know how to engage in the story? Are children's actions relevant to the story? They hoped at the time that they could develop this into an imaginary world that can be used in play on all different media, including the Internet and interactive TV.

6.4.3.3 KidsRoom

The MIT Media Lab's KidsRoom project also involves an interactive playroom with narrative playgrounds (see Figure 6.8) for children aged six to twelve [MIT Media Lab, 1996]. They can walk and run and interact with each other in the physical space which includes sounds, images of virtual characters and narration. The room resembles a children's bedroom that is transformed into a fantasy world through the children's actions.

The system uses action recognition, object tracking and event detection to enable interaction and sensor programs control the narrative.

Figure 6.8 KidsRoom [MIT, 1996]

6.4.3.4 PETS

PETS (Personal Electronic Teller of Stories) was developed by the HCI Lab at the University of Maryland [Montemayor et al., 2000]. Using this system, children assemble a robotic creature using available animal parts like wings, legs and ears (see Figure 6.9) and then use it to tell stories with the help of the accompanying My PETS software. With the software they can create emotions, draw emotive faces and compile a library of stories. When My PETS tells the story created by the children, the robot acts out the emotions involved. The designers believe that the child user should always be in control, since children want to decide their own activity patterns [Montemayor et al., 2000]. PETS can display different kinds of behaviour. Its head can follow or move away from a beam of light depending on its mood. If it is happy it moves its paws toward a person and if sad it pulls away. PETS communicates with the My PETS software via wireless radio frequency channels.

The researchers working on PETS have found that children love robots and are drawn to physical playthings [Montemayor et al., 2000]. They believe that in the same way that children who visit a zoo prefer interacting with squirrels and pigeons to watching more exotic animals behind bars, they are attracted by activities where they can be active participants. They would therefore prefer storytelling software that involves a robot that they can interact with, to basic screen-based software.

Figure 6.9 Examples of PETS creatures [University of Maryland HCI Lab, 2008]

Children are also naturally drawn to animals, hence the nature of the parts that they can use to build the robot [Montemayor et al., 2000]. The younger design partners were emphatic about their requirement that the robot should be huggable and cuddly, encouraging the designers to use materials like fur, foam and feathers. Children's play is always filled with emotions and it was natural to assume that they would also relate to PETS emotionally. Hence the focus on PETS's emotional reactions to the children's stories.

6.4.3.5 A Tangible Interface for KidPad

In section 5.4.1.2 I briefly discussed KidPad – a shared 2D drawing and story construction tool – on which the KidStory system [Benford et al., 2000] is based. Children draw story objects on an infinite drawing surface using a variety of screen-based drawing tools and then link the objects to represent a sequence of events [Stanton, Bayon, Neale, Ghali, Benford, Cobb, Ingram, O'Malley, Wilson and Pridmore, 2001]. KidStory is an extended version of KidPad that allows the use of multiple mice and collaborative use of screen-based tools to encourage collaboration.

Stanton et al. [2001] identified some problems with KidPad: creating links between objects and navigating a story are difficult for children to do with the mouse and keyboard; mechanisms to encourage collaboration are not 'natural' in the sense that they make it obvious for user what to do; it allows collaboration between groups of two or three children; and it only supports story creation and not retelling of existing stories. Stanton et al. [2001] wanted to address these issues by integrating the system with traditional story creation materials and making the interaction physical or tangible. They believed a tangible interface would support children's collaborative physical interactions, encourage interaction between larger groups, and provide opportunities to retell stories in front of an audience [Stanton et al., 2001].

In collaboration with children they designed a prototype interface that included a real paintbrush and coloured containers with which children could paint on-screen objects, and a carpet with pressure detectors (Figure 6.10) for moving left and right in the story using full body movement. The functionality of keyboard keys was simply transferred to these tangible input 'devices'. For the display they used a large flat screen. Through a process of testing and refinement the end product consisted of a large carpet with twelve pressure sensors around the sides. The sensors were covered with coloured rectangles, so that children could easily see where they had to step. Children requested that arrows be drawn on the carpet to indicate the direction of movement. Whereas previously the children jumped from square to square, they now stepped on the arrows carefully – sometimes so carefully that no pressure was detected. Children's altered behaviour caused by the arrows show that simple changes to an interface can result in unexpected user behaviour changes [Stanton et al., 2001].

Figure 6.10 The carpet and flat screen devices [Bayon, 2008]

The carpet of the final system consists of separate blocks that can be assembled easily and in different patterns allowing different styles of interaction. For vertical movement of objects on the screen Stanton et al. [2001] experimented with motion detection. Crouching and jumping actions for downward and upward movement were not successful as children looked up while crouching causing the downward movement to stop. As an alternative they let children show the camera a card with a green side facing for upward movement and a red card for downward movement. This was also difficult as they were required to keep the card in a certain position and they forgot which colour caused which movement. An obvious solution would be to use an arrow on the card, but children could still hold the card upside down.

To select a specific existing story object children suggested that they could 'show' the computer a picture of that object. To accomplish this, bar-codes were assigned to previously created objects and printed on cards with pictures of the corresponding objects (see Figure 6.11). When the child 'showed' that picture, the system scanned the barcode to identify the object. The bar-code component did, however, not encourage collaboration.

Figure 6.11 The barcoded objects [Bayon, 2008]

On reflection, Stanton et al. [2001] pointed out the following design guidelines:

1. The physical size of the interface elements and their tangibility influence interaction by encouraging collaboration between the children using the interface, as well as between the users and those observing the interaction. Large elements (e.g. the carpet) slow down interaction, allowing time to interact with co-users. Onlookers can easily see what is happening giving a sense of ‘audience’. The size makes it difficult for a single child to manipulate objects, so that they naturally work together to accomplish the intended results. Having to move physical props around also slows down interaction. Using more (and different) props together promotes collaboration as it avoids turn-taking and it encourages different collaboration styles.
2. Changes in the interface lead to differences in interaction. Stanton et al’s changes to KidPad’s interface worked well in terms of promoting collaboration, but the changes also introduced problems. When moving objects around children often overshot the target using the carpet interface and playing back a story was much easier with the mouse and keyboard. Designers of tangible interfaces should consider the influence of physical interaction on the interaction process and should only use them when they actually contribute to improved interaction.
3. Slight changes in how the tangible interface is presented can cause huge differences in user behaviour. For example, drawing arrows on the carpet had the effect that instead of jumping vigorously, children stepped carefully on the arrows.
4. Much can be achieved with low tech technology. It is more important for tangible systems to be adaptable to the users’ needs than to look high tech and shiny.

I84 Children should be in control in the sense that they can design their own activity patterns. They are attracted to activities where they can be active participants. (For example, they can assemble their own robotic toy.)

I85 Children prefer storytelling software that interacts with a robot to simple screen-based software (for example, the PETS robot acts out the emotions that children include in their stories).

I86 Young children prefer huggable or cuddly robotic pets.

I87 Stanton et al. [2001] derived the following design guidelines for tangible storytelling technologies:

- The physical size of the interface elements and their tangibility influence interaction by encouraging collaboration between the children using the interface, as well as between the users and those observing the interaction. Large elements slow down interaction, allowing time to interact with co-users. Onlookers can easily see what is happening giving a sense of 'audience'. The size makes it difficult for a single child to manipulate objects, so that they naturally work together to accomplish the intended results. Having to move physical props around also slows down interaction. Using more (and different) props together promotes collaboration as it avoids turn-taking and it encourages different collaboration styles.
- Designers of tangible interfaces should consider the influence of physical interaction on the interaction process and should only use tangible interface elements when they actually contribute to improved interaction.
- Slight changes in how the tangible interface is presented can cause huge differences in user behaviour and should always be tested with users.
- Much can be achieved with low tech technology. It is more important for tangible systems to be adaptable to the users' needs than to look high tech and shiny.

6.4.4 Robotic Pets

Personal robots is a new genre in human-computer interaction [Kahn(Jr.) et al., 2004]. It combines the research area of computer personas on desktop computers with research on physical computational artifacts that include augmented reality and tangible computing. The robotic toy that both Kahn et al. [2004] and Bartlett et al. [2004] used in their research is Sony's robotic dog named Aibo (Figure 6.12), that, at the time, was the most advanced robotic animal on the retail market.

Bartlett et al. [2004] tried to find out what makes children perceive a robotic pet more as a pet than as a robot. Sony's Aibo is a robotic dog that can perform playful behaviour, wag its tail, walk, fall down and stand up, sit and shake hands. Despite its robotic appearance, these features are enough to convince a child that it is a being with feelings, even if their attention is drawn to its robotic features. Making Aibo speak with a human voice changes children's perception – if it performs behaviour that normal dog would not do, its 'robotness' becomes more pronounced. Although children enjoy watching animated animals that talk, they prefer Aibo to act dog-like. Bartlett et al's [2004] experiment was conducted with preschool children as well as children from the first four school years.

Figure 6.12 AIBO [Hughes, 2001]

The children's first exposure to the robotic pets, was watching the robots play soccer. Although this is not dog-like behaviour they still described what they saw as dogs and not robots [Bartlett et al., 2004]. As the children's age increased they were more likely to acknowledge Aibo's robotic features. In general, the children believed Aibo had emotions and some thought he had a mind. After the children were shown other kinds of robots and Aibo's robotic features were pointed out, they agreed that Aibo meets the criteria for being a robot, but they still called it a dog. According to this study, Aibo fulfils biological animistic requirements to be regarded as a dog and children attribute intentions, feelings, emotional states, wishes and goals to the robotic pet. Although this was not specifically investigated, Bartlett et al. [2004] suspect that children also attribute emotional connection and companionship to Aibo. There are many physical characteristics of a dog that Aibo does not have, such as a wet nose and two eyes, but the basic physical structure is enough evidence that Aibo is a dog.

This research was conducted with the broader aim of determining to what extent a robot such as Aibo can assist blind people and people with other disabilities [Bartlett et al., 2004]. To be effective in providing assistance, the robot's features must match the expectations and attitudes of those that they are supposed to help. If it appears smarter or more alive than it actually is, this may cause unrealistic expectations and frustration.

Kahn et al. [2004] wanted to find out how robotic pets make it difficult to distinguish between different ontological categories (for example, animate and inanimate) and how they may impact children's social and moral development. Their research involved eighty children in the three to six years age group and two artifacts, namely Aibo and a soft stuffed dog. The two dogs were more or less the same size and both were black. They found that children engaged in imaginary play with both toys, although there were some differences in their conduct towards the toys. With Aibo there were more instances of exploratory and apprehensive behaviour and more attempts at reciprocity (in other words, the children expected the artifact to respond to their verbal and other behaviour). The children were more inclined to mistreat the stuffed dog and manipulate it in an animated way. The results show that children treated Aibo more as if it was a real dog. They reacted to Aibo's movements as if it might be a threat. The research suggests that children may form some type of moral relationships with robotic pets, but caring for such a pet would be somewhat like a man who 'cares' deeply about his car [Kahn(Jr.) et al., 2004].

Kahn et al. [2004] believe that artifacts such as Aibo represent a new technological genre that does not fall into traditional ontological categories. These artifacts are autonomous, adaptive, conveys a persona and they are embodied in the sense that computation is embedded in the artifacts. So, in one sense they behave as if alive, but they are definitely inanimate. Kahn et al. suggest that the HCI question should not be whether children treat such technologies as animate or inanimate, just as you would not ask of an orange object whether it is red or yellow. They would rather see a 'more nuanced psychology of human-robotic interaction' that can investigate children's understanding of and relationships with this new technological genre [Kahn(Jr.) et al., 2004].

With regard to HCI this research shows that with minimal social cues computational artifacts can initiate social responses [Kahn(Jr.) et al., 2004]. Another robotic pet that supports social development is Paro – a soft seal-like robot. In section 6.5, when I discuss interface elements aimed at children with disabilities, I will discuss Paro, whose therapeutic qualities were tested with children who are severely cognitively disabled.

I88	Children regard a robotic dog with basic dog-like features as a dog rather than a robot as long as it only performs dog-like actions. (For example, they disliked it when Aibo's designers let the robot speak.)
I89	Young children still believe it is a being with feelings and a mind if they know it is a robot. Older children are more likely to acknowledge robotic features.
I90	Characteristics of robotic pets designed to assist people with disabilities must match the expectations and attitudes of the intended users. It should not appear smarter or more alive than it actually is and create unrealistic expectations in the users.
I91	With minimal cues robots can initiate social responses. When comparing children's behaviour toward a robotic dog with their behaviour toward a stuffed dog, more exploratory and apprehensive behaviour was observed with the robot, they expected the robot to respond to their talking, they saw its movements as threatening and they handled it with more respect.

6.4.5 Mixed Reality Environments

In mixed reality environments (also referred to as augmented reality) views of the real world are combined with views of a virtual world [Rogers, Scaife, Gabrielli, Smith and Harris, 2004]. It provides a way to interact with digital information by manipulating familiar physical objects or acting in physical spaces, reflecting the way we interact with the everyday world. It has benefits such as enriching the user experience and enhancing learning and collaborative work [Rogers et al., 2004]. Rogers et al. examined how specific new forms of physical/digital embodiment might help children to explore more and to reflect on what they are doing. They conceptualise mixed reality environments in terms of transforms. These are changes in the state of the world encountered through perception, action or cognition. They define four kinds of transform types which link different combinations of actions and effects, which vary along physical (real) and digital (virtual) dimensions (see Table 6.2).

Table 6.2 Transform types and children's level of familiarity with them [Rogers et al., 2004].

Transform Type	**Example**	**Level of familiarity**
Physical action → Physical effect (PPt)	Mark on paper with pencil.	Highly familiar
Physical action → Digital effect (PDt)	Move wand and animation appears.	Unfamiliar
Digital action → Digital effect (DDt)	Click mouse to change display.	Familiar
Digital action → Physical effect (DPt)	Manipulate digital image of windmill to make physical windmill rotate.	Highly unfamiliar

The aim of Rogers et al's [2004] experiment was to understand how different combinations of the physical and the digital affect children's behaviour. Twenty children aged five to six were observed using the different configurations in pairs. The PDt transform (coloured blocks to create digital colour mixes) provided the most interesting observations. The physical blocks were easy to hold and manipulate and children could swap them, turn them and combine them in different ways. Actions were reversible and the feedback was immediate. Children could perform the actions together. They experimented by putting their faces on the tag reader to see if their faces appear on the screen; they tried stacking several blocks to see what colour emerges and they tried to intensify the colour projection by pressing hard on the blocks. There was a lot of collaboration. Although the DDt transform (digital colour and light mixing) also provided reversibility and immediate feedback, its interface was not as inviting and the input devices not so 'ready-at-hand' as the blocks. Only one child could use an input device at a time, forcing them to take turns. This is not a simple requirement when young children are involved. They may not want to hand over control or may forget about the other child. Rogers et al. conclude their findings as follows: '...the benefits of designing novel mixed realities in the context of play and learning is that by juxtaposing the ‘unexpected’ with the highly familiar promotes ‘richer’ experiences, prolonged interest and more reflection’ (p. 685).

6.4.6 Tabletop Environments

Sluis et al. [2004] investigated whether augmented tabletops can provide useful learning environments for young children. Earlier studies have shown that tabletop environments support collaboration and they offer a more natural interaction style. Sluis et al. [2004] give the two reasons why desktop environments are still generally preferred over tabletops. Firstly, current augmented tabletops are designed for specific application domains and mostly aimed at adult users. They are not available to the broader user population, and are not suitable for novice computer users. Secondly, very few software applications exist that exploit the benefits of a tabletop environment.

Sluis et al. [2004] developed Read-It, a tabletop application with a multimodal, tangible interface that helps children aged five to seven to learn to read. The system is based on the *Learning to Read Safely* method used in the Netherlands and specifically on its implementation in the *speelleesset* – a set of games that combines playing and reading – and in computer-based activities. According to Sluis et al., the game set has limitations – it is completely visual, with no audio or other feedback; to play the game together, children have to share a common orientation so that they do not confuse letters such as ‘p’ and ‘d’; some of the games are not self-corrective and requires adult supervision. The computer activities provide direct feedback, incorporate humour and check the user’s actions automatically. The computer application runs on a desktop computer which is not ideal for collaborative play.

Read-It was implemented on the Visual Interaction Platform. It uses a computer, two beamers, an infrared light source, an infrared-sensitive camera and a table with a reflective surface. The beamers project the display on the table. The camera captures what happens on the table and computer-vision software on the computer analyses the images providing tag positions, orientations and identifications as output that are used during the game. The chosen game is a memory game where the goal is to find matching pairs of hidden pictures on twenty tangible brick elements. The 55x55mm bricks are tagged with infrared reflecting tape for identification by the computer vision system. Sluis et al. [2004] found that children prefer bricks that are six millimetres thick and that have a notch for grasping. They flip pairs of bricks to show the pictures. Two pictures are a match if the words start with the same letter. This implementation combines the advantages of the *speelleesset* and the desktop application. Visual as well as audible cues are used, performance is automatically monitored and the game can be modified easily. Both physical and mental collaboration is promoted. The application allows for different ‘cards’ to be turned simultaneously or in succession. To confirm a match someone must press a designated button. The system can detect cheating and responds appropriately when, for example, a third ‘card’ is turned face up. What happens on the shared workspace is reflected on personal work spaces on two sides of the tabletop so that children on both sides of the table can see the words that describe the pictures with the correct orientation.

With regard to reading education, Read-It supports the need to attach individual graphemes to phonemes [Sluis et al., 2004]. When a ‘card’ is turned over the word is displayed grapheme by grapheme and spelled out at the same time. The children’s attention has to shift constantly between the play area and the personal work space keeping them stimulated and focussed. They do not need to concentrate on one aspect of the game for a long time. Good matches are rewarded with images of carrots in the personal work space. The level of difficulty can be modified easily and the game can be configured to match first sounds, middle sounds or last sounds of words. Two to four children can play the game together.

Testing of the system revealed that children understood the concept of augmented reality well and preferred the tabletop version to the paper version [Sluis et al., 2004]. They liked the inclusion of sound and the carrots as scoring method. There was less collaboration than was expected, but the reasons for this are not clear. The researchers felt that the recognition processes of the system should be replaced by a different technique such as radio-frequency tags and that it should be located underneath the tabletop to be less intrusive. The equipment is expensive, but they believe that it may become affordable.

I92	In mixed reality environments designers can use physical to digital transforms (where children manipulate physical objects to create a digital effect), digital to physical transforms (where digital manipulation has an effect on physical objects) or familiar digital to digital transforms.
I93	Physical to digital transforms can easily be designed to promote collaboration (for example, children manipulate different coloured blocks on a tabletop to mix colours on a screen).
I94	A tabletop interface can be used to transform familiar board or card games into interactive experiences by providing interesting feedback, preventing cheating and keeping track of player performance.

6.4.7 Existing Frameworks for the Design of Tangible Interaction

Hornecker and Buur [2006] and Antle [2007] studied the design of tangible systems. I end my discussion of tangible interfaces for children with a summary of their respective frameworks.

Hornecker and Buur [2006] present a framework for the design of tangible systems that are structured around four themes, namely tangible manipulation, spatial interaction, embodied facilitation and expressive representation. I summarise their framework in Table 6.3, giving a short description of each theme and some design questions that Hornecker and Buur associate with the themes.

Table 6.3 A Framework for Tangible Interaction [Hornecker and Buur, 2006]

Theme	**Description**	**Related design questions suggested by Hornecker and Buur [2006]**
Tangible manipulation	The actual tactile, material representations that are physically manipulated to establish interaction.	
Spatial interaction	The space and the movement in space of interaction-producing components.	Do people and objects meet? Is it a meaningful space? Does moving things (or your body) in space have meaning? Is the interaction visible to all participants? Which parts of the body can a user use? Do the movements communicate something?
Embodied facilitation	The effect of the configuration of material objects in space on user group behaviour.	Does the physical set-up constrain users in a way to compel users to collaborate? Can all users get their hands on interaction-producing components? Does the interaction build on users' skills in a way that invites them to interact?
Expressive representation	The expressive qualities and legibility of representations.	Are representations meaningful and of long-lasting importance? Are physical and digital representations equally strong and discernable? Can users think or talk with objects, using them as props? Do these props provide focus and a record of decisions? Are there clear links between user actions and the effect thereof? Is there a natural mapping between physical and digital representations?

Hornecker and Buur [2006] do not specifically address tangible technologies for young children, but their framework is formulated in a way that makes it applicable to all tangible technologies.

Like Hornecker and Buur [2006], Antle [2007] formulates her suggestions for the design of tangibles for children in terms of design-related questions. She identified three areas of cognitive development as particularly important in relation to the design of tangible technologies for children. These are embodied cognition (cognition grounded in bodily experience), spatial cognition, that is viewed by some as the foundation for abstract reasoning, and symbolic reasoning. She also uses the theory of development as a non-linear dynamic system to inform her suggestions. Grounding her design considerations in the theory relating to these aspects of development, Antle [2007] presents specific questions that may guide the design of tangibles. I summarise these in Table 6.4.

Table 6.4 Antle's [2007] design-related suggestions for tangibles

Cognitive Development Area	**Associated aspects that apply to tangibles**	**Related design questions as formulated by Antle**
Embodied cognition	Cognition is grounded in bodily experience, including sensation, perception, action and reflection. The concept of balance can, for example be applied to colours in a picture or balancing numbers on two sides of an equation sign. Mental rotation of objects is grounded in bodily rotation.	How can interactions be based on the ways children naturally solve problems using their bodies? How can we leverage children's understandings of bodily-based concepts to help them understand abstract concepts? How can we support parallel (not competing) use of motor, perceptual and cognitive processes?
Spatial cognition	Organisational structures of spatial schemata can facilitate memory, communication and reasoning. Abstract schemata can be developed from spatial schemata.	How can we base abstract concepts on children's understandings of spatial concepts and relationships? How can the physical and digital aspects of tangibles be used to support reciprocal mappings between spatial and mental representations?
Symbolic reasoning	Preschool children have difficulty seeing the same object as a model of two different things.	How can we support children to build up meaning actively through explorations of the relationships between representations and actual entities which are being represented? How can we make mappings between representations easily understood? How can we design representations to communicate how they are coupled to the world in ways that allow children to manipulate and understand multiple levels of meaning?
Development as a non-linear dynamic system	Physical growth, environmental factors, brain maturation and learning all interact to make development happen.	How can we create a system that allows flexible interactions and intelligent adaptive responses which allow children to adapt thinking over time? When and how should we provide local, fast, direct, real time feedback? How can tangible qualities of objects and spaces be utilised as adaptable, external aids which support the development of new understandings of schemata over time?

From Hornecker and Buur's [2006] framework:

I95 Four elements of tangible interaction are tangible manipulation, spatial interaction, embodied facilitation and expressive representation.

I96 When designing tangible systems, designers should be clear on:

- Which movements of a user's body have meaning.
- Which movements of physical representations have meaning.
- What these movements communicate.
- What should be visible to participants at specific times during the interaction.
- Whether physical elements should provide focus and a record of decisions

I97 If collaboration is required the physical set-up should constrain users so that they are compelled to collaborate.

I98 The interaction should build on users' skills in a way that will invite interaction.

I99 There should be clear links between user actions and the effect of these actions.

I100 Physical and digital representations should both be visible and there should be a natural mapping between them.

From Antle's [2007] framework:

I101 Base interactions on the ways children naturally solve problems using their bodies.

I102 Use children's understandings of bodily-based concepts to help them understand abstract concepts.

I103 Support parallel (not competing) use of motor, perceptual and cognitive processes.

I104 Base abstract concepts on children's understandings of spatial concepts and relationships.

I105 Use the physical and digital aspects of tangibles to support reciprocal mappings between spatial and mental representation.

I106 Make mappings between representations easily understood.

I107 Design representations to communicate how they are coupled to the world in ways that allow children to manipulate and understand multiple levels of meaning.

6.5 Interaction Environments for Children with Disabilities

Accessibility is an aspect of HCI that designers of technology are nowadays expected to take in to consideration. Preece et al. [2007] define accessibility as 'the degree to which an interactive product is usable by people with disabilities' (p.438). There is large range of disabilities, including severe conditions such as blindness, deafness and paralysis, and less severe ones such as dyslexia and colour blindness. Interaction design for young children with disabilities is a vast research field that could probably not be dealt with comprehensively in a complete thesis. My modest aim here is therefore to provide some insight into the complexities of designing for children with disabilities. I do this by reviewing a selection of research projects that investigated solutions to design problems relating to use by disabled children. The disabilities of interest here are visual impairments, hearing impairments and cognitive disabilities.

6.5.1 Haptic Interfaces for Visually Impaired Children

Patomäki, Raisamo, Salo, Pasto and Hippula [2004] believe that computers can provide visually impaired children with opportunities to learn and play. They found that computer applications for visually impaired children can be successful if tasks are well-designed and make use of haptic and auditory interfaces. Whereas text-based interfaces could easily be adapted for use by blind people, graphical interfaces that depend almost completely on visual feedback are difficult to translate into a form that is accessible to the visually impaired.

Patomäki et al. [2004] conducted a two-year study to test designs for haptic applications for young visually impaired children. They developed three multimodal learning and play environments that included a Phantom desktop device and a display with stereoscopic CrystalEyes 3D glasses (Figure 6.13). The Phantom is a haptic interface that generates accurate force feedback to simulate touch. It is operated with a pen-like stylus attached to a robotic arm which produces the force feedback. They also used a Magellan space mouse that consists of a large handle and several buttons. For some of the virtual objects in the applications the developers provided similar tangible real-world objects through which the children were made familiar with the virtual objects. They first touched the object with their hands and then with the stylus, to develop a feeling for the haptic feedback.

Figure 6.13 The Phantom device [LECO Research Group, 2005], Crystaleyes 3D glasses [Stereoscopy.com, 2000] and a Magellan space mouse

The subjects were eleven severely visually impaired children between three and a half and six and a half years of age. Eight had residual sight and five could see well enough to make partial use of visual feedback. They were allowed ample practice time with the devices before testing began – sometimes with assistance and sometimes on their own. During the tests they had to use the system as independently as possible. When they had completed the tasks the children taught their parents to use the devices. Observing this gave researchers the chance to assess how well the children's understood the system [Patomäki et al., 2004].

The first task involved recognition of virtual textures like sandpaper, a sponge, glass and a mouse mat. The context was a story about bears having to select a carpet for their home. The children found the sandpaper easiest to identify and the hardest was the mouse mat. They also did well on the glass and sponge. The audio feedback on this task imitated the sound made when moving something across the surface, so that harder and rougher textures produce more sound.

In the second task they had to track differently shaped paths or patterns, namely direct line, rooftop, sawtooth and castle wall. Here children were given haptic, auditory and visual feedback and the context was to guide baby moles through tunnels to find grandma's house. They found the direct line and the rooftop easy to trace, but the strict corners of the sawtooth and castle wall caused problems.

What Patomäki et al. [2004] learnt from these two tasks are:

- For tracing tasks, use rounded corners and wide paths.
- Children must learn to hold and position the stylus correctly (at this age the children's fine motor skills were not found sufficiently developed to use the Phantom device effectively – the device is intended for use by adults).
- For texture tasks, surfaces must be rougher with enough friction.
- Visual feedback did not really help children with partial vision, since it moved their attention away from the tactile feedback which was more important. Since the visual feedback and the tactile feedback in these experiments did not correspond perfectly (for safety reasons a mirror was removed from the display device so that the image was incorrectly orientated), it sometimes caused children to move the stylus in the wrong direction.

The next phase of Patomäki et al's [2004] research involved a game environment. The context was a story about a family of badgers and an ant. Here the feedback was mainly auditory, with a variety of sounds, and partly haptic. Magnetic objects allowed agents (story characters) to grab the stylus or create areas that attract the stylus to help with navigation. The interface also included a button with which children could open doors and make selections in the game environment. Tasks included finding an alarm clock in a room, with a ticking sound leading the user to the clock; finding a mailbox by moving through a tunnel; comparing various surfaces; and finding and popping balloons.

All the tasks in this phase were too hard. Children experienced problems holding the stylus and pressing the button at the same time. The design of the Phantom made the stylus slide downwards due to gravity. Children could not locate the balloons and when they did they moved the stylus in the wrong direction for the popping action. Children could move between the rooms easily and had no difficulty finding and opening the doors (door magnets helped getting the stylus in the correct position). Children enjoyed the lively

characters and they took part with great interest, sometimes talking back. The implementation was realistic in the sense that children believed there was actually something real (not virtual) there. Some tried to touch the objects with their free hand and one partially sighted child looked under the stylus, not believing that there was nothing there.
From this phase Patomäki et al. learnt the following:

- The magnets worked well for guiding the user to move or hold the stylus at a certain place.
- Some visual feedback would have helped the partially sighted children in these tasks.

In the third phase of the research the tests involved a learning environment where the children had to identify and find animals. The results from the previous two phases were taken into account and tasks were made easier. Visual feedback was provided for partially sighted children (but using a different device than before). Five of the children benefited from this. In these tasks children needed less assistance and were generally more successful than in the first two phases.

In general, girls were more patient and proficient when using the Phantom device. They held the stylus more correctly than the boys. The researchers suspect that the boys might have more undiagnosed handicaps than the girls, hence the results. The children's level of linguistic development also has an impact since objects that can be named are easier to remember. Sometimes children made up completely new stories or changed the given story to suit their perception of the haptic elements. The use of the Phantom is influenced by motor skills, ability to concentrate and the cognitive process of understanding the tactile experience. Patomäki et al. [2004] believe children over seven years of age will perform better than younger children. The children in the study varied substantially in their proficiency with the device. Often they could 'feel' the shape of an object but could not identify it correctly. Friction and surface texture make embossed graphs easier to distinguish from one another using the Phantom.

6.5.2 Sound Tools for Visually Impaired Children

McElligott and Van Leeuwen [2004] investigated the use of surround sound to provide blind children with opportunities to explore spatial relationships and distances. They designed a game in which children could use a joystick with force feedback to move sounds to fixed positions in space, to catch moving sounds, to place sounds and to 'throw' sounds. The idea is that these activities can be applied in complex games such as 'travelling the globe'. The results of experiments with seven to nine-year-old children showed that children could easily locate sounds and identify single sounds. They did have problems moving to the centre of a sound. They could remember up to four sounds from a scene. The children enjoyed being immersed in an aural space and the sounds associated with scary of forbidden actions caused high levels of excitement.

McElligott and Van Leeuwen's [2004] second experiment was concerned with visually impaired children's desire for autonomy, self-confidence and self-expression. With the help of five blind and visually impaired children they designed a recording device with which they could record their own voices, tell stories and make multitrack recordings that included sound effects, and record and edit real world sounds for use in games like sound guessing. This so-called Sounding Self workstation was implemented using a KORG Kaoss Pad designed for DJs which allows the user to tactually choose and control sound filters. In the experiment children took part in three activities, namely application and design of special effects, voice editing and DJ scratching. In the special effects task they had to simulate events, contexts and objects or creatures. In the voice editing task they recorded their own voices with real-time sound effects or they recorded their voices and edited it afterwards. For the scratching task they brought their own CDs and learnt the basics of scratching. The results showed that the children were able to use the technology with a high level of enjoyment and concentration. This may partly have been due to novelty.

In their third experiment, McElligott and Van Leeuwen [2004] provided children with opportunities to explore relationships between audio and tactile information. Sensory integration is important for the development of memory, consciousness, anticipation and intelligence [McElligott and Leeuwen, 2004]. McElligott and Van Leeuwen believe that receiving information through different senses and the availability of multiple modes of expression enhance involvement, attention and enjoyment. Using a monsters theme, they designed a system to support audio-tactile exploration and construction. Children could use their voices to create sounds that reflected the tactile characteristics of creatures. First, the children discussed with experimenters how a monster should sound and feel so that they could later use their ideas to create the monster. Children enjoyed using the sound processor to change the sound of their voices to make suitable monster voices. They had to express the monster’s characteristics and moods through sound. They were given fabrics and fillings from which to choose tactile characteristics for their monsters. Children could intuitively link textures to personality traits and verbalised emotional states that they associated with the textures. The technology gave them the opportunity to demonstrate and explore their ideas using a variety of sounds.

The games discussed above were all specially designed for visually impaired children. In the next section I look at the possibility of adapting existing software for use by blind children.

6.5.3 Adapting Existing Games for Use by Visually Impaired Children

Visually impaired users have unique requirements when interacting with game technology and it therefore makes sense to design games specifically for them. Archambault and Olivier [2005] have a different perspective, believing that allowing visually impaired children to play the same games as their sighted peers can reduce their feelings of isolation.

Archambault and Olivier [2005] discuss three modalities that can be used when adapting existing games for use by visually impaired users: sound, tactile devices and enlarged graphics:

- Correct interpretation of a sound can rely on the visual context to such an extent that when the visual aspects are removed the sound cannot be recognised. Designers should therefore use sounds that are recognisable out of the game context, or when adapting the game for visually impaired users, the necessary contextual information should be made available through modalities other than vision.

- Tactile input can be provided through Braille devices (for users who can read) or tactile overlays mounted on a tactile board that is connected to the keyboard port. The functions on the board are mapped to keyboard shortcuts and can thus only be used with applications that have keyboard shortcuts for all functions. Ideally only one overlay should be used for a whole game.
- For visually impaired children who have some degree of sight a zoom function that displays enlarged views of the display can be used. Since these children often cannot see moving images, designers should avoid using animations and can replace them with slide shows.

Often modification requires some changes to the content of a game [Archambault and Olivier, 2005]. For example, the help facility must be changed to refer to the tactile input rather than to mouse clicks or keyboard functions. Blind users also need immediate feedback on their actions, and subtle sound feedback will often be sufficient.

Archambault and Olivier [2005] propose a design model that allows designers to design games that are independent from specific interaction devices and modalities. It separates the logic of the game from the data needed to interact with the user. They used this model to design the TiM game engine (called Blindstation) that supports the design of games that can work with devices such as the standard keyboard and joystick, or with specialised input devices such as tactile boards or braille devices [Archambault and Olivier, 2005]. They have used this tool successfully to adapt existing games such as *Reader Rabbit: Toddler* and *Mudsplat*.

Adapting existing games for use by deaf children is less problematic. Sound elements that carry information that is crucial for interaction can be translated into text or graphic elements that convey the required message. There is, however, justification for designing games especially for children who are hearing impaired. In the next section I discuss a game that was designed to develop deaf children's language and communication skills.

I108	For tracing tasks, use rounded corners and wide paths.
I109	Fine motor skills of children younger than eight years are not sufficiently developed to use the Phantom device effectively – children must learn to hold and position the stylus correctly. Magnets can be used to guide the user to move or hold the stylus at a certain place.
I110	For texture tasks, surfaces must be rougher with enough friction.
I111	Visual feedback can help partially sighted children. Designers should, however, use it with care as in some applications it may move such users' attention away from tactile feedback which is more important.
I112	Girls are more patient when using the Phantom device.
I113	Phantom device use is influenced by motor skills, concentration and understanding of the tactile

experience.

I114 Blind children can use a joystick to move sounds to fixed positions in space, to 'catch' moving sounds and to 'throw' sounds.

I115 Activities that are suitable for blind children are:

- locating and identifying sounds,
- recording and editing their own voices, and
- creating their own tangible characters and giving them voice by recording/manipulating their own voices. The relation between tactile and audio information helps with sensory information.

I116 Correct interpretation of a sound can rely on the visual context to such an extent that when the visual aspects are removed the sound cannot be recognised. Use sounds that are recognisable out of the game context, or make the necessary contextual information available through modalities other than vision.

I117 Tactile input can be provided through Braille devices (for users who can read) or tactile overlays mounted on a tactile board that is connected to the keyboard port. The functions on the board are mapped to keyboard shortcuts and can thus only be used with applications that have keyboard shortcuts for all functions. Ideally only one overlay should be used for a whole game.

I118 For visually impaired children who have some degree of sight a zoom function that displays enlarged views of the display can be used. Since these children often cannot see moving images, designers should avoid using animations and can replace them with slide shows.

I119 Often modification requires some changes to the content of a game. For example, the help facility must be changed to refer to the tactile input rather than to mouse clicks or keyboard functions.

I120 Blind users also need immediate feedback on their actions, and subtle sound feedback will often be sufficient.

I121 To design games that are independent from specific interaction devices and modalities, use a design model that separates the logic of the game from the data needed to interact with the user.

6.5.4 Sign Language Tools for Hearing Impaired Children

Most deaf children have hearing parents and therefore do not grow up in a house where sign language is the first language. These children's language development is hampered by the lack of exposure to language models. Most computer games designed for deaf children focus on *understanding* sign language rather than *producing* it [Henderson, Lee, Brashear, Hamilton, Starner and Hamilton, 2005]. CopyCat [Brashear, Henderson, Park, Hamilton, Lee and Starner, 2006; Henderson et al., 2005] is a computer game aimed to help deaf children aged six to eleven to practice American Sign Language (ASL) Skills. It uses gesture recognition to 'listen to' and correct young children's use of ASL.

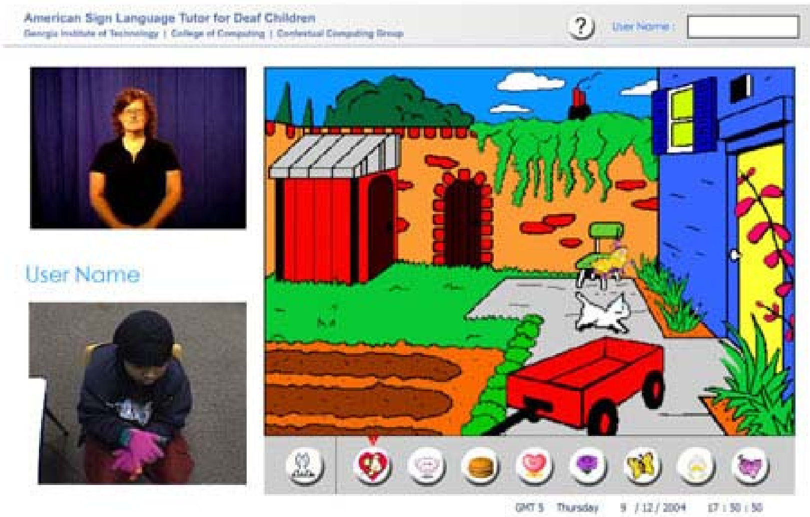


Figure 6.13 The CopyCat Interface [GVU Center, 2005]

The interface is made up of a window displaying video footage of a person demonstrating ASL phrases, a window displaying live video of the user and the game window with Iris, a cat that responds to the child's ASL gestures (see Figure 6.13). Action buttons allow the user to activate video clips that demonstrate phrases. When users are ready to sign they click an Attention button that activates the camera and they click again to indicate the end of their input. The 'push-to-sign' function helps to eliminate fidgeting and chatter as purposeful input. The interface does not include text so that it can be used by non-English speakers.

When users sign, their hands are video recorded and wireless accelerometers mounted inside gloves they wear provide additional data for recognising their signing (Figure 6.14). Multiple modes of data increases the accuracy of ASL recognition substantially [Henderson et al., 2005]. Bright pink gloves improve recognition as natural skin tone become problematic when the hands are, for example, moved in front of the face. Data gloves that are generally used to measure flexion and movement of the hands are expensive and not available in small children's sizes. Most importantly, the system is inexpensive and easy to configure.

Figure 6.14 Gloves with wrist-mounted accelerometers (left) and an accelerometer (right) [GVU Center, 2005]

The recognition engine could, at the time Brashear et al. [2006] reported on it, only recognise a subset of ASL, but the game uses a predetermined set of phrases which makes the recognition process somewhat easier. To train the recognition engine the researchers gathered a large amount of data from five children of the Atlanta Area School for the Deaf over nine days. The data is thus of signers conversing in a spontaneous unscripted manner [Brashear et al., 2006]. This data gathering is a time consuming and tedious task as a large number of samples must be recorded and labelled.

A detailed discussion of image processing and use of the accelerometer data can be found in Brashear et al. [2006]. The overall word accuracy of the recognition engine was over 90%. At the time of writing their recognition engine did not handle problems such as long pauses, hesitations and false starts, but that will be dealt with in further research.

I122 Gesture recognition can be used to detect sign language input.

I123 A push-to-sign function that allows users to indicate when signing starts and ends will help to eliminate the detection of fidgeting and chatter.

I124 Using a video recording together with accelerometers fitted into gloves worn by the signer can improve recognition rates as multiple modes of data for recognition increases the accuracy.

I125 If the signer wears brightly coloured gloves recognition is easier, especially for hand movements done in front of the face.

I126 A gesture recognition system for sign language recognition requires lots of training data.

6.5.5 Therapeutic Interaction with a Robotic Pet

Robots are usually designed to help people perform tasks or save time. There are, however, some robot designers who are more interested in the emotional effects of robots on humans, and how robots can be used to encourage social interaction. Marti, Pollini, Rullo and Shibata [2005] conducted a case study to investigate the therapeutic effect of a robotic pet on children with severe cognitive disabilities. The pet, Paro (Figure 6.14), is a baby seal that looks like a soft toy and has a complex network of sensors that react on input from the environment. It detects and responds to light, sound (including a speech recognition system), balance and touch. It can make vertical and horizontal neck movements, paddle movements and movements of the eyelids. Examples of its behaviour are: turning its head in the direction of a sound, becoming sleepy or tired (when its battery power is low), reacting to being stroked with coordinated movements of the body and head, fluttering its eyelids and making purring sounds.

In Marti et al's [2005] study, Paro was used in group therapy sessions with three patients with cognitive disabilities. The subjects are much older than the age group that I am interested in but I still regard the results as applicable to younger disabled children. Two are affected by Down syndrome and the other by Hanhart and Moebius syndromes. Chiara – a twenty-seven year old Down syndrome patient – is aggressive by nature and has poor collaboration and social skills. She resists taking part in group activities and does not make physical contact with others. The main goals of her treatment were to help her to produce context-relevant talk and improve her visual and physical contact with others. Emanuele was twenty-three and also a Down syndrome patient. He displays poor language skills and takes no initiative in social relationships. His therapy aimed to improve his relationship and communication skills. Paolo, the third patient was fourteen years old and severely physically and mentally handicapped. He cannot focus his attention even for very short periods and would start talking about events that has no relevance to the context. His therapists wanted to help him direct his conversation using appropriate content.

Figure 6.14 Paro [Keferl, 2007]

In the context of these patients' therapy, Marti et al. [2005] conducted an exploratory study over three months to see if Paro could mediate social interaction, help in focussing attention and stimulate sensory exploration. They made the following important observations:

- Initially the patients explored Paro individually, each waiting for his or her turn, but with their attention kept on the robot. In their usual therapy sessions this respectful behaviour was absent. Previously the patients had to be stimulated continuously by the therapist to remain focussed on an activity or to pay attention to someone else's activity.
- The patients did not only caress Paro with their hands, but also touched it with their noses and faces. They also acted as if protecting it from the cold or feeding it.
- Articulated social exchanges occurred between the patients in the presence of Paro. Chiara even touched Emanuele's hand – the first time she touched another person in a therapy session.
- The three patients worked together on a task where they created a story that involved taking photographs of, and with, Paro.
- For the first time Chiara used words that express emotion, such as 'sad', 'happy' or 'angry' when referring to Paro's state of mind. She also said about herself that she was tired and wanted to leave, whereas before she would just leave the room.
- When the robot was switched off, its effect on the patients' social, emotional and sensory behaviour was much less pronounced.

Marti et al. [2005] found that a robotic pet can mediate social exchange, stimulate engagement and support sensory exploration, especially if it displays reactive and proactive behaviour.

I only touched on the topic of designing for children with disabilities, but in doing so I demonstrated the important role that technology can play in treating, developing and entertaining these users.

I127	Robotic pets can support social development of children with severe cognitive disabilities.
I128	Robotic pets can mediate social interaction through reactive and proactive behaviour.

6.6 Web Applications for Young Children

The last topic of this chapter is the Internet and the World Wide Web. The WWW is largely text based and requires users to be able to read. Unless pre-reading children use web applications that are specifically designed for them, they cannot use the web without assistance of an adult or peer who can read.

Very little research has been done on the use of the WWW and the Internet by children younger than nine. One of the most useful contributions is that of Uden and Dix [2000] who investigated the design of an interface for an Internet search tool suitable for five to six-year-old children. The United Kingdom's National Curriculum requires that children in Keystage 1 (aged five to seven) are taught to use information technology (including the Internet) to supplement their problem-solving, information processing and creativity. The research of Uden and Dix [2000] was motivated by the concerns of teachers about the suitability of, for example, the Internet search facilities for young children. Without a suitable front the teachers would not be able to include the Internet into their curriculum. The reading abilities of children of five and six are not sufficient to use the general text-based interfaces. Iconic interfaces would be more suitable, but since icons that work for older children and adults are not necessarily appropriate for young children, the choice and design of these icons need to be investigated.

The aim is to design an interface in such a way that the user's mental model maps completely onto the designers conceptual model [Uden and Dix, 2000] and an obvious way to achieve this is to use metaphors. Metaphors can facilitate transfer of knowledge from a familiar domain to one that is less familiar, they can provide ideas for icon design and, when used successfully, they can help the user recognise the functionality of an icon. The challenge in icon design is to avoid ambiguity in the meaning of the icon. In Chapter 7, section 7.4.5 I return to the research by Uden and Dix [2000] to discuss their finding with regard to icon design for young children's web interfaces.

In contrast to Uden and Dix [2000], the research of Mitra and Rana [2001] showed that children can learn to use the Internet even if they cannot understand the text displayed. Data gathered from the famous 'Whole in the wall' project in India showed that children can teach themselves to become computer literate with no assistance other than the help they get from each other [Mitra and Rana, 2001]. Unfortunately these researchers do not distinguish between data relevant to children of different ages. The children who used the facility were from five to sixteen years old.

After ten years of WWW existence, Mioduser and Nachmias [2002] discuss education on the WWW in a survey of 436 educational web sites. They do not specify the age range of the targeted learners but I infer from the report that the web sites involved are aimed at secondary or tertiary education learners. We can still learn from their survey:

- In general the strengths of the Internet and the WWW are not utilised optimally. The teaching models resemble those that worked in the traditional classroom or for classic CAI lessons. Mioduser, Nachmias, Oren and Lahav [1999] explain that this is common and that it will take some time to make a complete transition to new possibilities offered by the Internet and the WWW.
- There are learning experiences that can be supported particularly well through the WWW – even for very young children. Mioduser et al. [1999] refer, for example, to a web site maintained by the University of Illinois at Urbana-Champaign where amongst other things, users can view a day-by-day multimedia tour of the life-cycle of a chicken's embryonic development.

Mioduser and Nachmias [2002] identified four functions of the WWW in teaching and learning, namely content delivery, instruction delivery, communication support and creation support. In terms of content delivery the video of the embryonic development mentioned above is suitable for children aged five to eight. Another example is Google Earth that can be employed to improve young children's spatial skills and their conception of geography. In terms of instruction delivery children can search for pictures or video material to support their school education, but, as Uden and Dix [2000] explained, young children will need a front-end that fulfils their needs to make these resources accessible to them. Once children have learnt to write they can use email facilities to communicate with friends and family, but with the assistance of a knowledgeable person. With regard to creation support children can, at a young age, learn to create a basic web page with pictures and symbols. Children who have access to the facilities can, for example, be encouraged to create such pages to keep grandma and grandpa up to date with their activities.

Several dangers are generally associated with children's use of the WWW. Web sites that are popular with younger children are often linked to popular television shows or toy companies who use the WWW to promote their products [Shields and Behrman, 2000]. Until about nine years of age, children find it difficult to distinguish between advertising and well-designed software or web sites with good intentions [Plowman and Stephen, 2003]. They should be taught the realities of Internet advertising. Children left to use the Internet unsupervised can (accidentally or on purpose) access unsuitable content such as graphics and video material of a sexual or violent nature. These issues are related to the circumstances under which children use the Internet rather that to design of applications aimed at children.

I129	There are learning experiences that can be supported particularly well through the WWW – even for very young children. For example, a day-by-day multimedia tour of the life-cycle of a chicken's embryonic development and Google Earth.
I130	In terms of instruction delivery, young children can search for pictures or video material to support

their school education, but they will need a front-end that makes these resources accessible to them.

I131 Once children have learnt to write they can use email facilities to communicate with friends and family, but with the assistance of a knowledgeable person.

I132 Young children can learn to create a basic web page with pictures and symbols. Children who have access to the facilities can, for example, be encouraged to create such pages to keep grandma and grandpa up to date with their activities.

6.7 Conclusion

With this chapter I concluded the part of my study that involved analysing existing literature on young children and technology with the aim of extracting guidelines for the design of technology.

The contributions of this chapter to my study are similar to those of Chapter 5 (see section 5.6), but where Chapter 5 dealt with technology aimed at supporting the development of specific cognitive skills, this chapter focused on the way children interact with technology. Based on a wide-ranging literature investigation, I here provided a detailed description of the interaction devices and environments available for children aged five to eight. This survey is in itself already a valuable 'tool' for designers of children's technology, as it provides a contained account of the wide range of suitable interaction devices and environments available. I took from this review the specific knowledge that pertains to the design of technology for young children and presented it as 132 potential guideline-generating ideas. These will be processed in Chapter 8 with the aim of integration into the intended framework of guidelines.

So far I have constructed a picture of the cognitive make-up of a five-to-eight year old child, I have learnt how technology can be designed to support young children's skill development and I have provided a decent impression of the range of interaction devices and environments available that are suitable for young children. Before I can analyse the assortment of potential guidelines extracted so far and organise them into a useful framework, one step remains. To complete phase 2 of my study I have to critically investigate existing guidelines for the design and evaluation of children's technology and evaluate the applicability of existing usability principles and design guidelines for adult products to technology aimed at children. In Chapter 7 I will give a systematic, case-by-case discussion of existing guidelines for the design of technology in general and for the design of children's technology in particular.

CHAPTER 7

Existing Guidelines

Table of Contents

7.1 Introduction

The final step in my literature study is to review the existing guidelines for the design of technology and, for those guidelines that are not specifically aimed at design for young children, discuss their applicability to children's technologies.

Various researchers and HCI specialists have proposed guidelines (or principles) for the design (or evaluation) of technology in general. The cases that I present in section 7.2 are Dix et al. [2004], Preece et al. [2007], Shneiderman [1998] and Nielsen [1994]. The work of authors that have presented guidelines specifically for the design of young children's technology is less familiar. Of these I have identified the following to include in section 7.3: Malone [1982], Clements and Battista [2000], Fishel [2001], Grammenos and Stephanidis [2002], Baumgarten [2003], Gilutz and Nielsen [2002], and Wyeth and Purchase [2003]. In the same section I also discuss the work of authors who formulated guidelines for the evaluation of children's products, namely Shade [1996], Haugland and Shade [1988], Buckleitner [1999] and Siraj-Blatchford and Siraj-Blatchford [2001]. Section 7.4 contains a variety of design guidelines for children's technology that were not presented by the authors in the form of guidelines, and that I have not dealt with in previous chapters. I conclude the chapter in section 7.5.

7.2 General Guidelines, Usability and Design Principles and Heuristics

Guidelines are usually presented as lists of rules telling designers what will work in a design and what will not. They can be high-level guiding principles that are widely applicable or low-level design rules that are detailed, specific and leave little room for interpretation by the designer. As I have explained in Chapter 2, the difference between design principles and usability principles are that design principles informs the design of a system, while usability principles are mostly used as the basis for evaluating prototypes and complete systems [Preece et al., 2007]. Usability principles can be more prescriptive than design principles. When used in practice, design or usability principles are often referred to as heuristics [Preece et al., 2007]. In this section I give an overview of four widely used sets of design guidelines, principles and heuristics.

7.2.1 Dix, Finlay, Abowd and Beale

Dix et al. [2004] provide interface designers with a comprehensive set of high-level directing principles with the aim of improving the usability of interactive systems. They divide their principles into three categories, namely Learnability principles, Flexibility principles and Robustness principles. They summarise their principles in three tables that I reproduce below as tables 7.1, 7.3 and 7.5. In these tables their words appear in a normal font while my added explanations are in italics. Following each of these tables respectively, are tables 7.2, 7.4 and 7.6 that explain some principles related to the core principles. For each category I also discuss the applicability of the principles to products for young children.

7.2.1.1 Learnability

Learnability refers to the ease with which users can enter a new system and reach a maximal level of performance [Dix et al., 2004]. Dix et al. identified five principles that affect the learnability of a computer-based system. They are defined in Table 7.1.

Table 7.1 Principles that affect Learnability (from Dix et al. [2004], p. 261)

Principle	**Definition**	**Related principles (explained in Table 7.2)**
Predictability	Support for the user to determine the effect of future action based on past interaction history.	Operation visibility
Synthesisability	Support for the user to assess the effect of past operations on the current state. *To be able to predict future behaviour, a user should know the effect of previous actions on the system. Changes to the internal state of the system must be visible to users so that they can associate it with the operation that caused it.*	Immediate/eventual Honesty
Familiarity	The extent to which a user's knowledge and experience in other real-world or computer-based domains can be applied when interacting with a new system. *The user's first impression is important here. Familiarity can be achieved through metaphors and through affective use of affordances that exist for interface objects. Clickable objects must look clickable, for example.*	Guessability, affordance
Generalisability	Support for the user to extend knowledge of specific interaction within and across applications to other similar situations.	
Consistency	Likeness in input-output behaviour arising from similar situations or similar task objectives.	

I explain the related principles mentioned above in Table 7.2.

Table 7.2 Principles that relate to Learnability principles

Principle	**Explanation**
Operation visibility	The way in which the availability of possible next operations are shown to the user and how the user is informed that certain operations are not available.
Honesty	The ability of the user interface to provide an observable and informative account of any change an operation makes to the internal state of the system. It is immediate when the notification requires no further interaction by the user. It is eventual when the user has to issue explicit directives to make the changes observable.
Guessability and affordance	The way the appearance of the object stimulates a familiarity with its behaviour or function.

7.2.1.2 Learnability and Young Children's Technology

All of the Learnability principles can be applied to products aimed at young children. Surprise is often a desirable element in children's games and can increase the experience of fun and engagement, but when it comes to learning how to use a system and navigating through the available functions and activities, predictability is very important. If they performed an action before, they will expect the system to behave similarly when they perform that action again. It is important that the operations that a young user can perform next are made known through age appropriate means.

When young children perform a printing operation and the printer does not respond, they tend to keep selecting the print option with the hope that it will eventually print. This is an example of the importance of synthesisability for children's products – the system should immediately provide age appropriate feedback telling the child the print document is in a queue but the printer is possibly not connected, so that they do not keep sending the print document.

Familiarity has a different meaning for children than for adults – they 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. Uden and Dix [2000] made some interesting observations in this regard that I discuss in section 7.4.5.

7.2.1.3 Flexibility

Flexibility refers to the many ways in which interaction between the user and the system can take place. Dix et al.'s [2004] main principles that relate to flexibility are explained in table 7.3 and other principles that relate to these, in table 7.4.

Table 7.3 Principles that affect Flexibility (from Dix et al. [2004], p. 266)

Principle	**Definition**	**Related principles**
Dialogue initiative	Allowing the user freedom from artificial constraints on the input dialogue imposed by the system.	System/user pre-emptiveness
Multi-threading	Ability of the system to support user interaction pertaining to more than one task at a time.	Concurrent vs. interleaving, modality
Task migratability	The ability to pass control for the execution of a given task so that it becomes either internalised by user or system or shared between them. *For example, a spell checker does some of the work but should ultimately let the user decide which words to replace.*	
Substitutivity	Allowing equivalent values of input and output to be arbitrarily substituted for each other.	Representation multiplicity, equal opportunity

Principle	Definition	Related principles
Customisability	Modifiability of the user interface by the user or the system.	Adaptivity, adaptability

Table 7.4 Principles that relate to Flexibility principles

Principle	Explanation
System pre-emptiveness	This occurs when the system initiates all dialogue and the user simply responds to requests for information. It hinders flexibility, but may be necessary in multi-user systems where users should not be allowed to perform actions simultaneously.
User pre-emptiveness	This gives the user freedom to initiate any action towards the system. It promotes flexibility, but too much freedom may cause the user to loose track of uncompleted tasks.
Concurrent/ interleaved multi-threading	Concurrent multi-threading allows simultaneous communication of information pertaining to separate tasks. Interleaved multi-threading permits temporal overlap between separate tasks, but at any time the dialogue is restricted to a single task.
Multi-modality	Separate modalities (channels of communication) are combined to form a single input or output expression.
Representation multiplicity	Flexibility for rendering of state information, e.g. in different formats or modes.
Equal opportunity	Blurs the distinction between input and output at the interface – the user has the choice of what is input and what is output; in addition, output can be reused as input.
Adaptability	Refers to user-initiated modification to adjust the form of input and output. Users may for example choose between different languages or complexity levels.
Adaptivity	Refers to system-initiated modification to customise the user interface automatically. Here the system should observe the users' behaviour (for example, repeated attempts at tasks) and determine their level of expertise in order to adjust the complexity level of tasks.

7.2.1.4 Flexibility and Young Children's Technology

The way the Dialogue initiative principle is applied to children's software will depend on the type of software (for example, educational or game software) and the type of tasks that make up the software (for example, open-ended or time-restricted). Software that requires children to perform specific actions at specific times or places in the program may require some degree of system pre-emptive dialogue, while software that just allows children to freely explore a simulated environment (such as outer space) would be largely user pre-emptive. From the research reviewed so far it is clear that young children want to control the interaction. A user pre-emptive style of interaction would therefore be preferable to a system pre-emptive style. System pre-emptiveness may, however, still be necessary to prevent users from causing damage or completely losing track of the tasks they have initiated [Dix et al., 2004].

Multi-threading does not apply to single user systems aimed at young children as they do not have the memory capacity and attention skills to work at different tasks at the same time. Multi-threading may, potentially, be employed in systems aimed at collaborative use where different users can simultaneously use different, complementing components of the system. Multi-modal communication between the user and the system can be particularly helpful in children's technology. For example, simultaneous audio and text cues can make a system accessible for reading as well as pre-reading children. I have also discussed how different modalities can be used to help children with disabilities (see Chapter 6, section 6.5).

The remaining three principles can all be applied to children's software. Task migratability applies, for example, to handwriting input where children should always have the keyboard available to correct recognition errors. Substitutivity is important as children have varying skill levels that will influence the type of input or output that is suitable for a specific user. From an educational point of view representational multiplicity is important, since providing children with different visualisations of the same information can support their learning of the concepts or knowledge involved. Customisability, and particularly adaptivity, is very relevant to children's software, since their development up to eight years is very rapid. A product aimed at children of different ages should be able to adapt to the users' varying levels of knowledge and development.

7.2.1.5 Robustness

Robustness refers to the level of support the user is given for successful achievement and assessment of their goals [Dix et al., 2004]. Table 7.5 summarises Dix et al.'s Robustness principles and table 7.6 some supporting principles.

Table 7.5 Principles that affect Robustness (from Dix et al. [2004], p. 270)

Principle	**Definition**	**Related principles**
Observability	Ability of the user to evaluate the internal state of the system from its perceivable representation. The user compares the current state with his or her intention within the task-action plan.	Browsability, static/dynamic defaults, reachability, persistence, operation visibility
Recoverability	Ability of the user to take corrective action once an error has been recognized.	Reachability, forward/backward recovery, commensurate effort

Principle	Definition	Related principles
Responsiveness	How the user perceives the rate of communication with the system. Response time is the duration of time needed by a system to inform the user of state changes. When this is not instantaneous the system should give some indication that the task is in progress.	Stability
Task conformance	The degree to which the system services support all of the tasks the user wishes to perform and in the way the user understands them.	Task completeness, task adequacy.

Table 7.6 Principles that relate to Robustness principles

Principle	Explanation
Browsability	This allows the user to explore the current internal state of the system via the limited view provided at the interface. The user should be able to browse to some extent to get a clear picture of what is going on, but negative side-effects should be avoided.
Static/Dynamic defaults	Static defaults are defined within the system or acquired at initialisation. Dynamic defaults evolve during the interactive session (for example, the system may pick up a certain user input preference and provide this as the default input where applicable).
Reachability	The possibility of navigation through the observable system states.
Persistence	Deals with 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 memory while visual communication remains available as long as the user can see the display.
Backward recovery	Involves an attempt to undo the effects of previous interaction in order to return to a prior state.
Forward recovery	Involves the acceptance of the current state and negotiation from that state towards the desired state.
Commensurate effort	If it is difficult to undo a given effect on the state, then it should have been difficult to do in the first place.
Stability	The invariance in response times for identical or similar computational resources.
Task completeness	Refers to the coverage of all the tasks of interest and whether or not they are supported in a way the user prefers.
Task adequacy	This addresses the user's understanding of the tasks.

7.2.1.6 Robustness and Young Children's Technology

It is difficult to say to what extent children aged five to eight are able to evaluate the internal state of a system and whether they relate the current perceivable interface with the internal state. Browsability is not a feature that is suitable for young children as they cannot be expected to use browsing as a way to get a clear picture of the system's current state. If the application involves reaching a series of sub goals to achieve some central goal, it may, however, be necessary to give them (or an adult assisting them) the option to view their current progress. In *I Spy Spooky Mansion* [Scholastic, 2002] this functionality is built into the game. Reaching a sub goal gives users a jigsaw puzzle piece to insert into a puzzle that must be completed before they receive the ultimate reward. The jigsaw puzzle also reflects the child's progress in the game. In *TimezAttack* [BigBrainz, 2005] children can click on a 'map' icon that displays their progress in the form of a map indicating which tables they have completed and which not. Unless a specific sequence of actions is necessary, children should be allowed to go directly to their favourite parts of the system. Children like to play their favourite games over and over and will find it very frustrating if they cannot get to those easily. Reachability is thus important. Persistence is also relevant in children's products. Providing children with audio instructions in the beginning of the game may not be adequate as children's short-term memory may not be sufficiently developed to remember the instructions until they are needed. The fact that pre-reading children have to rely on audio instructions make this an important aspect to consider.

With regard to Recoverability, young children should not be expected to know how to use Undo/Redo commands – the system should help them recover from an error through help that fits their level of understanding. Error prevention is essential in children's software and, due to their limited attention span, responsiveness is crucial. A lack of instantaneous feedback evokes repeated clicking or hitting of keys that may influence the program's execution.

I have shown above that most of Dix et al.'s [2004] Learnability, Flexibility and Robustness principles are applicable to technology aimed at young children.

7.2.2 Preece, Rogers and Sharp

Preece, Roger and Sharp [2007] discuss two types of design goals in interaction design, namely usability goals and user experience goals. The usability goals focus on aspects such as effectiveness and learnability, while the usability goals are concerned with the quality of the user's experience with the system and focus on things like aesthetics and enjoyment.

7.2.2.1 Usability Goals

Preece et al. [2007] have identified six usability goals that will ensure that people's interaction with technology is effective and enjoyable. I summarise these goals in Table 7.7.

Table 7.7 Preece et al.'s [2007] usability goals

Usability goal	**Explanation**
Effectiveness	A general goal that refers to how well a system is doing what is what designed for.
Efficiency	This has to do with how well a system supports users in carrying out their work. The focus is on productivity.

Usability goal	Explanation
Safety	Protecting the user from dangerous conditions and undesirable situations.
Utility	The extent to which a system provides the required functionality for the tasks it was intended to support. Users should be able to carry out all the tasks in the way they want to do them.
Learnability	How easily users learn to use the system.
Memorability	How easy it is to remember how to perform tasks that have been done before.

7.2.2.2 Applicability of Usability Goals to Children's Products

Any product should be effective – a product aimed at teaching a five-year-old to identify the first letters of words will be effective if children actually learn this skill successfully by using the system. In young children's products the focus is rarely on productivity, so that efficiency as defined by Preece et al. is less important. In the discussion of children's preferences with regard to input devices (section 6.2.5) it became clear that efficiency does not play a role in preschoolers' choices. They are focussed on the process rather than the end product and do not naturally work against time.

Safety is an important factor in children's products. This includes how children are affected physically by using the system (the equipment should not require them to perform actions or sit in positions that may harm them physically), how they can be affected by accessing material that is not appropriate (such as pornography or images of violence) and how they are psychologically influenced by the content of a computer game or marketing material.

To determine whether a product has adequate utility, designers should ask whether it allows children to carry out tasks in the way that they would like to do them. As Uden and Dix [2000] learned, children know what they want and they expect adults to know too. For the sake of utility, it is therefore important that designers do not make assumptions about children's preferences.

Learnability and memorability apply to children's products in the same way that they apply to any other product.

7.2.2.3 User Experience Goals

According to Preece et al. [2007], how the user feels about a product irrespective of its efficiency, effectiveness, learnability and so on, plays an important role in it being well accepted or not. For a system to provide users with positive experiences of interaction, designers should attend to features that will make the product satisfying, enjoyable, engaging, pleasurable, exciting, entertaining, helpful, motivating, aesthetically pleasing, supportive of creativity, cognitively stimulating, rewarding, fun, provocative, surprising, emotionally fulfilling, challenging, and enhancing sociability [Preece et al., 2007]. Features that make a product boring, frustrating, annoying or cutesy should be avoided.

Clearly one would not spend too much design effort on making a spreadsheet application entertaining or emotionally fulfilling, but these user experience goals are applicable to many new technologies in different application areas. Factors that may support the fulfilment of these user experience goals include attention, pace, interactivity, engagement and style of narrative [Preece et al., 2007].

7.2.2.4 Applicability of User Experience Goals to Children's Products

Technology for young children is almost always aimed at entertainment and learning, therefore all of these goals are relevant. There is often a trade-off between user experience goals and usability goals [Preece et al., 2007]. An action that requires more effort may contribute towards making a product more enjoyable and engaging. MacFarlane et al. [2005] investigated the relationship between fun and usability when children are using educational software. They found a correlation between observed fun and observed usability, as well as between reported fun and reported usability, although there was no correlation between observed usability and reported usability, or between observed fun and reported fun. Clearly, usability and fun are complex concepts and designers should be aware of the consequences of combining experience and usability goals and make sure that they address the needs of the user.

Adults and children have different ideas about what is boring or exciting and designers should be careful when using their own definitions of such concepts to guide their design decisions.

7.2.2.5 Design Principles

According to Preece et al. [2007], design principles are prescriptive suggestions to help designers to explain or improve their designs. Instead of telling the designer exactly how to design an interface, they inspire careful design, telling the designer what will work and what not. Preece et al. discuss a number of design principles that I summarise in Table 7.8.

Table 7.8 Summary of Preece et al.'s [2007] discussion of design principles

Principle	**Explanation**
Visibility	The more visible the available functions are, the better users will be able to perform their next task.
Feedback	This involves providing information (audio, tactile, verbal or visual) about what action the user has performed and what the effect of that action was.
Constraints	These restrict the actions a user can take at a specific point during the interaction. This is an effective error prevention mechanism.
Mapping	This has to do with the relationships between interface elements and their effect on the system. For example, clicking on a left-pointing arrow at the top left hand corner of the screen takes the user to the previous page and a right-pointing arrow in the right hand corner take the user to the next page.

Principle	Explanation
Consistency	This is similar to consistency as defined by Dix et al. [2004].
Affordance	This refers to an attribute of an object that tells people how it should be used. In an interface it is the perceived affordance of an interface element that helps the user see what it can be used for. Whereas a real button affords pushing, an interface button affords clicking. A real door affords opening and closing, but an image of a door on an interface affords clicking in order to 'open' it.

7.2.2.6 Applicability of the Design Principles to Young Children's Technology

All of these principles apply to children's technology. Visibility and feedback have been discussed at various points in this thesis. In children's products constraints can be used effectively to support a child in performing a task. For example, when a child has to build a virtual jigsaw puzzle by dragging pieces to the correct spot, incorrectly placed pieces can just snap back to their original position outside the puzzle. This is an example of a physical constraint. A logical constraint would be, for example, not allowing a child to place a sea creature on dry land in a storytelling environment that requires only real world characters.

7.2.3 Shneiderman

Shneiderman's [1998] principles for user-centred design are divided into three groups. I discuss these and mention their applicability to children's products where relevant.

7.2.3.1 Recognise Diversity

Before the task of designing a system can begin, information about the intended users, tasks, environment of use and frequency of use must be gathered. According to Shneiderman, this involves the characterisation of three aspects relating to the intended system: usage profiles, task profiles and interactions styles. I explain these in Table 7.9.

Table 7.9 Three aspects relating to recognition of diversity [Shneiderman, 1998]

Aspect	Explanation
Usage profiles	Designers must understand the intended users. Shneiderman lists several characteristics that should be described. Those that apply to young children are age, gender, physical abilities, level of education, cultural or ethnic background, and personality. Designers should find out whether all users will be novices or if they will have experience with the particular kind of system, or if a mixture of novice and expert users are expected. Different levels of expertise will require a layered approach whereby novices are given few options to choose from and are closely protected from making mistakes. As their confidence grows they can move to more advanced levels. Users who enter the system with knowledge of the tasks should be able to progress faster through the levels.
Task profiles	A complete task analysis should be done and all task objects and actions identified.

Aspect	Explanation
Interaction styles	Suitable interaction styles should be identified from those available. Here Shneiderman mentions menu selection, form fill-in, command language, natural language and direct manipulation. Of these, menu selection, natural language and direct manipulation are suitable for interfaces aimed at young children. In Chapter 6 I discussed many interaction styles that are suitable for young children. Of course, if a product should be accessible to children with disabilities, special interaction styles must be considered.

7.2.3.2 The Eight Golden Rules for Interface Design

Shneiderman [1998] suggests eight principles of design that are applicable to most interactive systems. They overlap to some extent with those of Dix et al. [2004] and Preece et al. [2007] and are mostly self-explanatory. In Table 7.10 I list them and elaborate on those that require explanation. Where appropriate I refer to their applicability to children's products.

Table 7.10 Shneiderman's [1998] eight golden rules

Rule	Discussion
1. Strive for consistency.	Similar to Dix et al.'s learnability principles.
2. Enable frequent users to use shortcuts.	Children's software often begins with an introduction accompanied by music and/or a voice welcoming the child to the game. The voice usually continues with instructions on how to get started. Many such products repeat the introduction every time the product is used. Some allow the child to interrupt the introduction with, for example, a mouse click, but never tell them about this possibility. If they do not find this out by accident they will listen to the introduction every time.
3. Offer informative feedback.	This is a particularly important principle for children's products. The content of the feedback should be understandable by a young child and the format in which it is presented should be suitable for the targeted age group.
4. Design dialogues to yield closure (the completion of a group of actions).	Users should know (through proper feedback) when they have completed a set of tasks.
5. Offer error prevention and simple error handling.	Design so that users cannot make mistakes.
6. Permit easy reversal of actions.	
7. Support internal locus of control.	Let users initiate actions instead of always just responding.
8. Reduce short-term memory load.	Shneiderman proposes this as a rule that applies to any software, but it is particularly important when designing for young children.

7.2.3.3 Prevent Errors

The last group of principles proposed by Shneiderman [1998] pertain to designing to prevent the user from making errors. It is imperative that designers of children's software anticipate every possible error that may occur and build a mechanism into the system that will prevent it. Error messages cannot be read by preschool children, so unless instructions for error recovery are presented at their level and in a way that will not intimidate them, they will not be able to recover on their own. Shneiderman suggest three techniques for error prevention, namely correct matching pairs, complete sequences and correct commands. In terms of children's software 'complete sequences' can be applied to reduce a complete introductory sequence to a short introduction when the child indicates in some way that he or she has seen the introduction before. The 'correct commands' techniques may, for example, be used in software aimed at teaching a child to read and write. When a child has to write a word requested by the system and do not know how, the system may provide some alternatives between which the child has to choose, or it may provide part of the word that the child should them complete. The 'correct matching pairs' technique is only relevant when writing text or programming commands where users may forget to close a parenthesised statement or a quotation, for example, and therefore only applicable where users are required to provide text input.

7.2.4 Nielsen's Heuristics

An empirical analysis of 249 usability problems led Nielsen [1994] to formulate a set of heuristics for the evaluation of a system's user interface. Preece et al. [2007] list a revised set of heuristics that I reproduce in Table 7.11.

Table 7.11 Nielsen's heuristics [Preece et al., 2007]

Heuristic	Description
Visibility of system status	Provide appropriate feedback within reasonable time to keep users informed of what is happening.
Match between the system and the real world	The interface should use language and concepts that the user is familiar with. Follow real-world conventions so that information appear natural and in logical order.
User control and freedom	Provide users with exits so that they can recover from unintentional or incorrect actions. Support redo and undo.
Consistency and standards	Follow known standards and conventions.
Error prevention	Design to prevent errors rather than to help users recover from errors. Require users to confirm potentially erroneous actions before performing them.
Recognition rather than recall	Minimise memory load by making objects, actions and options visible. Users should not need to remember instructions or previous choices when interacting with a system.

Heuristic	Description
Flexibility and efficiency of use	Provide expert users with shortcuts and allow users to tailor frequent actions.
Aesthetic and minimalist design	Only include relevant information in the interface.
Help users recognise, diagnose and recover from errors	Give error message in language that users can understand. Describe the problem precisely and suggest a solution.
Help and documentation	The user should be able to use the system without documentation, but when it is required make it easy to search and give clear steps to accomplish tasks.

7.2.4.1 Applicability of Nielsen's Heuristics to Children's Technology

All of these heuristics have come up in earlier discussions and they all apply to children's products. Some can be interpreted in slightly different ways when applying them to interfaces for children and adults respectively. For example, what counts as 'feedback within reasonable time' for adults is not necessarily quick enough for young children. If children do not see any effect when they expect something to happen, they will immediately try it again (and again, and again). Children's 'real world' includes fantasy characters or creatures and what appears far-fetched or ridiculous to an adult may seem perfectly natural to a young child. With regard to help and documentation, it is particularly important that young children are able to use the system without documentation.

To conclude my discussion of existing guidelines, principles and heuristics, I present one list of guidelines that hopefully captures all the issues raised by Dix et al. [2004], Preece et al. [2007], Shneiderman [1998] and Nielsen [1994] that can be related to the design of technology for young children.

E1	Children should be able to determine the effect of future action based on past interaction history. Changes to the internal state of the system must be visible so that users can associate them with the operations that caused them.
E2	Provide adequate feedback in the form of information (audio, tactile, verbal or visual) about what action the user has performed and what the effect of that action was. Content of the feedback should be understandable and in a format that is suitable for the targeted age group.
E3	Children should be able to apply their real-world or other computer-based knowledge when interacting with a new system.
E4	Familiarity can be achieved through metaphors and through affective use of affordances that exist for interface objects. The appearance of the object should promote familiarity with its behaviour or function.
E5	Surprise is often a desirable element in children's games and can increase the experience of fun and engagement. However, when it comes to learning how to use a system and navigating through the available functions and activities, predictability is very important. If they performed an action before,

	they will expect the system to behave similarly when they perform that action again.
E6	An interface should use language and concepts that the user is familiar with. Designers must follow real-world conventions so that information appear natural and in logical order. Familiarity has a different meaning for children than for adults – they 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.
E7	Interaction and input-output behaviour should be consistent within a system as well as across systems. The user should be able to extend knowledge of specific interaction within and across applications to other similar situations.
E8	Only operations that are available should be visible or it should be very clear which operations are not available. Available operations should be made known through age appropriate means.
E9	It should be very clear to the user what the next required action is. The more visible the available functions are the better users will be able to perform their next task.
E10	Allow equivalent values of input and output to be arbitrarily substituted for each other. Children have varying skill levels and preferences that will influence the type of input or output that is suitable for a specific user.
E11	The user interface should be modifiable by the user or the system. It should be adaptable (allowing user-initiated modification to adjust the form of input and output) and adaptive (allowing system-initiated modification to customise the user interface automatically).
E12	User pre-emptiveness is preferable to system pre-emptiveness in children's interfaces. In other words, the user should have freedom to initiate any action (although too much freedom may cause the user to loose track of incomplete tasks). System-pre-emptive dialogue is appropriate if children have to perform specific actions at specific times or places in the program.
E13	Allow multi-modality in terms of input and output mechanisms. Different modalities (channels of communication) can be combined to form a single, more effective input or output expression.
E14	Allow state information to be rendered in different formats or modes. Providing children with different visualisations of the same information can support their learning of the concepts or knowledge involved.
E15	Multi-modal communication between the user and the system can be particularly helpful in children's technology. For example, simultaneous audio and text cues can make a system accessible for reading as well as pre-reading children. It can also help to make a system accessible to children with disabilities.
E16	Response time must be quick. When it is not instantaneous the system should give clear indication that the task is in progress. Lack of instantaneous feedback evokes repeated clicking or hitting of keys, which may influence the program's execution.
E17	Response times for identical or similar tasks should be comparable.
E18	Use static and dynamic defaults to support interaction. Static defaults are defined within the system or acquired at initialization. Dynamic defaults evolve during the interactive session (for example, the system may pick up a certain user input preference and provide this as the default input

where applicable).

E19 Unless a specific sequence of actions is necessary, children should be allowed to go directly to their favourite parts of the system. Children like to play their favourite games over and over and will find it very frustrating if they cannot reach them easily.

E20 Audio communication persists only in the user's memory while visual communication remains available as long as the user can see the display. Designers should not rely on children's accurate recall of audio instructions (especially if given in the beginning of a session).

E21 If it is difficult to undo an unwanted effect on the system state, then it should have been difficult to get there the first place.

E22 Children cannot be expected to use browsing as a way to get a clear picture of the system's current state.

E23 If the application involves reaching a series of sub goals to achieve some central goal, it may be necessary to give them (or an adult assisting them) the option to view their current progress.

E24 Help users to recognise, diagnose and recover from errors. Give error message in language that users can understand. Describe the problem precisely and suggest a solution.

E25 Permit easy reversal of actions. Young children should not be expected to know how to use Undo/Redo commands – the system should help them recover from an error through help that fits their level of understanding.

E26 Any product should be effective – a product aimed at teaching a five-year-old to identify the first letters of words will be effective if children actually learn this skill successfully by using the system.

E27 For young children the focus is rarely on productivity, so that efficiency is not as important in young children's products as in systems designed for adults.

E28 Safety is an important factor in children's products. It involves how children are affected physically by using the system, how they can be affected by accessing material that is not appropriate such as pornography or images of violence, and how they are psychologically influenced by the content.

E29 To determine whether a product has adequate utility, designers should ask whether it allows children to carry out tasks in the way that they would like to do them. Children know what they want and they expect adults to know too. For the sake of utility, it is therefore important that designers do not make assumptions about children's preferences.

E30 Attention, pace, interactivity, engagement and style of narrative are factors that may influence the fulfilment of user experience goals such as fun, engagement and being emotionally fulfilling.

E31 There is a trade-off between user experience goals and usability goals. An action that requires more effort may contribute towards making a product more enjoyable and engaging. Designers should be aware of the consequences of combining user experience and usability goals and make sure that they address the needs of the user.

E32 Adults and children have different ideas about what is boring or exciting and designers should avoid using their own definitions of such concepts to guide their design decisions.

E33 Constraints that restrict the actions a user can take at a specific point during the interaction are

	effective error prevention mechanisms.
E34	There should be a clear mapping between interface elements and their effect on the system.
E35	Designers should create a profile of the intended user using information about their age, gender, physical abilities, level of education, cultural or ethnic background and personality.
E36	Find out whether users will be novices or if they will have experience with the particular kind of system (or if a mixture of novice and expert users is expected).
E37	Users with different levels of expertise will require a layered approach. Give novices options to choose from and protect them from making mistakes. As their confidence grows they can move to more advanced levels. Users who enter the system with knowledge of the tasks should be able to progress faster through the levels.
E38	Enable frequent users to use shortcuts and allow them to skip introductions and instructions that they already know. Make sure children are made aware of the possibility to skip these or create shortcuts.
E39	Reduce short-term memory load. Do this by making objects, actions and options visible. Users should not need to remember instructions or previous choices.
E40	Design to prevent errors rather than to help users recover from errors. Require users to confirm potentially erroneous actions before performing them.
E41	Young children should be able to use the system without documentation.

7.3 Existing Guidelines Aimed at Technology for Young Children

In this section I discuss the work of researchers whose aim was to formulate guidelines for the design of technology for young children.

7.3.1 Malone

Malone [1982] provide guidelines for designing educational computer programs. He organised them into three categories, namely Challenge, Fantasy and Curiosity. The guidelines that fall under Challenge include having a clear and personally meaningful goal, a variable level of difficulty, randomness and selectively revealed hidden information. According to Malone, a goal with an uncertain outcome is what makes a game challenging.

The Fantasy guidelines call for emotionally appealing fantasy, fantasy that is intrinsically related to the skill associated with the activity and fantasy that provides a useful metaphor. Malone [1982] gives ‘scoring in baseball’ as an example of a fantasy goal towards which a player can progress. A player can also try to avoid some fantasy catastrophe as in the Hangman game. To motivate players to answer questions as fast as possible their progress can be reflected as race cars moving along a racing track. The three fantasies mentioned are all extrinsic fantasies in the sense they can be used effectively for different kinds of problems, such as spelling or arithmetic. With intrinsic fantasies on the other hand, the skill relies on the fantasy.

Malone [1982] gives as example the *Adventure* game in which a fantasy underground cavern system is explored through the skills of reading cave descriptions and typing commands. He claims that intrinsic fantasies are more interesting and more instructional because the fantasy context can tell the learner how to apply the skills learnt in a real world context. They can also provide analogies that may help the child apply existing knowledge to new situations. Vivid fantasy images related to the learning material can help learners remember what they have learnt. The same learning content and problems can be presented in different fantasy environments. Since children's preferences for fantasies differ, Malone suggests that designers try to give them a choice of fantasy in which to embed a specific learning task. They can also let children participate in the creation of the fantasy by, for example, letting them choose names for the characters or places.

With regard to Curiosity, Malone [1982] says that designers should provide children with an optimal level of information complexity – the presentation should be neither too complicated nor too simple with respect to the child's existing knowledge. He suggests that software should include novel elements, surprises and interesting audio and visual effects. Music, animation and other audio and visual effects can be used as decoration, to enhance fantasy, as reward and for representation. A sequence of increasingly complex tasks can also sustain curiosity by introducing a surprising complication at each level.

Unfortunately Malone [1982] did not distinguish between different age groups and it is therefore difficult to know which apply to five to eight-year-old children.

7.3.2 Grammenos and Stephanidis

In the context of designing the user interface of a collaborative application for children between four and eight years old, Grammenos and Stephanidis [2002] identified the following design guidelines:

- Convey available functionality through highly visual interface components and not through textual representations, so that children who cannot yet read can use it.
- Create a system that is needs driven, learner-initiated and conceptually and intellectually engaging. It must be easily adaptable to children's preferences, cultural background and skills.
- Use intuitive metaphors.
- Make all the user interface components active or interactive. Provide feedback through audio effects and animation to indicate successful interaction. Feedback should facilitate comprehension of the concepts as well as promote exploratory interaction.
- The environment must be forgiving and should provide guidance when needed.
- Make the system gender-neutral or gender-adaptable.
- Avoid cumbersome input devices and interaction techniques.
- Create a transparent interface that enables children to focus on what must be done and not on how they should use the interface.

Although these guidelines seem sensible, they are all very general. There is, for example, no clear indication what an intuitive metaphor would be or which input devices children will find cumbersome.

7.3.3 Baumgarten and Fishel

I discuss Fishel [2001] and Baumgarten [2003] under one heading as they both discuss development-related guidelines and there is some overlap between their work. Baumgarten [2003] gives a summary of the developmental stages of children from age two to fourteen, relating the developmental characteristics to children’s use of electronic media. Fishel [2001] discusses design for children in general, also summarising relevant developmental characteristics for specific age groups. I discuss their findings that apply to children aged five to eight.

7.3.3.1 Physical Development

At five years of age a child’s brain weighs more or less 90% of its adult weight. Myelinization (the development of a fatty sheath needed for improved transmission of nerve impulses) of nerve fibres in the spinal chord is complete around the age of two, while brain neurons are only completely myelinated by the end of adolescence [Baumgarten, 2003]. Before six, most children are farsighted to some extent. By the age of six increased myelinization has improved visual motor functioning and eye movement control to the point where most children have good focussing and scanning skills. Organization of the brain and its functions into a left and right hemisphere is complete by age six, and with this the child’s hand preference is defined. From age six onwards the growth rate becomes slower than in the preschool years [Baumgarten, 2003].

According to Fishel [2001] the small motor movements of four-year-olds are sufficiently refined for them to use toys that are less chunky and hold pencils or crayons. From four their play can require some body control – they can run, jump, hop and skip and dance to music. Their balance also improves. Increased hand-eye coordination allows them to play board games, build jigsaw puzzles and string beads. At six or seven, large motor skills are well developed and children can balance well. They like competitive physical play. Small motor skills also improve and they get better at working with crayons, scissors, building with small interlocking blocks and knitting or weaving. Fishel [2001] suggests that children of six or seven like to be challenged, by, for example, timing them to see if they have improved at an activity.

7.3.3.2 Cognitive Development

During the preschool years, children develop language, demonstrating the ability to use symbols [Baumgarten, 2003]. They acquire a huge vocabulary and learn the rules of grammar. Their pretend play is further demonstration of their improved use of symbols. They move from mimicking the real use of objects (drinking tea from a cup) to using those objects to represent other objects (using a cup as a fairy bathtub). Preschoolers also display more control over attention and memory.

From age six upwards children start to reason logically, can see a situation from another's perspective (decenter) and grasp the concept of conservation. Language skills are strengthened by the ability to read and children develop the ability to adjust their speech to different listeners [Baumgarten, 2003].

In Fishel's [2001] view the ability of four and five-year-olds to think logically is sufficiently developed to play simple strategy and memory games. They can do simple patterning and understand clear and simple rules. They know the eight basic colours and can mix colours. Fishel [2001] believes that children are often ahead of their physical selves and should be presented with toys that allow them to practice grown-up actions (for example, giving them a play car console that allows them to 'drive'). At four and five children become interested in writing and words. Their self-esteem is developing, so they need games and toys that reinforce their ability to succeed on their own and there should be some system of reward [Fishel, 2001].

According to Fishel [2001], six and seven-year-old children prefer bright contrasting colours that create patterns. Since they can read more text they can be presented with more complex story plots and characters and designs can be based on a theme. They prefer predictable activities or activities that they can direct. Since at six and seven, children like to talk, they should be allowed to express themselves orally. They like singing and recording their own voices [Fishel, 2001].

7.3.3.3 Psycho-Social Characteristics

During the preschool years children learn about their social world through play, they develop friendships and relationships and learn social rules [Baumgarten, 2003]. They also develop a strong sense of identity, realising that there are different genders and characteristics that belong to a specific gender. They begin to behave more pro-socially, modelling the pro-social behaviour of parents, caregivers and characters they encounter on television and other media [Baumgarten, 2003]. From age six onwards, when children start their schooling, they have to learn new codes of conduct, new rules of friendship and have to re-establish a sense of self-worth outside their homes. They now have the reasoning skills to test their self-concept against reality and their self-esteem affects their social interactions [Baumgarten, 2003]. Children with a supportive family and good experiences are generally more resilient to stressful situations.

Fishel [2001] believes that four and five-year-old children already display gender-related preferences in their play. They can imitate adults in their world through pretend-play, but will generally not play non-traditional roles. They can separate themselves mentally from their physical surroundings and can therefore engage in absurd fantasy play or accept strange characters and events in stories. By six or seven, children develop deeper relationships with people outside their homes and designs can model such relationships and allow children to role play [Fishel, 2001]. They are now able to cooperate with other children and wait for their turns. They can play group games without the help of adults. At this age children can plan ahead and they can solve 'how much' and 'how often' problems [Fishel, 2001]. They have also developed better prosocial behaviour and understand more sophisticated emotions. They can argue about fairness.

7.3.3.4 Likes, Dislikes and Fears

Preschoolers want to show their independence and do this by, for example, demanding to choose their own clothes and books [Baumgarten, 2003]. This is part of finding out who they are. This is an age of discovery and frustration, where they try new things and often experience failure because of their limitations. Preschool children’s fears include being alone, getting lost, being in the dark and losing a parent and they are fascinated by stories of children (or other young creatures) who overcome these fears [Baumgarten, 2003].

Preschoolers’ sense of humour is still quite unsophisticated and what they find funny, older children will regard as silly. They enjoy putting on their clothes in the wrong way or mispronouncing words and expect people to find such actions funny even when they have done it repeatedly [Baumgarten, 2003].

Beyond the preschool years children still enjoy activities from which they can learn and test their mastery, especially if it involves social interaction with their peers [Baumgarten, 2003]. Although children start to strive towards independence from the age of six, they still desire nurturing. They develop a greater social awareness that leads to fear of exclusion by their peers. Six and seven-year-old children still prefer fun, humour, simplicity and familiarity. They like jokes and riddles and characters that behave as expected. Only after the age of seven do children prefer challenge and competition, and do they become very aware of their gender [Baumgarten, 2003].

7.3.3.5 Children and the Internet

From the characteristics described above, Baumgarten [2003] comes to certain conclusions regarding children’s use of the internet. Preschoolers are attracted to activities that offer opportunities for learning, mastery and silly fun. They want to see their favourite characters and will enjoy seeing them on the internet. They are drawn to programs that will help them learn useful things like the letters of the alphabet. They want to demonstrate their abilities to their parents or caregivers.

Problems that preschoolers may have with internet use are: they do not tolerate technical difficulties, they have a small attention span and their lack fine motor skills needed to manipulate the mouse [Baumgarten, 2003]. Material should therefore be simple to access and visuals should be clear and bright. Activities must allow them to achieve the goal within a short time. Feedback should be immediate and understandable at their level. Preschool children should not be required to read and directions should be verbal.

By six years of age children have sufficient fine motor skills to manipulate input devices without problems. They have now started reading and can follow simple written directions or give written feedback. They are now ready for memory-based games and their ability to reason logically may allow them to perform strategy-based activities and contests. They start to prefer novelty and challenge to familiarity [Baumgarten, 2003].

Baumgarten [2003] summarises kids’ likes with regard to internet content as follows. They like things that

- are new and different

- they can learn from
- they can relate to
- they can use to confront real fears and resolve them
- are well done technically
- are easy to use
- are forbidden and/or secret
- are fun.

The above list applies to children from two to fourteen.

7.3.4 Gilutz and Nielsen

Gilutz and Nielsen [2002] conducted usability studies with fifty five children aged six to twelve on twenty four web sites designed for children and three sites designed for adults. They found that children generally did better on the adult sites that were based on the design principles of utter simplicity and ease of use. The children's sites tended to be cluttered with more complicated interaction requirements that, combined with children's lack of patience, caused children to leave websites often. Some of the problems they identified are:

- Unclear navigational confirmation so that users were often confused about their locations.
- Inconsistent navigation options whereby the same destination could be reached in different ways, causing children to unintentionally revisit sites they have already been to. Like adults, many children have invested time in learning to use standard navigational tools. Designers should take advantage of this instead of confusing children with very creative interfaces that diverge from the standard interfaces.
- Lack of perceived clickability affordances that made users miss links that they would otherwise have followed.
- Fancy wording that made it difficult to understand the different choices.
- Text sections that were above the reading level of the intended audience.

Gilutz and Nielsen [2002] found that children's preferences sometimes contradict design and usability principles that are appropriate for adult websites. These include:

- Children like animation and sound effects. They want content that is entertaining, funny, and colourful, and uses multimedia effects.
- Children are willing to 'mine-sweep' (that is, click on whatever seems clickable on the screen).
- Children like geographic navigation metaphors such as pictures of rooms, villages and 3D maps.
- They rarely scroll down and mainly interacted with whatever was visible on the screen.
- Children who can read are willing to read instructions before playing a game.
- Children want to be in control of the interaction and do not like it when they cannot stop an introduction or unwanted animation, or when the navigation tools are hidden.

The differences between the requirements of children and adults can partly be explained by the fact that adults typically use the Web for business and goal-oriented tasks, whereas children often use it for entertainment. Children have also been found to be even more impatient with performance problems than adults and will very easily leave a site that crashes, displays and error message or is very slow [Gilutz and Nielsen, 2002].

An important finding by Gilutz and Nielsen is that children cannot distinguish between content and advertising and regard advertisements as relevant site elements. They will especially click on advertisements where the banner contains popular characters or a seemingly 'cool' game.

7.3.5 Wyeth and Purchase

The research of Wyeth and Purchase [2003] led to a set of six design criteria that are based on theories of development and learning. These are:

1. Activities should be open-ended and discovery oriented. Children should be actively involved in the learning process.
2. Interaction should encourage child-initiated play.
3. Experiences should involve active manipulation and transformation of real materials.
4. Entry-level knowledge and experience should be minimal.
5. Provision should be made for children's varied skill and ability levels.
6. Construction activities that involve design, creation and evaluation processes should form the basis of interactions.

These criteria can be useful during the initial planning stages of the design process, but they do not provide practical guidelines on the actual implementation of computer-based activities. How, for example, can interaction encourage child-initiated play? It gives no indication of what kind of construction activities would be appropriate. It also does not tell us what kind of design or creation processes will engage a preschool child or whether a typical six-year-old boy will like the same construction activity as a six-year-old girl.

The work of Wyeth and Purchase [2003] is based on the design of a product called Electronic Blocks that are LEGO®, Duplo™ or Primo™ blocks with electronics placed inside. The design of the Electronic Blocks incorporated the six design guidelines given above. The blocks have input and outputs through which they interact when connected. There are sensor blocks (seeing, hearing or touching blocks that detect light, sound and touch respectively), logic blocks ('not', 'delay', 'toggle' and 'and' blocks that can alter the signal passed between two blocks) and action blocks (making sounds, lighting up or moving in a straight line). Visual cues on the blocks inform the children of their functions. They designed the system so that no construction could produce incorrect results, only unexpected outcomes, so that the children could use the blocks without much intervention.

Experiments with preschool children using the Electronic Blocks showed that children found the blocks easy to use, especially for constructing stacks that included sensor actions. When a task required them to use logic blocks they tended to use trial-and-error to get the desired result. Blocks with input-output capabilities were most difficult to use. Children could not understand the relationship between invisible signals passed between blocks and the behaviours of the logic elements. The interaction need to be made visible for them to grasp it.

At primary school level the children fared better in using the logic blocks. Although they also found the tasks that involved logic more difficult than the simple sensor-action constructions they displayed improved understanding of the purpose of the ‘not’ and ‘and’ blocks. All the children displayed high levels of interest and excitement when their constructions performed any kind of behaviour. Older children remained interested even when they experienced difficulties. The Electronic Blocks provide a variety of opportunities for interaction, catering for different play styles, skill levels and interests.

The results of the experiments that Wyeth and Purchase [2003] conducted to test the success of applying their proposed guidelines in practice, show these guidelines are not sufficient. So, although useful at a high level of design more detailed guidance will be helpful.

Some authors writing about children and computers provide guidelines for the evaluation rather than the design of children’s software. I next discuss three such cases, rephrasing their evaluation guidelines as guidelines for the design of children’s software.

7.3.6 Druin, Bederson, Boltman, Miura, Knotts-Callahan and Platt

Druin, Bederson, Boltman, Miura, Knotts-Callahan and Platt [1999] investigated three research methodologies (contextual enquiry, technology immersion and participatory design) for collecting data on children’s use of technology. Comparison of the data gathered through the use of these three methods led them to useful conclusions about what children want in technology and to an understanding of what children notice when using technology. I formulate their results in the form of guidelines for designers:

- Children want to have control over what they use technology for, when they use and how they use it. Technology that offer limited paths of interaction do not keep a child’s interest.
- Technology should offer opportunities for social interaction. Children want to use technology in interaction with other children. Children who know one another want to share and show technology with each other and when children who do not know one another work together the technology becomes a shared interest that will help them become friends [Druin et al., 1999]. Cultural differences do not prevent children from sharing experiences with technology.
- Technology should give children the chance to express themselves in different ways. They want to tell stories, make up games and build things.

- Children want to have technology that is ‘cool’ in the sense that it is state of the art. Using headphones is ‘cooler’ than listening through speakers. They do not want their technology to compare badly with that of their peers.
- Children want technology that is easy to learn so that they can start using it quickly.
- The appearance of technology is as important to children as what it does. Technology should not appear as if it ‘talks down’ to the user.
- Children’s exposure to video games, television, movies and other media increases their expectations with regard to multimedia. They want a multi-sensory experience.

7.3.7 Shade

Shade [1996] divide his evaluation criteria in three groups, namely child features, teacher features and technical features.

1. Child features:
   - Avoid drill-and-practice type activities and create open-ended, discovery-oriented products by incorporating opportunities for active learning, child controlled interaction and experimentation. Let the children decide what they want to do, set the pace and stop at any time.
   - Software that will be used in school settings or where children will not have constant assistance available must be designed for independent use.
   - Children should be ‘the agents of change’ rather than the programmer. They should have the power to move graphics elements on the screen and observe the outcome of their actions, in other words, experience cause and effect processes actively.
   - The complexity level of software should be low at entry and should have a high ceiling to allow children of different ages and levels of cognitive development to benefit.
   - The activities should progress through a logical learning sequence.
   - Software should teach powerful ideas and include verbal instructions and help where appropriate.

2. Teacher features:
   - Software content should be representative of the diversity of the intended user population. Culture, ethnicity, gender, age, skill level, computer experience, family background and disability are all variables that need to be taken into account.
   - Software should include elements that are familiar to all children.
   - Teachers need some control over the software to customise it according to their or the children’s needs. For example, when young children are first introduced to a new program they may be overwhelmed by too many available options or tools. In such a case teachers should be able to restrict the number of options available and adapt it according to the children’s improving dexterity.

3. Technical features:
   This category did not yield any meaningful design guidelines other than obvious things such as easy installation, realistic sound effects and clear and distinct speech.

7.3.8 Haugland and Shade

Haugland and Shade [1988] formulated ten criteria for judging the developmental appropriateness of software for young children. These guidelines were published almost twenty years ago and most of them were incorporated in Shade's [1996] evaluation guidelines that I discussed above. I now summarise only those that have not been mentioned before or are described in a way that can add to the discussion:

- Software should display expanding complexity – complexity at entry level must be low enough so that children can easily learn to manipulate the software, the learning sequence must be clear and expanding complexity should support the children in building knowledge structures.
- The process of using the software should be engaging and should rely on intrinsic motivation rather than praise, smiling faces, or prizes.
- Software should be a 'process highlighter' that allows children to view processes and cause and effect relationships that are difficult to observe in reality.

Haugland and Shade [1988] also included 'trial and error' as a criterion, requiring software to allow children to try out alternative actions. More recent research has shown that promoting trial-and-error type behaviour to solve problems may be detrimental to children's development of good problem-solving strategies [Klein et al., 2000].

7.3.9 Buckleitner

In his discussion of children's software evaluation, Buckleitner [1999] 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 that designers 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.

In addition to the requirement that software must be age appropriate from a cognitive development point of view, Buckleitner [1999] points out that graphic themes, style of music and choice of characters should fit the preferences of the intended age group. Here it is important that designers consult children, because adults' judgement of children's preferences can be inaccurate. Software should take advantage of features of state of the art technology such as text-to-speech conversion, voice recognition, high quality sound and full-motion video.

Buckleitner [1999] believes that the underlying educational theory should be clearly identifiable, and should support the intended purpose of the software. Behaviorist reinforcement strategies may work well in some contexts while a constructivist approach may be more suitable for others.

7.3.10 Siraj-Blatchford and Siraj-Blatchford

In 1999 IBM donated computer equipment and educational software to fourteen nurseries in the UK and Sctoland as part of their so-called Kidsmart programme. Siraj-Blatchford and Siraj-Blatchford [2001] conducted a study to determine whether the IBM initiative had improved the provision of information and communications technology (ICT) to children aged three and four. As a result they formulated the eight general principles for assessing the suitability of ICT applications. I give these principles here as they are not equally applicable to children aged five to eight.

1. Applications should be educational.
   For an application to be educational its learning aims should be clearly identifiable. Drill and practice products should be used with caution because if children rely too much on this kind of learning style it will reduce their intrinsic motivation to learn.

2. Encourage collaboration.
   It is important that not all children's interaction with technology involve working alone. Collaboration can provide opportunities for children to learn how to resolve conflicts and for working together to reach a solution to a problem.

3. Integrate the use of ICT with other practices such as play and project work.
   It is better to bring computers into the normal classroom situation than to have a separate computer room. Children must learn to use computers for real purposes and to complete part of a larger project (e.g. they can use a draw program to make a birthday card and then use other material to decorate it further). Computer applications can be used to introduce children to virtual technologies that simulate real environments to which they would normally not have access to. Young children can even learn to use computers to gather and manage information.

4. The child should be in control.
   Sometimes it is necessary for an application to control the interaction, especially when helping the child to learn skills such as knowing the alphabet, early number skills and reading. There is, however, agreement that programmed learning does not support a positive attitude towards literacy or numeracy and that open-ended applications that allow a child to reach a solution in different ways are preferable.

5. Applications should be transparent and intuitive.

6. Applications should not contain violence or stereotyping.

7. Applications should display awareness of health and safety issues.
   Siraj-Blatchford and Siraj-Blatchford [2001] believe that a three-year-old should use a computer for longer than twenty minutes at a time and an eight-year-old not for longer than forty minutes. This will help to avoid problems such as repetitive strain injury and carpal tunnel damage that have been associated with extended computer use.

8. Parents should be involved.
   This guideline relates more to the educational aspects of communication around computers than the design of specific applications.

From Malone [1982]:

E42 Designers can place activities in a fantasy environment that can be related to the skill involved and provide a useful metaphor. With intrinsic fantasy the skill relies on the fantasy (e.g. navigate through an underground cavern by reading cave descriptions). An extrinsic fantasy is not linked to the skill and can be used for different kinds of problems, such as spelling or arithmetic (e.g. progress reflected as race cars moving along a racing track).

E43 Intrinsic fantasies are more interesting and more instructional because the fantasy context can tell the learner how to apply the skills learnt in a real world context.

E44 Vivid fantasy images related to the learning material can help learners remember what they have learnt.

E45 Children's preferences for fantasies differ, so designers should try to give them a choice of fantasy in which to embed a specific learning task. They can also let children participate in the creation of the fantasy by, for example, letting them choose names for the characters or places.

E46 Provide children with an optimal level of information complexity – the presentation should be neither too complicated nor too simple with respect to the child's existing knowledge.

E47 A sequence of increasingly complex tasks can sustain curiosity by introducing a surprising complication at each level.

From Grammenos and Stephanidis [2002]

E48 Convey available functionality through highly visual interface components and not through textual representations, so that children who cannot yet read can use the system.

E49 Make all the user interface components active or interactive.

E50 Provide feedback through audio effects and animation to indicate successful interaction. Feedback should facilitate comprehension of the concepts as well as promote exploratory interaction.

E51 Make the system gender-neutral or gender-adaptable.

E52 Create a transparent interface that enables children to focus on what must be done and not on how they should use the interface.

From Baumgarten [2003] and Fishel [2001]:

E53 By the age of six, visual motor functioning and eye movement control have reached the point

	where most children have good focussing and scanning skills.
E54	From age four children's activities can require some body control – they can run, jump, hop and skip and dance to music. Increased hand-eye coordination allows them to play board games, build jigsaw puzzles and string beads.
E55	By six or seven, large motor skills are well developed, children can balance well and they like competitive physical play.
E56	Six and seven-year-olds like to be challenged by, for example, timing them to see if they have improved at an activity.
E57	During preschool, children move from mimicking the real use of objects (drinking tea from a cup) to using those objects to represent other objects (using a cup as a fairy bathtub).
E58	From five or six, children start to reason logically, can see a situation from another's perspective (decenter) and grasp the concept of conservation.
E59	From age six, language skills are strengthened by the ability to read and children develop the ability to adjust their speech to different listeners.
E60	The ability of five-year-olds to think logically is sufficiently developed to play simple strategy and memory games; they can do simple patterning and understand clear and simple rules.
E61	Most five-year-olds know the eight basic colours and can mix colours.
E62	Young children are often mentally ahead of their physical selves and should be presented with products that allow them to practice grown-up actions.
E63	At five, children are interested in writing and words.
E64	To develop their self-esteem children need games and toys that reinforce their ability to succeed on their own. There should be some system of reward.
E65	Six and seven-year-old children prefer bright contrasting colours that create patterns.
E66	From six they can be presented with more complex story plots and characters and designs can be based on a theme.
E67	Six and seven-year-olds prefer predictable activities or activities that they can direct.
E68	At six and seven children like to talk and they should be allowed to express themselves orally. They like singing and recording their own voices.
E69	During preschool years children develop a strong sense of identity, realising that there are different genders and characteristics that belong to a specific gender.
E70	They begin to behave more pro-socially, modelling the pro-social behaviour of parents, caregivers and characters they encounter on television and other media.
E71	From age six onwards children have the reasoning skills to test their self-concept against reality and their self-esteem affects their social interactions. Children with a supportive family and good experiences are generally more resilient to stressful situations.
E72	From four or five, children can separate themselves mentally from their physical surroundings and can engage in absurd fantasy play or accept strange characters and events in stories.
E73	By six or seven, children develop deeper relationships with people outside their homes and designs can model such relationships and allow children to role play.

E74 They are now able to cooperate with other children and wait for their turns. They can play group games without the help of adults and can argue about fairness.

E75 Preschool children's fears include being alone, getting lost, being in the dark and losing a parent and they like stories of children (or other young creatures) who overcome these fears.

E76 Preschoolers' sense of humour is still quite unsophisticated. For example, they enjoy putting on their clothes in the wrong way or mispronouncing words and expect people to find such actions funny when they have done it repeatedly.

E77 Although children start to strive towards independence from the age of six, they develop a greater social awareness that leads to fear of exclusion by their peers.

E78 Six and seven-year-old children still prefer fun, humour, simplicity and familiarity. They like jokes and riddles and characters that behave as expected.

E79 From the age of eight children prefer challenge and competition, and they become very aware of their gender.

From Gilutz and Nielsen [2002]:

E80 With regard to web applications:

- Preschoolers are attracted to activities that offer opportunities for learning, mastery and silly fun.
- They want to see their favourite characters and will enjoy seeing them on the internet.
- They are drawn to programs that will help them learn useful things like the letters of the alphabet.
- They want to demonstrate their abilities to their parents or caregivers.

E81 Web-based material aimed at preschoolers should be simple to access, visuals should be clear and bright, and activities must allow them to achieve the goal within a short time.

E82 By six years of age children have sufficient fine motor skills to manipulate input devices without problems. They have now started reading and can follow simple written directions or give written feedback. They are now ready for memory-based games and their ability to reason logically may allow them to perform strategy-based activities and contests. They start to prefer novelty and challenge to familiarity.

E83 Reasons children leave web sites are:

- Unclear navigational confirmation.
- Inconsistent navigation options whereby the same destination could be reached in different ways, causing children to unintentionally revisit sites they have already been to.
- Lack of perceived clickability affordances that made users miss links that they would otherwise have followed.
- Fancy wording that made it difficult to understand the different choices.
- Text sections that are above the reading level of the intended audience.
- When a site crashes, displays and error message or is very slow.

E84 Children's preferences sometimes contradict design and usability principles that are appropriate for adult websites. For example:

- They like animation and sound effects. They want content that is entertaining, funny, and colourful, and uses multimedia effects.
- Children are willing to 'mine-sweep'.
- Children like geographic navigation metaphors such as pictures of rooms, villages and 3D maps.
- They rarely scroll down and mainly interacted with whatever was visible on the screen.
- Children who can read are willing to read instructions before playing a game.

E85 Children cannot distinguish between content and advertising and regard advertisements as relevant site elements. They will especially click on advertisements where the banner contains popular characters or a seemingly 'cool' game.

From Wyeth and Purchase [2003]:

E86 Construction activities that involve design, creation and evaluation processes should form the basis of interactions.

E87 Design construction systems so that children's constructions cannot produce incorrect results – only unexpected outcomes (thus reducing the need for adult intervention).

E88 Preschool children find it difficult to use logic operations such as 'and', 'or' and 'not' in construction tasks where the construction represents a logic operation.

E89 Primary school children are better at building logic constructions and have a better understanding of the purpose of 'not' and 'and' components.

E90 Children display high levels of interest and excitement when their constructions perform any kind of behaviour.

From Druin et al [1999]:

E91 Technology that offers limited paths of interaction do not keep a child's interest.

E92 Technology should offer opportunities for social interaction.

E93 Cultural differences do not prevent children from sharing experiences with technology as the technology becomes a shared interest.

E94 Technology should give children the chance to express themselves in different ways. They want to tell stories, make up games and build things.

E95 Children want state of the art technology. Headphones are 'cooler' than speakers.

E96 What technology looks like is as important to children as what it does. Technology should not appear as if it 'talks down' to the user.

E97 Children's exposure to video games, television, movies and other media increase their expectations with regard to multimedia. They want a multi-sensory experience.

From Shade [1996]:

E98 Let the children decide what they want to do, how fast they want to do it and when they want to end.

E99 Children should have the power to move graphics elements on the screen and observe the outcome of their actions. Let them experience cause and effect processes actively.

E100 The complexity level of software should be low at entry and should have a high ceiling to allow children of different ages and levels of cognitive development to benefit.

E101 Software content should be representative of the diversity of the intended user population in terms of culture, ethnicity, gender, age, skill level, computer experience, family background and disability.

E102 Products aimed at classroom use should give teachers some control over the software to customize it according to their or the children's needs (e.g. restricting the number of available options according to children's improving dexterity).

From Haugland and Shade [1988]:

E103 The process of using the software should be engaging and should rely on intrinsic motivation rather than praise, smiling faces, or prizes.

E104 Software should be a 'process highlighter' that allows children to view processes and cause and effect relationships that are difficult to observe in reality.

From Buckleitner [1999]:

E105 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 that designers have a clear idea of the context in which the software will be used.

E106 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.

E107 The underlying educational theory should be clearly identifiable, and should support the intended purpose of the software. Behaviourist reinforcement strategies may work well in some contexts while a constructivist approach may be more suitable for others.

Siraj-Blatchford and Siraj-Blatchford [2001]

E108 An application can only be educational if its learning aims are clearly identifiable.

E109 Drill and practice products promote a learning style that may reduce their intrinsic motivation to learn.

E110 Technology should be designed in a way that allows integrating its use with other practices such as play and project work.

E111 Children must learn to use computers for real purposes.

E112 Computer applications can be used to introduce children to virtual technologies that simulate real environments to which they would normally not have access to. Young children can even learn to use computers to gather and manage information.

E113	Applications should not contain violence or stereotyping.
E114	Applications should display awareness of health and safety issues.
E115	An eight-year-old should use a computer for longer than forty minutes.

7.4 Additional Design Guidelines Emerging from the Literature

This section represents the final step of the guideline gathering process. It can be seen as a tidying up process whereby I review a collection of guideline-producing studies that have not been incorporated into earlier chapters. I group the emerging guidelines according to the following themes: developmental issues, cultural issues, gender issues, learning and problem solving, icon design, humour and praise.

7.4.1 Developmental Issues

Grover [1986] discusses the effect of software designed in accordance with certain cognitive-developmental principles versus software that do not incorporate these principles. Based on research with children aged two and half to eight years, she found that:

- Software should provide cues for coordination between symbols on screen and the keyboard to address the problem young children have with relating two-dimensional representations to their three-dimensional referents. For example, in an experiment conducted by MacFarlane et al. [2005] children did not know what to when they typed their names and were requested to press Enter, as the laptop computer did not have a key marked 'Enter'.
- Material should be presented in a familiar or meaningful context using suitable graphics, to improve the comprehensibility of instructional content and reduce the information processing. Appropriate context cues activate prior knowledge that children can apply to make sense of what they perceive and their interpretation thereof.
- There should be a match between the content of the material and children's cognitive competencies. This match can be achieved through additional cues for difficult concepts. For example, in conservation of number tasks the spatial arrangement of objects to be counted can influence children's performance.
- Software should be personalised and interactive to make it friendlier. If feedback on correct or incorrect actions mentions the child's name, he or she may pay more attention to it.
- Graphics should be simple enough so that children are not distracted from the relevant components of the interface. Young children do not always grasp complex interrelationships between components of an interface.
- Neutral feedback after errors (such as a triangle instead of a sad face) will increase the incentive values of reinforcers. Children lose interest in activities if they perceive themselves to be motivated (or rather manipulated) by external rewards rather than engagement. The informational aspect of reinforcers is at least as important as the reward value.

The results of Grover's [1986] research showed that software that incorporate cognitive-developmental principles enhanced learning significantly more than software that do not. She also found that when speech was added to the interface, children's performance deteriorated. Reasons for this are that the children were often too impatient to wait for the voice instructions, found the slow-paced voice frustrating and could sometimes not follow the staccato voice of the voice synthesiser. The software was developed before 1983 and I am therefore certain that Grover's results are not relevant anymore. Sound technology and processing speed of computers have improved to such an extent that children's responses to outdated forms of technology have no bearing on current technology. Although her results with regard to the benefits of cognitive-developmental principles seem believable, I suspect that the basic appearance of the two software products had much to do with the results. From Grover's descriptions, I infer that the developmentally appropriate software was visually and content-wise much more interesting than the other test software and that its success can be partially attributed aesthetics and visual engagement rather than cognitive-developmental aspects.

Brouwer-Janse et al. [1997] believe that software for children should exploit children's natural intellectual curiosity and their eagerness for acquiring new knowledge. Children do not simply take in what they observe as it is presented – they reconstruct it according to their mental models [Brouwer-Janse et al., 1997]. Software designers should therefore support the development of children's reasoning by enabling them to build on their existing worlds. They must be encouraged to be creative and inventive and this can be done partly through the things we provide them with and partly through their own discovery.

Using Piaget's theory of cognitive development as point of departure, Schneider [1996] discusses the design of information visualisation applications for children. Children in the pre-operational stage (ages two to seven) prefer cursors that are automatically drawn to items through discrete movements to cursors that need sliding onto an item [Schneider, 1996]. Referring to research by Strommen, Schneider [1996] explains that preoperational children should be allowed to interact through direct manipulation using a mouse, rather than indirectly through keyboard commands, as such commands require additional information to be held in working memory. Around four years of age children are highly imaginative and enjoy things that are obviously incongruent [Schneider, 1996]. By seven they begin to understand organisational schemes such as the system for organising books in a library. Because preoperational children do not yet understand conservation of liquid, for example, an interface should not rely on changing size of objects to convey new knowledge. In general, perceptual literacy depends on experience and domain knowledge that preoperational children lack [Schneider, 1996].

In the concrete-operational stage (seven to eleven years) children are able to organise items into groups and categories [Schneider, 1996]. They enjoy browsing activities and due to the quick development of working memory, they can manipulate more than one item in memory at the same time [Schneider, 1996]. Motor skills now allow the use of different types of input devices and perceptual acuity reaches a level so that children can understand the way two or more software tools can be used together to accomplish a task (for example, creating an image with one application and dropping it into another). According to Schneider, children can also learn to use a metaphor that has no relation with their own experience, as long as it is internally consistent. She speculates that concrete-operational children's literalness and rule-driven nature allow them to learn arbitrary rules and behaviours, and that this makes them adaptable to different kinds of computer interfaces.

Schneider's [1996] discussion of various information visualisation software yielded the following guidelines for software designers: 3D interfaces tend to be abstract, busy, information-dense and they rely on metaphors (such as office space and equipment) that children are not familiar with. For children, 3D interfaces should be stripped down, with a limited number of familiar objects that resembles a familiar setting such as a child's bedroom or playroom. It should be an imitation of their lives with games in a toy chest, homework on a desk and a dust bin for discarding unwanted objects. 3D workspaces could also be modelled on well-known children's books, so that when there is a lot of detail it will still be familiar to the child user. Interface tools that reflect how often they have been used or accessed may help children to see the relationship between actions and outcomes. This can be achieved through the use of colour, size, shape or pop-up messages.

7.4.2 Cultural Issues

The US National Association for the Education of Young Children [NAEYC, 1996] suggests that software should reflect a child's context and should therefore come in multiple languages, reflect gender equity, avoid racial discrimination and portray diverse families and experiences. It should also promote positive social values and no violence. For example, when a child has created something on a computer, it might seem like good fun to let the child 'blow up' the creation when they have finished or is not satisfied with it. The NAEYC [1996] states that since the child is in control, such actions may have the general effect of suppressing the child's feeling of responsibility for violent outcomes. Many computer games encourage pro-social behaviour, but the most popular software involves competition and aggression [Subrahmanyam et al., 2000]. Fortunately this does not apply to most applications aimed at younger children.

A study by Sundholm and Dahlbäck [2002] involving the evaluation of digital cuddly toy-like museum guides for young children yielded useful results. One of their main conclusions was that when designing a guide character, the children's culture should be kept in mind. For example, if their cultural experience has taught them that an owl is a knowledgeable creature, a designer should not present an owl that just pretends to be clever. Similarly, a shy and worried bat will not fit their expectation that will probably be based on Batman and ghost stories.

Chimbo and Gelderblom [2008] 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.

7.4.3 Gender

Boys and girls aged four and five spend an equal amount of time at computers [Haugland, 2000] and by ten years of age boys spend significantly more time at computers than girls. Designers should acknowledge differences between boys and girls in their choice of activity or setting. For example, girls prefer pretend play based on reality while boys prefer pretend play based on fantasy [Subrahmanyam et al., 2000]. A review by Clements [1987] show that preschool boys prefer creative problem-solving programs, while girls rather use drill and practice programs, but this is refuted by other research that have found no difference in the preferences of girls and boys at this age. Plowman and Stephen [2003] report on more recent studies that show little gender differences at preschool level, although some suggest that girls lean more toward education and strategy games and boys toward combat and sport games. Egloff [2004] looked at gender differences in the use of interactive computer-based games with a tangible interface. Contrary to Egloff's expectations no gender differences were observed in the children's actions and preferences.

In their experiments on children's use of the Web, Gilutz and Nielsen [2002] found bigger differences between boys and girls than is usually found between adult men and women. The boys complained significantly more about verbose pages than the girls. It may be that girls are better readers at this age. Girls on the other hand, complained more than boys about sites that lacked good instructions.

More guidelines relating to gender differences have emerged in earlier chapters. For example, in Chapter 6 (section 6.2.6) I discussed Inkpen et al.'s [1997] findings on the differences between turn taking behaviour of boys and girls when working together at one computer. Although the differences were not clear cut, the research showed that designers should at least offer different turn taking protocols for users to choose from.

7.4.4 Learning and Problem Solving

Price, Rogers, Scaife, Stanton and Neale [2002] believe that playful learning should include the following interrelated learning activities: engagement, exploration, reflection, imagination or creativity, and collaboration. Along similar lines Roschelle et al. [2000] report that research in general has shown that learning is most successful when it involves active engagement, participation in groups, frequent interaction and feedback and connections to real-world contexts. According to Roschelle et al. [2000], various studies have shown that computer-based applications that encourage serious reasoning about mathematics are more successful than applications that look good but are based on repetitive skill practice.

A software product that promises to support the development of a whole range of cognitive skills (mathematics, reading, music, art, and so on) or a complete curriculum should be considered with suspicion

[Shade, 1996]. It would serve the users better if designers focussed their attention on one or only a few learning areas and did it thoroughly and with the necessary consideration of developmental issues.

Liang and Johnson [1999] studied the effect of technology on early literacy, and from this formulated several guidelines for the development of successful computer-based material. In accordance with current play theory, the material should foster positive affect and intrinsic motivation; the activity should be more important than the end result and there should be a maintained 'as if' stance. They found that children will lose interest in games that concentrate on successfully reaching the end-product rather than on the pleasure of performing the activity. Drill-and-practice activities and trite electronic books should be avoided. There must be a match between the challenges offered by the material and the abilities and interests of the users. Mindless 'click-and-see' activities that require no mental effort, and to which small children unfortunately tend to be drawn to, should be avoided.

7.4.5 Metaphors and Icon Design

Children between five and eight are just starting to read and do not have the necessary skills to use applications such as Internet search engines that are largely text-based. Uden and Dix [2000] did research on the design of a search tool for use by young and preschool children. To make the facility usable they focussed on the design of an iconic interface based on metaphors that are suitable for the age group concerned. For the interface to be easy to use and learn it should be consistent with the user's mental model of the system [Uden and Dix, 2000]. Drawing on existing knowledge, carefully selected metaphors can help a child form an accurate mental model of the system. The problem with iconic interfaces is the difficulty to design icons that convey the correct meaning without invoking other connotations [Uden and Dix, 2000]. Users may recognise unintended aspects of the metaphor depicted by the icon.

For the interface of their system, Uden and Dix [2000] wanted something familiar that was intuitive, engaging and encouraged exploration. They began by focussing on two functions of the system – an email facility and a tool for watching movies on the Internet – and finding suitable metaphors that will help children understand the functionality and assist the designers in creating a suitable structure. To match the design model with the children's mental model of these functions, they conducted interviews with the children to gain insight into their existing knowledge. Metaphors that will work for adults will not necessarily work for children. The metaphors should not require the children to learn and remember many rules and procedures, they should draw on the children's knowledge of the world and allow children to predict the outcomes of their actions [Uden and Dix, 2000].

About the design of icons, Uden and Dix [2000] found the following:

- The icons used to depict the metaphors must closely reflect the children's mental model.
- They must be recognised as familiar objects to which the children can relate.

- Relationships between the icons and the functions that they represent must be clear (these functions should be analogous to the functions that can be performed on the physical objects that the icon resembles). Icon design is determined by the chosen metaphor, but if the metaphor has no clear iconic mapping it is obviously not suitable for the interface.

There are different categories of icons such as representational icons (for example, a knife and fork to represent a restaurant), abstract icons (such as a cracked wine glass to convey fragility) and arbitrary icons (a symbol that has no direct relation to the underlying concept), static icons and animated icons. Uden and Dix [2000] tested several full colour representational icons, some static and some animated, for their email and movie watching functions. (From prior experience they new that children do not like black and white icons and that they find coloured icons easier to recognise). Children could identify all the animated icons but not all the static icons. Many of the icons were meaningless to a five-year-old. For example, a printing press, a fountain pen, an old-fashioned typewriter, 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 [2000] speculate that the reason for the use of old fashioned images rather than pictures of modern versions, is that the new equipment is not easily distinguishable in a small image. 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 [2000] experiment the children surprisingly found a cameraman with an old-fashioned camera as the best indication of the movie watching function. 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.

In summary, children prefer animated icons and understand them better. The animation helps to demonstrate what can be done by selecting that icon and it provides entertainment. It is crucial to understand the children's mental models and even when these seem clear, thorough testing of the interface may reveal slight mismatches between the user's mental model and the design model. Children are not self-aware about their needs and they usually cannot tell a designer what they want before they see some options [Uden and Dix, 2000]. They do, however, like to feel in control and make their own choices. They do not have patience to struggle with something and soon lose interest if they cannot get it right. They do not want to be limited in their interaction with a system and are motivated by varied interaction. Children like to tell designers what an icon should look like and expect designers to read their thoughts to know exactly what they mean if they cannot articulate properly. Finally, Uden and Dix [2000] say that designers should not underestimate children's intelligence. They can learn a lot about children's playfulness, enthusiasm and the metaphors that they can identify with by spending time with them, trying to understand them and their mental models. 'They know what they like and want.' (p. 285).

7.4.6 Humour

Coleman [1992] discusses how humour can be used to enhance learning. The five variables that determine people's reaction to humour are:

- Social context (humour is more effective in a group situation as laughter feeds laughter),.
- Cognitive challenge (a joke that challenges the intellect is appreciated more).
- Novelty (the element of surprise plays a role in many humorous situations).
- Timing (this refers to building anticipation and delivering the punch line at the correct moment).
- Degree of detachment (jokes that are detached from personal issues are enjoyed more).

People's preferences with regard to humour depend to a large extent on their intellectual ability [Coleman, 1992]. Coleman says that children also enjoy meeting the intellectual challenge to understand and make jokes, but that their appreciation of humour depends on their factual knowledge, their symbolic, logical and abstract reasoning abilities, and their level of language development. Children with better cognitive and social skills are more able to produce, understand and appreciate humour.

Coleman [1992] looks at the use of humour in children's educational television programs, but we can transfer his findings to other forms of media. According to Coleman, fast-paced humour in educational television improves the acquisition of information as it helps to hold the child's attention. This is also true for children with short attention spans. Positive reaction to humorous incidents can make the child enjoy the programme as a whole. According to Coleman [1992], the more humour the better for children.

According to Coleman [1992], humour that employs irony or satire may be detrimental to young children's learning experience as they are not yet able to understand sophisticated forms of humour. Non-relevant humour with no direct relation to the subject matter is more effective in supporting preschool and early primary school children's knowledge acquisition [Coleman, 1992]. This changes as children grow older and become intellectually mature. For adults, non-relevant humour does not help knowledge acquisition and may even interfere negatively with it.

7.4.7 Praise

According to the Computers as Social Actors (CASA) research paradigm, adults apply their interpersonal interaction rules to their interactions with computers [Bracken and Lombard, 2004]. Results of research in this paradigm provide evidence that people attribute gender to computers, they perceive computers with different 'voices' as different social actors and they act politely toward computers.

Praise involves feedback that express positive affect, such as surprise, delight and excitement, and gives a person information about the value of his or her action [Bracken and Lombard, 2004]. Dispositional praise applies positive trait labels by making comments such as ‘good girl’. Non-dispositional praise evaluates a specific action or behaviour with statements like ‘you have neat handwriting’. Mills and Grusec [1989; cited by Bracken and Lombard, 2004] demonstrated that dispositional praise had significant behavioural, cognitive and affective consequences with children whereas non-dispositional praise did not. Another study found that girls are more intrinsically motivated when given praise than boys.

Bracken and Lombard [2004] investigated whether children’s social responses to computers are comparable to their social responses to people. They tested children aged eight to ten with two versions of a software product, where the one version responded to the child’s actions with praise and the other with neutral feedback. They were interested in the effect of praise from a computer on children’s perceived ability, their intrinsic motivation, recall and recognition. They found that children who received praise rated their own ability as greater and performed better at recall and recognition tasks than those who received neutral feedback. They also found that intrinsic motivation was not influenced by praise. Their results indicate that perceived ability or intrinsic motivation is not necessary for an increase in recall or recognition. Their was also an age distinction in that the older children who received neutral feedback performed better on the recall tasks, while younger children who received praise performed better. In general, Bracken and Lombard [2004] showed that children respond to a computer as a social actor rather than a machine and that these experiences can have useful consequences. On the basis of their research results, Bracken and Lombard [2004] warn designers that young children are affected by the relationships within their lives. Since they display intense reactions to computers, their human-computer relationships should be handled with as much care as their human-human relationships.

According to Bracken and Lombard [2004], young children react in a positive way to praise, but when they get older they tend to see praise as a negative. Although they refer to other research that claims this change occurs around the age of eleven, Bracken and Lombard were not able to link the change with a specific age. Older children and adults will consider the task context and the difficulty of the task when interpreting feedback that they receive, so that praise is only effective as a direct motivational strategy with younger children.

Developmental Issues:

E116 Software should provide cues for coordination between symbols on screen and the keyboard to address the problem young children have with relating two-dimensional representations to their three-dimensional referents.

E117 Appropriate context cues activate prior knowledge that children can apply to make sense of what they perceive and their interpretation thereof.

E118 There should be a match between the content of the material and children’s cognitive competencies. This match can be achieved through additional cues for difficult concepts. For example, in conservation of number tasks the spatial arrangement of objects to be counted can influence children's performance.

E119 If feedback on correct or incorrect actions mentions the child's name, he or she may pay more attention to it.

E120 Neutral feedback after errors (such as a triangle instead of a sad face) will increase the incentive values of reinforcers. Children lose interest in activities if they perceive themselves to be motivated through external rewards rather than engagement. The informational aspect of reinforcers is at least as important as the reward value.

E121 Software should exploit children's natural intellectual curiosity and their eagerness for acquiring new knowledge.

E122 Preoperational children should be allowed to interact through direct manipulation using a mouse, rather than indirectly through keyboard commands, as such commands require additional information to be held in working memory.

E123 By seven they begin to understand organisational schemes such as the system for organising books in a library.

E124 Because preoperational children do not yet understand conservation of liquid, for example, an interface should not rely on changing size of objects to convey new knowledge.

E125 From seven, children are able to organise items into groups and categories, they enjoy browsing activities and due to the quick development of working memory, they can manipulate more than one item in memory at the same time.

E126 From seven, motor skills allow the use of different types of input devices and perceptual acuity reaches a level at which children can understand the way two or more software tools can be used together to accomplish a task (for example, creating an image with one application and dropping it into another).

E127 3D interfaces for young children should be stripped down, with a limited number of familiar objects that resembles a familiar setting such as a child's bedroom or playroom. It could be an imitation of their lives with games in a toy chest, homework on a desk and a dust bin for discarding unwanted objects.

E128 3D workspaces could also be modelled on well-known children's books, so that when there is a lot of detail it will still be familiar to the user.

E129 Interface tools that reflect how often they have been used or accessed may help children to see the relationship between actions and outcomes. This can be achieved through the use of colour, size, shape or pop-up messages.

Cultural Issues:

E130 Software should reflect a child's context and should therefore come in multiple languages, reflect gender equity, avoid racial discrimination and portray diverse families and experiences. It should also promote positive social values and no violence.

E131 When designing a guide character, the children's culture should be kept in mind. For example, in some cultures an owl is a knowledgeable creature, so a designer should not present an owl that just pretends to be clever. Similarly, a shy and worried bat will not fit their expectation that will probably be based on Batman and ghost stories.

E132 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.

Gender:

E132 Girls prefer pretend play based on reality while boys prefer pretend play based on fantasy. Girls lean more toward education and strategy games and boys toward combat and sport games.

E134 Boys have been found to complain more about verbose web pages than the girls. It may be that girls are better readers at this age. Girls on the other hand, complain more about websites that lack good instructions.

Learning and Problem Solving:

E135 Playful learning should include engagement, exploration, reflection, imagination or creativity, and collaboration.

E136 Learning is most successful when it involves active engagement, participation in groups, frequent interaction and feedback, and connections to real-world contexts.

E137 A single software product cannot adequately support the development of a whole range of cognitive skills (mathematics, reading, music, art, and so on) or a complete curriculum. Designers should focus their attention on one or only a few learning areas and do it thoroughly and with the necessary consideration of developmental issues.

E138 The activity should be more important than the end result. Children will lose interest in games that concentrate on successfully reaching the end-product rather than on the pleasure of performing the activity.

E139 Mindless 'click-and-see' activities that require no mental effort, and to which small children unfortunately tend to be drawn to, should be avoided.

Metaphors and Icon Design:

E140 Metaphors should draw on children's existing knowledge so that they can easily see what to do and predict the outcomes of their actions.

E141 Icons must look like things children will recognise. Successful interpretation of an icon depends on its caption (what it is meant to communicate), the context in which it will appear, and the image.

E142 Uden and Dix [2000] learnt the following about icon design for young children:

- Children do not like black and white icons and find them difficult to recognise.
- They prefer boxed icons.

- They recognise animated icons more easily than static icons.
- They prefer animated icons that come alive when the mouse moves over them.
- Icons with linguistic cues are not suitable for this user group.
- Icons should not be culturally specific.
- Images of outdated objects such as a fountain pen or typewriter may not be recognisable by five or six-year-olds unless they have seen them in picture books or cartoons (which is quite possible).
- Do not make assumptions about what a five or six-year-old will understand and should always involve them in their selection of interface icons. Adults cannot predict what children will think or like.

E143 Uden and Dix [2000] also learnt some things that do not relate directly to icon design:

- Young children are not self-aware and articulate about what they want, but they expect designers to read their thoughts and give them what they want. Designers should give them options to choose from.
- When an interface offers them limited interaction options, they lose interest.
- Children expect an interface to be interactive and animated.
- Children want to control their environment.
- Children do not have patience to struggle with something and soon lose interest if they cannot get it right.

Humour:

E144 Five variables that determine people's reaction to humour are:

- Social context (humour is more effective in a group situation as laughter feeds laughter).
- Cognitive challenge (a joke that challenges the intellect is appreciated more).
- Novelty (the element of surprise plays a role in many humorous situations).
- Timing (this refers to building anticipation and delivering the punch line at the correct moment).
- Degree of detachment (jokes that are detached from personal issues are enjoyed more).

E145 Children also enjoy meeting the intellectual challenge to understand and make jokes, but that their appreciation of humour depends on their factual knowledge, their symbolic, logical and abstract reasoning abilities, and their level of language development.

E146 Children with better cognitive and social skills are more able to produce, understand and appreciate humour.

E147 Fast-paced humour in educational television improves the acquisition of information as it helps to hold the child's attention. This is also true for children with short attention spans.

E148 The more humour the better for children.

E149 Humour that employs irony or satire may be detrimental to young children's learning experience as they are not yet able to understand sophisticated forms of humour.

E150 Non-relevant humour with no direct relation to the subject matter can be effective in supporting preschool and early primary school children's knowledge acquisition.

Praise:

E151 Praise involves feedback that expresses positive affect, such as surprise, delight and excitement, and gives a person information about the value of his or her action.

E152 Dispositional praise applies positive trait labels by making comments such as ‘good girl’. Non-dispositional praise evaluates a specific action or behaviour with statements like ‘you have neat handwriting’. The former has significant behavioural, cognitive and affective consequences with children whereas non-dispositional praise does not.

E153 Children who receive praise have been found to rate their own ability as greater and performed better at recall and recognition tasks than those who received neutral feedback.

E154 Young children are affected by the relationships within their lives. Since they display intense reactions to computers, their human-computer relationships should be handled with as much care as their human-human relationships.

E155 Young children react in a positive way to praise, but when they get older they tend to see praise as a negative.

7.5 Conclusion

In this chapter I discussed respected design guidelines and principles for the design of technology in general and explained their applicability to technology aimed at young children. I also reviewed the literature that present design and usability guidelines for products aimed at young children. Throughout the chapter I pulled together the different proposed guidelines and organised them into lists of potential guidelines for the design of technology for young children.

The main product of my study will be a set of guidelines for the design of technology for children aged five to eight that are based on the literature in different relevant fields. In formulating these guidelines I cannot ignore the design guidelines and principles that are already available. The purpose with this chapter was not to create new scientific knowledge with regard to interaction design, but rather to take the guidelines for design that are already there and harvest those that are applicable to technology aimed at five to eight-year-olds. Where the existing guidelines were not aimed at young children’s technology I had to contemplate their applicability to such technologies. The contribution of this chapter then lies in the process of reflection and sifting and, because there are a lot of overlap, the fusion of different sets of guidelines.

I have now completed phase 2 of my study, namely to review the literature on young children and technology to gather guidelines for the design of technology for children aged five to eight. My next task (in Chapter 8) is to analyse all the design-related factors and potential guidelines gathered and organise them into an integrated framework.

CHAPTER 8

Classification of the Emerging Guidelines

Table of Contents

8.1 Introduction

The overarching purpose of my study is to demonstrate that it is possible to develop a comprehensive and useful set of guidelines for the design and evaluation of children's technology, by studying 1) psychological theories of children's development, 2) existing research results on children's cognitive development, 3) existing research results on children's use of technology and 4) existing design guidelines and usability principles. I have now reached the point where I can complete this undertaking by displaying the overall results of my literature investigation. This third phase of my study involves the classification and organisation of the guidelines that emerged in phases 1 and 2.

The literature that I investigated in Chapters 4, 5, 6 and 7 yielded 502 data items representing potential guidelines for the design of technology for children aged five to eight. I collected and labelled these in what I refer to as 'data boxes' throughout those chapters. Because the emerging guidelines address a wide range of design factors, constructing a classification framework requires thorough inspection and intense commitment to ensure credibility and usefulness. In this chapter I report on the outcome of a laborious process of analysing, coding, dissecting, integrating, grouping, questioning and reformulating the 502 data elements to come up with an agreeable classification.

As described in Chapter 3 (section 3.6.1.3) I structured the data analysis process according to a five step process proposed by Terre Blanche and Kelly [1999]. My discussion of the classification of the guidelines (in section 8.2) is organised according to their structure. The familiarisation and immersion step is described in section 8.2.1. Inducing themes involves a discussion of the initial categories that emerged after a first round of data analysis (section 8.2.2) and coding is described in section 8.2.3. In section 8.2.4 I explain how the initial categories were reduced to a refined classification scheme. In section 8.2.5 I present the complete classification framework containing the resulting 350 guidelines organised into 6 categories and 26 subcategories. I then briefly discuss (in section 8.2.6) the emerging guidelines that were excluded from the framework and the reasons for excluding them, and conclude the chapter in section 8.3.

8.2 Classification of Guidelines Emerging from the Literature

8.2.1 Familiarisation and Immersion

The way in which I gathered the design-related data elements, required a thorough understanding of the theory or literature from which they emerged. Consequently, at this point I have already progressed a long way towards familiarisation with the data. To prepare for the next step I printed each of the 502 data elements on coloured cards and, while editing and formatting the text for this purpose, studied them again attentively.

8.2.2 Inducing Themes (Initial Categories)

The process of identifying categories of guidelines already began when I planned the literature investigation and decided on how I should organise my discussion of the literature in Chapters 4, 5, 6 and 7. The initial set of categories therefore corresponds to a large extent to the sections of those chapters. They are:

- Designing developmentally appropriate activities.
- The role of children's prior knowledge in interaction with technology.
- The role of gender, culture, context.
- Zone of Proximal Development and scaffolding.
- Children's memory development.
- Transfer of skills between knowledge domains.
- Literacy development (learning to read and write).
- Development of storytelling skills.
- Development of mathematics skills.
- Development of problem solving skills.
- Development of computer literacy.
- Providing feedback.
- Controlling the interaction (system vs. user).
- Encouraging and supporting collaborative use of technology.
- The supporting role of interface agents.
- Input devices.
- Speech input and output.
- Robotic interfaces.
- Tangible interfaces.
- General interface/interaction design issues.
- Supporting users with disabilities.
- Designing for the Internet and the WWW.

8.2.3 Coding

The coding process began when I collected the potential guidelines. The label assigned to the data elements link them to the associated parts of the literature study. Instead of using further codes to link my data items to the themes identified in the previous step, I used a card system whereby I physically organised the data into the categories where they belong. Figure 8.1 shows examples of the cards with the labelled data elements and Figure 8.2 illustrates how the cards were organised according to the initial categories.

Where a guideline statement belonged to more than one category I made copies of the cards and placed them in both groups.

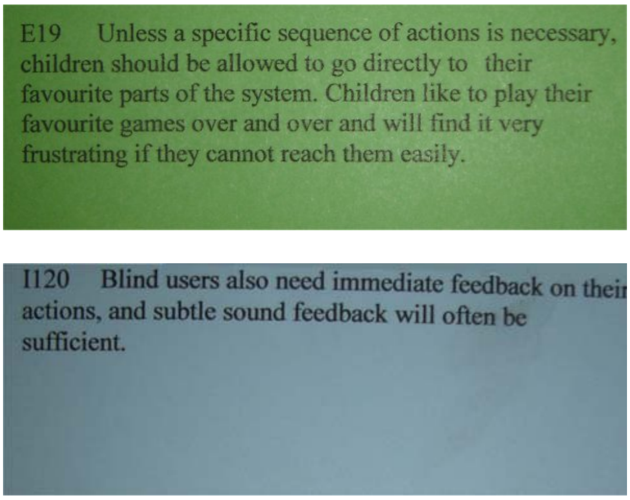


Figure 8.1 Data cards

Figure 8.2 Organising the cards into themes

8.2.4 Elaboration (Refined Categorisation and Classification Scheme)

My first attempt at refining the categorisation produced broad or abstract categories such as ‘functional requirements’, ‘environmental requirements’ and ‘usability requirements’ such as those used, for example, by Read et al. [2004] and Preece et al. [2006]. This was not successful as it led to the separation of groups of guidelines that logically belonged together. I consequently followed a more practical classification approach based on questions that arise from design practice. Below is the list of questions that I used as the basis for the final categorisation:

1. What does ‘developmentally appropriate technology’ mean when designing for five to eight-year-olds?
2. How do we design technology to support the development of mathematics (reading, writing, storytelling or problem solving) skills?
3. How can scaffolding (or other kinds of support) be built into software or technology aimed at young users?
4. How should interface agents that support young users look and sound, and how should they behave towards the user?
5. How can we design technology that encourages collaboration between young users?
6. To what extent should a product aimed at five to eight-year-olds cater for diversity in terms of culture, ethnicity, language, context and skill level?
7. What input and output devices and mechanisms are suitable for young users?
8. What kinds of tangible interfaces are suitable for young users and how do we design them?
9. How do we design Internet and Web applications that are suitable for young children?
10. What interface elements or interaction environments are suitable for young children?
11. How does interaction design for you children differ from interaction design for adults?

Consideration of these questions led to the following classification scheme:

Table 8.1 Classification scheme for emerging guidelines

Category of guidelines	**Related concepts/skills**
A. Guidelines that will ensure developmental appropriateness.	Age specific abilities Biological maturation Existing knowledge structures Memory capacity
B. Guidelines aimed the development of specific skills	Mathematics Problem solving Reading Writing Storytelling Computer literacy
C. Guidelines about the design of built-in support	General support, scaffolding and ZPD Feedback Support through interface agents
D. Guidelines about encouraging collaborative use of technology	Peer collaboration Environments or interfaces that invite or inhibit collaboration
E. Guidelines that address the diversity of users	Identity (socio-economic and cultural context, personality, gender and language) Existing knowledge and experience Users with disabilities
F. Guidelines about the use and design of interaction environments and devices	Input/output devices Control (user vs. system) Engagement Tangible interfaces Robotic interfaces Internet and WWW Interface design

8.2.5 A Framework of Guidelines for the Design of Technology for Children aged 5 to 8

With all the groundwork done, the last step in the data analysis process was the 'interpretation and checking' stage proposed by Terre Blanche and Kelly [1999]. This involved formulating the final set of guidelines. In doing this I had to make sure there were no weak points, contradictions or holes in the articulation of the results.

The final framework of guidelines is presented in the tables below. After each guideline appearing in the framework, I list, in brackets, the labels referring to the data elements that form the basis of this guideline. This makes it possible to trace each guideline back to its original source (or sources) in the literature. The reader is reminded that the first character of a label refers to the associated thesis chapters as follows:

P – Chapter 4 (Psychological theories).
T – Chapter 5 (Technology for children).
I – Chapter 6 (Interaction environments).
E – Chapter 7 (Existing guidelines).

A. GUIDELINES THAT WILL ENSURE DEVELOPMENTAL APPROPRIATENESS

A.1 Age specific guidelines	
A.1.1	Do not use age as the only indicator of knowledge – experience can have a significant effect (P84).
A.1.2	When designing for five-year-olds, designers can use the following list as an indication of their competencies. Children aged five: • Have the logic ability to play simple strategy and memory games, to do simple patterning and to learn clear and simple rules (E58, E60). • Find it difficult to use logic operations such as ‘and’, ‘or’ and ‘not’ (E88). • Start to see a situation from another’s perspective (decenter) and can consequently switch between character viewpoints (T60, E58). • Can separate themselves mentally from their physical surroundings to imagine and accept absurd fantasies or strange characters in stories (E72). • Start to understand the concept of conservation (P11, E58). • Can behave pro-socially and model pro-social behaviour of adults or media characters (E70). • Start to develop a strong sense of identity and realise that they belong to a specific gender (E69). • Know the eight basic colours and can mix colours (E61). • Are interested in writing words (E63). • Are able to use interfaces that integrate tactile and screen-based elements (T120).
A.1.3	When designing for six-year-olds, designers can use the following list as an indication of their competencies. Children aged six: • Like fun, humour and simplicity – they enjoy jokes and riddles (E78). • Prefer bright contrasting colours that create patterns (E65). • Mostly still prefer predictable activities that they can direct, but they begin to appreciate a challenge to familiarity (E67, E78, E82). • Can begin to follow simple written directions and give simple written feedback such as typing their names (E82). • Have sufficient logic ability to perform strategy-based or competitive activities (E82). • Begin to enjoy the challenge of being timed to see if they have improved at an activity (E56).

	• Can play memory-based games (E82). • Like to talk, sing and record their own voices (E68). • Are able to cooperate with other children and wait for their turns – they can reason about fairness and play group games without adult supervision (E74). • Can perform tasks that combine counting and comparison of quantities (P58).
A.1.4	When designing for seven-year-olds, designers can use the following list as an indication of their competencies. Children aged seven: • Have the perceptual acuity to understand how two or more software tools can be used together to accomplish a task (e.g. creating an image in one application and dropping it into another) (E126). • Can start to organise items into groups and categories and can understand organisational schemes such as that used for books in a library (E123, E125). • Enjoy browsing activities (E125). • Should be given tasks that involve combining, ordering and separating objects mentally (P07). • Start to understand construction activities based on logic operations such as 'and'. 'or' and 'not' (E89). • Can begin to write simple programming commands to direct movement of an on-screen object (T21). • Are very aware of their gender (T79). • May enjoy competitive games (but these should not be designed in a way that only winners will want to continue playing) (E56, E79, P89).
A.1.5	When designing for five to eight-year-olds, designers can use the following list as an indication of their competencies. (Note that these characteristics were not pinned to a specific age in the literature.) Children aged five to eight: • Can coordinate structures for dealing with more than one aspect of a situation (P27). • Do not fully understand the functioning of biological systems, but they understand that living things grow and toys do not (P82). • Have a good idea of the materials things are made of and can, for example, predict whether something will break when it falls (P78). • Understand psychological causality and realise that characters have goals and beliefs that may influence their behaviour (P80). • Can reason to some extent about false beliefs (P81). • Can distinguish between physical things and things that only exist in their or someone's mind, and between software-based characters or objects and those in the real-world (P03, P79). • Can interpret symbols and images that represent real-life situations (P04). • Fears being alone, getting lost and losing a parent and therefore like stories about characters who overcome these fears (E75).
A.1.6	Take children's exposure to video games, television, movies and other media into account when designing technology. It influences their expectations of technology as follows: • They want a multi-sensory experience (E97). • They want state of the art technology (e.g. headphones are 'cooler' than speakers) (E95). • They do not want technology that 'talks down' to them (E96).

A.2 Biological maturation (excluding memory development)	
A.2.1	Know the minimum requirements of a specific task with regard to biological maturation and do not expect children to perform actions that they are not physically capable of (P35).
A.2.2	When designing for children aged six or seven, designers can assume that they have the following biological competencies: • Their visual motor functioning and eye movement control is sufficient for good focussing and scanning skills (E53). • Their large motor skills allow balancing tasks and competitive physical play (E55). • They can manipulate input devices without problems and are able to use different types of input devices (E82, E126).
A.2.3	Design activities for eight-year-olds so that they can complete them comfortably in a forty minute session. The beginning and end of an activity should be clearly defined so that restricting their time at the computer will not interfere with an ongoing activity (E114, E115, T122).
A.2.4	Young children are often mentally ahead of their physical selves, so give them opportunities to practice grown-up actions (E62).

A.3 Existing knowledge structures in skill development	
A.3.1	Identify all the skills involved in an activity and understand how these will be coordinated and integrated in the activity (P23, P36).
A.3.2	When all the underlying skills and operations have been identified, determine whether the user will be able to perform each of these and whether they have the mental capacity for the new skill (P23).
A.3.3	Be well-informed of all the knowledge structures that underlie every activity presented to a child (P01).
A.3.4	When two or more skills must be integrated to acquire a new skill, provide activities that require application of those skills or operations before presenting users with activities that combine them (P23).
A.3.5	Be informed of the development sequence of every skill that will be supported in an application (P37).
A.3.6	Do not assume that if a child can solve a specific kind of problem in one domain that they can transfer that skill to a different domain. Support independent development of skills in different domains, but provide explicit links between the domains (e.g. some children may make the connection between music timing and mathematical fractions, but others will not grasp the link without instruction) (T16, P29, P30, P32).
A.3.7	Make any skill's connection with real life explicit. An activity chosen to develop a skill must be one that can be naturally associated with that skill (P31, P32).

A.4 Memory capacity	
A.4.1	Strive to relieve a child's working memory of extra processing that may prevent them from coordinating their knowledge structures – that is, reduce the cognitive load of interaction so that they have sufficient cognitive resources for learning to take place (interpreting and navigating the user interface should require as little working memory capacity as possible) (P25, T40).
A.4.2	Find ways to free up processing or storage capacity in working memory. For example: • Help children to practice a skill until it becomes automatic – when an operation becomes automatic some working memory is freed up (P24, P74). • Make objects, actions and options visible so that users need not remember instructions or previous choices (E39). • Teach children explicitly how to use memory strategies, since choosing between and using such strategies consume processing capacity (P75). • Let young children interact through direct manipulation with a mouse, as keyboard commands require information to be held in working memory (E122).
A.4.3	Use vivid fantasy images to help children remember what they have learnt (E44).
A.4.4	If the aim is to capture information or knowledge in long term memory some drill-and-practice may be used, but take care to keep the child engaged (P69).
A.4.5	Audio communication persists only in the user's memory, so do not rely on children's accurate recall of audio instructions (especially when given in the beginning of a session) (E20).
A.4.6	Children in the same age group may have different upper bounds of memory capacity, so make support adaptable to this variation (P26).
A.4.7	Help children to construct scripts of everyday situations and task-specific circumstances. This will facilitate the retrieval of information relating to such circumstances from memory (P76).
A.4.8	Children who are familiar with narrative structures are better at recalling events in the correct order – if the aim is to improve this skill, provide them with activities that will develop their knowledge of narrative structure (P77).
A.4.9	Also keep in mind with regard to children's memory capacity: • The part of working memory that stores phonological information (the phonological store) is sensitive to similarity – children recall different sounding items better than sounds that are similar (P72). • Because the visiospatial and verbal short term memories work together, the image that a child associates with a sound may influence the perception of the sound – so, the way children process what is presented to them is influenced by the associations it triggers (P73). • From seven, children can manipulate more than one item in memory at the same time (E125).
A.4.10	The phonological store keeps information for only two seconds – do not expect a child to act on audio cues that occurred longer than two seconds ago (unless the information is repeated often). Verbal instructions should be short (P71).

A.5 General high-level guidelines relating to children’s development	
A.5.1	Make the learning goals of the application clear and base the design on a clearly identifiable educational approach that supports these intended goals (E107, E108).
A.5.2	Do not attempt to address a range of cognitive skills in one application. Focus on one or two skill domains and do it well (E137).
A.5.3	Design an application so that its use can be integrated with other practices such as play and project work (E110).
A.5.4	Construction activities provide an effective basis for interaction, so give children activities that allow them to design, create and evaluate (E86).
A.5.5	Encourage play behaviour, keeping in mind that play is not only motivated by feelings of pleasure – it is sometimes inspired by the need to be something that children are not, or to have something that they do not have (P88).
A.5.6	Avoid click-and-see activities that require no mental effort. The fact that young children are drawn to these does not mean they should be included (E139).
A.5.7	Promote positive social values and no violence (E130).
A.5.8	If the aim is to reduce the need for adult intervention, make sure that children cannot produce incorrect results – only unexpected outcomes (E87).
A.5.9	Children benefit significantly from support by teachers who closely guide children’s interaction and continually encourage, question, prompt and demonstrate. There is thus a place for technology designed for collaborative use with an adult (T88).
A.5.10	Respect preschoolers’ natural need to demonstrate their abilities to parents and caregivers – give them opportunities to do so (e.g. allow them to print things they have made so that they can show and tell) (E80).

B. GUIDELINES AIMED AT THE DEVELOPMENT OF SPECIFIC SKILLS

B.1 Mathematics	
B.1.1	Computer-based activities should not be screen-based versions of paper-based drill-and-practice sheets. If practice is required (e.g. with multiplication tables), place such activities in a fun and engaging environment which links the concepts to their real-life uses (T01, P69).
B.1.2	Do not convey a message that mathematics is about memorisation (T02).
B.1.3	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 the symbols in their school workbooks with real quantities or operations (P52, P61).

B.1.4	Present activities that gradually teach children to associate number symbols with the correct number of objects – they need to attain the procedures for translating between formal arithmetic code and concrete situations (P53, P62).
B.1.5	Help children make the connection between formal representations and dynamic visual representations, thereby supporting construction of mathematical ideas through visual approaches (T06, T19, T28).
B.1.6	Only introduce operator symbols such as + and – their associated operations when children can use the number symbols confidently (P54).
B.1.7	From six, children can be presented with activities involving counting, comparison of quantities and simple arithmetic (P58).
B.1.8	When rewarding a child with a game, use the learned mathematical concept in the game so that the child links the fun with the mathematics. Children should not get the idea that fun only starts after completing the mathematics. (T03).
B.1.9	When separating the game module and the instructional module of a product that supports mathematics skill development: • Children should only be able to accomplishing the goals of the game if they master the underlying mathematical concepts (knowledge they need for this is provided by the instructional module). • Do not force the instructional module on the child, but make it accessible at any time – motivation to get the mathematical knowledge must come from the desire to accomplish the goal of the game (T09, T10, T11, T12).
B.1.10	Only use hypothetical statements if users will be older than five and make sure such statements do not contradict aspects of the context (e.g. do not say 'suppose Zac has 10 cookies' if Zac appears to have a different number) (P57).
B.1.11	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 mind how the context may influence their understanding (P55, P56).
B.1.12	From around eight years, children can perform operations on quantities that involve two variables (e.g. rands and cents, hours and minutes) and they can mentally solve double-digit addition problems (P59, P60).
B.1.13	Encourage children to apply their mathematical skills and knowledge in real-world situations and provide links between different cognitive domains (e.g. mathematics and music). They must be able to detach the skill from the game context (P68, T04, T16).
B.1.14	Help children to turn their intuitive spatial knowledge into real knowledge by providing activities that require them to write simple program instructions that will move an object along a specified path (T23).
B.1.15	Let children draw shapes with simple programming instructions to teach them about measurement, inverse operations and geometry. Make sure there is an explicit connection between these activities and the mathematics involved (T25).

B.1.16	Teach children how they can use visual representations to understand and solve mathematical problems. I.e. Include activities that help children build a visual representation of a mathematical problem (T29, T30, T31).
B.1.17	Use activities that require children to recognize and manipulate shapes to teach them about symmetry, patterns, spatial order and fractions (T26).
B.1.18	To teach children about fractions, design on-screen blocks in a way that allows children to divide them in equal sized, smaller blocks (T27).
B.1.19	When choosing on-screen manipulatives, consider how children with different backgrounds and levels of experience will react to them. Cognitive resources must be used for the mathematical content, not to understand the visual representations (T32).
B.1.20	Support development of spatial knowledge and skills by letting children build, draw or follow simple maps of familiar places (P51).
B.1.21	Provide activities that teach children about organising and displaying data through representations such as bar graphs (P51).
B.1.22	Show children a strategy in action and then let them practice the strategy – practice and play with different kinds of mathematical problems will teach them to use the strategy and will lead to more sophisticated strategy use (P63, P64, P65, P67).
B.1.23	To support development of temporal awareness in children aged six to eight, provide activities that address the following skills: • Giving the correct sequence in which events occur. • Incorporating cyclical patterns (like the days of the week) into the bigger time system, so that they can, for example, understand that two Tuesdays are similar but also different. • Co-ordinating different temporal systems (for example, days with weeks). • linking temporal systems to number concepts (so that they can learn to read a watch or calendar) (T106).
B.1.24	To help children to reason about temporal concepts, provide them with an integrated set of explicit representations so that they can see how the different concepts (e.g. time, days, months) relate to each other (T107, T112).
B.1.25	Choose graphical representations that constrain reasoning about a concept in a way that helps children to understand the concept (T109, T110). To illustrate: When supporting the development of time concepts a circular time-chart is better than a time line so that children can, for example, see that midnight is also the start of a new day. Use linear structures to represent a unique sequence of events and circular structures to represent recurring properties of a temporal system (for example, the seasons of the year).
B.1.26	Use re-representation (different representations of the same temporal concept) only if it simplifies the interpretation. Re-representation can also complicate a concept if not use correctly (T108).
B.1.27	When referring to the time event occurred in the past, use 'the number of years ago' rather than a date, as young children still struggle to understand the concept of a date (T111).
B.1.28	Support children's understanding of time sequences with activities that require them to correctly order a jumbled sequence of events (T111).

B.2 Problem solving	
B.2.1	To support reversibility skills, include activities that require mental reversing of actions such as combining, ordering, separating and recombining of elements (P05).
B.2.2	When children have to follow a number of steps or states to solve a problem, record their steps so that they can be played back to them on request (using, for example, a standard replay metaphor that includes play, rewind, pause, etc.). They can also be allowed to change their actions during replay to see how this will affect the outcome (T05, T33, T34, T35).
B.2.3	To help children grasp cause and effect relationships: • Give younger children opportunities to move objects around on the screen by, for example, dragging them with the mouse (P06, E99). • Create opportunities for children to view processes and cause and effect relationships that are difficult to observe in reality (E104). • Provide activities that allow children to experiment with changes of state in a way that explains the differences (P10). • Software can be a 'process highlighter' that allows children to view processes and cause and effect relationships that are difficult to observe in reality (E104).
B.2.4	The content of an application should progress from simple to complex and from concrete to abstract (T96).
B.2.5	Do not expect young children to solve logic problems presented abstractly or symbolically. Bring meaning and purpose into the problem-solving situation (P83, P87). To illustrate: Children cannot infer from 'if A then B' and 'A' that 'B' is true, but they understand 'if you are naughty, no ice cream' and being naughty leads to no ice cream.
B.2.6	Compensate for children's lack of meta-cognitive knowledge that they need to choose between possible solutions to a problem (P85).
B.2.7	When children are required to use analogies to solve problems, make sure that they have sufficient knowledge of the two areas involved (P86).
B.2.8	Activities aimed at teaching children to use a strategy to solve problems should also teach them the concepts on which the strategy is built – children will only use a strategy if they understand how and why they work (P66).
B.2.9	Make children aware of the connection between different strategies and operations by giving them different views on a specific problem, but do not use re-representations that may complicate interpretation (T18, P65, T108).
B.2.10	Children have different preferences with regard to strategy use – allow them to use different strategies in problem-solving activities (P64).
B.2.11	Encourage children to think about and discuss their plans before doing an activity to help them to collaborate more, plan better and perform tasks more efficiently (T89).
B.2.12	Encourage reflection on actions (T15).
B.2.13	Support higher order thinking skills by allowing children to create, save, retrieve and change their ideas (T14).

B.2.14	Activities that require children to write simple programming commands to move an on-screen object, will force them to plan a solution – having to imagine the movements develops their thinking and problem-solving skills (T24).
B.2.15	Do not allow children to solve all problems through trial-and-error. Include mechanisms to focus their attention and to get them to plan their solutions (T93, T94). Children who never receive support from adults in using a computer may internalise a trial-and-error way to solve problems – they do not learn to focus their attention or to plan their actions.
B.2.16	To help children to see a situation from another's point of view, provide them with activities where they must help on-screen characters to solve problems or make decisions that may have consequences for other characters (P09).
B.2.17	To improve children's perspective taking skills, teach them to imagine physical spaces from different points of view and to compare different states of the world by presenting them with: • Three-dimensional images that they can manipulate and rotate with the mouse. • Virtual physical spaces through which they can navigate with the mouse, keyboard and other input devices (P08).
B.2.18	Provide opportunities for children to: • Formulate their own problems and get feedback on them. • Make choices. • Explore and manipulate different kinds of representations interactively (T13, T18, T20).

B.3 Reading	
B.3.1	Learning to read involves integration of visual graphic skills, phonological skills and semantic skills. Technology aimed at teaching children to read should therefore address all of these skills separately. When children are competent in the separate skills, present them with activities that require all three (P39, P44, T41).
B.3.2	Develop semantic skills through activities that improve a child's vocabulary (P43).
B.3.3	Support visual graphic skills with activities that teach children to distinguish between different letter forms (P40).
B.3.4	Simultaneous application of different senses is advantageous for acquiring reading skills. Combine tactile, visual and auditory elements to support learning of letter-sound relationships – for example, immediate audio feedback when a child touches a letter or word (T43, T44).
B.3.5	To support phonological skills, provide activities that: • Involve recognition of rhyme (P42). • Improve letter-sound association (T41). • Present children with words in audio format and ask them to identify words beginning with the same letter (P41).
B.3.6	Only when children have adequate phonological awareness should they be presented with activities that require them to transform letter forms into phonological codes and vice versa (T42).

B.3.7	Reading must become an automatic skill, so include some practice activities and activities that aim to increase the speed with which children decode words. The main focus should, however, be on skill acquisition rather than practice (T51, T53).
B.3.8	Design reading support activities to have varying levels of difficulty and to be adaptable to different skill levels (T49).
B.3.9	Include the following components in a reading support application: • A management module that keeps record of children's progress, the tasks performed and their competency levels (this can provide information for adapting to differing needs) (T47, T52). • A diagnostic system that picks up errors and responds in supportive ways (T52).
B.3.10	Use speech recognition to help children with reading out loud, pronunciation and foreign language learning (I38).
B.3.11	Use a text-to-speech facility to help children with the pronunciation of an on-screen word. They can decide which words they want to hear and select them with mouse clicks (T45).
B.3.12	The following are examples of suitable activities to support acquisition of reading skills: • Matching pictures with spoken words. • Indicating where a sound is heard in a spoken word. • Indicating which letter sound is heard. • Pointing out a specific letter in a word. • Filling in a missing letter. • Matching pictures with written words. • Selecting a word by its sound. • Spelling a word that is already written on the screen. • Spelling a word by its sound (T54).
B.3.13	If audio output and speech recognition technology are available, use the following sequence as an example of activities that will help children acquire reading skills: 1. Improve children's vocabulary so that they understand the meaning of words presented to them. 2. Present them with activities that will develop the skill to recognise rhyme words amongst words presented to them in audio format. 3. Provide them with examples of sounding out words letter-by-letter using audio combined with visual cues. 4. Help them recognise the same word in different contexts. 5. Present them with activities to develop the skill to produce rhyme words. 6. Present them with reading tasks. When they struggle to read the word, help them by providing semantic cues, giving words that rhyme with the particular word or, if they are still unsuccessful, sounding the word out for them (P45, P49).

B.4 Writing	
B.4.1	Include activities that allow concrete manipulation of letters and words (through touch, hearing, seeing, constructing, playing and replaying auditory constructs) to help children with decomposition, re-composition and creation of words (T46).
B.4.2	From age seven, children can be presented with activities that teach them about punctuation (P50).
B.4.3	To support learning to spell: • Present children with different spelling options for a word given in audio format so that they have to select the correct one (P46). • Use a spellchecker-like facility that helps children recognise and correct their own mistakes (it should not automatically correct spelling mistakes) (P47). • Draw children's attention to the relevant spelling rules and repeatedly demonstrate the rule (P48). • Use analogies with similar sounding words that children already know (P48).
B.4.4	Use speech feedback for writing support, but let children decide when and whether it should be given (T56, T57).
B.4.5	Children should have control over the level of speech feedback – that is, whether it should be at letter, phoneme or word level (T58).
B.4.6	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 (T59).
B.4.7	Allow easy adaptability to different languages (T55).

B.5 Storytelling	
B.5.1	Keep storytelling interfaces very simple and include a training module to help children acquire a good enough mental model to use the system (T65).
B.5.2	The following are suitable activities to include in a storytelling application: • Reading existing stories. • Sequencing jumbled stories. • Finishing partly written stories. • Changing existing stories. • Helping an agent to write a story. • Creating a new story with characters and a recorded voice-over or space to type the story (for children who can write) (T75, P50).
B.5.3	Keep children engaged by allowing them to implement their own ideas, but remember that they usually do not want to create a whole story from scratch (T61).
B.5.4	Include external mechanisms to help children with planning, organisation and sequencing of events (e.g. if they initially include a character that never plays a role in the story, ask them whether the character is still needed). They need affirmation and they are not always proficient in talking about stories in a structured way (T62, T69).

B.5.5	Include supportive elements such as the following in storytelling products: • Text-to-speech technology. • Spelling support. • Structured word banks that children can access easily. • A software agent that provides proactive assistance (T70).
B.5.6	Provide a text or ideas bank that: • Gives children ideas and suggestions. • Supports unusual combinations of scenes and characters. • Encourages the use of imagination. • Challenges stereotypes. • Changes power relations (T72).
B.5.7	Typical elements that can be included in a word or ideas bank are story titles, story starters, story stirrers, story events, endings and feelings (T72).
B.5.8	When designing storytelling applications, keep in mind that: • Children do not require a high degree of realism in presentation. Even crude forms of movement and exaggerated facial expressions are acceptable (T63). • They are able to switch between editing (writing) mode and acting (playing) mode if the change in mode is very clear (T64). • From six, children can adjust their speech to different listeners and from seven they can be presented with activities that require them to shift between character voices and the voice of the narrator (E59, T80).
B.5.9	Create a balance between familiar and imaginative elements by allowing, children to select props, scenes and characters from real life, familiar environments or from fairy tale or space scenes (e.g. mix real world elements such as family photos with fantasy world elements, and elements from the past, present and future) (T71).
B.5.10	Through storytelling software, designers can provide opportunities for dramatic and creative play – although children do not physically carry out the actions, they create the characters and the story line and, through recording facilities, give the characters the voice (P91).
B.5.11	Use a combination of peer and agent help to provide emotional support during story creation. Support should be adaptable to individual needs (T68).
B.5.12	In terms of content, a supporting agent can initially give broad suggestions – if the child does not show progress the agent can provide more detailed help (T74).
B.5.13	Use an embodied conversational agent that resembles a young child and has better storytelling skills than the user, to replace real peer collaboration in single user applications (T76).
B.5.14	An ideal agent will respond to the content of the child's story, will prompt the child to continue or nod to show interest. This has been done through audio threshold detection (silences and pauses in the child's story) but should ideally rely on keyword recognition (T77).
B.5.15	An agent should provide a linguistic model, use constructive criticism and support perspective taking (T79).

B.5.16	Emotional interactions should be consistent with the personality of the agent (I76).
B.5.17	Children prefer storytelling software that interacts with a robot, to simple screen-based software (I85).

B.6 Computer literacy	
B.6.1	If children cannot be assumed computer literate, give beginners the opportunity to practice the following skills: • Moving the cursor with the mouse. • Left, right and double clicking the mouse buttons. • Dragging and dropping screen-based objects. • Locating specific keys on the keyboard (T37, T39).
B.6.2	Also, do not take the following computer literacy skills for granted: • Interpreting the computer's interface and being familiar with conventions in this regard. • Navigating electronic text. • Locating and retrieving appropriate information. • Having the sensorimotor skills necessary for computer interaction (T38, T39).
B.6.3	Use computer-based activities to introduce children to technologies that they do not normally have access to, so that they can understand the extended role computers play in our everyday lives (E112, T36).
B.6.4	Teach children to use computers for real purposes (E111).
B.6.5	Present children with activities that involve simplified gathering and managing information (E112).

C. GUIDELINES ABOUT THE DESIGN OF BUILT-IN SUPPORT

C.1 Support, scaffolding and the zone of proximal development (ZPD)	
C.1.1	Children of the same age do not necessarily have the same ZPD. Assess children's level of understanding of a concept or their competence in a skill to determine their ZPD for this specific concept or skill. This can be done as follows: 1. First give them random tasks of different levels of difficulty. 2. Then present them with examples that are below their ZPD in order to build their confidence (may be tricky as children may become bored if not challenged). 3. Now move to problems that are just beyond their capacity and provide scaffolding where they need it. 4. When a child succeeds with the help of scaffolding, give a similar task without scaffolding at first to determine if the relevant skill has been acquired. 5. When this has been achieved the application may provide a task that falls beyond their ZPD (P19, P20).

C.1.2	When a child makes an error, provide scaffolding through a series of hints that guides the child to the correct answer (T99).
C.1.3	If different scaffolding options are available, let users know what they are and how to access them (T103).
C.1.4	Providing encouragement is not sufficient – make children be aware of the processes underlying success or failure. Tell children what specific actions or choices led to the correct or incorrect result to help them to generalise from an experience to future ones (T95).
C.1.5	There are three types of scaffolding that can be incorporated in technology: 1. Supportive scaffolding supports a task without changing the task itself and includes guiding (through messages that appear when the software detects that the user needs advice), coaching and modeling (by providing examples that explain concepts). Guiding scaffolding allows fading by displaying a button that the user can click to switch off the support. Coaching and modeling examples only appear on the user's request, so they fade by not being used. 2. Reflective scaffolding encourages users to think about a task before doing it. It doesn't change the task, but asks the user to provide plans, predictions or evaluations. Fading involves reducing the requests for reflection. 3. Intrinsic scaffolding is built into tasks by, for example, starting at an easy level and gradually increasing the complexity of the tasks. Fading is implemented as changes in the task. For each activity there can be multiple levels of difficulty – beginning levels address fundamental cognitive skills and then one or more higher-level skill is added per level (T98, T105, E100).
C.1.6	As children become more proficient, gradually remove support until they can succeed on their own. This fading of support can be implemented through adaptive or adaptable scaffolding: • Adaptive support changes automatically based on the system's model of the user's understanding. This may be difficult to implement (T101). • Adaptable support is faded by the user. To help the user with fading decisions, allow self-evaluation so that the user can judge their own progress and understanding. Limit the fading options because too many will confuse users (T101, T102). • A combination of adaptive and adaptable support allows the user to control the fading, but with guidance from the software (T100, T104).
C.1.7	Make sure the application and the user share the required common knowledge and that this knowledge determines the choice of scaffolding (P21).
C.1.8	Specific mediation variables that can potentially be incorporated into software are: • Focusing (ensuring that the child focuses on the right interface element using mechanisms such as selecting, exaggerating, accentuating, grouping and sequencing). • Affecting (through verbal or non-verbal appreciation or affect). • Expanding (focusing the children's attention on the concepts they used to solve the problem). • Encouraging (establishing feelings of competence through verbal or non-verbal expression of satisfaction with specific components of a child's behaviour – through immediate vocal, musical and/or visual feedback) (T91, T97).

C.1.9	Support should ideally change the environment according to the child's needs, interests and abilities. Changes can involve: • Changing the intensity, frequency, order, form or context of stimuli. • Arousing the child's curiosity, attention and perceptual acuity (T90).
C.1.10	To develop their self-esteem, present children with activities that reinforce their ability to succeed on their own (E64).
C.1.11	Do not require children to get support from documentation (E41).

C.2 Feedback	
C.2.1	If there is a possibility that users cannot read fluently, provide feedback in audio format (T50).
C.2.2	Provide adequate feedback in the form of information (audio, tactile, verbal or visual) about what action the user has performed and what the effect of that action was. Content of the feedback should be understandable and in a format that is suitable for the targeted age group (E2).
C.2.3	Use multi-modal feedback to improve the comprehensibility and accessibility of children's technology (E15). To illustrate: Simultaneous audio and text cues can make a system accessible for reading as well as pre-reading children. It can also help to make a system accessible to children with disabilities.
C.2.4	Response time must be quick. When it is not instantaneous the system should give clear indication that the task is in progress. Lack of immediate feedback leads to repeated clicking or hitting of keys, which may influence the program's execution (E16, T07).
C.2.5	Response times for similar tasks should be comparable (E17).
C.2.6	Let feedback facilitate comprehension of the concepts as well as promote exploratory interaction (E50).
C.2.7	Use feedback that mentions the user's name to capture his or her attention (E119).
C.2.8	Provide state information in different formats or modes. Providing children with different visualisations of the same information can support their learning of the concepts or knowledge involved (E14).
C.2.9	Choose a format for feedback that is suitable for the context in which a product will be used. For example, in a classroom setting audio feedback may disturb classmates. This means designers should have a clear idea of the physical context in which their product will be used (E105, E106).
C.2.10	Let children control when speech feedback is given and give them the option to turn it off (I78, T57).
C.2.11	Praise can be given in the form of feedback that expresses positive affect such as surprise, delight and excitement, and it should give the users information about the value of their actions (E151).
C.2.12	Use praise and flattery carefully. Praise can have a positive effect on interaction, but children will not be convinced if they hear the same praise words every time they do well (I77, T118).
C.2.13	If used correctly, praise (as opposed to neutral feedback) can improve children's confidence in their own ability and help them to perform better at recall and recognition tasks (E153).

C.2.14	Rather use non-dispositional praise that evaluates a specific action or behaviour (e.g. 'your handwriting is neat') than dispositional praise such as 'good girl' (E152).
C.2.15	Neutral feedback after errors (such as a triangle instead of a sad face) will increase the incentive values of reinforcers (E120).

C.3 Support through interface agents	
C.3.1	Handle children's relationship with computer-based agents with as much care as their human-human relationships. Young children are affected by the relationships within their lives and they can display intense reactions to computers (E154).
C.3.2	If the child should build up a long term relationship with an agent, let the agent's interaction acknowledge previous encounters. It cannot greet children with exactly the same words every time they start up the system (T78).
C.3.3	In general, children prefer advice from a real toy to that from an on-screen agent – the effectiveness does not only depend on the content of the help provided, but also on the way the help is made available (T113, T117).
C.3.4	If support is provided by an electronic toy, keep the following in mind: • Children will only ask for help when they have become comfortable with the toy or agent (and usually when prompted by an adult to do so) (T114). • If children take notice of the support offered by the toy and this helps them to succeed it causes much pleasure (T115). • If the toy gives incorrect help, children will not use its help again and they may even become abusive towards the toy (T116).
C.3.5	When designing an interface agent, keep the intended user's cultural context in mind (e.g. in some cultures and owl is a wise creature, so it should not be presented as one that just pretends to be clever) (E131).
C.3.6	In addition to step-by-step assistance in content, form and structure, provide emotional support that is adaptive to children's individual needs (T68).
C.3.7	Let animated software agents facilitate conversational interaction through: • High fidelity moving lips that improve intelligibility of speech output. • Realistic gazing and gestures that improve the efficiency of dialogue. • An appropriate dialogue style and personality (I42).
C.3.8	Make sure the emotional aspects of an agent's interaction with a child fit the agent's personality. Speech patterns convey critical information about personality and feelings, and children's perceptions of technology are strongly influenced by the emotional tone of speech output (I71, I73, I76).
C.3.9	Adapt a synthesized voice to the user's personality to make the interaction more pleasurable. Users prefer voices that are similar to their own. Extrovert users prefer extrovert voices and introvert users prefer introvert voices (I44).
C.3.10	Let software agents support children in a positive and caring way by showing interest, affirming them and making suggestions for further action. They should not criticize or correct (T73).

C.3.11	To improve collaboration, planning and efficiency in performing tasks, let an interface agent that models an adult providing scaffolding, do the following: • Present the child with structured, but open-ended tasks. • 'Discuss' problem-solving strategies. • Encourage the child to collaborate with peers. • Suggest that they think about their plans before performing a task. • Help children to focus on a problem. • Seek precise information. • Compare different perceptions (T89, T94).

D. GUIDELINES FOR ENCOURAGING COLLABORATIVE USE OF TECHNOLOGY

D.1 Supporting peer collaboration	
D.1.1	Collaboration does not come naturally with all children. Offer children opportunities for social interaction where they can learn to work together with their peers (E92, T81, T87).
D.1.2	Do not assume that cultural differences inhibit collaboration –technology can become a shared interest that reduces cultural barriers (E93).
D.1.3	Determine which kinds of applications or activities facilitate social behaviour at a computer and give preference to those (e.g. children collaborate more when using drawing programs than when using face construction software) (T82).
D.1.4	Children prefer advice from peers to that from adults with regard to computer use (T81).
D.1.5	Keep in mind that from age six: • Children are able to cooperate with other children and wait for their turns (E74). • They can play group games without the help of adults and can argue about fairness (E74). • They develop a social awareness that leads to fear of exclusion by peers (E77).
D.1.6	Tactile toys encourage social interactions and collaboration between peers, but the toys will only succeed as collaborative learning partners if the help they provide is adequate and appropriate (T121).

D.2 Environments or interfaces that invite/inhibit collaboration	
D.2.1	Encourage co-construction of solutions to problems so that children are required to interact with each other to come to an agreement on what should be done (T83).
D.2.2	With tangible interfaces the physical set-up can constrain users so that they are compelled to collaborate (I97). (See guidelines F.4.11 and F.4.12 below).
D.2.3	If collaboration is desired, design activities so that children will gain something by choosing to work

	together – that is, doing it with someone will make it easier and more fun. Children should ideally discover for themselves the benefits of working together (T84).
D.2.4	Make the enhanced effect of collaboration versus lesser effect of individual use clearly noticeable in advance (T85).
D.2.5	Encourage collaboration by: • Making the effects of collaboration interesting and not completely predictable. • Making behaviour with multiple users a natural extension of the behaviour with a single user. • Using sound (or other rewarding) effects as feedback only on collaborative efforts (T86).
D.2.6	Avoid touchscreen interfaces when the objective is to encourage collaboration (I18, I32).
D.2.7	Avoid interfaces where both the touchscreen and the mouse are available at the same time. If one child is using the mouse another can easily interfere on the touchscreen causing conflict (I33).
D.2.8	Two-mouse interfaces can work if the sharing protocol is implemented carefully. Note that: • The best way to get children to take turns when sharing a mouse interface is to use a two mouse 'take' protocol: the child who does not have control clicks his or her mouse's right button to indicate that he or she wants control. The other child then responds by giving control over. • In a two mouse 'give' protocol the child who has control decides when to give control to the partner. • Boys have been found to perform better with the 'take' protocol, while girls do better with the 'give' protocol. • Designers should include different turn-taking protocols so that users can choose one that suits them best (I34, I35, I36).
D.2.9	Use a tabletop interface to transform familiar board or card games into interactive experiences by providing interesting feedback, preventing cheating and keeping track of player performance (I94).

E. GUIDELINES THAT ADDRESS THE DIVERSITY OF USERS

E.1 Identity (socio-economics, family and cultural context, gender, personlaity and language)	
E.1.1	Design technology to reflect a child's context – ideally it should come in multiple languages, reflect gender equity, avoid racial discrimination and portray diverse families, abilities and experiences. (E101, E113, E130, E132).
E.1.2	Design technology to cater for children's variable play preferences that is influenced by their age, gender, socio-economic status, personality, taste, special needs and experience (P93).
E.1.3	Create a profile of the intended user using information about their age, gender, physical abilities, level of education, cultural or ethnic background and personality (P13, E35).
E.1.4	The specific learning or entertainment goals of the product must fit the context of different kinds of users (P15).

E.1.5	If a product is aimed at children from different cultural groups, first investigate how these cultures use and teach the skills that the product will support (P34).
E.1.6	Embed tasks in scenarios that the users can relate to. It may be difficult to find a generic scenario that suits users from different contexts, so, in the same way as some applications let users choose their language of choice, give children a choice of scenarios (P16, P17, P18).
E.1.7	Acknowledge the culture and sub-culture of the intended users. Identify particular problems that are important in that culture and the tools typically used to solve that kind of problem (P28). To illustrate: 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 of information processing structures for 'eating in a restaurant'.
E.1.8	Keep in mind that the level of support and encouragement that children require depend on their context – children with a supportive family and good experiences are generally more resilient to stressful situations (E71).
E.1.9	From age six, children develop deeper relationships with people outside their homes (designs can model such relationships and allow children to role play) (E73).
E.1.10	Make activities gender-neutral or gender adaptable: • Girls prefer pretend play based on reality while boys prefer pretend play based on fantasy. • Girls lean more toward education and strategy games and boys toward combat and sport games. • Girls prefer a greater variety of play materials than boys. • Boys have been found to complain more about verbose web pages than the girls. It may be that girls are better readers at this age. • Girls on the other hand, complain more about websites that lack good instructions (P93, E51, E133, E134).
E.1.11	Give children the chance to express themselves in different ways, allowing a variety of approaches to perform an activity – sometimes they will want to tell stories, sometimes they will want to make up games and sometimes they may want to build things (E94, T66).
E.1.12	Designers must acknowledge their own context and how that may consciously or subconsciously influence their design practice (P14).

E.2 Existing knowledge and experience	
E.2.1	Determine whether users will be novices, experts or a mixture of beginners and advanced users and design accordingly. Users with different levels of expertise will require a layered approach. Give novices options to choose from and protect them from making mistakes. As their confidence grows they can move to more advanced levels. Users who enter the system with knowledge of the tasks should be able to progress faster through the levels (E36, E37, E46).
E.2.2	Keep complexity levels low for beginners but provide a high enough ceiling to allow children of different levels of cognitive development to benefit. Matching children's cognitive competencies can be achieved through additional cues for difficult concepts (E100, E118).
E.2.3	Enable frequent users to use shortcuts and allow them to skip introductions and instructions that they already know. Make sure children are aware of these options (E38).

E.2.4	Allow children to apply their real-world or other computer-based knowledge when interacting with a new system (E3).
E.2.5	Use appropriate context cues to activate prior knowledge that children can apply to make sense of what they perceive. Keep in mind how the context may influence their interpretation and response (P56, E117).
E.2.6	Children's appreciation of humour depends on their factual knowledge, language skills, cognitive ability and their social skills (E144, E145, E146).
E.2.7	Three-dimensional interfaces for young children should be stripped down, with a limited number of familiar objects that resembles a familiar setting such as a child's bedroom or playroom (E127).
E.2.8	Three-dimensional workspaces could also be modelled on well-known children's books, so that when there is a lot of detail it will still be familiar to the child (E128).
E.2.9	Activities must help the child to fit the information presented into existing knowledge schemes, adapt existing schemes to incorporate the new information, or to combine existing schemes to form more complex schemes (P02).
E.2.10	Help children to construct accurate scripts (or mental schemata) for everyday situations as well as for task-specific circumstances. Children use such scripts of familiar situations when retrieving related information from memory (P76).
E.2.11	When designing screen-based manipulatives, consider how children with different backgrounds and levels of experience will react to these. Children with scripts that support the interpretation of the visual representation of the manipulative, will have an advantage (T32, T49).
E.2.12	In products designed for classroom use, allow teachers to customise activities according to the children's needs and abilities (e.g. restricting the number of available options) (E102).

E.3 Users with disabilities	
E.3.1	When using gesture recognition to detect sign language input, keep the following in mind: • A push-to-sign function that allows users to indicate when signing starts and ends will help to eliminate the detection of fidgeting and chatter. • Using video input together with data from accelerometers fitted into gloves worn by the signer, can improve recognition rates – multiple modes of data for recognition increase the accuracy. • If the signer wears brightly coloured gloves, recognition is easier, especially for hand movements done in front of the face. • A gesture recognition system for sign language recognition requires lots of training data (I122, I123, I124, I125, I126).
E.3.2	The following applies when designing for visually impaired children: • Blind users need immediate feedback on their actions – subtle sound feedback will often be sufficient. • Blind children can use a joystick to move sounds to fixed positions in space, to 'catch' moving sounds and to 'throw' sounds. • Tasks that require visually impaired children to trace a path should use rounded corners and wide paths.

	• For tasks that require children to respond to textured surfaces, these surfaces must be sufficiently rough with enough friction (I108, I110, I114, I120).
E.3.3	A Phantom device that uses force feedback can be used as input device for visually impaired children. When designing technology for use with this device, keep the following in mind: • Fine motor skills of children younger than eight years are not sufficiently developed to use the Phantom device effectively – children must learn to hold and position the stylus correctly (I109). • Magnets can be used to guide the user to move or hold the stylus at a certain place (I109). • Girls are more patient when using the device (I112). • Successful use depends on motor skills, concentration and understanding of the tactile experience (I113).
E.3.4	Activities that are suitable for blind children are: • Locating and identifying sounds. • Recording and editing their voices. • Creating their own tangible characters and giving them voice by recording/manipulating their own voices (I115).
E.3.5	Use visual feedback for partially sighted children with care: • It may distract their attention from tactile input that is more important (I111). • These children often cannot see moving images. Slide shows are preferable to animations (I118).
E.3.6	Tactile input can be provided through: • Braille devices (for users who can read). • Tactile overlays mounted on a tactile board that is connected to the keyboard port. The functions on the board are mapped to keyboard shortcuts and can thus only be used with applications that have keyboard shortcuts for all functions. Ideally only one overlay should be used for a whole game (I117).
E.3.7	Use multimodal communication (such as simultaneous audio and text feedback) between the user and the system to make a system accessible to visually impaired children (E15).
E.3.8	To designing technology that is easily adaptable to visually impaired users, designers should: • Use sounds that are recognisable without the visual context, or make the necessary contextual information available through modalities other than vision (I116). • Make the Help facility adaptable in the sense that it can refer to the tactile and audio input/output instead of mouse or keyboard functions (I119). • Use a design model that separates the logic of the game from the interaction mechanisms (I121).
E.3.9	Robotic pets can support social development of children with severe cognitive disabilities (I127). (See guideline F.5.3 below.)

F. GUIDELINES FOR DESIGN OF INTERACTION ENVIRONMENTS AND DEVICES

F.1 Input and output devices	
The Mouse	
F.1.1	Children aged five to eight can use the standard mouse as well as a smaller mouse successfully – the size is not an obstacle (I01).
F.1.2	Marquee-type selection is hard for young children as they find it difficult to select the initial corner correctly. A more suitable way to implement this is to allow the child to select a group of objects by circling the objects with the mouse (I02).
F.1.3	When designing for children younger than six, give both mouse buttons the same functionality. They find it difficult to distinguish between left and right and will therefore find it difficult to click the left mouse button consistently (I03).
F.1.4	Help children to stop mouse movement on target by: • Using large hotspots (clickable areas) that are widely spaced. • Placing frequently used hotspots in a corner where it is easy to stop the mouse (I04).
F.1.5	Point-and-click (or click-and-carry) is a quicker and more accurate way for children to move objects on the screen than drag-and-drop (with the mouse button held down) (I05).
F.1.6	Drag-and-drop may be better for tasks where the kinaesthetic connection between holding the mouse button down and ‘holding on’ to the object involved contributes to successful performance of the task (I06).
F.1.7	In an interface where an object or character (or the user him or herself) moves along a trail and the user has to start and stop the movement, a click-and-go (click to start and click again to stop) or slide-and-go (move the mouse as long as the object has to move) interface is preferable to a hold-and-go (hold the mouse button down until movement must stop) interface. The choice between click-and-go and slide-and-go will depend on how well they respectively support the goals of the interaction (I07, I08)
F.1.8	‘Gain’ refers to the relationship between the distance the mouse moves and the distance of the movement on screen (a 2:1 gain means the mouse moves twice the distance of the screen object). When the child has to track movement with the mouse a 1:1 gain is best, a 2:1 gain is acceptable, but a 1:2 gain is unacceptable for children under eight years of age (I20).
F.1.9	When children are required to move the mouse and the mouse buttons should not be pressed during movement, deactivate the buttons if possible. Children often press the mouse buttons accidentally while moving the mouse (I24).
The Touchscreen	
F.1.10	Children’s competence with the touchscreen improves over time, so let them start off with activities that will give them practice in using the touchscreen (I14).

F.1.11	Make screen objects bigger when a touchscreen will be used – it is more difficult to make fine selections with a finger (I15).
F.1.12	The touchscreen is not ideal for tasks that require collaboration between users (I18, I32).
F.1.13	The touchscreen is particularly suitable for tasks where children have to track movement of an object on the screen (I19).
Game Controllers	
F.1.14	With regard to arrow key devices such as the Sony Playstation controller, young children's interaction using the device can be improved as follows: • Limit the input choices available. • Match the choices with the colour/shape icons on the keys of the controller. This means the choices must be limited to the number of keys available. If the options can be arranged on screen to match the arrangement of the keys on the controller, even better. The child will be able to select an option with one keypress (I11).
F.1.15	Another way to adapt the interface is to allow children to use the up and down arrow keys to move a selection highlight between the choices. Cursor movement occurs in discrete steps rather than in a continuous path to prevent overshooting. This allows for more options, but it requires more keypresses from the user (I12).
F.1.16	Do not expect young children to perform tracing tasks with a joystick – they find it difficult to control the joystick and often overshoot the target. From around seven they can learn to use their free hand as a brake to prevent overshooting (I09, I10, I25).
Speech Recognition and Speech Output	
F.1.17	Speech recognition input is suitable for: • Teaching pre-reading children concepts such as colours, shapes and the alphabet (I37). • Helping children who can read to practice pronunciation, reading, foreign language tuition or anything that would normally involve the assistance of an adult listener (I38).
F.1.18	Remember that training a speech recognition engine usually involves reading of training text, but with pre-reading children this is not possible. An adult can whisper the training text into their ears or it can be played to them through headphones (I40).
F.1.19	Two common problems that can occur when using speech recognition are out-of-turn speech and incorrect recognition errors. These can be solved as follows: • Out-of-turn speech – the user presses the space bar while speaking. Visual feedback tells the child when the system is in 'listening' mode and when not. • Incorrect recognition – when the child utters a word the system responds with 'Did you say …?' If the child says 'no' the system asks for the word again (I41).
F.1.20	Users adapt the amplitude and duration of their speech to the speech style of synthesized speech. This can be used to subtly guide users to speak within a range that the software can recognise. An extrovert voice leads to increased amplitude and shorter utterances while an introvert voice leads to reduced amplitude and longer utterances (I43).
F.1.21	Users prefer voices that are similar to their own. Extrovert users prefer extrovert voices and introvert users prefer introvert voices. Adapting a synthesized voice to the user's personality can make the

	interaction more pleasurable (I44).
F.1.22	A conversational interface should be adaptable in terms of response time. Children's response time is slower than that of adults and their response time increases with age (I45). (See guideline C.3.10.)
F.1.23	Children's perceptions about technology are strongly influenced by the emotional tone of speech output (I71).
F.1.24	Include natural variation in speech output (I75).
F.1.25	If characters are based on familiar characters the voices must be consistent with the known voices (I74).
Handwriting Input	
F.1.25	As soon as children start writing, handwriting interfaces are suitable for them. A handwriting tablet better supports fluent writing than a keyboard (I46, I47).
F.1.26	Children have been found to prefer the tablet PC to a digital pen, and the tablet PC produces a better recognition rate (I53).
F.1.27	Make sure children can distinguish between the top and bottom of a tablet or digital paper. If used upside down no words will be recognized (I54).
F.1.28	Indicate the recognition area on the tablet or digital page. Children tend to start writing near the top edge that may be outside the recognition area (I55).
F.1.29	A keyboard should always be available to correct errors that occur with handwriting interfaces. Typical errors are: • Spelling errors. • Construction errors (incorrectly formed letters). • Execution errors (failing to touch the tablet with the pen or adds spurious characters). • Software induced errors (recognition errors and incorrectly changed capitalisation) (I48).
F.1.30	Unwanted spaces appear between letters because children pause too long. Designers should make them aware of this so that they can learn to prepare a word in their heads before typing (I50).
F.1.31	Provide clear advice on efficient error correction strategies. Children will, for example, delete a whole word if only one letter is incorrect (I51).
F.1.32	Children do not always notice errors such as incorrect capitalisation caused by the software, spelling errors and recognition errors. Help them to look critically at the results and make sure children realise that recognition errors are not their mistakes (I49, I52).
F.1.33	Systems that use handwriting input and are aimed at young children should: • Support the planning, translation (writing) and review phases of the writing process, providing ideas for planning, allowing fast and accurate transcription and allowing for easy movement, alteration and deletion of text. • Include spelling support and file handling facilities. • Provide help through speech. • Take ten minutes or less for a child to learn to use. • Not assume that there will be expert adult help available. • Not assume that users can read well, spell well or write well (I58).

Movement Input	
F.1.34	Camera-based input can be used to map users' movements to onscreen characters in the following two ways: • Through indirect or disconnected mapping – a predetermined set of physical gestures is used to initiate character action. The user has to learn these gestures. • Through direct manipulation of the on-screen character. The character mimics all the movements of the user in real time (I59).
F.1.35	In direct control interfaces users may find it difficult to move in exactly the way they want the character to move. To solve this, the system can detect partial movement and then take over control to complete the movement in a realistic way (I66).
F.1.36	To help the user to stay within the camera's view, display a small web cam image of the user in a corner of the screen (I60).
F.1.37	Require movements that are intuitive and physically appropriate. Use the Wizard of Oz method to test whether the movements they require the children to perform are appropriate (I61).
F.1.38	Try to identify movements that do not vary too much from child to child (e.g. children perform swimming movements in a variety of ways, but their jumping movements are very similar) (I62).
F.1.39	Dynamic movements are recognised more successfully than subdued ones (I63).
F.1.40	If movement of the feet should be recognized, make sure the camera does not focus mostly on the upper body (I64).
F.1.41	When children tire their movements become less pronounced. Do not expect them move vigorously for long periods. Five to six-year-olds should rest every four or five minutes (I65).
F.1.42	In the same way that interactional synchrony (coordination of behaviour) improves interaction between people it will improve interaction between a child and a robotic toy that reacts on movement input. The toy's behaviour should therefore be synchronized with the child's (I67).
F.1.43	Interactional synchrony between a robot and the user is particularly important in a game where they dance together (I68).
F.1.44	When movement input involves manipulating objects in a two-dimensional game world by making movements in a three-dimensional space, the interface must help young children to bridge the conceptual discordance between the movements and their effects (I69).
Comparison of Input Devices	
F.1.45	For preoperational children (around age five), direct manipulation with the mouse is preferable to indirect manipulation through keyboard commands, as they lack the working memory capacity the latter requires (E122).
F.1.46	Children fare better with the mouse than the touchscreen when moving things around on the screen (I13).

F.1.47	Speed of mouse use is the same for boys and girls, but boys are faster with the joystick (I23).
F.1.48	Children’s response time with the joystick is generally slower than with other input devices. This can be because they find it difficult to change direction with the joystick. Restricting direction changes to between 90 and 180 degrees may help (I26).
F.1.49	The mouse, trackball and touchscreen map the user’s movement in terms of direction, but the joystick and keyboard have more abstract connections with screen objects (I27).
F.1.50	For tasks that require the user to pick up, move and drop objects, the mouse is most efficient and the joystick is more efficient than a keyboard (I28).
F.1.51	When choosing between the keyboard, mouse, handwriting input or speech recognition for text input, there is a trade-off between efficiency (speed) and effectiveness (correctness): • Correctness of input is equally good with the keyboard and mouse. • Correctness of handwriting input is significantly better than speech input. • Speech input is by far the fastest, with handwriting slightly faster than the keyboard and the mouse the least efficient (I56).
F.1.52	Although preschoolers (age five) do as well with the mouse as grade one to grade three children (ages six to nine), they prefer the keyboard. They still see input devices as objects of enquiry rather than tools and for them the process is more important than the end product. They do not care about efficiency yet (I29, I30, I31).
F.1.53	Efficiency starts to play a role in preference from grade one onwards. In grade one (around six years) children are equally divided between the mouse and the keyboard. In grade three (from eight) they clearly prefer the most efficient device (I29).
F.1.54	For tasks that involve tracing or tracking an object on the screen: • The touchscreen is better than the mouse as there is a direct relationship between the required action and the on-screen effect (I17). • Continuous control is better than discrete control (involving repeated keypresses to move the cursor along). When children fall behind in the tracking task using the discrete option, they will quickly press the key a number of times to catch up and then overshoot (I21). • Five and six-year-olds can trace large on-screen letters with the mouse but not with the joystick (I22).
F.1.55	If a system is used once-off or only occasionally, the touchscreen is good, but for applications that will involve extended use, the mouse is preferable. When introduced to the mouse and touchscreen, children initially fare better with the touchscreen but with practice the mouse is more effective (I19).
F.1.56	Ultimately, the effectiveness of the relationship between the task and the device will determine which device is best for the task (I17).

F.2 User control vs. system control	
F.2.1	Children are attracted to activities where they can be active participants. They want to design their own activity patterns. Let them decide what they want to do, how fast they want to do it and when they want to end (I81, I84, E67, E98).

F.2.2	Giving the child control over what happens on screen will promote engagement and self-directed exploration (T22).
F.2.3	User pre-emptiveness is preferable to system pre-emptiveness in children's interfaces. In other words, the user should have freedom to initiate any action. A system-pre-emptive dialogue is appropriate if children have to perform specific actions at specific times or places in the program (keep in mind that too much freedom may cause the user to loose track of incomplete tasks) (E12).
F.2.4	Unless a specific sequence of actions is necessary, allow children to go directly to their favourite parts of the system. Children like to play their favourite games over and over and will find it frustrating if they cannot reach them easily (E19).

F.3 Engagement	
F.3.1	Use the computational capacity of the computer to enhance learning and engagement – that is, visual displays, animated graphics, speech, recording progress, detection of and adaptation to individuals (T05).
F.3.2	Factors that influence engagement are attention, pace, interactivity and style of narrative (E30).
F.3.3	Adults and children have different ideas about what is boring or exciting and designers should avoid using their own definitions of such concepts to guide their design decisions (E32).
F.3.4	Make interfaces more engaging by letting them evoke emotional responses from users (I70).
F.3.5	Forestall situations where children can repeatedly fail at a task as this will cause them to lose interest. There must, however, be a balance between the challenge provided and the child's ability to perform the activity (P22, T08).
F.3.6	A sequence of increasingly complex tasks can sustain curiosity by introducing a surprising complication at each level (E47).
F.3.7	Offer children a variety of paths of interaction (E91, E143).
F.3.8	Drill-and-practice products promote a learning style that may reduce children's intrinsic motivation to learn (E109).
F.3.9	Task-sensitive praise affects task performance and motivation and it helps with task persistence when children are learning new material (I72).
F.3.10	Provide children with opportunities to construct things that perform some kind of behaviour, as this causes high levels of interest and excitement (E90).
F.3.11	Activities should concentrate on the pleasure of performing the activity and not on successfully reaching the end product – if the end result is more important than the activity, children lose interest (E103, E120, E138).
F.3.12	Include humour, warmth and spontaneity in an interface to increase children's motivation to use it (I79).

F.4 Tangible interfaces	
F.4.1	When designing tangible systems, designers should be clear on: • Which movements of a user's body will have meaning. • Which movements of physical representations will have meaning. • What these movements communicate. • What should be visible to participants at specific times during the interaction (I96).
F.4.2	The interaction should build on users' skills in a way that will invite interaction (I98).
F.4.3	There should be obvious links between user actions and the effect of these actions (I99).
F.4.4	If an interface involves associated physical and digital representations, both should be visible and there should be a natural mapping between them (I100, I106).
F.4.5	Base interactions on the ways children naturally solve problems using their bodies (I101).
F.4.6	To help children to understand abstract concepts, base these concepts in their understanding of: • Their own bodily-based concepts (I102). • Spatial concepts and relationships (I104).
F.4.7	Support parallel (not competing) use of motor, perceptual and cognitive processes (I103).
F.4.8	Design representations so that children can easily see how they relate to the world (I107).
F.4.9	A soft toy can be used to manipulate an on-screen character. This is possible through a recognition engine that recognises the toy's movements through wireless sensors embedded in the toy. The software should include one or more models for each possible action to aid the recognition process (I82).
F.4.10	When using a soft toy to manipulate an on-screen character, a sympathetic interface can improve interaction. This means the interface interprets the user's intentions based on the context and takes over control to let the character complete the action (I83).
F.4.11	The physical size of interface elements and their tangibility influence interaction and, specifically, collaboration. Large elements slow down interaction, allowing time to interact with co-users. Onlookers can easily see what is happening giving a sense of 'audience'. If the size makes it difficult for a single child to manipulate objects, they naturally work together to accomplish the intended results (I87).
F.4.12	Requiring children to move physical props around slows down interaction. Using more (and different) props together promotes collaboration as it avoids turn-taking and it encourages different collaboration styles (I87).
F.4.13	Designers of tangible interfaces should consider the influence of physical interaction on the interaction process and should only use tangible interface elements when they actually contribute to improved interaction (I87).
F.4.14	Slight changes in how the tangible interface is presented can cause huge differences in user behaviour and should always be tested with users (I87).
F.4.15	Designers can achieve a lot with low tech technology. It is more important for tangible systems to be adaptable to the users' needs than to look high tech and shiny (I87).

F.4.16	In mixed reality environments designers can use: • Physical to digital transforms (where children manipulate physical objects to create a digital effect), • Digital to physical transforms (where digital manipulation has an effect on physical objects) or • Digital to digital transforms (e.g. click of the mouse changes the display) (I92).
F.4.17	Physical to digital transforms can easily be designed to promote collaboration (for example, children manipulate different coloured blocks on a tabletop to mix colours on a screen) (I93).

F.5 Robotic interfaces	
F.5.1	Young children prefer huggable, cuddly robotic pets (I86).
F.5.2	Young children see a robotic pet as a being with a mind and feelings and social responses to such a pet can be achieved with minimal cues (I89, I91).
F.5.3	Characteristics of robotic pets designed to assist people with disabilities must match the expectations and attitudes of the intended users. It should not appear smarter or more alive than it actually is and create unrealistic expectations in the users (I90).
F.5.4	Robotic pets can mediate social interaction through reactive and proactive behaviour (I128).
F.5.5	To enhance personalised interaction between humans and robotic toys: • The robot should be able to recognise the human user as an individual and it should be able to alter its behaviour according to characteristics and behaviour of that particular user. • It should be able to accumulate experiences in its memory so that future behaviour can take earlier experiences into account. • It should express its emotions in a way that is clearly identifiable to the user (I80).

F.6 Internet and the WWW	
F.6.1	Web-based material aimed at preschoolers should: • Be easy to access. • Contain clear and bright visuals. • Contain activities that allow children to achieve the goal within a short time (E81).
F.6.2	When designing for the Internet and the WWW, avoid: • Unclear navigational confirmation. • Inconsistent navigation options whereby the same destination could be reached in different ways, causing children to unintentionally revisit sites they have already been to. • Lack of perceived clickability affordances that made users miss links that they would otherwise have followed. • Fancy wording that made it difficult to understand the different choices. • Text sections that are above the reading level of the intended audience. • Content that slows down the ineraction (E83).
F.6.3	Use the WWW to provide children with learning experiences that cannot be provided by any other means, such as viewing a drinking hole in a game park somewhere in Africa, or exploring the world through Google earth (I129).

F.6.4	When designing applications that involve searching the Internet, provide children with an age-appropriate a front-end. Pre-reading children will require an iconic interface (I130).
F.6.5	As soon as children can write they can be given activities that involve sending and receiving email messages (I131).
F.6.6	Young children can be given the task to create basic web pages with pictures and symbols (I132).
F.6.7	Preschoolers enjoy web sites where they can interact with familiar characters (E80).
F.6.8	Remember that children's preferences sometimes contradict design and usability principles that are appropriate for adult websites. For example: • They like animation and sound effects. • They want content that is entertaining, funny, and colourful, and uses multimedia effects. • Children are willing to 'mine-sweep' (that is, clicking on whatever seems clickable on the screen). • Children like geographic navigation metaphors such as pictures of rooms, villages and 3D maps. • They rarely scroll down and mainly interact with whatever is visible on the screen. • Children who can read are willing to read instructions before playing a game (E84).
F.6.9	Children cannot distinguish between content and advertising and regard advertisements as relevant site elements. They will especially click on advertisements where the banner contains popular characters or a seemingly 'cool' game (E85).

F.7 Interface design	
Interface Elements	
F.7.1	An interface for preoperational children should not rely on changing size of objects to convey information (e.g. a progress bar) as at this age children do not necessarily understand conservation of liquid (on which the metaphor draws) (E124).
F.7.2	Interface tools that reflect how often they have been used may help children to see the relationship between actions and outcomes. This can be achieved through the use of colour, size, shape or pop-up messages (E129).
F.7.3	Use static and dynamic defaults to support interaction. Static defaults are defined within the system or acquired at initialization. Dynamic defaults evolve during the interactive session (E18).
F.7.4	Successful interpretation of an icon depends on its caption (what it is meant to communicate), the context in which it will appear, and the image (I141)
F.7.5	When designing icons for children's interfaces, keep in mind that: • The design of icons must draw on children's existing knowledge (including knowledge gained from watching cartoons). • Children prefer icons in colour and find black and white icons more difficult to recognize. • They prefer boxed icons. • Children recognize animated icons more easily than static ones and they like it when the icons come alive when the mouse moves over them. • Icons with linguistic cues are not suitable for five to eight-year olds. • Icons should not be culturally specific (E141, E142).

F.7.6	Always involve users when selecting images for icons. Adults cannot predict what children will think or like. Give them options to choose from (E142, E143).
F.7.7	Children expect an interface to be interactive and animated (E49, E143).
F.7.8	There should be a clear mapping between interface elements and their effect on the system (E34).
F.7.9	Provide cues for coordination between symbols on screen and the keyboard to address the problem young children have with relating two-dimensional representations to their three-dimensional referents (E116).
F.7.10	Fantasy can be included in technology in two ways: • Through intrinsic fantasy where application of a skill is directly related to the fantasy. • Through extrinsic fantasy, where the fantasy is detached from the skill and can be used as the context for a variety of skills. Intrinsic fantasies are more interesting and instructional as the fantasy context indirectly teaches children to apply a skill as they would in the real world (E42, E43).
F.7.11	Children have different preferences for fantasies, so it is advisable to provide them with a choice of fantasy in which to embed activities. Designers can also give children the opportunity to help create the fantasy by, for example, letting them choose names for characters of places (E45).
Humour	
F.7.12	Use humour in abundance. It supports children's learning and creativity as it relaxes the brain (P33, T67, E148).
F.7.13	Keep in mind that preschoolers' sense of humour is still unsophisticated – they think that mispronouncing words or putting clothes on the wrong way is funny (even when they have done it repeatedly) and they do not understand humour that involves irony or satire (E76, E149).
F.7.14	Non-relevant humour with no direct relation to the learning content can support preschoolers' knowledge acquisition (E150).
F.7.15	When including humour in activities, be aware of the intended users' cognitive and social skills, as this determines how well children are able to produce, understand and appreciate humour. Specific skills associated with humour are factual knowledge, symbolic, logical and abstract reasoning and language (E145, E146).
F.7.16	When designing activities that involve humour, remember that reaction to humour is influenced by: • Social context (more effective in a group situation). • Cognitive challenge (a joke that has more is more appreciated). • Novelty (the element of surprise). • Timing (building anticipation and a well-timed punch line). • Degree of detachment (further removed from personal issues is better) (E144).
Visibility	
F.7.17	Only operations that are available should be visible, or it should be very clear which operations are not available. Use age appropriate means to make available operations known (E8, E9).
F.7.18	Make it very clear to the user what the next required action is (E9).

F.7.19	Do not expect children to use browsing as a way to get a clear picture of the system's current state. If the application involves reaching a series of sub goals to achieve some central goal, it may be necessary to give them (or an adult assisting them) the option to view their current progress (E22, E23).
F.7.20	Convey available functionality through highly visual interface components and not through textual representations, so that children who cannot yet read can use the system (E48).
F.7.21	Create a transparent interface that enables children to focus on what must be done and not on how they should use the interface (E52).
Consistency and Familiarity	
F.7.22	Children should be able to determine the effect of future action based on past interaction history. Make changes to the internal state of the system visible so that users can associate them with the operations that caused them (E1).
F.7.23	Interaction and input-output behaviour should be consistent within a system as well as across systems. The user should be able to extend knowledge of specific interaction within and across applications to other similar situations (E7).
F.7.24	Familiarity can be achieved through metaphors and through affective use of affordances that exist for interface objects. The appearance of the object should promote familiarity with its behaviour or function (E4).
F.7.25	Metaphors should draw on children's existing knowledge so that they can easily see what to do and predict the outcomes of their actions (E140).
F.7.26	Surprise is often a desirable element in children's games and can increase the experience of fun and engagement. However, when it comes to learning how to use a system and navigating through the available functions and activities, predictability is very important. If they performed an action before, they will expect the system to behave similarly when they perform that action again (E5).
F.7.27	Follow real-world conventions so that information appear natural and in logical order. Familiarity has a different meaning for children than for adults – they 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 (E6).
Error Prevention	
F.7.28	Help users to recognise, diagnose and recover from errors. Give error message in language that children can understand. Describe the problem precisely and suggest a solution (E24).
F.7.29	Make it easy to reverse an action. Young children should not be expected to know how to use Undo/Redo commands – the system should help them recover from an error through help that fits their level of understanding (E21, E25).
F.7.30	Use constraints that restrict the actions a user can take at a specific point during the interaction as error prevention mechanisms (E33).
F.7.31	Design to prevent errors rather than to help users recover from errors. Require users to confirm potentially erroneous actions before performing them (E40).
F.7.32	Sometimes children prefer an interface that is easier to use and needs correcting to an accurate, but difficult one. This is true, for example, for handwriting interfaces (I57).

Flexibility and Adaptability
F.7.33 Allow equivalent values of input and output to be arbitrarily substituted for each other, or provide or multi-modal input and output. Children have varying skill levels and preferences that will influence the type of input or output that is suitable for a specific user. Different modalities (channels of communication) can be combined to improve articulation of input or output or to make the system accessible to more users (E10, E13, E15).
F.7.34 The user interface should be: • Adaptable (allowing user-initiated modification to adjust the form of input and output). • Adaptive (allowing system-initiated modification to customise the user interface automatically) (E11).
General Interface Guidelines
F.7.35 Children's products rarely focus on productivity, therefore efficiency is not as important as in systems designed for adults (E27).
F.7.36 Create products that are safe to use. Safety in children's products involves: • How children are affected physically by using the system. • How they can be affected by accessing material that is not appropriate such as pornography or images of violence. • How they are psychologically influenced by the content (E28).
F.7.37 To determine whether a product has adequate utility, designers should ask whether it allows children to carry out tasks in the way that they would like to do them. For the sake of utility, it is therefore important that designers do not make assumptions about children's preferences (E29).
F.7.38 There is a trade-off between user experience goals and usability goals. An action that requires more effort may contribute towards making a product more enjoyable and engaging. Designers should be aware of the consequences of combining user experience and usability goals and make sure that they address the needs of the user (E31).

8.2.6 Potential Guidelines that were Excluded

Of the 502 potential guidelines only 22 were not incorporated in the above set. The reasons for excluding them were:

1. They use terminology or refer to concepts that have meaning only in very specific contexts (P12, P38, P70, I95, T92).
2. They are so general or obvious that they do not really contribute any value (T17, I16, E26, E66).
3. They involve a combination of aspects that are dealt with separately in other guidelines (P92, T48, E54, E135, E136).
4. They are not clearly applicable to the design of technology for young children (P90, E57, I39).
5. They are vague or difficult to reformulate as useful guidelines (E155, I105, T119, E147, E121).

8.3 Conclusion

In this chapter I presented the primary contribution of my study, namely an integrated framework consisting of guidelines for the design or evaluation of technology for children aged five to eight. I have identified six broad categories and twenty-six subcategories to use as an organisation scheme for 350 guidelines that deal with a broad range of design issues.

On a scientific level, the integration of knowledge from several different research fields – theory of development, cognitive skill development, young children and technology, interaction design for children and principles and guidelines for design – into one coherent framework of design guidelines is probably the most profound contribution of my study. The framework not only provides designers of technology for the five to eight age group with practical guidelines, but the assimilated knowledge on children's cognitive development will help them create developmentally appropriate applications without having to embark on an in-depth investigation into the cognitive competencies of their users.

The range of technologies addressed by the framework is so broad that no single product will require application of all the guidelines. The categorisation scheme used in the framework will make it easy for designers or evaluators to identify which subset of guidelines applies to their particular product, as will be demonstrated in the next chapter.

Comparing this framework with existing sets of guidelines for the design of technology for young children (as discussed in Chapter 7) the following can be said:

- This is the most comprehensive set of guidelines available.
- It covers a much broader range of technologies than any of the existing sets. Despite the broadness, each category provides a depth of guidance that is at least comparable to that provided by the available guidelines for that category.
- The process through which this framework was constructed elicits the specific theoretical and empirical knowledge from which the guidelines emerged and each individual guideline can be traced back to its original source.

I have now completed phase 3 of the research. What remains to be done is phase 4 – the validation of the proposed guidelines. In Chapter 9 I will evaluate the credibility and usefulness of the proposed guidelines by demonstrating how the guidelines can be used to evaluate and re-design an existing software application.

CHAPTER 9

Application of the Emerging Guidelines

Table of Contents

9.1 Introduction

This chapter is concerned with the remaining research question, namely: How credible and useful are the proposed guidelines? Having answered this question I will be in a position to defend or contest the accuracy of the thesis statement that directed my study.

Here then, I aim to demonstrate the practical value of the proposed set of guidelines. Using the proposed guidelines presented in Chapter 8, I evaluate an existing software application aimed at young children and then explain how parts of the application can be improved by re-designing them according to the guidelines. The product I have chosen for this purpose is Storybook Weaver Deluxe 2004 [Broderbund, 2005]. I begin the discussion with a brief description of the software in section 9.2. In section 9.3 I describe the results of the evaluation of this application. This is followed by some detailed recommendation for re-designing the software in the light of the evaluation and the proposed guidelines (section 9.4). Based on the evaluation and re-design, I end the chapter in section 9.5 by drawing conclusions with regard to the usefulness of the guidelines.

In the remainder of this chapter as well as in Chapter 10, 'Storybook Weaver Deluxe 2004' is abbreviated as 'SBW'.

9.2 An Overview of Storybook Weaver Deluxe 2004

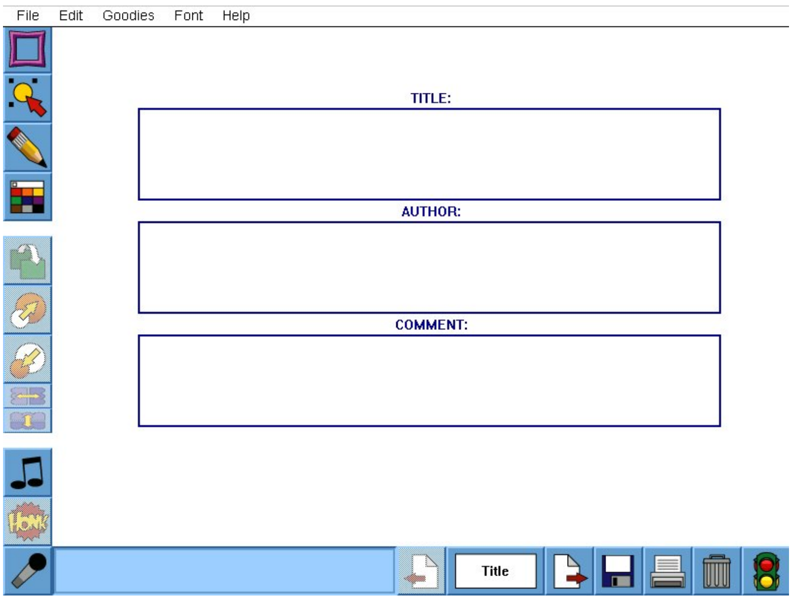


Figure 9.1 The Title page

SBW is a software application for creating story books. Children can choose from a large selection of backgrounds to create scenes on the pages of their electronic storybook and select from thousands of story characters and objects to create illustrations. Each story begins with a title page where they provide the title, author and comments (Figure 9.1), and on every subsequent page they can create a scene, type story text (if they can write), add background music, attach sounds to objects and characters, or record their own voices to attach to a page or story character. Figure 9.2 gives an overview of the story page and its functions.

The program is packed with 140 sceneries, over 1800 images with which to populate the pages, 37 colours, 69 page borders, 99 sound effects and 60 songs. Each page contains a text pane where the child can type the story text. Children can create voice-overs to attach to pages or specific characters by recording their own voices, or they can use existing sound files for this purpose. Additional functions such as a text-to-speech facility, a spell checker and a thesaurus are available on the 'Goodies' menu.

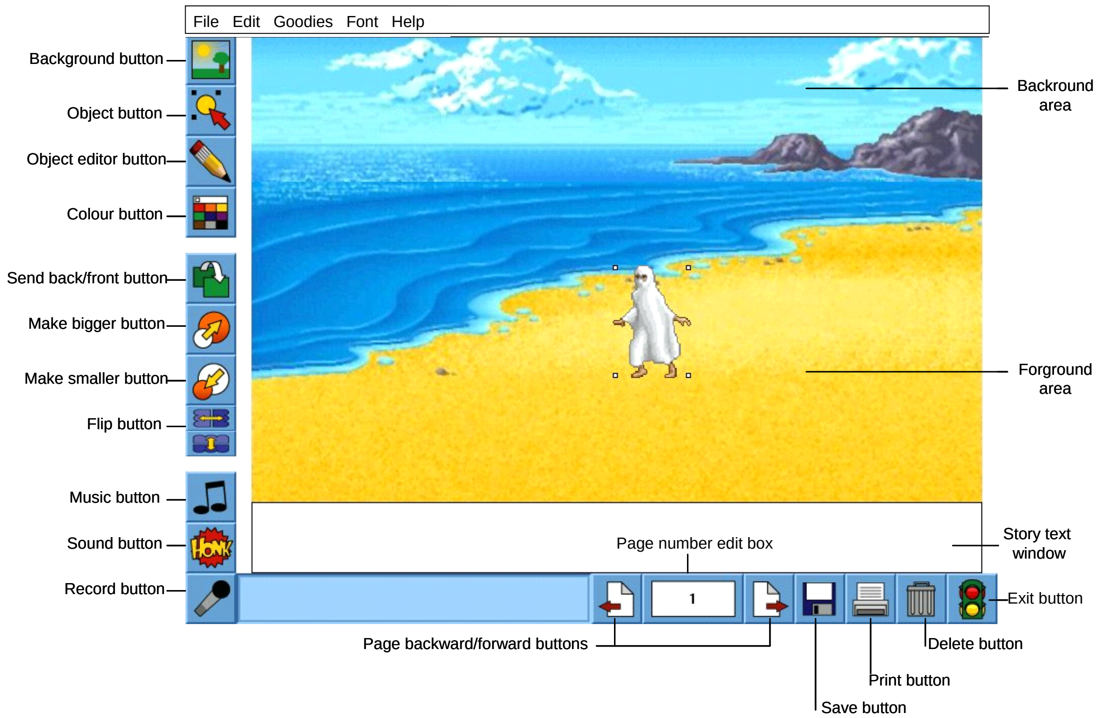


Figure 9.2 An overview of the Story page

From the opening screen (Figure 9.3) children can start a new story, open an existing story for reading, open an existing story for editing, get ideas from a 'story starters' list, or view a quick tour of the application.

A typical session will proceed as follows: The child starts a new story, types a title and his or her name on the title page and selects a border for the title page. Next, users go to the first story page where they pick a background scene and add story characters and objects. If they can write, they can type the story text in the story window. They can add music that should play in the background when the page is opened, and they can attach sounds to the page or story objects. If there is a microphone attached to the computer they can use the Record button to open the recording function where they can record a voice over for the story page.

Continuing with this, users can create as many pages of their story book as they like. Using the Save button they then save the story to disk. They also have the option (from the File menu) to publish the story as a Web document.

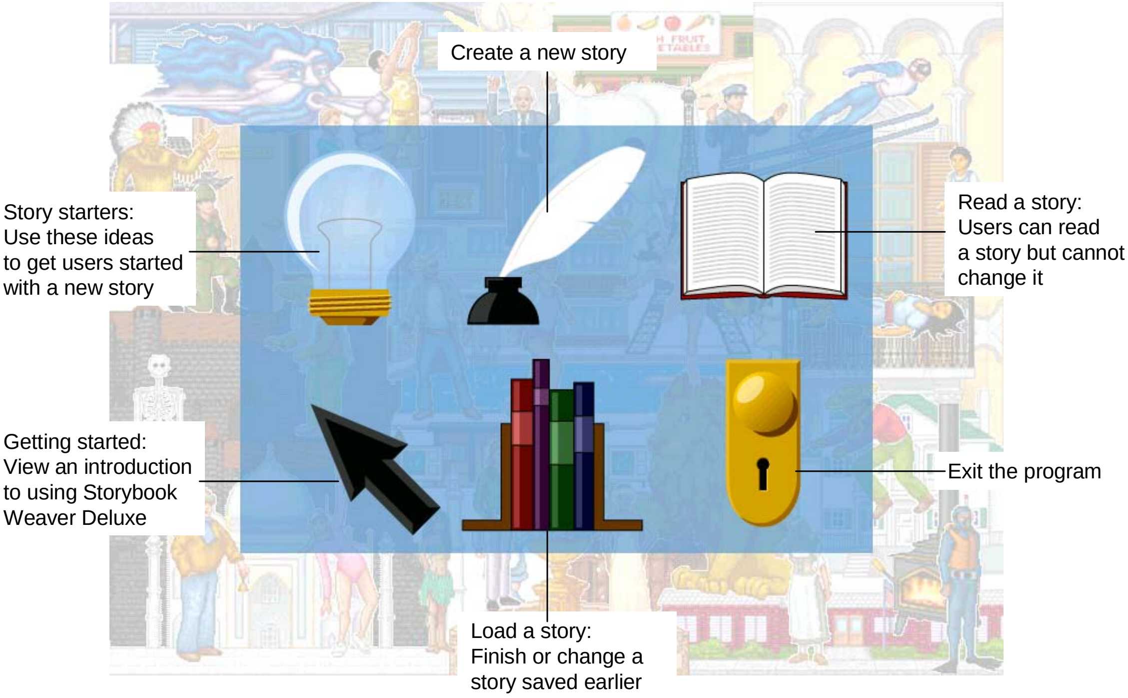


Figure 9.3 Opening screen

9.3 Evaluation of Storybook Weaver Deluxe 2004

This section should be read together with Appendix 1 which contains the evaluation notes on which this discussion is based. The guidelines presented in Chapter 8 address a large range of applications and interaction environments and designers or evaluators will never apply the complete set when designing or evaluating a specific application. The first step in this evaluation was then to identify those guidelines in the set that are applicable to a non-tangible desk top application for story creation. The left hand columns of the tables in Appendix 1 list the guidelines that were identified as such. The numbers used in those tables correspond to the numbers in the complete set of guidelines. The right hand columns contain the evaluation notes and suggestions for improvement.

Together, sections 9.3.1 to 9.3.6 represent an evaluation report based on the evaluation notes in Appendix 1. These sections correspond to the six categories of guidelines identified in Chapter 8. Throughout the evaluation I include the numbers of the relevant guidelines in parentheses.

9.3.1 Developmental Appropriateness

9.3.1.1 General Issues

The designers of SBW set out the learning objectives clearly in the accompanying documentation (A.5.1). These are:

- To use a user's inherent creativity to write and illustrate a story.
- To explore the writing process using a simple word processor and a variety of graphics.
- To create illustrations that depict the storyline.
- To write with a purpose.
- To share writing with an audience.
- To enhance vocabulary by associating a word with its picture.
- To develop story sequencing skills.

They do not explicitly state their educational approach but it is clearly constructivist as children learn the above skills by actively constructing stories (A.5.1).

SBW does not attempt to address a large range of skill areas – it focuses exclusively on storytelling (A.5.2). A solid understanding of narrative structure helps with the writing process in any subject. The software can easily be used to complement schoolwork in disciplines such as learning to read, learning to write, creative writing and even history (A.5.3).

SBW successfully uses construction activities as the basis of interaction. It allows children to design a story, a setting and the individual characters, and to create the story. There is, however, no support for evaluation of what they have created (A.5.4). None of the characters or objects promotes or portrays violence, but children are also not actively encouraged to create stories about social values and pro-social behaviour (A.5.7).

SBW should ideally be used with some help from a parent or teacher as children may miss a lot of the functionality if it is not explicitly pointed out to them (A.5.9). Research by Chimbo and Gelderblom [2008] has shown that children only need to be shown once how a function in SBW works to be able to use it. Once they have mastered the basic functionality, children are completely in charge of what happens and nothing they do will produce 'incorrect' results (A.5.8).

SBW allows children to demonstrate their stories to others by printing it on paper or by creating a web-based version of their story (A.5.10). There is no distinction between 'story writing' mode and 'play back' mode. It would be useful include a playback option that shows the story in the form of a continuous slide show (B.5.8, C.2.8).

9.3.1.2 Age Specific Features

SBW was designed for use by children aged six to twelve. The following features will also appeal to five-year-olds (A.1.2):

- Absurd fantasy characters and background scenes are available.
- Beginning writers can type their names in the author field.
- Its open-ended nature that gives children the opportunity to model the pro-social behaviour of adults or media characters.
- There is ample opportunity to express their gender-related preferences.

SBW is developmentally appropriate for six year olds in the following sense (A.1.3):

- Borders and backgrounds include options with bright contrasting colours that create patterns.
- Characters and scenes include familiar characters as well as strange and fantasy characters.
- Children can write their names, story titles and simple story text, and they can record their own voices to attach voice-overs to pages or story objects.

One feature of SBW that will appeal to children from around seven years of age is the possibility to import graphics or photos created in a different application into SBW for use as story objects (they can, in other words, import photographs of themselves, their friends, family or of their favourite celebrities) (A.1.4). Also appropriate for seven-year-olds is the way children can browse easily through story characters, backgrounds, borders, sounds, and so on, and the fact that the story creation process involves combining, ordering and separating objects mentally (A.1.4)

SBW uses a simple categorisation scheme for story objects that is presented in text format. The categories are therefore only understandable by children who can read and thus not suitable for children younger than seven (A.1.4).

Other features that make SBW developmentally appropriate for five to eight-year-olds are (A.1.5):

- Through creating stories, children have to see situations through the characters' eyes and learn to deal with more than one aspect of a situation.
- It gives children the opportunity to use their understanding of psychological causality and their realisation that characters have goals and beliefs that may influence their behaviour.
- SBW gives children opportunity to use symbols and images to represent real-life situations.
- They can write stories about their favourite topics such as the fear of being alone, getting lost and losing a parent, and overcoming these fears.

Many children have experience in using applications such as the SIMS™ series with characters that move around and that allows them to modify the appearance of characters. This may raise their expectations about what is possible in SBW and they could be disappointed by the static nature of the characters (A.1.6). On the other hand, if designers allow children to dress and modify characters' appearance in SBW it may distract them from the story creation process – they may get so involved in creating and modifying characters that they never get to write a complete story.

9.3.1.3 Biological Maturation

With regard to the physical use of the input and output devices, children find it difficult to use the menu in SBW as it requires them to keep the mouse button down when moving the cursor down to the required option (A.2.2). This is not consistent with how menu selection usually works in a Windows environment and can easily be corrected. From age six, children can use the recording function without difficulty (A.2.2).

Writing a story in SBW is an open-ended activity and a child can take hours to complete a story. SBW has no mechanism for restricting the length of a session and it is impossible to direct the writing process in a way to limit the sessions to forty minutes. It is up to the parents, teachers or care givers to monitor the time spent at the computer (A.2.3).

Through the story characters, SBW gives children opportunities to implement grown-up actions that they are not yet physically capable of (A.2.4).

9.3.1.4 Existing Knowledge Structures

The skills involved in story telling are: deciding on a theme, choosing the setting and characters, choosing objects, ordering events, identifying a suitable beginning, development and ending, dividing the story into pages, focussing on the theme and completing the story (A.3.1). SBW allows children to develop these skills through actively constructing stories by themselves or with the help of a more skilled story writer. There is no built-in coaching or support other than a simple and, mostly, logical interface that divides the story into pages and promotes easy construction of story scenes. SBW also provides a list of story starters by way of support.

In SBW, creating a story also involves the following skills: typing or recording the story text, placing the illustrations on the page and organising the characters and objects in the chosen scene, and writing for an audience (A.3.1). SBW makes it easy for children to adapt their use of the software according to the skills they already have – if they cannot yet write they just create a picture book without any text. It does not require children to perform specific activities. It is merely a set of tools that children can use in whatever way they want to and are capable of. Children will naturally only use the tools that they have mastered and therefore combine operations that they are familiar with (A.3.4).

Of the skills mentioned above there are two that five to eight-year-olds may find difficult. These are to type the story (as they may not be able to read or write yet) and to stay focused on the chosen topic (A.3.2). The problem with staying focused is aggravated by the fact that SBW allows children to browse thousands of story scenes, objects and characters. They can get so involved in browsing that they forget what their story is about, or they may stumble upon interesting characters that have no link to their story and become side-tracked (A.3.2).

SBW supports one overarching skill, namely story creation. All the possible activities fulfil some purpose in the aim to help children to create a story and can be naturally associated with it (A.3.7). There are functions that children from five to eight will not understand or discover unless they are explained to them explicitly (e.g. the recording function, the spell checker and the thesaurus).

9.3.1.5 Memory Capacity

The fact that most of the interaction takes place through direct manipulation with the mouse reduces the strain on storage and processing capacity (A.4.2). The keyboard is used to type story text, but navigation is primarily done with the mouse. There is still room for improvement with regard to relieving a child's working memory of extra processing – many of the icons do not clearly convey their purpose and some of the functions are 'hidden' in the menus (A.4.1, A.4.2). This will be discussed further in the suggestions for re-designing SBW in section 9.4.

If doing the quick tour is the only way a child will learn to use SBW (in other words, when there is no adult support), there is reliance on children's recall of audio instructions (A.4.5). The audio instructions are, however, linked to a visual display which children should recognise once they use the system. The child will learn a lot through the quick tour, but audio cues can be used much more extensively to help (especially pre-reading) children to interpret icons correctly (C.2.3).

9.3.2 Development of Specific Skills

The aim of SBW is to develop children's storytelling skills, but as a side-effect it also supports some problem solving and writing skills.

9.3.2.1 Problem Solving

Creating scenes, adding story characters and objects and editing these in SBW sometimes require mental combination, separation and re-combination while children are planning their stories, thereby supporting the development of reversibility skills (B.2.1). SBW offers many opportunities for observing cause and effect relationships. For example, dragging objects around on the screen and using the 'move forward', 'flip', 'make bigger' and 'make smaller' buttons. Since all of these operations can easily be reversed they also support children's reversibility skills (B.2.3).

SBW allows children to make choices and to explore and manipulate different kinds of representations interactively (B.2.18). It compensates to some extent for children's lack of meta-cognitive knowledge by providing story starters, but there are many other ways in which they could be supported to make choices (B.2.6) – for example, an ideas bank that contains story endings, story emotions, and so on. SBW does not provide users with a platform to plan a story in advance (B.2.11) – that is, a ‘place’ where they can think about the beginning, climax and ending or where they can consider possible characters and suitable sceneries.

SBW gives children endless opportunities to create, save, retrieve and change their ideas, but its open-ended nature makes it easy for children to lose focus (B.2.13, B.2.15).

SBW can potentially teach children about the concept of perspective in the sense that objects in the front of a scene can be made bigger and those at the back smaller. Currently children have to discover this for themselves or with the help of an adult (B.2.17). SBW also does not allow children to rotate objects and characters – they can flip them vertically and horizontally but not to face in an opposite direction (B.2.17). (I have observed children trying to create a family around a dining room table and getting frustrated if they could not include characters with their backs to the viewer.)

9.3.2.2 Writing

SBW includes a spell checker that children can use to check the spelling of text on a page (if nothing is highlighted), single words or a piece of text. They highlight the text and activate the spell checker. When the spell checker shows the correct spelling, they can either double-click in the spell checker to replace a word or correct the spelling themselves. Children find this function difficult to use [Chimbo and Gelderblom, 2008]: firstly, children have to highlight the text, which is already difficult for younger children, and secondly they have to select the Goodies menu, keep the mouse button down and move to the spell checker. This function should be much more accessible (B.4.3).

SBW has a text-to-speech (TTS) facility that reads back what the child has written on the current page, or what the child has highlighted (B.4.4). A misspelled word leads to strange utterances by the TTS facility, but this may help a child to identify spelling mistakes (B.4.6). I discuss later how this facility can be improved.

SBW is currently available in English and Spanish and one can easily change the language in the Preferences option on the menu. The software seems easily adaptable to different languages (B.4.7).

9.3.2.3 Storytelling

SBW allows children to read existing stories, to complete partly written stories and to create a new story with characters, music, recorded voice-overs and typed story text (B.5.2). It has a training module that is sufficient to get a child started and the interface is simple to use [Chimbo and Gelderblom, 2008]. It does, however, not display all the functionality clearly (B.5.1).

SBW gives children complete control to implement their own ideas, but do provide them with story starters if they want help (B.5.3). It includes TTS technology and spelling support, but not a structured word bank or a software agent that provides proactive assistance (B.5.5).

SBW does not have high quality graphics. From my observations this does not seem to bother children, but some users have indicated that they would like to be able to rotate people and objects (B.5.8). SBW allows children to use their own voices to speak for characters as well as to narrate – they can change their voices when speaking on behalf of different characters (B.5.8). They can also ask friends or family members to provide voices for different characters.

SBW allows users to mix real world elements such as family photos with fantasy world elements, or with elements from the past, present and future (B.5.9). They can import a photograph and use the object editor to make any part or parts of it transparent. Figure 9.4 shows an example where the author imported a photograph of her two daughters dancing. Everything in the background of original photograph was made transparent in the Object editor so that they seem to be dancing in the SBW scene.

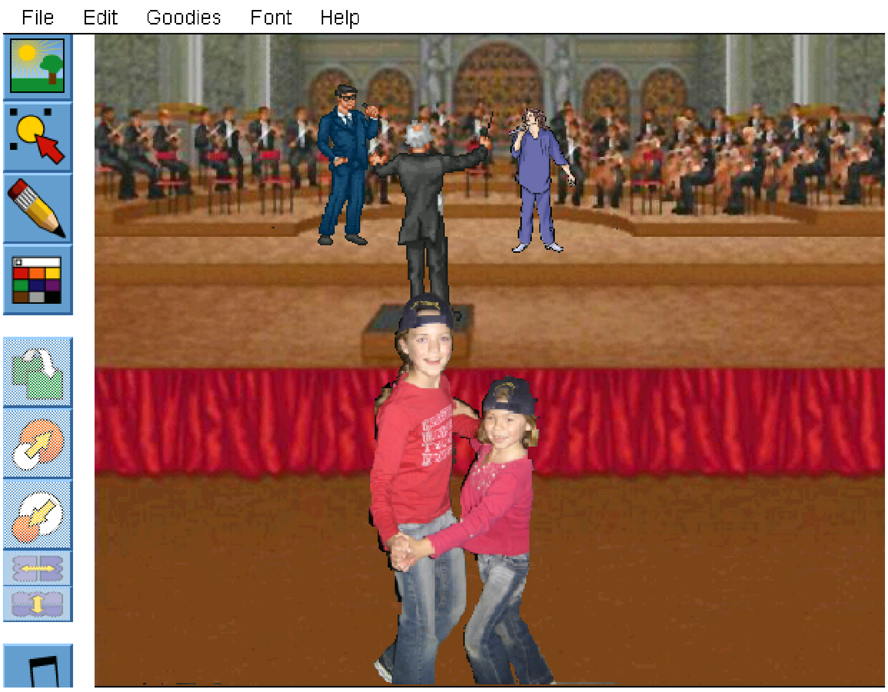


Figure 9.4 Importing a photograph into a story page

SBW provides opportunities for dramatic and creative play – although children do not physically carry out the actions, they create the characters and the story line and, through recording facilities, give the characters voice (B.5.10).

SBW does not include agent support (B.5.11). There is a danger that adding an agent may interfere with children's creativity, but and agent may be useful for the initial tutorial. Despite some obvious shortcomings, SBW works so well without an agent that adding one will not necessarily add to the usability of, and engagement with, the software.

9.3.3 Built-in Support

9.3.3.1 Support and Scaffolding

No user action is ever interpreted as an error in SBW and therefore there is no need for scaffolding that guides a user to correct an error (C.1.2). SBW offers support in the form of a quick tour and some story starters that can be accessed from the opening screen (although children will probably not interpret the icons that represent these correctly). Also, a TTS facility, a spell checker and a thesaurus can be accessed from the Goodies menu by children who can read (C.1.3). Children who cannot read will probably not need the latter two functions anyway.

No reflective or intrinsic scaffolding is obviously included in SBW (C.1.5). Intrinsic scaffolding is not applicable as children can decide for themselves how complex their stories will be. Children with fewer skills will naturally use less of the functionality, while older or more skilled users will use the more sophisticated functionality offered. Some reflective scaffolding could be useful in encouraging children to think about and plan their stories (C.1.5). SBW does not include any adaptive or adaptable support. Whether users are six or twelve, they all get the same functionality (C.1.6). SBW does not give enough support to pre-reading children. Audio cues attached to icons will help them, but users must have the option to switch these off.

There is no way in SBW to keep children focused on a theme or to keep them on track in the broader story construction process when they tend to get side-tracked (C.1.8). Ideally the software should detect when children are diverting from their topic and subtly guide them back on track. SBW provides sufficient intrinsic motivation so that praise is not required to keep children engaged. It may, however, be a good idea to reward children in some way if their stories fulfil some basic requirements (C.1.8).

SBW has a positive effect on children’s self-esteem in the sense that they can, on their own and without much effort, create something that looks like a story book (C.1.10). It is, however, impossible for a child of eight and younger to discover and understand all the functionality without the help of an adult or older assistant who can read the accompanying manual (C.1.11).

9.3.3.2 Feedback

Other than the TTS facility, SBW does not provide any audio feedback (C.2.1, C.2.15). The TTS is useful in that it allows users to listen to their story and hear what it will sound like to another listener. In general, the SBW interface relies to a large extent on reading ability – many functions are only available through the menu and there are no audio cues attached to buttons and menu items (C.2.1, C.2.10).

Feedback during story creation is sufficient (C.2.2). Children add objects or backgrounds with immediate, visible effect (C.2.4). There is a problem though with feedback on basic functions such as exporting a story as a Web document. A wordy text message appears telling the user where the web document has been saved. Children who cannot yet read may interpret this as an error message (C.2.2). Simultaneous audio and text cues can make a system accessible to reading as well as pre-reading children.

When using the recording function, SBW shows a graph of the sound wave (see Figure 9.6). The form of the wave tells the child when something has been recorded, even if they cannot hear it for some reason (C.2.8). However, SBW falls short in conveying state information with regard to the story as a whole (C.2.9).

SBW has limited and optional TTS feedback that children can use if they are aware of the facility (C.2.9, C.2.10). The 'Adding sound' function may become irritating to non-users or in a classroom setting because children browse the sounds and try them all out before selecting one (C.2.9). In such circumstances it should be possible for a teacher to deactivate this function (although it is one that the children enjoy a lot). Recording of voices and sounds also requires a quiet environment. SBW is thus not ideal for use in a classroom, unless users use earphones and do not require the recording facility. It is ideal for home use where one, two or three children can, noisily, use it together (C.2.9).

SBW does not praise users. Praise is not suitable in this application because the activities are completely open-ended and no specific actions are expected. The system can therefore not judge how 'well' the child has performed (C.2.11, C.2.12, C.2.13, C.2.15).

9.3.3.3 Support through Interface Agents

As discussed earlier, SBW does not include agent support and the nature of the software and its success without the agent support suggests that including an agent will not necessarily improve interaction (C.3.2).

9.3.4 Collaborative Use of the Application

As has been observed (e.g. by Chimbo and Gelderblom [2008]), cultural differences do not necessarily influence collaboration when using SBW (D.1.2) and gender has a stronger effect. SBW naturally encourages collaboration (D.1.3). Children want to show their friends their stories and invariably the friends start making comments and suggestions [Chimbo, 2006]. The voice recording function is especially conducive to collaboration – different children represent different characters and they have to work together to record the voices when these characters speak [Chimbo, 2006; Le Roux, 2006].

SBW naturally requires children to co-construct stories if they are using it collaboratively (D.2.1). Giving different characters different voices by using the voices of different users in a story is obviously better than using one voice for all characters (D.2.3, D.2.4, D.2.5). Designers could have made children aware of this and encouraged them to work together.

9.3.5 Addressing the Diversity of Users

9.3.5.1 Identity Issues

A diverse user group will find SBW appealing. The available story characters, scenes and objects represent a good mix of cultural elements (E.1.1, E.1.2), but there is still room for improvement. For example, there are no characters with disabilities and there are some gender stereotypes (e.g. no male ballet dancers). More suggestions for improvement are made in the re-design discussed in section 9.4.

SBW is intended for a wide range of users and the open-ended nature of the software makes the content suitable for a wide range of ages, all genders, and different cultures and backgrounds (E.1.3, E.1.5, E.1.10). Storytelling is a universal skill that is practised in many contexts (e.g. play, school, home, therapy) and SBW is suitable for use in any of these contexts (E.1.4).

Stories created in SBW can model children's relationships with different people outside their homes (E.1.9). Users can, for example, import pictures of their teachers, idols and classmates and make up stories about them.

SBW allows children to express themselves in a variety of ways (E.1.11): they can use existing sound clips or record their own; they can use existing objects or import their own graphics or photos; they can edit objects; they can export a story as a web document; or they can print it.

9.3.5.2 Existing Knowledge and Experience

SBW has a variety of potential users – novices, pre-readers, readers, even teenagers. They can choose to use or ignore functions that are too simple or too complicated for them (E.2.1). SBW allows a complete novice to create a basic story very easily, but it also allows expert users to create complex stories with many different story elements (including voice recordings and imported graphics) (E.2.1, E.2.2). Users can skip the introductions and go directly to their favourite parts of the application (E.2.3).

Children can apply their real-world knowledge when choosing scenes, sounds and objects (E.2.4). The recording function uses icons that resemble the buttons on a recording device (E.2.4). A child who has never seen or used a recording device will probably not understand the metaphor (E.2.11).

When choosing a menu item, the mouse button must be held down while moving down an option list – this is not the standard way of using a menu (E.2.4). The Print, Font and Save dialogs are internally consistent with the appearance of other SBW functions, but they differ from the standard Windows dialogs. It is advisable to follow the standard format (E.2.4). I address this in the suggestion for re-design in section 9.4.

9.3.6 The Interaction Environments and Devices

9.3.6.1 Input and Output Devices

SBW uses the mouse as primary input device – all interaction happens through straightforward mouse clicks. The right mouse button has no special function, so it can just as well have the same functionality as the left one (F.1.1, F.1.3). The only other input device is a microphone used for voice recording.

In the object editor users can select part of an image using marquee-type selection. This can be replaced with circling the objects, but children do not seem to struggle with the marquee-type selection required in SBW as this is the standard way of doing it (F.1.2).

SBW icons are large enough – even five-year-old children do not seem to have a problem clicking on them [Chimbo, 2006; Le Roux, 2006] (F.1.4).

To move objects around on a page, users drag and drop them. This seems natural and users do not seem to prefer click-and-carry instead. This may be a case where there is a kinaesthetic connection between holding the mouse button down and holding on to the object to move it (F.1.6, F.1.7).

9.3.6.2 Speech Recognition and Speech Output

SBW's TTS voice is a monotone adult male voice (F.1.24) – no differentiation is possible with regard to the voices of different story characters.

SBW does not use any familiar cartoon characters so there is not a possibility that familiar characters have incorrect voices (F.1.25). To include familiar characters may have copyright issues that will complicate the re-design. No user comments have been recorded in this regard. If they have access to digital photos of such characters, users can easily import them into the story, but the voice issue is then not relevant.

9.3.6.3 User Control vs. System Control

SBW is completely open-ended and children can decide in all respects what to do (F.2.1). This application is proof of the fact that children are engaged by being in control (F.2.2). Despite being engaged, children often do not complete their stories or do not keep to the original story line (F.2.3). It would be a good idea to build in a way to judge the completeness or logical flow to keep the child focused.

No specific sequence of actions is required and SBW allows the child to go directly to any function they prefer to use (F.2.4).

9.3.6.4 Engagement

SBW makes good use of the computational capacity of the computer (graphics, voice recording, TTS) to enhance engagement (F.3.1). It could be used even better, for example, to record children's progress so that they can be given advice on narrative structure.

SBW shows some evidence of choices made by adults instead of children (F.3.3). The icons on the opening screen can be interpreted by adults or older children but not necessarily by younger children. I come back to this in section 9.4. If all functions were tested with child users the designers would, for example, have realised that the Colour function is problematic. It refers to Zone 1 and Zone 2 that mean nothing to even an adult user and the only way to find out how the system will respond to the user's choices is through trial and error (see Figure 9.5).

SBW fulfils many of the requirements for engagement:

- Interaction with SBW often evokes emotional responses – when children add sounds to pages and objects or successfully record their own voices it creates lots of excitement [Chimbo and Gelderblom, 2008] (F.2.4).
- Activities are easy to perform – even when children struggle to edit an object they remain relatively patient (F.3.5). When it becomes an impossible task they will discard the object and use something else instead.
- There is no set path for story creation (F.3.7). Children can progress according to their own preferences.
- In terms of adding behaviour to characters and objects, attaching sounds or voices provides lots of excitement. It would however be supportive of engagement to allow children to let characters and objects perform other kinds of behaviour as well (F.3.10).
- The process of constructing a story is clearly more important than the end product (F.3.11).

9.3.6.5 Interface Design

SBW has no tools for telling users which objects, characters or functions have been used previously (F.7.2). It also does not use any defaults to support interaction (F.7.3).

SBW has many icons that may be difficult to interpret (F.7.4, F.7.5, F.7.6, F.7.7). This will be discussed in detail in the suggestions for re-design below. Some of the problem icons are the arrow for 'Getting started', (see Figure 9.3), the 'Undo' icon in the object editor (Figure 9.12) and the traffic light to return to the main menu from the story page (Figure 9.2).

Use of the same icon for different purposes is obviously problematic (F.7.8). The green checkmark icon appears on most dialogues, but for different purposes. In the Colour dialogue (Figure 9.5), the user's changes take effect when the user clicks on the green checkmark icon, while on the Record dialogue (Figure 9.6) it closes the dialogue – if the plus icon was not first selected, the recording will be lost.

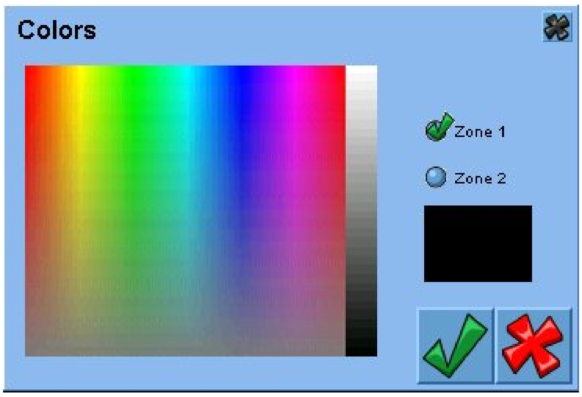


Figure 9.5 The Colour dialogue

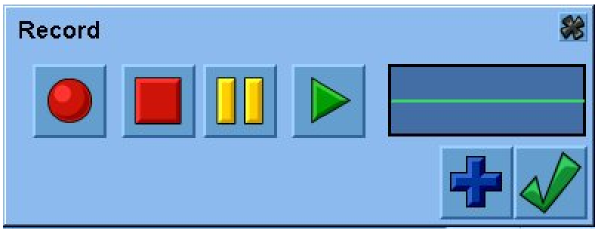


Figure 9.6 The Record dialogue

SBW provides children with a choice of fantasy in which to embed activities and they have many opportunities to choose names for characters and places (F.7.11).

9.3.6.6 Visibility

SBW does not consistently make it clear when functions are unavailable. Users can add objects to the title page and although these cannot be edited, the Object editor seems available (F.7.17). Also, some of the functions are only available through the menu and are thus not visible on the general story construction page. Children will not necessarily explore the menu options (especially if they cannot read) and will miss functions such as the spellchecker and the TTS facility (F.7.17, F.7.20). Many functions are only accessible to children who can read. When choosing objects the object categories and subcategories are presented in text only (F.7.20).

In SBW the next action is usually completely up to the user (F.7.18). One point in the application where the next action is not obvious is when selecting scenes or objects: the user has to click + to add an object and √ to close the window which can cause confusion about what to do (see Figure 9.7).

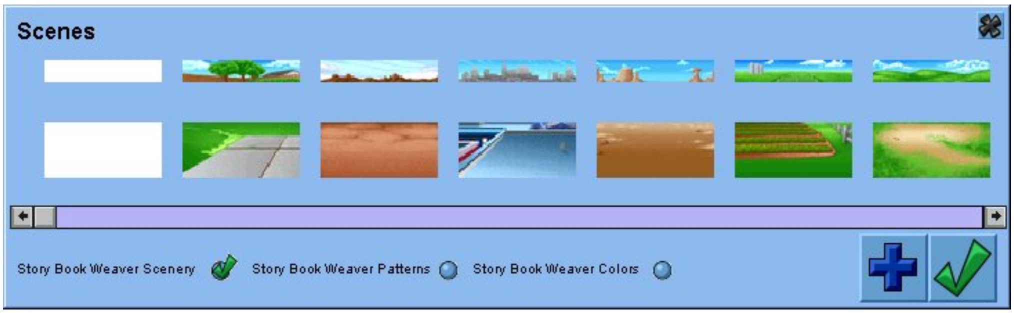


Figure 9.7 The Scene selector

SBW does not allow children to view their stories with more than one page on the screen. The only way to get an overview is to page through the story from beginning to end (F.7.19).

On the whole, SBW provides a transparent interface that enables children to focus on what must be done and not on how they should use it (F.7.21). The interface is very easy to use and learn, although children may initially need help to interpret the icons correctly or to get access to functions that are not made visible on the main screen. Weak readers (and pre-readers) will struggle more as some aspects require good reading abilities (e.g. using the menu, choosing categories of objects and characters). On the Colour dialogue the Zone 1 and Zone 2 options are completely meaningless and children can only discover through trial and error what these mean (F.7.21).

9.3.6.7 Consistency and Familiarity

The evaluation revealed the following inconsistencies:

- Inconsistent use of the x, + and √ buttons (F.7.22).
- The Save, Font and Colour dialogues differ substantially from the Windows standards (F.7.22, F.7.23). If designers want to give them a SBW look, they should still make their appearance and operation comparable with the standard.
- Screens have no title bars with minimise, maximise or close buttons as generally appear in the top right hand corner of a window. This makes it difficult to go to a different application without closing SBW first.

The appearance of objects in SBW does not always reflect their function or behaviour (F.7.24). Also, not all icons have a clear connection with their functions (e.g. those on the opening page). SBW makes use of the paper-based book metaphor in some of its icons, but it does not convey the message clearly enough (F.7.25). The connection can be made clearer through the use of more appropriate graphics or animated icons on the opening page.

The voice recording function uses the metaphor of a manual recorder successfully – the buttons for record, replay, pause, et cetera, follow the standard on most recording devices (F.7.25).

9.3.6.8 Error Prevention

SBW follows the principle of avoiding errors rather than handling them (F.7.28, F.7.31). Children can easily reverse their actions by, for example, removing and object that they have placed on the page (F.7.29). The object editor has an Undo button, but it is not easily interpretable as that. It is also not obvious how to remove a sound attached to an object or page – this can only be done through the menu. When overwriting an existing file or exiting without saving, SBW requires the user to confirm the action, but the warnings should also appear in audio format (F.7.31).

SBW does not always disable buttons that represent functions that should not be available (e.g. editing objects placed on the title page) (F.7.30).

9.3.6.9 Flexibility and Adaptability

SBW allows different kinds of input and output (F.7.33). Users of all ages and abilities can use the graphic part of the story pages to create the story illustrations. Those who can write and type can also provide the story text, but this is optional. Once children have learnt to use the recording function they can attach voice recordings to their story pages. With regard to output, SBW allows children to page through their stories in SBW, they can print the stories in story book format or they can export them as web documents.

Other than the fact that children can use whatever they feel comfortable with and ignore the rest, the interface is not adaptable or adaptive to children's individual needs (F.7.34). The Preferences function is very limited. It only gives a choice between two languages (Spanish and English) and it lets the child pick a layout for the printed version of their story (should they choose to print it). Users can also indicate whether objects must be brought to the front when they are moved on a page. It is not clear at all what the user is setting when selecting between the print options because there is no indication that this has to do with printing out the story (see Figure 9.8). This is only explained in the user guide [Kirchoff and Horne, 2003] that accompanies the software. There are many things that could potentially be included on the preferences dialogue, for example, a choice between different sounding TTS voices and whether audio cues should be switched on or off. I address this in the suggestions for re-design.

Figure 9.8 The Preferences dialogue

9.3.6.10 General Interface Guidelines

SBW does not focus on productivity or efficiency (F.7.35). It is safe in the sense that it does not expose children to violence or pornography or any other harmful material and it holds no psychological threat (F.7.36). Children may want to play for extended periods so it is advisable that a timer is built into the system to suggest to a child when it is time to take a break (F.7.36).

With regard to utility, SBW could provide children with more functionality (F.7.37) (e.g. allowing them to turn an object around to face backwards or to the front). SBW only shows characters and objects from the front or sometimes the side and children can flip them to face left or right or to be upside down or upright. They cannot change the rotation three-dimensionally.

Despite many obvious usability problems, SBW is a successful and engaging application. One should be careful when adding functionality or support facilities not to compromise the success of the application. The fact that it is so open-ended and all control lies with the user contributes a lot to the user's experience (F.7.38).

9.4 Suggestions for Re-designing Storybook Weaver Deluxe 2004

Following the proposed set of guidelines and addressing the results of the evaluation discussed above, I will now make suggestions for re-designing parts of the application. This partial re-design will be sufficient to demonstrate that the proposed set of guidelines can facilitate the design process. Whereas the evaluation was organised according to the main categories of guidelines, the re-design will consider the key screen designs of the application.

9.4.1 Main Menu (Opening Page)

Currently the opening page appears as in Figure 9.3. The icons initially appear without text. When a child moves over them with the mouse pointer, the function names appear.

With regard to the icons, the proposed guidelines suggest the following:

- Always display the name of the function with the icon. This will help users who can read to interpret the icons correctly (F.7.4, F.7.33). The name (or text) associated with an icon should be stored separately from the graphic part of the icon so that it can easily be adapted to different languages or for visually impaired users (B.4.7, E.3.8).
- Make the icons more meaningful (F.7.5). A child's head with the light bulb in a thought bubble will more clearly convey that this leads to story starters or an ideas bank.
- If the mouse pointer moves over the mouse, activate an audio cue that says what function the icon represents (C.2.3, F.7.33), but make it possible to switch the audio cues off.
- Using animated icons that come alive when the mouse moves over them will also help the user to interpret them correctly (F.7.5, F.7.7, E.2.5).
- The arrow icon for the help facility has no connection with the function it represents – a question mark would be more suitable (F.7.5, F.7.8).

Figure 9.9 represents a re-designed main menu (following guidelines F.7.4 to F.7.8). The arrow icon for the 'Getting started' and 'Story starters' functions are replaced with a single 'Help' button which will take the user to a screen where they can choose between the 'Quick tour' of SBW or an 'Ideas bank'. The 'Ideas bank' includes story starters, story ideas, incomplete stories that children can complete, story endings, feelings and emotions, examples of happy, sad, scary and exciting events, and so on (B.5.3, B.5.4, B.5.6, B.5.7). Children like stories about their own fears and overcoming them, so the ideas bank will include themes about fear of being alone, getting lost and losing a parent (A.1.5). Incomplete stories that already include a beginning, middle and end will help to teach children about narrative structure and can compensate for their lack of metacognitive knowledge (required when choosing between different options) (A.4.5, B.2.6).

To bring in humour, SBW can suggest storylines where, for example, boys wear dresses or where adults call children 'sir' and 'madam', or where the role of children and teachers are reversed (F.3.12, F.7.12, F.7.13). It should support unusual combinations of scenes and characters, encourage the use of imagination, challenge stereotypes and change power relations (B.5.6).

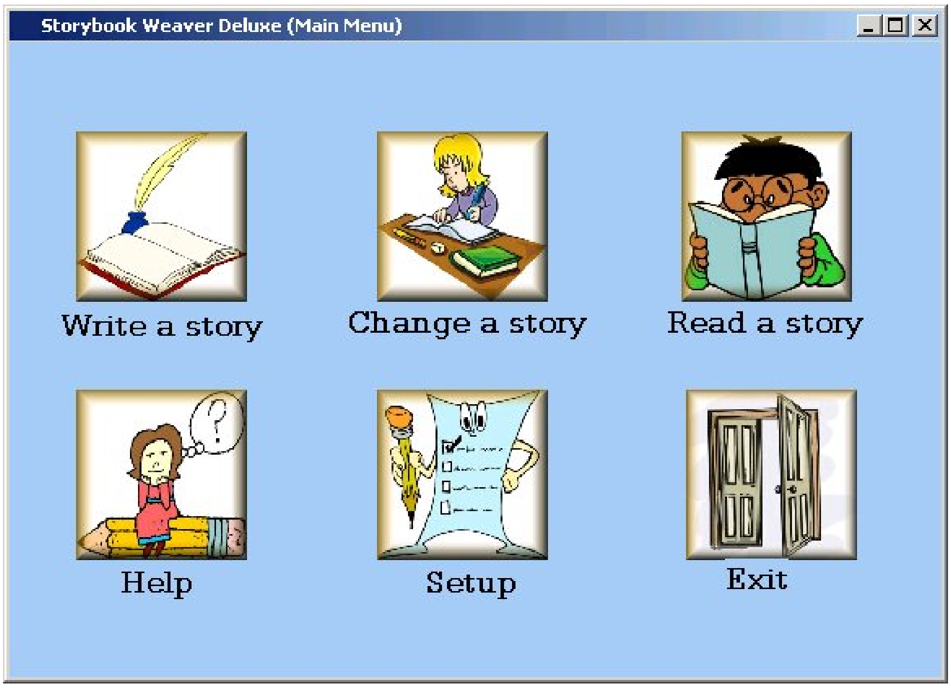


Figure 9.9 The new Main Menu dialogue

A Setup function was also added. This replaces the Preferences function which could only be accessed from the SBW's File menu and allows users to adapt the software according to their needs (F.7.34). To make SBW suitable for classroom use, teachers should be able to customise activities according to their own needs as well as to those of the learners (E.2.12). The Setup function should thus allow users or facilitators to do the following:

- Switch audio cues on or off (A.4.5, C.2.1, C.2.3, C.2.9, E.3.7).
- Change the default TTS voice or the voice used for audio cues (children should be allowed to listen to the available voices and select one) (E.1.10, F.1.21).

- Choose a language or dialect (B.4.7).
- Deactivate the TTS function or the Recording function if the use of sound will be problematic in the environment where the software will be used (C.2.9, E.2.12).

Knowing the user will help in setting up SBW according to their needs. When a child uses it for the first time the application can ask them for the following information: name, age, gender, nationality and language (E.1.2, E.1.3, E.1.4). Thereafter the child must click on his or her name on entering the program or indicate that he or she is a new user. New users are asked for the same information. This information can be used to set default values. For children of seven and younger audio cues can be switched on and for older children they can be switched off. If the user is a six-year-old girl the default TTS voice can be that of a young girl (F.1.21). When the user starts a new story, his name can appear in the Author field by default (F.7.3).

Children prefer advice from peers to that from adults with regard to computer use, therefore the voice used in the quick tour (now part of the Help facility) should be replaced with that of a child (D.1.4).

The system can also use information obtained from the operating system to set defaults (F.7.3). If there is no microphone installed, the Recording function can be made unavailable. If there are no speakers attached to the computer, the text-to-speech facility should not be available. If no printer is available, deactivate the Print function.

The ‘Change a story’, ‘Read a story’ and ‘Exit’ buttons have also been replaced with more meaningful graphics.

9.4.2 The Story Construction Page

The story construction page is explained in Figure 9.2. Once children have learnt what the icons mean they can use them with ease [Chimbo, 2006; Le Roux, 2006]. Some of the icons can, however, be improved to convey their functions better, especially the ‘Return to main menu’ button (F.7.4, F.7.8). All the buttons can be made more understandable by adding audio cues and text hints (C.2.3, F.7.33). Currently, if the user selects a button, an explanation of that function appears in the area (to the right of the ‘Record’ button) that is physically removed from the buttons. It will be more effective if this explanation appears as a hint directly attached to the button when the mouse pointer moves over it (F.7.8). Figure 9.11 illustrates this. For pre-readers an audio cue can be played simultaneously (C.2.1, C.2.3).

To expand the application’s functionality and to the make the available functions and support that are hidden in the menu more accessible (A.4.2, C.1.3, F.7.20, F.7.21), the area to the right of the Record button can be used for the following additional buttons:

- A button to switch audio cues on and off (A.4.5, C.1.9, C.2.1, C.2.3, C.2.10).
- A button to activate the TTS facility (B.4.4, B.4.6).

- A button for spell checking (B.4.3).
- A button to view the story in the form of a slide show (B.2.12, B.5.8, C.2.8).
- A button to reach the ideas bank from this page (using the same icon as on the main menu for consistency) (F.7.23).

Adding the buttons for functions currently only available through the menu will make actions and options visible so that users need not remember where to find them. This will help to free up processing and storage capacity in working memory (A.4.2). The availability of the slide show option will encourage children to play back their stories, thereby supporting reflection on their choices and thinking about possible changes (B.2.12).

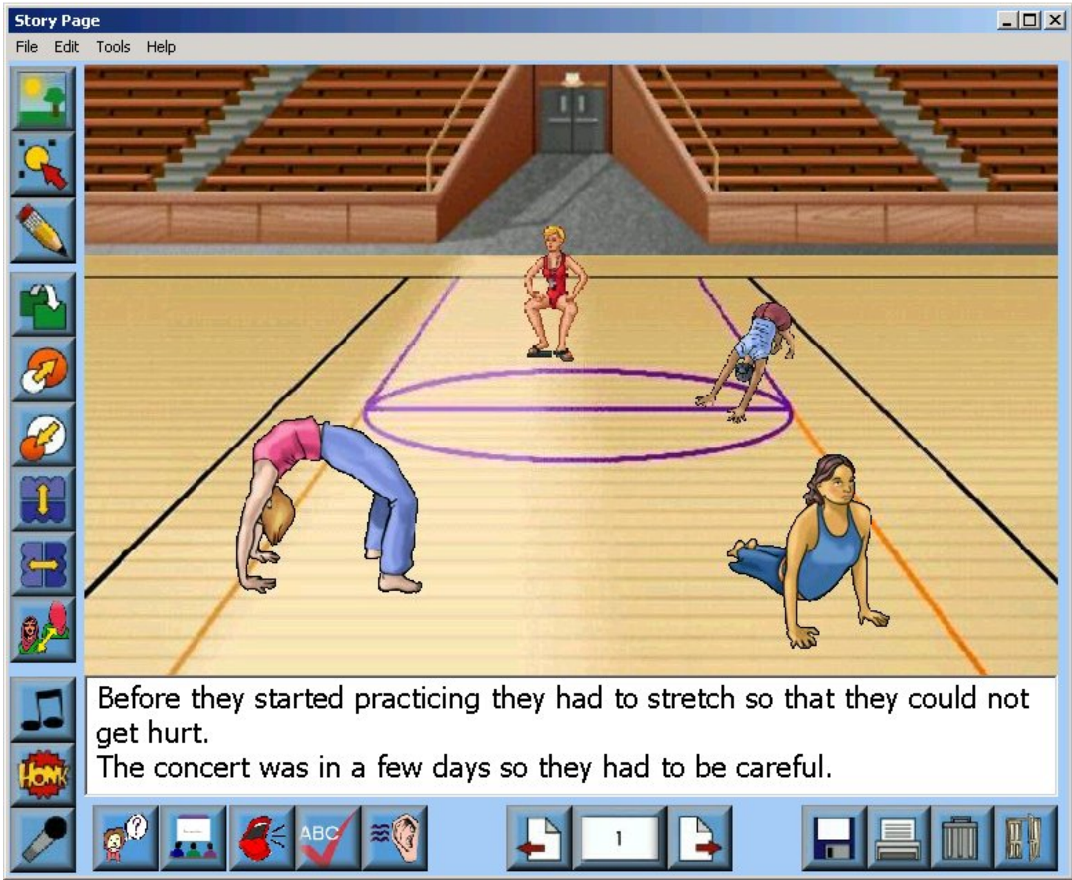


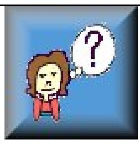
Figure 9.10 The re-designed story page

The Flip buttons currently allow vertical and horizontal flipping, but there is no way to flip an object so that it can be seen from behind or from the top or bottom (F.7.37). Children's perspective taking skills can be improved by providing them with opportunities to imagine physical spaces from different points of view and to compare different states of the world by presenting them with three-dimensional images that they can manipulate and rotate with the mouse (B.2.17). The current version of SBW does not require a lot of processing power. Making it possible for children to directly manipulate objects with the mouse in order to rotate them in any possible way will require a more sophisticated platform that will increase the system requirements substantially, thereby making the product less accessible to children who do not have access to powerful computers. My suggestion is therefore to only add a back-to-front flip function (E.1.2). This means that back views of all objects should be available and programmatically attached to the actual object in the object selector. Figure 9.10 shows a re-designed story page where the above aspects are included.

Figure 9.11 A hint attached to a button

Table 9.1 explains the newly added or modified buttons and their functionality.

Table 9.1 Explanation of new and modified buttons

Button	Explanation
 	Clicking on these buttons toggles between switching audio cues on or off.
	This button activates the spell checker. If a spelling error is found alternatives are given and the user can choose one with which to replace the misspelled word.
	This button activates the text-to speech facility.
	This presents the currently open story in the form of a continuous slide show.
	This takes the user to the ideas bank.
	This button is only available when an object is selected. Clicking on it turns the object around to face to the back or front.
	This button replaces the traffic light as the Exit button and is similar to the Exit button on the main menu.

The Colour button was removed as this function logically belongs in the Object editor.

The ‘Adding sound’ function may be problematic to non-users or in a classroom setting because children browse the sounds and try them all out before selecting one. Teachers or parents will now be able to de-activate this function using the Setup facility on the main menu (C.2.9).

SBW currently lacks a mechanism whereby users can evaluate their own stories (A.5.4). When a user has saved a story, the system can ask whether he or she wants to check how good the story is. They can then be presented with some simple yes/no questions designed to make them think about possible changes (B.2.11). While doing this the story can be displayed in multiple page format so that they have an overview on which to base their answers. Examples of suitable questions are:

- Does your story have a title?
- Is the whole story about that?
- Are there any characters that do not really do anything in the story?
- Does anything exciting or scary happen in your story? Or does someone have a big problem?
- Is the problem solved in the end?

If they answer ‘No’ to some of these the system can make suggestions about adding a page to accommodate that, or removing some unnecessary objects, and so on (B.2.15, B.5.4, C.1.8).

9.4.3 Object Selector

When a user clicks on the Object selector button, a form appears on top of the story page (as in Figure 9.12). To add an object the child selects a category from the left hand list and a corresponding subcategory from the right hand list. Images of all the available objects in that subcategory appear in the horizontally scrollable top panel. To add an object on the story page the user selects an object (as indicated by a blue frame) and then clicks on the + icon. They can add any number of objects in this way. To close the Object selector they click on the green tick mark button or on the cross button in the top right hand corner. Pre-reading children can only find objects by randomly selecting categories and subcategories until they see something they want.

Figure 9.12 The current Object selector

This facility can be improved as follows:

- Remove the checkmark button or replace it with red cross button for consistency (F.7.22, F.7.23, F.7.24).
- Re-design the categorisation of story objects to make it comprehensible to pre-readers, both in terms of layout and by including pictures (F.7.19, F.7.20, F.7.21).
- Address gender stereotypes (e.g. also include male ballet dancers) (A.1.4, E.1.1, E.1.10).
- Include possibilities for grown-up actions such as getting married or being president (A.2.4).
- Make the import of photos function more accessible and include a category of own objects where all previously imported graphics are kept. Add a button for importing graphics so that users do not need to do this from the menu (B.5.9).
- Make sure all races and nationalities are equally represented (E.1.1, F.7.37). In the Object editor children can be allowed to change the skin colour of any character. This will reduce the required number of objects in the Object selector. Donkey carts need not appear in the 'historical' subsection of the vehicle 'section' as some children's families who live in rural areas may still use that as primary vehicle. In South Africa all children wear school uniforms. In the School category, South African users will expect to find children with different kinds of school uniforms.
- Include people with disabilities and their supporting equipment (e.g. a wheelchair) in the choice of characters (E.1.1, F.7.37).
- Include a category for frequently used objects where children can get quick access to recently or often used objects (F.2.4, F.7.2).

The re-design in Figure 9.13 shows how most of the above can be addressed:

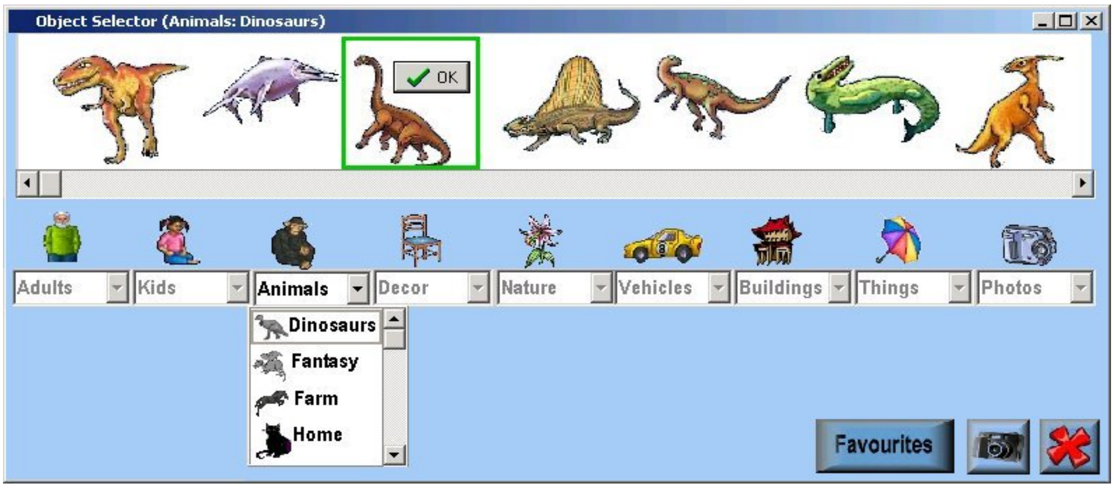


Figure 9.13 The re-designed Object selector

When a child now selects an object, an OK button appears next to it (F.7.17). Clicking on the OK button will place the object on the story page. If the user does not want to add that object she can just select another one. A child could also be allowed to add an object quickly by double-clicking on it (B.3.8, E.2.3). The addition of graphic icons at all levels makes the categories and sub categories accessible to pre-readers (A.1.4, B.3.8).

A Photos category is added where all photos that were imported previously are kept. When selecting this category, thumbnails of those photos appear in the selection bar (F.7.23). The Favourites button displays a set of the most often selected objects (F.2.4) and the Camera button represents the Import function. When a photo is imported it immediately becomes part of the Photos collection.

9.4.4 Scene Selector

The Scene selector is shown in Figure 9.7. Here the green checkmark icon should be replaced by a red cross (as explained above). Overall this function works well. Children can select the backgrounds and foregrounds in different combinations of scene, colour or pattern. Generally, children prefer to use the background-foreground pairs as given.

Designers should think outside of their own context when compiling the available scenes (E.1.12). Examples of scenes that can be added to the available collection are a music recording studio, a film studio, doctor's rooms and different styles of homes (e.g. a crowded informal settlement) (E.1.1, E.1.2, E.1.6, E.1.7). There should also be scenarios (and characters) that will encourage stories about social values and pro-social behaviour (A.5.7).

9.4.5 Object Editor

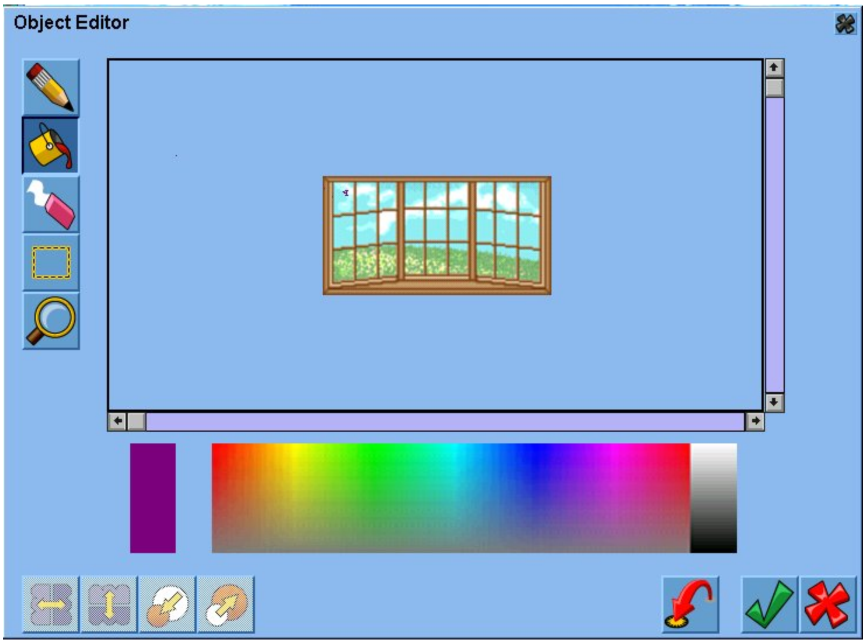


Figure 9.14 The Object editor

When an object is selected on the story page the user can activate the Object editor to modify that object. It opens up with the selected object in the editing pane (see Figure 9.14). Users can change the object by erasing parts of it, by using the pencil tool to draw on it, or changing the colour of selected parts of it. The most recent action can be undone with the 'Undo' button that appears to the left of the green checkmark. When clicking the green checkmark the changes are accepted and the user returns to the story page where the changes will be reflected. Clicking on the red cross will discard the changes and return to the story page.

The following changes to the Object editor are suggested:

- When different characters speak, children should ideally be able to choose a TTS voice for each different character. Include a 'Choose voice' button for attaching a specific voice to a character. This will then be that character's TTS voice when the story is played back. The choice should include male, female, adult, child, US or British pronunciation, et cetera (E.1.1, E.1.2, E.1.10, E.1.11, F.1.24, F.7.37).
- Story characters can be made representative of all races by allowing users to change their skin colour. SBW should detect when the selected object is a person and then make a 'Change skin colour' function available with which the skin tone can be made progressively darker or lighter (E.1.1).
- Make the Colour function part of the Object editor as this is where it logically belongs (F.7.27).
- Allow users to undo more than the last action and also add a redo button. The audio cues for these should explain clearly what their purposes are (F.7.29).

9.4.6 Preferences

The Preferences function of SBW (Figure 9.8) is very limited. It only gives a choice between two languages and it lets the child pick a format for the printed version of their story. As I have said before (section 9.3.6.9), it is not clear what the user is setting when selecting between the print options because there is no indication that this choice has to do with printing. This is only explained in the user guide that accompanies the software (not in line with C.1.11). The selection of the print format does not belong here and should be moved to the Print dialogue (see Figure 9.15) (F.7.23).

It is also unnecessary to indicate here whether objects should be brought to the front if moved. There is a Move to the front button available on the story page when an object is selected.

Because users should be allowed to set the preferences when they start using the application, the ‘Preferences’ (or ‘Set up’) function should be available on the opening screen as described in section 9.4.1 above. It should, however, be possible to change the preferences at any time during interaction, therefore the Setup function should also be available on the menu.

9.4.7 Recording Function

The only change necessary on the recording function (Figure 9.6) is to replace the green tick button with a red cross to be consistent with the rest of the system (E.2.4, F.7.23). A child who has never seen or used a recording device will probably not understand the metaphor (E.2.11, F.7.24, F.7.25). This can be solved by including a Help button that will display simple instructions and simultaneously play an audio help message (if audio cues are switched on) (C.2.1, C.2.3).

9.4.8 General Interface Design Issues

When choosing a menu item the mouse button must be held down while moving down an option list – this should be changed to the standard way of using a menu (A.2.1, E.2.4, F.1.5, F.7.23). The Save, Print, Font and Colour dialogs are internally consistent with the appearance of other SBW functions, but they differ from the standard Windows dialogs. It would be advisable to make them resemble the standard format (E.2.4, F.7.22, F.7.23). Figures 9.15 and 9.16 show how they can be improved.

In the Print dialogue users can choose the number of storybook pages they want displayed per page. Further, it includes standard functions that should appear on a Print dialogue such as Properties (for selecting the paper size, colour quality, and so on).

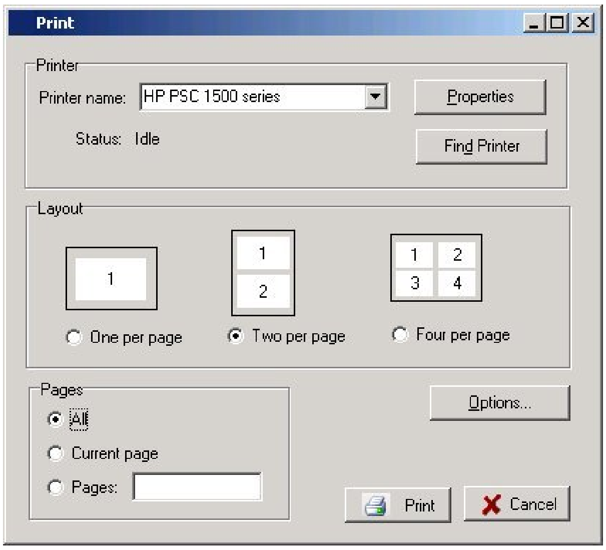


Figure 9.15 A suggested Print dialogue

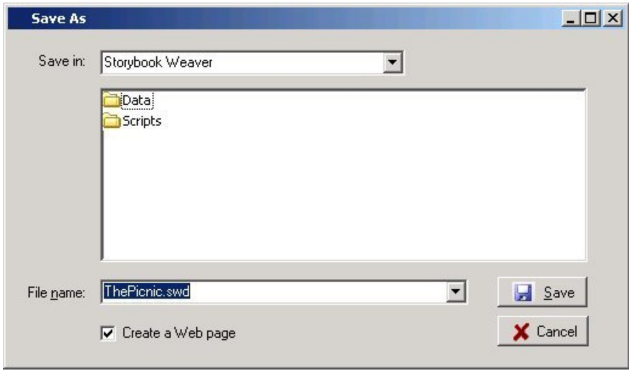


Figure 9.16 A suggested Save As dialogue

There is a problem with feedback on basic functions such as saving a story as a web document (C.2.2). A wordy text message appears telling the user where the web document has been saved (the user does not have the option to choose a name or location). Children who cannot yet read may interpret this as an error message. Simultaneous audio and text cues can solve this problem (C.2.1, C.2.3). The warning that a file may be overwritten when saving a story should also appear in audio format (but only if the audio cue option is turned on). I suggest that the option to publish the story as a Web page be included on the Save As dialogue. The story will be saved under the given File name in the selected folder and, if the Create a Web page check box is selected, a Web document with the same name will be saved in that same folder. The story title (as on the title page) should appear in the File name text box by default when the Save As dialogue opens up (F.7.3).

9.4.9 Suggestions for New Functionality

I have already mentioned some new functions that can be included, such as audio cues and choosing between voices for the TTS system. I end the suggestions for re-design with some brief suggestions for additional functionality without exploring how it can be implemented:

- Allow story characters to perform simple movements (e.g. jumping and dancing) (F.3.10).
- Provide users with a platform to plan a story in advance – this can be combined with an overview function to help them to think about the beginning, middle, climax and ending of their story (B.2.11, B.5.4).
- Build in some way to help children stay focussed – the system should notice when a child is aimlessly adding objects and pages and make suggestions about stories that involve those objects (B.2.15, B.5.4, C.1.8).
- The system could detect when children are diverting from their topic and subtly guide them back on track (B.5.4, C.1.8).
- It would be a good idea to build in a way to judge the completeness or logical flow of a story (F.2.3).

- Some non-dispositional praise when a child has completed a story could improve user experience (C.1.8, C.2.14).
- Include story examples and point out the different narrative elements to a child (e.g. beginning, climax and ending) (E.2.9).
- SBW does not teach specific concepts, but one can regard beginnings, climax, endings, sequencing, et cetera, as concepts related to storytelling. Include more explicit feedback on these concepts, but not at the cost of open-endedness (A.3.5).
- Draw a children's attention to objects or characters they never use (B.2.15, B.5.4, C.1.9).

9.5 Conclusion

In this chapter I used the framework of guidelines presented in Chapter 8 to evaluate and partially re-design Storybook Weaver Deluxe 2004. My aim was to test the usefulness of the guidelines. The evaluation report presented in section 9.3 demonstrates that systematically weighing up an application against the guidelines will elicit a large number of strong and weak points. The result of this evaluation brought to light many aspects of the system that could be improved and I have shown how a number of these can be re-designed in line with the proposed guidelines.

This demonstration highlights the practical contribution of my research. It shows that the proposed framework of guidelines for the design and evaluation of technology for children aged five to eight provides designers with the means to evaluate the developmental appropriateness, general usability and appeal of an existing application (or a prototype of a new product) aimed at young children. It offers a comprehensive collection of practical design guidelines to support designers in making design decisions and to stimulate new design ideas.

CHAPTER 10

Conclusion

Table of Contents

10.1 Introduction

In this last chapter I provide a brief overview of the research (section 10.2), state the conclusions (section 10.3) and summarise the contributions of the study (section 10.4). In section 10.5 I reflect on my research process using an assessment scheme suggested by Miles and Huberman [1994]. I end the chapter with suggestions for further research (section 10.6) and a short concluding remark (section 10.7).

10.2 Summary of the Research

The investigation into the possibility to develop a credible, dependable and useful set of guidelines for the design of technology for children aged five to eight years was guided by six research questions presented in section 1.3 of Chapter 1. None of these questions can be answered in one sentence or even one paragraph. The answers are rich descriptions based on the rigorous study of the relevant literature and each answer appeared as a separate chapter of this thesis. This summary then briefly explains the process whereby the questions were answered and provide pointers to the answers in the thesis.

Question 1: What are the cognitive and developmental characteristics of typical five to eight-year-old children with regard to skills relevant to the use of technology?
To answer this question I studied four prominent theories of children's cognitive development and four specific domains of development, namely literacy, mathematics, thought (including memory, knowledge representation and problem solving) and play. This literature study yielded a large set of potential design-related factors that, together, provide a comprehensive description of the relevant cognitive and developmental characteristics of children in the target age group. These statements appear in the data boxes of Chapter 4 and played and important part in the formulation of the proposed set of guidelines. The categories of guidelines on which this data had distinct bearing are category A (Guidelines that will ensure developmental appropriateness) and category B (Guidelines aimed at the development of specific skills).

Question 2: What can we learn from existing research into role of technology on skill development that can inform designers of technology for children aged five to eight?
The survey of the literature on the role of technology in the development of mathematics, literacy, storytelling and problem solving skills yielded numerous potential guidelines for the design of technology that support these skills. The preliminary guidelines (presented in the data boxes of Chapter 5) were processed and integrated into the set of guidelines to make up most of category B – Guidelines aimed at the development of specific skills – in the framework. Together, the guidelines emerging in Chapter 5 illustrate that computers and other technology provide unique opportunities for creating and learning things that traditional materials cannot.

Question 3: What does the literature on interaction environments for young children tell us in terms of the design of technology for five to eight-year-old children?

In answering this question I provided a comprehensive description of the range of interaction devices and environments available that are suitable for young children and formulated a large number of guidelines for the design of these. Potential guidelines for the design of interaction environments for young children appear in the data boxes of Chapter 6 and are integrated into the final set of guidelines, mainly as category F (Guidelines about the use and design of interaction environments and devices) but also in categories C, D and E.

Question 4a and 4b: What guidelines exist for the design of technology for children aged five to eight? Which existing guidelines not specifically aimed at the design of young children's technologies apply to technology for children aged five to eight?

Answering these questions involved: 1) an evaluation of the applicability of existing usability principles and design guidelines for adult products to technology aimed at children and 2) a critical investigation of existing guidelines for the design and evaluation of children's technology. The result is a systematic, case-by-case discussion of existing guidelines and an explanation of their applicability to technology aimed at young children. In the data boxes of Chapter 7 I pulled together the guidelines, design principles and usability principles that were found applicable, and organised them into lists of related potential guidelines. Together, these provide the answer to the stated questions.

Question 5: How can the guidelines emerging from the literature be organised into a structure that is useful for designers?

Analysing and synthesising the potential guidelines gathered in the data boxes of Chapters 4 to 7, I identified six broad categories and twenty-six subcategories in which to organise the resulting guidelines. Using these categories, a final set of 350 guidelines is presented in an integrated framework in such a way that designers and evaluators can easily identify those guidelines that apply to their applications. Each guideline in the framework is connected to the theory or literature from which it emerged through labels pointing to the corresponding factors in the data boxes.

Question 6: Is the proposed set of guidelines credible and useful?

The only way to answer this question honestly is by practically testing the credibility and usefulness of the guidelines. To do this I used the set of guidelines presented in Chapter 8 to evaluate and partially re-design a software application for children. The richness of the evaluation report (presented in Chapter 9) which was strictly based on the proposed set of guidelines alone, and the significance of the consequent re-design proves that the guidelines are indeed credible and useful.

10.3 Conclusions

While constructing the answers to each of the questions discussed above, I tested the sub-arguments which make up the thesis that was the basis for this investigation. Using the results I formulated the set of guidelines that now puts me in a position to make the following conclusions:

1. It is possible to derive credible, dependable and useful guidelines for the design of technology for children from psychological theories of children's development.
2. It is possible to derive credible, dependable and useful guidelines for the design of technology for children from existing research results on children's cognitive development.
3. It is possible to derive credible, dependable and useful guidelines for the design of technology for children from existing results on children's use of technology.
4. It is possible to derive credible, dependable and useful guidelines for the design of technology for children from existing design guidelines and usability principles.

With this I have provided the necessary evidence in favour of the thesis. The main conclusion of thus study is then: it is possible to develop a credible, dependable and useful set of guidelines for the design of technology for children aged five to eight years by studying 1) psychological theories of children's development, 2) existing research results on children's cognitive development, 3) existing results on children's use of technology and 4) existing design guidelines and usability principles.

The integrated framework that organises the resulting guidelines supports the following additional conclusion: It is possible to integrate design-related factors from different theoretical fields and research disciplines into one coherent and useful structure.

10.4 Summary of Contributions

I briefly discuss here what my research has contributed by way of new knowledge, what the implications of this are for research in the field of child-computer interaction as well as for designers of children's technology.

10.4.1 New Knowledge

The distinctive product of my research is a framework of guidelines for the design of technology for young children aged five to eight that assimilates a collection of guidelines from different knowledge domains. New research results on children's use of technology and the role of technology in their development are being published continuously in academic journals or conference proceedings. These usually report on empirical studies that focus on one aspect of children's interaction with technology. There is obviously value in this, but from a general design point of view there are usually many aspects to consider and few designers will take the trouble to search for and study all the academic publications that are relevant to their product. I have done here what few researchers in the field of designing for children have attempted, namely to draw together the results of a broad range of research topics such as children's cognitive development, acquisition of specific cognitive skills, the relationship between technology use and skill development, young children's use of technology and interaction environments for children.

When considering the existing guidelines for the design of young children's technology (most of which I discussed in section 7.3 and 7.4 of Chapter 7), my research adds in several respects:

1. The range of interaction environments addressed: The guidelines presented, for example, by Baumgarten [2003] and Gilutz and Nielsen [2002], focus only on children's use of the Internet and the WWW and Read et al.'s [2004] on handwriting interfaces. My guidelines cover a wide spectrum of interaction environments, including tangible interfaces, speech-based interfaces and normal desk top applications and they are organised so that designers that focus on a particular type of environment can easily find those guidelines that are applicable.
2. Age-specific guidelines: Many authors (for example, Malone [1982] and Shade [1996]) do not distinguish clearly between different age groups. Young children's abilities develop rapidly and it is essential that designers are informed about developmental differences between different age groups. Limiting the age group – as in my guidelines – provides guidance that supports the development of technology that is appropriate.
3. Depth of advice: Those who do differentiate between age groups (such as Grammenos and Stephanidis [2002], Fishel [2001] and Wyeth and Purchase [2003]) provide useful high-level advice but their guidelines lack depth. For example, instead of general advice such as 'avoid cumbersome input and output devices', I provide age-related advice on children's competencies and preferences with regard to a range of specific input and output devices (see subcategory F.1 in the framework) .
4. Theoretical grounding of age-related guidelines: No set of guidelines proposed by any of the authors cited in Chapter 7 involved a study of theories of children's development that is comparable to what I have done. By extracting design knowledge from these theories and incorporating it into the framework, I provide designers with an implicit developmental profile of the five to eight-year-old user. Following these guidelines will reduce the need for designers to consult psychological theory during the design process. Baumgarten's [2003] work is also well-grounded in psychological theory, but it does not explicitly link the guidelines to specific psychological theories. In my framework each guideline can be traced back to the theory or empirical results from which it emerged, giving the guidelines the necessary academic credibility.

Expert knowledge on the acquisition of specific cognitive skills (for example, mathematics, reading and writing) has been translated into knowledge that designers can use to create technology that supports the development of specific skills. From the existing body of knowledge on interaction devices and environments for young children I extracted aspects that can help designers to make decisions about such environments and incorporated these as design guidelines into the proposed framework.

From a scientific point of view, the most important contribution of my study lies in the integration of different fields of research, namely developmental psychology, young children and technology, interaction environments and design guidelines and principles.

With regard to the research process, I have demonstrated that a systematic, qualitative literature investigation can produce practical design guidelines that are grounded in respected theory or research results. In section 10.5 below, I reflect on this process in detail.

10.4.2 Theoretical Implications

Two of my initial assumptions were that 1) the knowledge base on child development and can help designers to understand young users and thereby help them in the design process, and 2) there is a large body of knowledge on technology for children that can provide insight into the design process. My point of departure was thus not to challenge existing theory but rather to find, in the existing theory, knowledge that are applicable to the design of technology for children aged five to eight. My re-interpretation and integration of the existing knowledge expands the theory on child-computer interaction by providing the most comprehensive set of guidelines for the design of young children's technology available thus far.

I also provide a foundation for further research into the design of specific topics in the field and offer a broad overview that will be useful for new researchers embarking on research in the field of technology for young children.

10.4.3 Practical Implications

The proposed guidelines are organised in a framework that makes them accessible to designers of children's technology and to researchers in the field of interaction design for children. Practical application of the results was demonstrated in Chapter 9 and Appendix 1. This demonstration served as proof that the results are both usable and useful for evaluation and design.

In terms of the design of new technologies the framework can be used as follows:

- To provide an overview of what children are capable of and what they expect from technology.
- To provide practical advice in the planning stages of design or when choosing between options during all the stages of design.
- To stimulate design ideas.
- To evaluate and develop prototypes.
- As a checklist for designers of technology for children aged five to eight, to ensure that their designs are developmentally appropriate.

The proposed framework of guidelines is not the only useful part of my research for designers. The abounding discussions of the theories of development and the rich descriptions of existing technologies will give readers a sound and substantial overview of the field of interaction design for young children.

With the work presented here I have succeeded in providing designers of technology for children aged five to eight, with a solid basis for design. The set of guidelines resulting from the research is accessible and practical, but, at the same time grounded in respectable theory or research results.

10.5 Evaluation of and Reflection on the Research Process

Miles and Huberman [1994] provide a set of questions whereby the confirmability, dependability, credibility, transferability and application of the results of qualitative research can be assessed. I have listed their complete set of questions in Appendix 2. Miles and Huberman state that these are merely suggestions and that some of the questions may not be applicable to the specific study being evaluated. Many of the questions relating to dependability, for example, apply specifically to field studies and observational research and not to a literature study. The assessment below was guided by those questions that are applicable to a qualitative literature investigation.

10.5.1 Confirmability

The question here is whether the study is replicable. This is only possible if the data collection and analysis methods are described in sufficient detail, giving a complete picture [Miles and Huberman, 1994]. Results must be explicitly linked to visible evidence in the data the researchers' assumptions and possible biases should be considered [Miles and Huberman, 1994].

The methods and procedures followed in this study are explained in detail in Chapter 3 and the background against which my research was conducted is fully described in Chapters 1 and 2. There is a visible trail through the thesis that shows how data were collected. All data appear explicitly in data boxes throughout Chapters 4 to 7 and the data boxes always follow directly on the discussion of the literature from which those data elements were derived. This is detailed enough to be followed as an audit trail. My progress from the literature to the initial data (data boxes) to the final result (proposed guidelines) can be followed with ease.

I demonstrate the process of coding, organising, analysing and transforming the data, and there is a clear description of the data as it appears at different stages of the analysis. Using a labelling system, each of the guidelines in the final set is explicitly linked to one or more data elements in the data boxes. All data gathered are included in the thesis for reanalysis.

All assumptions are clearly and explicitly stated in Chapter 1 (section 1.2.1).

10.5.2 Dependability

Dependability indicates how carefully the research is conducted [Miles and Huberman, 1994].

The proposal for my research was subjected to peer review and accepted for presentation as a poster at the 2004 Interaction Design and Children Conference [Gelderblom, 2004]. With this the worth of my planned effort was confirmed.

The research questions are stated clearly in Chapter 1 and the study was strictly guided by these questions (see section 10.2 above). Each of Chapters 4 to 7 addresses one of these questions. The research questions compelled me to collect data across a wide range of theories and research areas. Where contradictions arose in the data, this was further investigated and either resolved or discarded as potential guideline-generating data. The theoretical fields connected to the study and how the study relates to these are described in detail in Chapter 2.

10.5.3 Credibility

Credibility requires context-rich and meaningful descriptions [Miles and Huberman, 1994]. It is supported through links to existing theory and triangulation of research methods and data sources.

Throughout Chapters 4 to 7 I provide a rich and comprehensive account of the literature investigated. The way the information is laid out in those chapters clearly links the data in the data boxes with the theory or research results from which it emerged.

The nature of the study and the guiding thesis do necessitate triangulation of methods, but it naturally involved triangulation of data sources. The data sources included a diverse collection ranging from developmental theories to empirical results on children’s use of technology. Many of the resulting guidelines are based on a combination of data elements from different kinds of sources.

The emerging categories of guidelines are well linked to the related theory as well as to existing design frameworks or existing guidelines. Throughout, the researcher ensured that the thesis and the findings are coherent – all parts of the study are systematically related.

Areas of uncertainty or where further research may be necessary are discussed in section 10.6 below.

10.5.4 Transferability

Transferability refers to the extent to which the findings are applicable to other contexts [Miles and Huberman, 1994].

The study focused on technology for five to eight-year-olds and the intention was not to make the results transferable to technology for other user groups. The inclusion of four theories of development instead of one or two promotes generalisability within the age group. The choice of theories of development and other sources was fully justified in Chapter 3 (section 3.6.1.1). The chosen texts and theories were discussed in Chapters 4 to 7 in sufficient detail to allow comparison with other similar texts.

The complete research history is contained in this document.

10.5.5 Application

This refers to the practical applicability of the findings and their accessibility for potential users [Miles and Huberman, 1994]. I have already explained in section 10.4.3 above how the results of my research are applicable in practice.

10.6 Suggestions for Further Research

The set of guidelines that is the end product of my study is an ideal starting point for many potential studies. The groundwork has been one for developing a model for the design of young children's technology that will systematically guide designers through the process of planning, designing and development of technology for young children.

There is still a lot of scope for refinement of the proposed framework. Surveying the literature for results that may have bearing on the guidelines should be an ongoing process. Some areas of design were covered in less depth than others and could be studied further, for example, tangible and robotic interfaces and technology for children with disabilities. This was a very broad study. Each of the subfields covered by the literature investigation (that is, theories of development, mathematics skill development, literacy development, storytelling skills, tangible interfaces, existing guidelines, and more) is worthy of becoming the subject of and individual research project, to provide more depth.

Practical application of the guidelines and further testing against real experiences with children will increase their credibility.

The study of psychological theories and their potential contribution towards the design of technology can be extended by exploring theories other than the four chosen for my study. In my opinion Fischer's dynamic skills theory and its application to the design of technology in general, is worth investigating further.

In this study I focussed on the five-to-eight age group. Similar studies can be done for other age groups.

10.7 Concluding Remark

I have made it clear from the outset (section 1.2.2) that I do not believe that any product can be designed without user input and usability testing and that no set of guidelines alone can guarantee design success. User involvement in the design and evaluation process is imperative, but proper guidelines can reduce the required amount of usability testing and user involvement and can take designers far in the design process. I believe that the framework of guidelines that resulted from my research will provide functional and constructive support in the design of technology for children aged five to eight and that following these guidelines will contribute to the development of products that are developmentally appropriate.

REFERENCES

Ager, R. & Kendall, M. (2003) Getting it Right from the Start: A Case Study of the Development of a Foundation Stage Learning and ICT Strategy in Northamptonshire, UK. *Proceedings of the IFIP Working Group 3.5 Conference: Young Children and Learning Technologies, Parramatta, July 2003.* Australian Computer Society, Inc.

Alexander, K. & Strommen, E. (1998) Evolution of the Talking Dinosaur: The (Not So) Natural History of a New Interface for Children. *Proceedings of CHI 98.*

Alliance for Childhood (2000) Fool's Gold: A Critical Look at Computers in Childhood.

Alloway, N. (1994) Young Children's Preferred Option and Efficiency of Use of Input Devices. *Journal of Research on Computing in Education,* 27**,** 105-110.

Anderson, J. (2000) Developmentally Appropriate Practice, http://www.edenpr.k12.mn.us/ckc/develop1.htm, accessed 16 February.

Antle, A. (2007) Designing Tangibles for Children: What Designers Need to Know. *CHI 2007.* San Jose, California, USA, ACM Press.

Archambault, D. & Olivier, D. (2005) How to Make Games for Visually Impaired Children. *ACE 2005.* Valencia, Spain.

ARS Electronica (2006) QuiQui's Giant Bounce, http://www.aec.at/en/center/current_exhibition_detail.asp?iProjectID=13590, accessed 22 January 2008.

Attewell, P., Suazo-Garcia, B. & Battle, J. (2003) Computers and Young Children: Social Benefit or Social Problem? *Social Forces,* 82**,** 277-296.

Bartlett, B., Estivill-Castro, V. & Seymon, S. (2004) Dogs or Robots: Why do Children see them as Robotic Pets rather than Canine Machines? In Cockburn, A. (Ed.) *Conference on Research and Practice in Information Technology.*

Baumgarten, M. (2003) Kids and the Internet: A Developmental Summary. *ACM Computers in Entertainment,* 1**,** Article 2.

Bayon, V. (2008) Kidstory, http://www.virart.nott.ac.uk/idue/index.php?option=displaypage&Itemid=59&op=page&SubMenu=, accessed 17 January 2008.

Bekker, M. M., Markopoulos, P. & Kersten-Tsikalkina, M. (Eds.) (2002) *Interaction Design and Children,* Eindhoven, The Netherlands, Shaker Publishing.

Benford, S., Bederson, B., Akesson, K., Bayon, V., Druin, A., Hansson, P., Houcarde, J. P., Ingram, R., Neale, H., O'Malley, C., Simsarian, K. T., Stanton, D., Sundblad, Y. & Taxén, G. (2000) Designing Storytelling Technologies to Encourage Collaboration Between Young Children. *CHI 2000 Proceedings.* The Hague, Amsterdam.

Berk, L. E. (2000) *Child Development,* Boston, Allyn and Bacon.

Berkovitz, J. (1994) Graphical Interfaces for Young Children in a Software-based Mathematics Curriculum. *CHI 94.* Boston, Massachusetts, USA, ACM Press.

BigBrainz (2005) TimezAttack. BigBrainz, Inc.

Blakemore, S.-J. & Frith, U. (2000) The Implications of Recent Developments in Neuroscience for Research on Teaching and Learning, http://www.tlrp.org/pub/acadpub.html, accessed 18 November.

Boone, R., Higgins, K., Notari, A. & Stump, C. S. (1996) Hypermedia Pre-Reading Lessons: Learner-Centered Software for Kindergarten. *Journal of Computing in Childhood Education,* 7**,** 39-69.

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

Bracken, C. C. & Lombard, M. (2004) Social Presence and Children: Praise, Intrinsic Motivation, and Learning With Computers. *Journal of Communication***,** 22-37.

Brashear, H., Henderson, V., Park, K., Hamilton, H., Lee, S. & Starner, T. (2006) American Sign Language Recognition in Game Development for Deaf Children. *ASSETS '06.* Portland, Oregon, USA, ACM.

Broderbund (2005) Storybook Weaver Deluxe 2004. Riverdeep Interactive Learning Limited.

Brouwer-Janse, M. D., Suri, J. F., Yawitz, M., Vries, G. d., Fozard, J. L. & Coleman, R. (1997) User Interfaces for Young and Old. *Interactions***,** 34-46.

Bryant, P. (1993) Reading and Development. In Pratt, C. & Garton, A. F. (Eds.) *Systems of Representation in Children: Development and Use.* John Wiley and Sons, Ltd.

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

Byrne, B. & Fielding-Barnsley, R. (1991) Evaluation of a Program to Teach Phonemic Awareness to Young Children. *Journal of Educational Psychology,* 83**,** 451-455.

Carlson, A. D. (1996) Concept Books and Young Children. In Vandergrift, K. E. (Ed.) *Ways of Knowing: Literature and the Intellectual Life of Children.* The Scarecrow Press, Inc.

Carlson, S. L. & White, S. H. (1998) The Effectiveness of a Computer Program in Helping Kindergarten Students Learn the Concepts of Left and Right. *Journal of COmputing in Early Childhood Education,* 9**,** 133-147.

Carr, M. & Hettinger, H. (2003) Perspectives on Mathematics Strategy Development. In Royer, J. M. (Ed.) *Mathematical Cognition.* Information Age Publishing.

Carroll, J. M. (2003) Introduction: Toward a Multidisciplinary Science of Human-Computer Interaction. In Carrol, J. M. (Ed.) *HCI Models, Theories and Frameworks.* San Fransisco, Morgan Kaufman Publishers.

Case, R. (1992a) Advantages and Limitations of the Neo-Piagetian Position. In Case, R. (Ed.) *The Mind's Staircase.* Lawrence Erlbaum Associates, Publishers.

Case, R. (1992b) General and Specific Views of the Mind, its Structure, and its Development. In Case, R. (Ed.) *The Mind's Staircase.* Lawrence Erlbaum Associates, Publishers.

Case, R. (1992c) *The Mind's Staircase*, Lawrence Earlbaum Associated Publishers.

Case, R. & Okamoto, Y. (1996) *The Role of Central Conceptual Structures in the Development of Children's Thought,* Chicago, University of Chicago Press.

Casell, J. & Ryokai, K. (2001) Making Space for Voice: Technologies to Support Children's Fantasy and Storytelling. *Personal and Ubiquitous Computing,* 5**,** 169-190.

Chambers, B., Abrami, P., Slavin, R., Cheung, A. & Gifford, R. (2005) Computer-Assisted Tutoring in Success for All: Reading Outcomes for First-Graders.

Chimbo, B. (2006) Usability Evaluation of Storybook Weaver Deluxe 2004 Educational Software Among Culturally Diverse South African Children Between the Ages of Six and Eight Years (A research report in part fulfilment of the honours degree in Information Systems). University of South Africa.

Chimbo, B. & Gelderblom, H. (2008) The Suitability of American Storytelling Software for South African Diversity. *To appear in Proceedings of MCCSIS 2008.* Amsterdam.

Chomsky, N. (1986) *Knowledge of Language: Its Nature, Origin, and Use*, Greenwood Publishing Group.

Clements, D. & Nastasi, B. (1993) Electronic Media and Early Childhood Education. In Spodek, B. (Ed.) *Handbook on the Research on the Education of Young Children.* New York, Macmillan.

Clements, D. H. (1986) Effects of Logo and CAI environments on cognition and creativity. *Journal of Educational Psychology,* 78**,** 309-318.

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

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

Clements, D. H. & Battista, M. T. (2000) Designing Effective Software. In Kelly, A. E. & Lesh, R. A. (Eds.) *Handbook of research design in mathematics and science education.*

Clements, D. H. & Samara, J. (2002) Effects of a Preschool Mathematics Curriculum: Research on the NSF-funded *Building Blocks* Project, http://www.gse.buffalo.edu/org/buildingblocks/writings, accessed 22 Oct 2003.

Cole, M. & Scribner, S. (1978) Introduction. In Vygotsky, L. S. (Ed.) *Mind in Society.* Harvard\University Press.

Coleman, J. (1992) All Seriousness Aside: The Laugh-Learning Connection. *International Journal of Instructional Media,* 19**,** 269-276.

Control Centre (1999) The Control Centre, http://www.joy-stick.net/reviews/joysticks/precisio.htm, accessed 22 January 2008.

Cooper, B. & Brna, P. (2001) Fostering Cartoon-Style Creativity with Sensitive Agent Support in Tomorrow's Classroom. *Educational Technology & Society,* 4**,** 32-40.

Crook, C. (2000) computer Scaffolding? In Mcgurk, H. (Ed.) *Childhood Social Development: Contemporary Perspectives.* Lawrence Earlbaum Associates.

De Villiers, M. (2005) Three approaches as pillars for interpretive information systems research: development research, action research and grounded theory. *ACM International Conference Proceedings Series.* ACM.

Della Sala, S. & Logie, R. H. (2002) Neuropsychological Impairments of Visual and Spatial Working Memory. In Baddaley, A. D., Kopelman, M. D. & Wilson, B. A. (Eds.) *The Handbook of Memory Disorders.* Second ed., John Wiley & Sons, Ltd.

Demming, G. (2004) Sony EyeToy™: Developing Mental Models for 3-D Interaction in a 2-D Gaming Environment. In Masoodian Et Al. (Ed.) *Lecture Notes in Computer Science.* Springer-Verlag.

Dettori, G. & Lemut, E. (1995) External Representations in Arithmetic Problem Solving. In Sutherland, R. & Mason, J. (Eds.) *Exploiting Mental Imagery with Computers in Mathematics Education.* Berlin, Springer Verlag.

Dix, A., Finlay, J., Abowd, G. & Beal, R. (2004) *Human-Computer Interaction*, Pearson Education Ltd.

Donaldson, M. (1978) *Children's Minds,* London, Fontana.

Dreyfus, T. (1995) Imagery for Diagrams. In Sutherland, R. & Mason, J. (Eds.) *Exploiting Mental Imagery with Computers in Mathematics Education.* Berlin, Springer Verlag.

Druin, A. (1996) CHIKids: A Common Ground for Kids and Adults. In Tauber, M. J. (Ed.) *Common Ground: CHI 96 Conference Proceedings.* ACM.

Druin, A. (1999) Cooperative Enquiry: Developing New Technologies for Children with Children. *ACM CHI 99 Conference on Human Factors in Computing Systems.* ACM.

Druin, A., Bederson, B., Boltman, A., Miura, A., Knotts-Callahan, D. & Platt, M. (1999) Children as Our Technology Design Partners. In Druin, A. (Ed.) *The Design of Children's Technology.* San Fransisco, Morgan Kaufman.

Druin, A. & Solomon, C. (1996) *Designing Multimedia Environments for Children*, John Wiley & Sons, Inc.

Durkin, K. (1993) The Representation of Number in Infancy and Early Childhood. In Pratt, C. & Garton, A. F. (Eds.) *Systems of Representation in Children: Development and Use.* John Wiley and Sons, Ltd.

Durrheim, K. (1999) Research Design. In Terreblanche, M. & Durrheim, K. (Eds.) *Research in Practice: Applied Methods for the Social Sciences.* Cape Town, UCT Press.

Egloff, T. (2004) Edutainment: A Case Study of Interactive CD-ROM Playsets. *Computers in Entertainment,* 2**,** Article 4.

Fay, B. (1975) *Social Theory and Political Practice,* London, George Allen & Unwin Ltd.

Fisch, S. M. (2004) What's So "New" About "New Media"? Comparing Effective Features of Children's Educational Software, Television and Magazines. In Druin, A., Hourcade, J. P. & Kollet, S. (Eds.) *Building a Community. IDC 2004 Interaction Design and Children, Conference Proceedings.*

Fischer, K. & Bidell, T. R. (2006) Dynamic Development of Action and Thought. In Lerner, R. M. (Ed.) *Theoretical Models of Human Development.* Sixth ed., John Wiley & Sons, Inc.

Fischer, K. & Yan, Z. (2002) The Development of Dynamic Skill Theory. In Lewkowicz, D. J. & Lickliter, R. (Eds.) *Conceptions of Development: Lessons from the Laboratory.* Psychology Press.

Fischer, K. W. & Corrigan, R. (1981) A Skill Approach to Language Development. In Stark, R. (Ed.) *Language Behavior in Infancy and Early Childhood.* Amsterdam, Elsevier.

Fischer, K. W. & Granott, N. (1995) Beyond One-dimensional Change: Parallel, Concurrent, Socially Distributed Process in Learning and Development. *Human Development,* 38**,** 302-314.

Fischer, K. W. & Immordino-Yang, M. H. (2002) Cognitive Development and Education: From Dynamic General Structure to Specifc Learning and Teaching. Harvard Graduate School of Education.

Fischer, K. W. & Silvern, L. (1996) Stages and Individual Differences in Cognitive Development. *Annual Review of Psychology,* 36**,** 613-648.

Fishel, C. (2001) *Designing for Children*, Rockport Publishers, Inc.

Fogg, B., Cypher, A., Druin, A., Friedman, B. & Strommen, E. (1999) Is ActiMates Barney Ethical? The Potential Good, Bad, and Ugly of Interactive Plush Toys. *CHI 99.* ACM Press.

Frei, P., Su, V., Mikhak, B. & Ishii, H. (2000) Designing a New Class of Computational Toys. *Proceedings of CHI 2000.* The Hague, Amsterdam.

Fujita, M., Kitano, H. & Doi, T. T. (2000) Robot Entertainment. In Druin, A. & Hendler, J. (Eds.) *Robots for Kids: Exploring New Technologies for Learning.* Morgan Kaufman.

Gale (2001) Cognitive Development. *Gale Encyclopedia of Psychology.* 2nd ed., Gale Group.

Game Vortex (2008) EyeToy Play 2, www.psillustrated.com/gamevortex/soft_rev.php/2686, accessed 22 January 2008.

Gardner, H. (1983) *Frames of Mind: the Theory of Multiple Intelligences,* New York, Basic Books.

Gardner, H. (1991) *The Unschooled Mind: How Children Think and How Schools Should Teach*, BasicBooks (A division of HarperCollins publishers).

Gathercole, S. E. (2002) Memory Development During the Childhood Years. In Baddaley, A. D., Kopelman, M. D. & Wilson, B. A. (Eds.) *The Handbook of Memory Disorders.* Second ed., John Wiley & Sons, Ltd.

Gelderblom, H. (2004) Designing Software for Young Children: Theoretically Grounded Guidelines. In Druin, A., Hourcade, J. P. & Kollet, S. (Eds.) *Building a Community. IDC 2004 Interaction Design and Children, Conference Proceedings.*

Gilutz, S. & Nielsen, J. (2002) Usability of Websites for Children, www.nngroup.com, accessed 18 November.

Ginsburg, H. & Opper, S. (1969) *Piaget's Theory of Intellectual Development: An Introduction*, Prentice Hall.

Grammenos, D. & Stephanidis, C. (2002) Interaction Design of a Collaborative Application for Children. In Bekker, M., Markopoulos, P. & Kerstin-Tsikalkina, M. (Eds.) *Interaction Design and Children.* Eindhoven, The Netherlands, Shaker Publishing.

Graue, E. & Walsh, D. J. (1998) *Studying Children in Context,* Thousand Oaks, Sage Publications.

Greene, S. & Hill, M. (2005) Researching Children's Experience: Methods and Methodological Issues. In Greene, S. & Hogan, D. (Eds.) *Researching Children's Experience: Approaches and Methods.* London, SAGE Publications Ltd.

Griffin, S. (2003) The Development of Math Competence in the Preschool and Early School Years. In Royer, J. M. (Ed.) *Mathematical Cognition.* Information Age Publishing.

Grover, S. (1986) A Field Study of the Use of Cognitive-Developmental Principles in Microcomputer Design for Young Children. *Journal of Educational Research,* 79**,** 325-332.

GVU Center (2005) Gesture-based American Sign Language (ASL) Game for Deaf Children, http://gvu.cc.gatech.edu/ccg/projects/copycat/, accessed 23 January 2008.

Hanna, L., Risden, K. & Alexander, K. J. (1997) Guidelines for Usability Testing with Children. *Interactions***,** 9-14.

Hartley, R. (1986) Imagine You're Clever. *Journal of CHild Psychology and Psychiatry and Allied Disciplines,* 27**,** 383-398.

Harvey, B. (Accessed 23 Oct 2007) Harmful to Children? The *Alliance for Childhood* Report, http://www.cs.berkeley.edu/~bh/alliance.html, accessed 2007-10-23.

Haugland, S. W. (2000) Early Childhood Classrooms in the 21st Century. Using Computers to Maximize Learning. *Young Children,* 55**,** 12-18.

Haugland, S. W. & Shade, D. D. (1988) Developmentally Appropriate Software for Young Children. *Young Children,* 43**,** 37-43.

Haugland, S. W. & Wright, J. (1997) *Young Children and Technology: a World of Discovery,* Allyn and Bacon New York.

HCILab (2008) KidPad, www.cs.umd.edu/hcil/kiddesign/kidpad.shtml, accessed 17 January 2008.

Henderson, V., Lee, S., Brashear, H., Hamilton, H., Starner, T. & Hamilton, S. (2005) Development of an Americam Sign Language Game for Deaf Children. *Interaction Design and Children.* Boulder, Colorado.

Hoenderdos, R., Vermeeren, A., Bekker, M. & Pierik, A. (2002) Design for Experience: the "Look, Mama!" Experience. In Bekker, M., Markopoulos, P. & Kersten-Tsikalkina, M. (Eds.) *Interaction Design and Children.* Eindhoven, The Netherlands, Shaker Publishing.

Hornecker, E. & Buur, J. (2006) Getting a Grip on Tangible Interaction: A Framework on Physical Space and Social Interaction. *CHI 2006.* Montreal, Quebec, Canada.

Hourcade, J. P., Bederson, B. B. & Druin, A. (2004) Preschool Children's Use of Mouse Buttons. *Proceeding of CHI 2004, Vienna, Austria.*

Höysniemi, J., Hämäläinen, J. P. & Turkki, L. (2004) Wizard of Oz Prototyping of Computer Vision Based Action Games for Children. In Druin, A., Hourcade, J. P. & Kollet, S. (Eds.) *Building a Community. IDC 2004 Interaction Design and Children, Conference Proceedings.*

Höysniemi, J., Hämäläinen, P., Turkki, L. & Rouvi, T. (2005) Children's Intuitive Gestures in Vision-Based Action Games. *Communications of the ACM,* 48**,** 45-50.

Hughes, J. (2001) Sony AIBO, http://the-gadgeteer.com/review/sony_aibo_review, accessed 22 January 2008.

Hughes, M. (1991) What is Difficult About Arithmetic. In Light, P., Sheldon, S. & Woodhead, M. (Eds.) *Learning to Think.* London, Routledge.

Hutchinson, H., Druin, A. & Bederson, B. B. (2007) Designing Searching and Browsing Software for Elementary-Age Children. In Lazar, J. (Ed.) *Universal Usability: Designing Computer Interfaces for Diverse Users.* West Sussex, John Wiley and Sons Ltd.

Inkpen, K. M. (1997) Three Important Research Agendas for Educational Multimedia: Learning, Children and Gender. *AACE World Conference on Educational Multimedia and Hypermedia 97.* Calgary.

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

Inkpen, K. M., McGrenere, Booth, K. S. & Klawe, M. (1997) The Effect of Turn-Taking Protocols on Children's Learning in Mouse-Driven, Collaborative Environments. *Graphics Interface 97.* Canadian Information Processing Society.

Inwood, M. (2005) Hermeneutics. In Craig, E. (Ed.) *The Shorter Routledge Encyclopedia of Philosophy.* Routledge.

IRA & NAEYC (1998) IRA & NAEYC Position Statement: Learning to Read and Write: Developmentally Appropriate Practices for Young Children, http://www.naeyc.org/resources/position_statements/psread1.htm, accessed 18 February.

Jackson, S., Krajcik, J. & Soloway, E. (1998) The Design of Guided Learner-Adaptable Scaffolding in Interactive Learning Environments. *CHI 98.* Los Angeles CA USA, ACM Press.

John-Steiner, V. & Souberman, E. (1978) Afterword. In Vygostsky, L. S. (Ed.) *Mind in Society.* Harvard University Press.

Johnson, J. E. (1998) Play Development From Ages Four to Eight. In Fromberg, D. P. & Bergen, D. (Eds.) *Play from birth to twelve and beyond.* Garland Publishing, Inc.

Johnson, M., Wilson, A., Blumberg, B., Kline, C. & Bobick, A. (1999) Sympathetic Interfaces: Using a Plush Toy to Direct Synthetic Characters. *CHI '99.* Pittsburgh, PA, USA, ACM.

Kahn(Jr.), P. H., Friedman, B., Perez-Granados, D. R. & Freier, N. G. (2004) Robotic Pets in the Lives of Preschool Children. *Proceedings of CHI 2004.* ACM Digital Library.

Kaput, J. J. (1995) Overcoming Physicality and the Eternal Present: Cybernetic Manipulatives. In Sutherland, R. & Mason, J. (Eds.) *Exploiting Mental Imagery with Computers in Mathematics Education.* Berlin, Springer Verlag.

Keferl, M. (2007) Paro Robot Seal - Healing and Companionship Pet, http://www.kilian-nakamura.com/blog-english/index.php/paro-robotic-seal-healing-and-companionship-pet/, accessed 23 January 2008.

Kelly, K. (1999a) From Encounter to Text: Data-Gathering in Interpretive Research. In Terreblanche, M. & Durrheim, K. (Eds.) *Research in practice: Applied Methods for the Social Sciences.* Cape Town, UCT Press.

Kelly, K. (1999b) Hermeneutics in Action: Empathy and Interpretation. In Terreblanche, M. & Durrheim, K. (Eds.) *Research in Practice: Applied Methods for the Social Sciences.* Cape Town, UCT Press.

King, J. & Alloway, N. (1992) Preschoolers' Use of Microcomputers and Input Devices. *Journal of Educational Computing Research***,** 451-468.

Kirchoff, D. & Horne, S. (2003) *Storybook Weaver Deluxe 2004: User Guide*, Riverdeep Interactive Learning Limited.

Klein, H. K. & Myers, M. D. (1999) A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems. *MIS Quarterly,* 23**,** 67-94.

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

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

Kline, S. (1993) *Out of the Garden: Toys, TV, and Children's Culture in the AGe of Marketing,* London, New York, Verso.

Knight, C. C. & Fischer, K. W. (1992) Learning to Read Words: Individual Differences in Developmental Sequences. *Journal of Applied Developmental Psychology,* 13**,** 377-404.

Kotzé, P. & Johnson, C. (2004) *Human Computer Interaction: Study guide for INF120-8,* Pretoria, UNISA Press.

Le Roux, T. (2006) Young Children's Software-Based Storytelling (A research report in part fulfilment of the honours degree in Information Systems). University of South Africa.

LECO Research Group (2005) LECO Research Group, http://www.cs.uta.fi/hci/leco/research_children.php, accessed 22 January 2008.

Li, N., Moraveji, N., Kimura, H. & Ofek, E. (2006) Improving the Experience of Controlling Avatars in Camera-Based Games Using Physical Input. *MM '06.* Santa Barbara, California, USA, ACM Press.

Liang, P.-H. & Johnson, J. (1999) Using Technology to Enhance Early Literacy Through Play. *Computers in the Schools,* 15**,** 55-64.

Loke, L., Larssen, A. T., Robertson, T. & Edwards, J. (2007) Understanding Movement for Interaction Design: Frameworks and Approaches. *Peronal Ubiquitous Computing,* 11**,** 691-701.

Louw, D. A., Van Ede, D. M. & Louw, A. E. (1998) *Human Development,* Pretoria, Kagiso Publishers.

Luckin, R., Connolly, D., Plowman, L. & Airey, S. (2003) Children's Interactions with Interactive Toy Technology. *Journal of Computer Assisted Learning,* 19**,** 165-176.

MacFarlane, S., Sim, G. & Horton, M. (2005) Assessing Usability and Fun in Educational Software. *Interaction Design and Children.* Boulder, Colorado, USA, ACM.

Madej, K. (2003) Towards Digital Narrative for Children: From Education to Entertainment: A digital Perspective. *ACM Computers in Entertainment,* 1**,** Article 3.

Malone, T. W. (1982) Guidelines for Designing Educational Computer Programs. *Childhood Education,* 59**,** 241-247.

Martens, W. L. (1997) Designing the "Spatial Math Tutor": A Non-Symbolic, On-Line Environment for Teaching Mathematical Skills to Dyslexic Children. *Second International Conference on Cognitive Technology, 1997.* IEEE.

Marti, P., Pollini, A., Rullo, A. & Shibata, T. (2005) Engaging with Artificial Pets. *Annual Conference on European Association of Cognitive Ergonomics.* ACM Press.

Martin, F., Mikhak, B., Resnick, M., Silverman, B. & Berg, R. (2000) To Mindstorms and Beyond: Evolution of a Construction Kit for Magical Machines. In Druin, A. & Hendler, J. (Eds.) *Robots for Kids: Exploring New Technologies for Learning.* Morgan Kaufman.

Masterman, E. & Rogers, Y. (2002) A Framework for Designing Interactive Mutimedia to Scaffold Young Children's Understanding of Historical Chronology. *International Science,* 30**,** 221-241.

Matthew, K. (1997) A Comparison of the Influence of Interactive CD-ROM Storybooks and Traditional Print Storybooks on Reading Comprehension. *Journal of Research on Computing in Education,* 29**,** 263-275.

McElligott, J. & Leeuwen, L. v. (2004) Designing Sound Tools and Toys for Blind and Visually Impaired Children. In Druin, A., Hourcade, J. P. & Kollet, S. (Eds.) *Building a Community. IDC 2004 Interaction Design and Children, Conference Proceedings.*

Meece, J. L. (2002) *Child and Adolescent Development for Educators,* Boston, McGraw-Hill.

Meraka Institute (2007) The Digital Doorway Project, www.meraka.org.za/digitalDoorway.htm, accessed 23 October 2007.

Mercer, N. & Fisher, E. (1997) The Importance of Talk. In Wegerif, R. & Scrimshaw, P. (Eds.) *Computers and Talk in the Primary Classroom.* Multilingual Matters Ltd.

Michalowski, M., Sabanovic, S. & Kozima, H. (2007) A Dancing Robot for Rhythmic Social Interaction. *Human-Robot Interaction 2007.* Arlington, Virginia, ACM Press.

Miles, M. B. & Huberman, A. M. (1994) *Qualitative Data Analysis*, SAGE Publications.

Mioduser, D. & Nachmias, R. (2002) WWW in Education: An Overview. In Adelsberger, H., Collis, B. & Pawlowsky, M. (Eds.) *Handbook on Information Technologies for Education and Training.* Springer.

Mioduser, D., Nachmias, R., Oren, A. & Lahav, O. (1999) Web-Based Learning Environments (WBLE): Current Implementation and Evolving Trends. Tel-Aviv University, School of Education, Knowledge Technology Lab.

Mioduser, D., Tur-Kaspa, H. & Leitner, I. (2000) The Learning Value of Computer-Based Instruction of Early Reading Skills. *Journal of Computer Assisted Learning,* 16**,** 54-63.

MIT (1996) KidsRoom: An Interactive narrative Playspace, vismod.media.mit.edu/./kidsroom/kidsroom.html, accessed 22 January 2008.

MIT (2007) One Laptop per Child Project, www.laptopical.com/2b1-laptop.html, accessed 23 October

2007.

MIT (2008a) Past Projects: Storymat, http://www.media.mit.edu/gnl/, accessed 22 January 2008.

MIT (2008b) Sam, www.media.mit.edu/gnl/images/projects/sam.gif, accessed 17 January 2008.

MIT Media Lab (1996) Kidsroom: Action Recognition in an Interactive Story Environment, http://vismod.www.media.mit.edu/vismod/demos/kidsroom., accessed 5 June 2005.

Mitra, S. (2003) Minimally Invasive Education: A Progress Report on the "Hole-in-the-Wall" Experiments. *British Journal of Educational Technology,* 34**,** 367-371.

Mitra, S. & Rana, V. (2001) Children and the Internet: Experiments with Minimally Invasive Education in India. *British Journal of Educational Technology,* 32**,** 221-232.

Montemayor, J., Druin, A. & Hendler, J. (2000) PETS: A Personal Electronic Teller of Stories. In Druin, A. & Hendler, J. (Eds.) *Robots for Kids: Exploring New Technologies for Learning.* Morgan Kaufman.

Moreno, L. E. & Sacristán, A. I. (1995) On Visual and Symbolic Representations. In Sutherland, R. & Mason, J. (Eds.) *Exploiting Mental Imagery with Computers in Mathematics Education.* Berlin, Springer Verlag.

NAEYC (1995) NAEYC Position Statement: Responding to Linguistic and Cultural Diversity Recommendations for Effective Early Childhood Education, http://www.naeyc.org, accessed 17 February 2004.

NAEYC (1996) NAEYC Position Statement: Technology and Young Children - Ages 3 through 8, http://www.naeyc.org/resources/position_statements/pstech98.htm, accessed 17 February 2004.

NAEYC (1997) NAEYC Position Statement: Developmentally appropriate practice in early childhood programs serving children from birth through age 8, http://www.naeyc.org/resources/position_statements/dap3.htm, accessed 17 February 2004.

NAEYC (2002) NAEYC Position Statement: Early Childhood Mathematics: Promoting Good Beginnings, http://www.naeyc.org/resources/position_statements/psmath.htm, accessed 17 February 2004.

Nass, C. & Lee, K. (2000) Does Computer-Generated Speech Manifest Personality? An Experimental Test of Similarity-Attraction. *CHI 2000.* The Hague, Amsterdam, ACM Press.

Nicol, A., Casey, C. & MacFarlane, S. (2002) Children are Ready for Speech Technology - But is the Technology Ready for Them? *Interacting with Computers,* 15**,** 159-170.

Nicol, T., Read, J. & MacFarlane, S. (2005) Improving Speech Recognition in a Listening Interface for Young Children. *HCI 2005.* Edinburgh, Scotland, Springer.

Nicolson, R., Fawcett, A. & Nicolson, M. (2000) Evaluation of a Computer-Based Reading Intervention in Infant and Junior Schools. *Journal of Research in Reading,* 23**,** 194-209.

Nielsen, J. (1994) Heuristic Evaluation. In Nielsen, J. & Mack, R. (Eds.) *Usability Inspection Methods.* New York, John Wiley and Sons.

Nielsen, J. (2001) Ten Usability Heuristics, www.useit.com/papers/heuristic, accessed 16 January 2005.

Nunes, T. & Bryant, P. (2000) A Step After Phonics, http://www.literacytrust.org.uk/Database/morphemes.html, accessed 25 November.

Oviatt, S., Darves, C. & Coulston, R. (2004) Toward Adaptive Conversational Interfaces: Modeling Speech Convergence With Animated Personas. *ACM Transactions on Computer-Human Interaction,* 11**,** 300-328.

Ozoux.com (2007) Keepon Dancing Robot, http://www.ozoux.com/eclectic/archive/2007/03/26/keepon-dancing-robot, accessed 22 January 2008.

Papert, S. (1980) *Mindstorms: Children, Computers and Powerful Ideas,* New York, Basic Books.

Parkin, A. J. (1997) The Development of Procedural and Declarative Memory. In Cowan, N. (Ed.) *The Development of Memory in Childhood.* Psychology Press.

Patomäki, S., Raisamo, S. R., Salo, J., Pasto, V. & Hippula, A. (2004) Experiences on Haptic Interfaces for Visually Impaired Young Children. *Proceedings of ICMI'04.* Pennsylvania, USA.

Piaget, J. (1953) *The Origins of Intelligence in the Child.,* London, Routledge and Kegan Paul.

Pine, K. (1999) Theories of Cognitive Development. In Messar, D. & Millar, S. (Eds.) *Exploring Developmental Psychology: From Infancy to Adolescence.* Arnold.

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

Plowman, L. & Stephen, C. (2003) A 'Benign Addition'? Research on ICT and Pre-School Children. *Journal of Computer Assisted Learning,* 19**,** 149-164.

Preece, J., Rogers, Y. & Sharp, H. (2002) *Interaction design: beyond human-computer interaction*, John Wiley & Sons, Inc.

Preece, J., Rogers, Y. & Sharp, H. (2007) *Interaction Design: Beyond Human-Computer Interaction*, John Wiley & Sons, Inc.

Price, S., Rogers, Y., Scaife, M., Stanton, D. & Neale, H. (2002) Using Tangibles to Support New Ways of Playing and Learning. *Interacting with Computers,* 15**,** 88-92.

Puttergill, C. (2000) Strategies of Discovery. In Collins, K. J., Plooy, G. M. D., Grobbelaar, M. M., Puttergill, C. H., Terreblanche, M. J., Eeden, R. V., Rensburg, G. H. V. & Wigston, D. J. (Eds.) *Research in the Social Sciences: Only Study Guide for RSC201-H.* Pretoria, Unisa Press.

Read, J. (2005) The Usability of Digital Ink Technologies for Children and Teenagers. *HCI 2005.* Edinburgh, Scotland, Springer.

Read, J., Gregory, P., MacFarlane, S., McManus, B., Gray, P. & Patel, R. (2002a) Interaction Design and Children. In Bekker, M., Markopoulos, P. & Kersten-Tsikalkina, M. (Eds.) *International workshop on Interaction Design and Children.* Eindhoven, The Netherlands, Shaker Publishing.

Read, J., MacFarlane, S. & Casey, C. (2001) Measuring the Usability of Text Input Methods. *HCI 2005.* Lille, France.

Read, J., MacFarlane, S. & Casey, C. (2002b) Oops! Silly me! Error in Handwriting Recognition-based Text entry Interface for Children. *Proceedings of NordCHI, October 2002.*

Read, J., MacFarlane, S. & Gregory, P. (2004) Requirements for the Design of a Handwriting Recognition Based Writing Interface for Children. In Druin, A., Hourcade, J. P. & Kollet, S. (Eds.) *Building a Community. IDC 2004 Interaction Design and Children, Conference Proceedings.*

Resnick, M. J. (2000) Commentary in Five Commentaries: Looking to the Future. *The Future of Children - Children and Technology,* 10**,** 173-175.

Revelle, G. L. (2003) Educating via Entertainment Media: The Sesame Workshop Approach. *ACM Computers in Entertainment,* 1**,** Article 7.

Revelle, G. L. & Medoff, L. (2002) Interface Design and Research Process for Studying the Usability of Interactive Home-Entertainment Systems by Young Children. *Early Education and Development,* 13**,** 423-434.

Revelle, G. L. & Strommen, E. (1990) The Effects of Practice and Input Device Used on Young Children's Computer Control. *Journal of Computing in Childhood Education,* 2**,** 33-41.

Rittle-Johnson, B. & Siegler, R. S. (1999) Learning to Spell: Variability, Choice, and Change in Children's Strategy Use. *Child Development,* 70**,** 332-348.

Rogers, Y., Scaife, M., Gabrielli, S., Smith, H. & Harris, E. (2004) A Conceptual Framework for Mixed Reality Environments: Designing Novel Learning Activities for Young Children. *Presence,* 11**,** 677.

Rogoff, B., Gauvain, M. & Ellis, S. (Eds.) (1991) *Child Development in Social Context.*

Romeo, G., Edwards, S., McNamara, S., Walker, I. & Ziguras, C. (2003) Touching the Screen: Issues Related to the Use of Touchscreen Technology in Early Childhood Education. *British Journal of Educational Technology,* 34**,** 329-339.

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

Rossouw, H. (1980) *Wetenskap, Interpretasie, Wysheid,* Port Elizabeth, University of Port Elizabeth.

Ryokai, K. & Cassell, J. (1999) StoryMat: A Play Space for Collaborative Storytelling. *CHI 99.* Pittsburgh, Pennsylvania, ACM Press.

Ryokai, K., Vaucelle, C. & Cassell, J. (2003) Virtual Peers as Partners in Storytelling and Literacy Learning. *Journal of Computer Assisted Learning,* 19**,** 195-208.

Scaife, M. & Bond, R. (1991) Development Changes in Children's Use of Computer Input Devices. *Early Child Development and Care,* 69**,** 19-38.

Scaife, M. & Rogers, Y. (2001) Informing the Design of a Virtual Environment to Support Learning in Children. *International Journal of Human-Computer Studies,* 55**,** 115-143.

Schneider, K. G. (1996) Children and Information Visualization Technologies. *Interactions***,** 68-74.

Schneider, W. & Bjorklund, D. F. (1998) Memory. In Kuhn, D. & Siegler, R. S. (Eds.) *Cognition, Perception and Language.* 5th ed., John Wiley and Sons Inc.

Scholastic (2002) I Spy Spooky Mansion. *I SPY™.* Scholastic, Inc.

Sedighian, K. (1997) Challenge-Driven Learning: A Model for Children's Multimedia Mathematics Learning Environments. *ED-MEDIA 97: World Conference on Educational Multimedia and Hypermedia.* Calgary, Canada.

Shade, D. D. (1996) Software Evaluation. *Young Children,* 51**,** 17-21.

Shields, M. & Behrman, R. (2000) Children and Computer Technology: Analysis and Recommendations. *The Future of Children,* 10**,** 4-30.

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

Shneiderman, B. (1998) *Designing the User Interface: Strategies for Effective Human-Computer Interaction,* New York, Addison-Wesley.

Siegler, R. (1976) Three Aspects of Cognitive Development. *Cognitive Psychology,* 8**,** 481-520.

Siraj-Blatchford, J. & Siraj-Blatchford, I. (2001) Kidsmart: The Phase 1 UK Evaluation 2000-2001: Final Project report.

Sluis, R. J. W., Weevers, I., Van Schijndel, C. H. G., Kolos-Mazuryk, L., Fitrianie, S. & Martens, J. B. O. (2004) Read-It: Five-to-Seven-Year-Old Children Learn to Read in a Tabletop Environment. In Druin, A., Hourcade, J. P. & Kollet, S. (Eds.) *Building a Community. IDC 2004 Interaction Design and Children, Conference Proceedings.*

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

Sony (2008) Sony Playstation 2, http://www.us.playstation.com/PS2/Accessories/SCPH-10010U, accessed 22 January 2008.

Stanton, D., Bayon, V., Neale, H., Ghali, A., Benford, S., Cobb, S., Ingram, R., O'Malley, C., Wilson, J. & Pridmore, T. (2001) Classroom Collaboration in the Design of Tangible Interfaces for Storytelling. *CHI 2001.* Seattle, WA, USA, ACM Press.

Stereoscopy.com (2000) 3D News Archive, accessed 23 January 2008.

Strommen, E. (1994) Children's Use of Mouse-Based Interfaces to Control Virtual Travel. *CHI 94.* Boston, Massachusetts, USA, ACM Press.

Strommen, E. & Alexander, K. (1999) Emotional Interfaces for Interactive Aardvarks: Designing Affect into Social interfaces for Children. *Proceedings of CHI 99.* Pittsburgh, USA.

Subrahmanyam, K., Kraut, R. E., Greenfield, P. M. & Gross, E. F. (2000) The Impact of Home Computer Use on Children's Activities and Development. *The Future of Children,* 10**,** 123-144.

Sundholm, H. & Dahlbäck, N. (2002) An Evaluation of Digital Cuddly Toy Museum Guides. *Interacting with Computers,* 15**,** 171-175.

Swaminathan, N. (2007) Could Robots Become Your Toddler's New Best Friend? *Scientific American.*

Tanaka, F., Noda, K., Sawada, T. & Fujita, M. (2004) Associated Emotion and Its Expression in an Entertainment Robot QRIO. In Rauterberg, M. (Ed.) *Lecture notes in Computer Science, 3166.* IFIP.

Terre Blanche, M. & Durrheim, K. (1999) Histories of the Present: Social Science Research in Context. In Terre Blanche, M. & Durrheim, K. (Eds.) *Research in practice: Applied methods for the social sciences.* Cape Town, UCT Press.

Terre Blanche, M. & Kelly, K. (1999) Interpretive Methods. In Terre Blanche, M. & Durrheim, K. (Eds.) *Research in practice: Applied methods for the social sciences.* Cape Town, UCT Press.

Terre Blanche, M. & Durrheim, K. (1999) Histories of the Present: Social Science Research in Context. In Terre Blanche, M. & Durrheim, K. (Eds.) *Research in practice: Applied methods for the social sciences.* Cape Town, UCT Press.

Terre Blanche, M. & Kelly, K. (1999) Interpretive methods. In Terre Blanche, M. & Durrheim, K. (Eds.) *Research in practice: Applied methods for the social sciences.* Cape Town, UCT Press.

Thornton, S. (1995) *Children Solving Problems,* London, Harvard University Press.

Thornton, S. (2002) *Growing Minds,* London, Harvard University Press.

Uden, L. & Dix, A. (2000) Iconic Interfaces for Kids on the Internet. *Proceedings of IFIP WorldComputer Congress, August 2000.* Beijing.

University of Maryland HCI Lab (1997) KidPad.

University of Maryland HCI Lab (2008) Project Screenshots, www.cs.umd.edu/hcil/pubs/screenshots/PETS, accessed 22 January 2008.

Valinho, P. & Correia, N. (2004) oTTomer: An Interactive Adventure System for Children. *SRMC 2004.* New York, USA, ACM Press.

Van Daal, V. & Reitsma, P. (2000) Computer-Assisted Learning to Read and Spell: Results from Two Pilot Studies. *Journal of Research in Reading,* 23**,** 181-193.

Van Ijzendoorn, M. & Pellegrini, A. (1995) Joint Book Reading Makes for Success in Learning to Read: A Meta-Analysis on Intergenerational Transmission of Literacy. *Review of Educational Research,* 65**,** 1-21.

Van Kleeck, A. (1990) Emergent Literacy: Learning about Print before Learning to Read. *Topics in Language Disorders,* 10**,** 25-45.

Van Rensburg, G. H. (2000) Sampling. In Collins, K. J., Du Plooy, G. M., Grobbelaar, M. M., Puttergill, C. H., Terreblanche, M. J., Van Eeden, R., Van Rensburg, G. H. & Wigston, D. J. (Eds.) *Research in the Social Sciences: Only Study Guide for RSC201-H.* Pretoria, Unisa Press.

Vygotsky, L. (1978) *Mind in Society: The Development of Higher Psychological Processes,* Cambridge, Mass, Harvard University Press.

Wartella, E. A. & Jennings, N. (2000) Children and Computer Technology: New Technology - Old concerns. *The Future of Children,* 10**,** 31-42.

Web Japan (2004) Run, Robot, Run! *Kids Web Japan Monthly News.* Web Japan.

Wellman, H. M. & Gelman, S. M. (1998) Knowledge Acquisition in Foundational Domains. In Kuhn, D. & Siegler, R. S. (Eds.) *Cognition, Perception and Language.* 5th ed., John Wiley and Sons Inc.

Williges, B. H. & Williges, R. C. (1984) User Considerations in Computer-Based Information Systems. In Muckler, F. A. (Ed.) *Human Factors Review 1984.* Santa Monica, Human Factors Society.

Wyeth, P. & Purchase, H. C. (2003) Using Developmental Theories to Inform the Design of Technology for Children. *Interaction Design and Children.* University of Central Lancashire, Preston, UK, ACM Digital Library.

Appendix 1: Evaluation of Storybook Weaver Deluxe 2004

Following the framework of guidelines presented in Chapter 8 of this thesis, the table below contains notes on the evaluation of the Storybook Weaver Deluxe 2004 storytelling software for children. Of the 350 guidelines in the complete framework, I have identified 157 that are applicable to this kind of application. For example: I used some of the guidelines that relate to writing skills and problems solving, but excluded all of the guidelines associated with mathematics, reading, and computer literacy from the evaluation. Since this is a mouse-driven, PC based product I did not address any of the guidelines that refer to tangible and robotic interfaces or to the internet.

The relevant guidelines appear in the left hand column with numbers that correspond to their numbers in the complete framework. My evaluation notes appear in the right hand column with evaluation statements in the Times Roman font and suggestions for re-design in Arial. I abbreviate the name of the software as SBW. The structured evaluation report based on these evaluation notes appears in Chapter 9 of this thesis.

B. DEVELOPMENTAL APPROPRIATENESS

A.1 Age specific guidelines		
A.1.2	Children aged five: • can separate themselves mentally from their physical surroundings to imagine and accept absurd fantasies or strange characters in stories. • can behave pro-socially and model pro-social behaviour of adults or media characters. • start to develop a strong sense of identity and realise that they belong to a specific gender. • are interested in writing words.	Developmentally appropriate for five year olds: Absurd fantasy characters and background scenes are included in SBW. Characters represent different genders and gender preferences. Children can experiment with writing with the keyboard. E.g. SBW allows children to type title, author and story pages. SBW is completely open-ended. Children can create any story – also stories that where they can model pro-social behaviour of adults or media characters.
A.1.3	Children aged six: • prefer bright contrasting colours that create patterns. • mostly still prefer predictable activities that they can direct, but they begin to appreciate a challenge to familiarity. • can begin to follow simple written directions and give simple written feedback such as typing their names. • like to talk, sing and record their own voices. • are able to cooperate with other children and wait for their turns – they can reason about fairness and play group games without adult supervision.	Developmentally appropriate for six year olds in the sense that borders and backgrounds include options with bright contrasting colours that create patterns. Characters and scenes include familiar characters as well as strange characters. Can write their names and simple story text. Can record their own voices and attach the recordings to pages or story objects. SBW is ideal for collaborative story writing using a desktop computer – children can give characters different voices by working together.
A.1.4	Children aged seven: • have the perceptual acuity to understand how two or more software tools can be used together to accomplish a task. • can start to organise items into groups and categories	Developmentally appropriate for seven-year-olds: Children can import graphics or photos created in a different application into SBW and use this as story objects. SBW uses a simple categorisation scheme for story objects – currently only understandable by children who

	and can understand organisational schemes such as that used for books in a library. • enjoy browsing activities. • should be given tasks that involve combining, ordering and separating objects mentally. • can begin to write simple programming commands to direct movement of an on-screen object. • are very aware of their gender.	can read. Children can browse easily through story characters, backgrounds, borders, sounds, etc. The whole story creation process involves combining, ordering and separating objects mentally. Can include programming of simple movements. Address gender stereotypes (e.g. no male dancers). Re-design categorisation of story objects to make it understandable to pre-readers.
A.1.5	Children aged five to eight: • can coordinate structures for dealing with more than one aspect of a situation. • understand psychological causality and realise that characters have goals and beliefs that may influence their behaviour. • can reason to some extent about false beliefs. • can interpret symbols and images that represent real-life situations. • fears being alone, getting lost and losing a parent and therefore like stories about characters who overcome these fears.	Developmentally appropriate for five to eight-year-olds: Creating stories allow children to deal with more than one aspect of a situation. Can use their understanding of psychological causality and realisation that characters have goals and beliefs that may influence their behaviour. Can incorporate false beliefs in their stories. SBW gives them opportunity to use symbols and images to represent real-life situations. They can write their own stories about their fear of being alone, getting lost and losing a parent and overcoming these fears. Include in story ideas and story starters: stories about fear of being alone, getting lost and losing a parent; stories that include false beliefs.
A.1.6	Take children's exposure to video games, television, movies and other media into account when designing technology. It influences their expectations of technology as follows: • They want a multi-sensory experience. • They want state of the art technology (e.g. headphones are 'cooler' than speakers). • They do not want technology that 'talks down' to them.	Children may have been exposed to applications such as SIMS which allows them to modify characters and where characters are alive and move around. Allowing children to dress and modify characters' appearance may, however, distract children from creating stories – they may get so involved in creating characters that they never get to write a complete story. Designers should keep up with current trends and fashions (e.g. include an MP3 player as a story object).

A.2 Biological maturation		
A.2.1	Know the minimum requirements of a specific task with regard to biological maturation and do not expect children to perform actions that they are not physically capable of.	Can determine the user's age in the beginning and use that to make certain functions available or not (e.g. voice recording). If children are old enough to read, audio cues can be switched off by default.
A.2.2	When designing for children aged six or seven, designers can assume that they can manipulate input devices without problems and are able to use different types of input devices.	See above. Six-year-olds can use the recording function without difficulty. They do, however, find it difficult to use the menu as they have to keep the mouse button down when moving the cursor to the required option.
A.2.3	Design activities for eight-year-olds so that they can complete them comfortably in a forty minute session. The beginning and end of an activity should be clearly defined so that restricting their time at the computer will not interfere with an ongoing activity.	Activities are open-ended and storytelling is not really something you can put a time limit on. Can suggest to the child to take a break after 40 minutes.
A.2.4	Young children are often mentally ahead of their physical selves, so give them opportunities to practice grown-up actions.	Storytelling activities give children the opportunity to implement grown-up actions through their characters. Include possibilities for grown-up actions such as kissing and getting married, or being president, for example.

A.3 Existing knowledge structures in skill development		
A.3.1	Identify all the skills involved in an activity and understand how these will be coordinated and integrated in the activity.	The skills involved in story telling are: Deciding on a theme, choosing the setting and characters, choosing objects, ordering events, identifying a suitable beginning, development and ending, dividing the story into pages, focussing on the theme and completing the story. In SBW, creating a story also involves: Typing or recording the story text, placing the illustrations on the page and organising the characters and objects in the chosen scene, writing for an audience.
A.3.2	When all the underlying skills and operations have been identified, determine whether the user will be able to perform each of these and whether they have the mental capacity for the new skill.	Children of five to eight can learn most of the above skills. Only from six or seven can they be expected to start reading or writing. Children of this age may find it difficult to keep focused on the chosen theme, especially, since SBW gives them the opportunity to browse thousands of story scenes, objects and characters. They get so involved in browsing these that they forget what their story is about, or they may see interesting characters that have no link to their story and get side-tracked.
A.3.4	When two or more skills must be integrated to acquire a new skill, provide activities that require application of those skills or operations before presenting users with activities that combine them.	SBW does not present children with specific activities. It is merely a set of tools that children can use in whatever way they want. Children will naturally only use the tools that they have mastered and therefore combine operations that they are familiar with.
A.3.5	Be informed of the development sequence of every skill that will be supported in an application.	Investigate the development sequence for the skill to tell stories.
A.3.7	Make any skill's connection with real life explicit. An activity chosen to develop a skill must be one that can be naturally associated with that skill.	SBW supports one overarching skill, namely story creation. All the activities fulfil some purpose in the aim to help children to create a story and can be naturally associated with it. There are functions that children may not understand unless they are explained to them (e.g. the thesaurus).

A.4 Memory capacity		
A.4.1	Strive to relieve a child's working memory of extra processing that may prevent them from coordinating their knowledge structures – that is, reduce the cognitive load of interaction so that they have sufficient cognitive resources for learning to take place.	User interface can improve a lot in this respect. E.g. more suitable icons, make functions such as TTS and spell checking more visible, use audio cues to make icons 'visible' to pre-readers.
A.4.2	Find ways to free up processing or storage capacity in working memory. For example: • Make objects, actions and options visible so that users need not remember instructions or previous choices. • Let young children interact through direct manipulation with a mouse, as keyboard commands require information to be held in working memory.	Most of the interaction takes place through direct manipulation with the mouse. The keyboard is used to type story text, but navigation is primarily done with the mouse. Currently objects, actions and options are visible, but there is room for improvement. Also see above.
A.4.5	Audio communication persists only in the user's memory, so do not rely on children's accurate recall of audio instructions (especially when given in the beginning of a session).	SBW has a quick tour of the story creation process that explains how a story is created using a voice over. This is the only audio instructions currently available. The child will learn a lot through the quick tour, but audio can be used much more extensively to help (especially pre-reading) children to interpret (for example) icons correctly.

A.4.8	Children who are familiar with narrative structures are better at recalling events in the correct order – if the aim is to improve this skill, provide them with activities that will develop their knowledge of narrative structure.	The ideas bank can include incomplete stories that the children can complete (these can then include a beginning, middle and ending, instead of just the beginning).
A.4.10	The phonological store keeps information for only two seconds – do not expect a child to act on audio cues that occurred longer than two seconds ago (unless the information is repeated often). Verbal instructions should be short.	Use verbal instructions but keep them short.

A.5 General high-level guidelines relating to children's development		
A.5.1	Make the learning goals of the application clear and base the design on a clearly identifiable educational approach that supports these intended goals.	Learning objectives as set out by SBW designers: To use a student's inherent creativity to write and illustrate a story. To explore the writing process using a simple word processor and a variety of graphics. To create illustrations that depict their storyline. To write with a purpose. To share writing with an audience. To enhance vocabulary by associating a word with its picture. To develop story sequencing skills. The designers do not state their educational approach but it is clearly constructivist as children learn the above skills by actively constructing stories.
A.5.2	Do not attempt to address a range of cognitive skills in one application. Focus on one or two skill domains and do it well.	SBW focuses exclusively on storytelling skills.
A.5.3	Design an application so that its use can be integrated with other practices such as play and project work.	Can easily be used to complement schoolwork. E.g. reading and writing, creative writing, history.
A.5.4	Construction activities provide an effective basis for interaction, so give children activities that allow them to design, create and evaluate.	SBW allows children to design a story, a setting and the individual characters. They then create the story. More can be done in terms of evaluation of what they have created.
A.5.7	Promote positive social values and no violence.	SBW does not include characters or objects that promote or portray violence. Include scenarios and characters that will encourage stories about social values and pro-social behaviour.
A.5.8	If the aim is to reduce the need for adult intervention, make sure that children cannot produce incorrect results – only unexpected outcomes.	Children are completely in charge of what happens – nothing they do will produce 'incorrect' results.
A.5.9	Children benefit significantly from support by teachers who closely guide children's interaction and continually encourage, question, prompt and demonstrate. There is thus a place for technology designed for collaborative use with an adult.	SBW should ideally be used with some help from a parent or teacher as children may miss a lot of the functionality if it is not explicitly pointed out to them. They only need to be shown how a function works once
A.5.10	Respect preschoolers' natural need to demonstrate their abilities to parents and caregivers – give them opportunities to do so.	SBW allows children to print out their stories, but it would also be nice to give them a playback option that shows the story as a slide show.

B. THE DEVELOPMENT OF SPECIFIC SKILLS

B.2 Problem solving	
B.2.1 To support reversibility skills, include activities that require mental reversing of actions such as combining, ordering, separating and recombining of elements.	Creating scenes, adding story characters and objects and editing these, will sometimes require mental combination, separation and re-combinations while children are planning their stories.
B.2.3 To help children grasp cause and effect relationships: • Give younger children opportunities to move objects around on the screen by, for example, dragging them with the mouse. • Create opportunities for children to view processes and cause and effect relationships that are difficult to observe in reality.	SBW offers lots of opportunities for dragging objects around on screen. Also, clicking on the 'move forward', 'flip', 'make bigger' and 'make smaller' buttons allow children to see cause and effect relationships. Since all of these operations can be reversed they also support their reversibility skills.
B.2.6 Compensate for children's lack of metacognitive knowledge that they need to choose between possible solutions to a problem.	By providing story starters, children are helped along when they cannot get started. This can be extended to an ideas bank where they can also have story endings, story emotions, etc.
B.2.11 Encourage children to think about and discuss their plans before doing an activity to help them to collaborate more, plan better and perform tasks more efficiently.	Consider providing children with a platform to plan their whole story – sort of an overview that helps them to think about the beginning, middle, climax and ending.
B.2.12 Encourage reflection on actions.	Encourage them to play back their stories and think about possible changes.
B.2.13 Support higher order thinking skills by allowing children to create, save, retrieve and change their ideas.	SBW gives children lots of opportunities to create, save, retrieve and change their ideas.
B.2.15 Do not allow children to solve all problems through trial-and-error. Include mechanisms to focus their attention and to get them to plan their solutions.	SBW's open-ended nature makes it easy for children to lose focus. Could build in some way to help them stay focussed.
B.2.17 To improve children's perspective taking skills, teach them to imagine physical spaces from different points of view and to compare different states of the world by presenting them with three-dimensional images that they can manipulate and rotate with the mouse.	SBW can potentially teach children about perspective in the sense that objects in the front are bigger and objects at the back are smaller. Currently children have to discover this for themselves or with the help of an adult. SBW does not allow children to rotate objects and characters – they can only flip them. Could build in a mechanism to draw children's attention to the use of the 'make smaller' and 'make bigger' buttons to move object forward and backwards in the picture. Allow users to rotate objects so that they can be seen from behind, in front or from the top.
B.2.18 Provide opportunities for children to: • formulate their own problems and get feedback on them • make choices • explore and manipulate different kinds of representations interactively.	SBW allows children to make choices and to explore and manipulate different kinds of representations interactively.

B.4 Writing	
B.4.3 To support learning to spell, use a spellchecker-like facility that helps children recognise and correct their own mistakes (it should not automatically correct spelling mistakes).	SBW includes a spell checker that children can use to check the spelling of single words or a piece of text. They can either double-click in the spell checker to replace a word or correct the spelling themselves. Make the spell checking function more accessible and easier to use.

B.4.4	Use speech feedback for writing support, but let children decide when and whether it should be given.	SBW currently provides a text to speech facility that reads back what the child has written on the current page. This facility can be improved (see C.3.9, F.1.21 and F.1.24 below).
B.4.5	Children should have control over the level of speech feedback – that is, whether it should be at letter, phoneme or word level.	SBW allows children to highlight the text that they want to hear.
B.4.6	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.	A misspelled word leads to strange utterances by the text to speech facility in SBW, but this may help a child to identify spelling mistakes.
B.4.7	Allow easy adaptability to different languages.	SBW is currently available in English and Spanish. It seems to be easily adaptable to different languages. If text is included with icons (as I will recommend later), the text should not be part of the graphic so that it can be translated easily.

B.5 Storytelling		
B.5.1	Keep storytelling interfaces very simple and include a training module to help children acquire a good enough mental model to use the system.	SBW has a simple interface as well as a training module that is good enough to get a child started. It does, however, not tell them about all the functionality.
B.5.2	The following are suitable activities to include in a storytelling application: • reading existing stories • sequencing jumbled stories • finishing partly written stories • changing existing stories • helping an agent to write a story • creating a new story with characters and a recorded voice-over or space to type the story (for children who can write).	SBW includes the following activities: • reading existing stories • finishing partly written stories • creating a new story with characters and a recorded voice-over or space to type the story (for children who can write). The following can be added: • sequencing jumbled stories • helping an agent to write a story.
B.5.3	Keep children engaged by allowing them to implement their own ideas, but remember that they usually do not want to create a whole story from scratch.	SBW gives children complete control to implement their own ideas, but do provide them with story starters if they want help.
B.5.4	Include external mechanisms to help children with planning, organisation and sequencing of events. They need affirmation and they are not always proficient in talking about stories in a structured way.	If they initially include a character that never plays a role in the story or do not appear again, ask them whether the character is still needed. Include and 'ideas bank' with story starters, happy, sad, scary and exciting events, endings, feelings, etc.
B.5.5	Include supportive elements such as the following in storytelling products: • text-to-speech technology • spelling support • structured word banks that children can access easily • a software agent that provides proactive assistance.	SBW includes text-to-speech technology and spelling support. But not • structured word banks that children can access easily • a software agent that provides proactive assistance.
B.5.6	Provide a text or ideas bank that: • gives children ideas and suggestions • supports unusual combinations of scenes and characters • encourages the use of imagination • challenges stereotypes • changes power relations.	Include these in the 'ideas bank' (see B.5.4 above).
B.5.7	Typical elements that can be included in a word or ideas bank are story titles, story starters, story stirrers, story events, endings and feelings.	SBW only gives story starters. Can also include titles, stirrers, events, endings and feelings.
B.5.8	When designing storytelling applications, keep in mind	SBW does not have high quality graphics but this does

	that: • Children do not require a high degree of realism in presentation. Even crude forms of movement and exaggerated facial expressions are acceptable. • They are able to switch between editing (writing) mode and acting (playing) mode if the change in mode is very clear. • From six, children can adjust their speech to different listeners and from seven they can be presented with activities that require them to shift between character voices and the voice of the narrator.	not seem to put off users. Users have indicated that they would like to be able to rotate people and objects. Currently there is no differentiation between writing and playing back mode. SBW allows children to use their own voices to speak for characters as well as to narrate – they can change their voices when recording different characters. Should include a 'play' option whereby the story is played back like a slide show.
B.5.9	Create a balance between familiar and imaginative elements by allowing, children to select props, scenes and characters from real life, familiar environments or from fairy tale or space scenes.	SBW allows children to mix real world elements such as family photos with fantasy world elements, as well as elements from the past, present and future. Can make the import of photos function more accessible.
B.5.10	Through storytelling software, designers can provide opportunities for dramatic and creative play – although children do not physically carry out the actions, they create the characters and the story line and, through recording facilities, give the characters voice.	SBW provides ideal opportunity for this.
Agent support in storytelling technologies		
B.5.11	Use a combination of peer and agent help to provide emotional support during story creation. Support should be adaptable to individual needs.	SBW does not include agent support. Adding an agent may interfere with children's creativity. It is something that can be considered for the initial tutorial. Despite some obvious shortcomings, SBW works so well without an agent that I feel it will not necessarily add to the usability of and engagement with the software.

C. BUILT-IN SUPPORT

C.1 Support, scaffolding and the zone of proximal development (ZPD)		
C.1.2	When a child makes an error, provide scaffolding through a series of hints that guides the child to the correct answer.	No action of a child is ever interpreted as an error in SBW.
C.1.3	If different scaffolding options are available, let users know what they are and how to access them.	It is not obvious to SBW users that a text-to-speech facility is available. Also, the story starters are not adequately visible as children may not interpret the icon correctly.
C.1.5	There are three types of scaffolding that can be incorporated in technology: 4. Supportive scaffolding supports a task without changing the task itself and includes guiding (through messages that appear when the software detects that the user needs advice), coaching and modelling (by providing examples that explain concepts). Guiding scaffolding allows fading by displaying a button that the user can click to switch off the support. Coaching and modelling examples only appear on the user's request, so they fade by not being used. 5. Reflective scaffolding encourages users to think about a task before doing it. It doesn't change the task, but asks the user to provide plans, predictions	Supportive scaffolding is available in the form of story starters. Reflective and intrinsic scaffolding is not obviously included in SBW. Intrinsic scaffolding is not really relevant as children can decide for themselves how complex their stories will be. Children with fewer skills will naturally use less of the functionality, while older or more skilled users will use the more sophisticated functionality offered. Reflective scaffolding can be added to encourage children to think about and plan their stories.

	or evaluations. Fading involves reducing the requests for reflection. 6. Intrinsic scaffolding is built into tasks by, for example, starting at an easy level and gradually increasing the complexity of the tasks. Fading is implemented as changes in the task. For each activity there can be multiple levels of difficulty – beginning levels address fundamental cognitive skills and then one or more higher-level skill is added per level.	
C.1.6	As children become more proficient, gradually remove support until they can succeed on their own. This fading of support can be implemented through adaptive or adaptable scaffolding: • Adaptive support changes automatically based on the system's model of the user's understanding. This may be difficult to implement. • Adaptable support is faded by the user. To help the user with fading decisions, allow self-evaluation so that the user can judge their own progress and understanding. Limit the fading options because too many will confuse users. • A combination of adaptive and adaptable support allows the user to control the fading, but with guidance from the software.	Currently SBW does not give enough support to pre-reading children. Audio cues attached to icons will help them, but users must have the option to switch these off. Can make support adaptive by initially asking the user if he or she can read. If yes – switch audio cues off, otherwise leave them on.
C.1.8	Specific mediation variables that can potentially be incorporated into software are: • focusing (ensuring that the child focuses on the right interface element using mechanisms such as selecting, exaggerating, accentuating, grouping and sequencing) • affecting (through verbal or non-verbal appreciation or affect) • expanding (focusing the children's attention on the concepts they used to solve the problem) • encouraging (establishing feelings of competence through verbal or non-verbal expression of satisfaction with specific components of a child's behaviour – through immediate vocal, musical and/or visual feedback).	Would be nice if the software can pick up when children are diverting from their topic. Enough intrinsic motivation so that praise is not required. It may, however, be a good idea to reward children in some way if their stories fulfil some basic requirements.
C.1.9	Support should ideally change the environment according to the child's needs, interests and abilities. Changes can involve: • changing the intensity, frequency, order, form or context of stimuli • arousing the child's curiosity, attention and perceptual acuity.	Applies to audio cues – change the frequency according to the child's needs. Can draw a children's attention to objects or characters they never use.
C.1.10	To develop their self-esteem, present children with activities that reinforce their ability to succeed on their own.	SBW is already good in this regard.
C.1.11	Do not require children to get support from documentation.	Some functions are not clearly visible and its purpose and how it works can be read in the pdf manual. Users will always need some help of an adult to get started.

C.2 Feedback		
C.2.1	If there is a possibility that users cannot read fluently, provide feedback in audio format.	Text-to-speech is available but will only be useful when the child has typed some text. Relies to a large extent on reading ability. Many functions are only available on the menu. Should add audio cues to icons.
C.2.2	Provide adequate feedback in the form of information (audio, tactile, verbal or visual) about what action the user has performed and what the effect of that action was. Content of the feedback should be understandable and in a format that is suitable for the targeted age group.	Feedback during story creation is good. Children add objects or backgrounds with immediate, visible effect. There is a problem though with feedback on basic functions such as saving a story as a web document. A wordy text message appears telling the user where the web document has been saved. Children who cannot yet read may interpret this as an error message.
C.2.3	Use multi-modal feedback to improve the comprehensibility and accessibility of children's technology.	In the example above, simultaneous audio and text cues can make a system accessible for reading as well as pre-reading children.
C.2.4	Response time must be quick. When it is not instantaneous the system should give clear indication that the task is in progress. Lack of immediate feedback leads to repeated clicking or hitting of keys, which may influence the program's execution.	No response time problems in SBW.
C.2.5	Response times for similar tasks should be comparable.	All feedback is immediate.
C.2.6	Let feedback facilitate comprehension of the concepts as well as promote exploratory interaction.	SBW doesn't teach specific concepts, but one can regard beginnings, climax, endings, sequencing, etc. as concepts related to storytelling. Maybe there should be more explicit feedback on these concepts, but not at the cost of open-endedness.
C.2.7	Use feedback that mentions the user's name to capture his or her attention.	SBW does not ask any information about the user. Using the child's name can improve engagement – SBW does not need it as it is engaging enough as it is. Also, when used by more than one user simultaneously, using a specific name may be a problem.
C.2.8	Provide state information in different formats or modes. Providing children with different visualisations of the same information can support their learning of the concepts or knowledge involved.	When using the recording function, SBW shows a graph of the sound wave. The form of the wave tells the child when something has been recorded, even if they cannot hear it for some reason. SBW falls short in conveying state information with regard to the story as a whole. Allow child to view a slide show of story so far. Allow display of many pages at once so that children can get an overview of the current status. This will help them to evaluate the structure and flow of their stories.
C.2.9	Choose a format for feedback that is suitable for the context in which a product will be used. For example, in a classroom setting audio feedback may disturb classmates. This means designers should have a clear idea of the physical context in which their product will be used.	SBW has limited speech feedback which is an optional function. Adding sounds may become irritating to non-users as children browse the sounds by trying them all out. Recording own voices and sounds requires a quiet environment. SBW is thus not ideal for use in a classroom, unless users use earphones and do not require the recording facility. More suitable for home use where one, two or three children can use it together.
C.2.10	Let children control when speech feedback is given and give them the option to turn it off.	SBW does not use audio feedback, only a text to speech facility that children can choose to use or ignore.

	Speech feedback (e.g. audio cues when moving over an icon with the mouse) will enhance usability and make more functions accessible to younger users. But the option to turn it off must be very clear.
C.2.11 Praise can be given in the form of feedback that expresses positive affect such as surprise, delight and excitement, and it should give the users information about the value of their actions.	SBW currently does not give audio feedback such as praise. Not suitable as the activities are completely open-ended an no specific actions are expected. Can therefore not judge how 'well' the child has performed.
C.2.12 Use praise and flattery carefully. Praise can have a positive effect on interaction, but children will not be convinced if they hear the same praise words every time they do well.	See previous comment.
C.2.14 Rather use non-dispositional praise that evaluates a specific action or behaviour (e.g. 'your handwriting is neat') than dispositional praise such as 'good girl'.	SBW does not include audio feedback such as praise. Can consider including some non-dispositional praise when a child has completed a story.
C.2.15 Neutral feedback after errors (such as a triangle instead of a sad face) will increase the incentive values of reinforcers.	Users cannot make errors in SBW.

C.3 Support through interface agents	
C.3.2 If the child should build up a long term relationship with an agent, let the agent's interaction acknowledge previous encounters.	See B.5.11. Could use an agent for the initial orientation.
C.3.9 Adapt a synthesized voice to the user's personality to make the interaction more pleasurable. Users prefer voices that are similar to their own. Extrovert users prefer extrovert voices and introvert users prefer introvert voices.	If audio feedback will be provided, can determine, at setup time, whether the user is an extrovert or introvert. Set the TTS parameters accordingly. This means you will have to identify the specific user in the beginning of a session. Will not really be advisable when groups of children will be using the software together. Can let children listen to a few voices and let them choose one for TTS function or audio cues.
SBW does not use an agent and I suggest that the software remains without an agent. The remainder of the category C guidelines is therefore excluded from the evaluation.	

D. GUIDELINES FOR ENCOURAGING COLLABORATIVE USE OF TECHNOLOGY

D.1 Supporting peer collaboration	
D.1.1 Collaboration does not come naturally with all children. Offer children opportunities for social interaction where they can learn to work together with their peers.	When recording voices for different characters, suggest that they ask someone else to represent different characters to give them distinguishable voices.
D.1.2 Do not assume that cultural differences inhibit collaboration – technology can become a shared interest that reduces cultural barriers.	As has been observed (e.g. Chimbo and Gelderblom [2008]) cultural differences do not necessarily influence collaboration when using SBW. Gender has a stronger effect.
D.1.3 Determine which kinds of applications or activities facilitate social behaviour at a computer and give preference to those.	SBW naturally encourages collaboration. The recording function is especially conducive to collaboration.
D.1.4 Children prefer advice from peers to that from adults with regard to computer use.	Can use a child voice in the tutorial and for audio cues.

D.2 Environments or interfaces that invite/inhibit collaboration		
D.2.1	Encourage co-construction of solutions to problems so that children are required to interact with each other to come to an agreement on what should be done.	SBW naturally requires children to co-construct stories if they are using it collaboratively.
D.2.3	If collaboration is desired, design activities so that children will gain something by choosing to work together – that is, doing it with someone will make it easier and more fun. Children should ideally discover for themselves the benefits of working together.	Giving different characters different voices by using the voices of different users in a story is obviously better than using one voice for all characters. Can, however, make children aware of this and encourage them to do this to see for themselves the results.
D.2.4	Make the enhanced effect of collaboration versus lesser effect of individual use clearly noticeable in advance.	See above.
D.2.5	Encourage collaboration by: • making the effects of collaboration interesting and not completely predictable. • making behaviour with multiple users a natural extension of the behaviour with a single user. • using sound (or other rewarding) effects as feedback only on collaborative efforts.	See above.

E. THE DIVERSITY OF USERS

E.1 Identity (socio-economics, family and cultural context, gender, personality and language)		
E.1.1	Design technology to reflect a child's context – ideally it should come in multiple languages, reflect gender equity, avoid racial discrimination and portray diverse families, abilities and experiences.	SBW does well in this respect, but could be improved. E.g. no characters with disabilities. Make sure all races are equally represented (there should be a black and a white doctor, a black and a white rugby player, etc.) or children should be allowed to change the skin colour of any character. Include people with disabilities in the choice of characters. Donkey carts need not appear in the 'historical' subsection of the vehicle 'section' as some children's families who live in rural areas may still use that as primary vehicle. Also include a wheel chair. In South Africa all children wear school uniforms. In the School category South Africa users will therefore expect to find children with different kinds of school uniforms.
E.1.2	Design technology to cater for children's variable play preferences that are influenced by their age, gender, socio-economic status, personality, taste, special needs and experience.	See above.
E.1.3	Create a profile of the intended user using information about their age, gender, physical abilities, level of education, cultural or ethnic background and personality.	SBW is intended for a wide range of users and the nature of the software makes it possible to make the content suitable for a wide range of ages, all genders, different cultures and backgrounds, etc.
E.1.4	The specific learning or entertainment goals of the product must fit the context of different kinds of users.	Storytelling is a universal skill that is practiced in many contexts (e.g. play, school, home, therapy). SBW is suitable for use in any of these contexts.
E.1.5	If a product is aimed at children from different cultural	See example of donkey carts and school uniforms

	groups, first investigate how these cultures use and teach the skills that the product will support.	above.
E.1.6	Embed tasks in scenarios that the users can relate to. It may be difficult to find a generic scenario that suits users from different contexts, so, in the same way as some applications let users choose their language of choice, give children a choice of scenarios.	Children decide on the scenarios for their stories and a large selection of possible scenes is available in SBW. Examples of backgrounds that can be added: Recording studio. Doctor's rooms. Different types of homes (e.g. Ndebele village).
E.1.7	Acknowledge the culture and sub-culture of the intended users. Identify particular problems that are important in that culture and the tools typically used to solve that kind of problem.	Examples of subcultures to consider: Skateboarders Musicians Children with specific learning or cognitive disabilities.
E.1.9	From age six, children develop deeper relationships with people outside their homes (designs can model such relationships and allow children to role play).	SBW allows this. Can import pictures of their teachers, idols and classmates and make up stories about them.
E.1.10	Make activities gender-neutral or gender adaptable: • Girls prefer pretend play based on reality while boys prefer pretend play based on fantasy. • Girls lean more toward education and strategy games and boys toward combat and sport games. • Girls prefer a greater variety of play materials than boys. • Boys have been found to complain more about verbose web pages than the girls. It may be that girls are better readers at this age. • Girls on the other hand, complain more about websites that lack good instructions.	SBW is completely open-ended. Remove gender stereotypes (e.g. only female ballet dancers).
E.1.11	Give children the chance to express themselves in different ways, allowing a variety of approaches to perform an activity – sometimes they will want to tell stories, sometimes they will want to make up games and sometimes they may want to build things.	SBW allows this. Children can use existing sound clips or record their own. They can use existing objects or import their own graphics or photos. They can edit objects. They can export their story as a web document or they can print it. Include a 'slide show' option.
E.1.12	Designers must acknowledge their own context and how that may consciously or subconsciously influence their design practice.	The story scenes, objects and characters included in SBW is obviously influenced by the context of the designers. Make sure all possible user groups are accommodated.

E.2 Existing knowledge and experience		
E.2.1	Determine whether users will be novices, experts or a mixture of beginners and advanced users and design accordingly. Users with different levels of expertise will require a layered approach. Give novices options to choose from and protect them from making mistakes. As their confidence grows they can move to more advanced levels. Users who enter the system with knowledge of the tasks should be able to progress faster through the levels.	SBW has a variety of potential users, novices, pre-readers, readers, even teenagers. Children can choose to use or ignore functions that are too simple or too complicated for them. SBW allows a complete novice to create a basic story very easily. But it also allows expert users to create complex stories with many different story elements (including voice recordings and imported graphics).
E.2.2	Keep complexity levels low for beginners but provide a high enough ceiling to allow children of different levels of cognitive development to benefit. Matching children's cognitive competencies can be achieved through additional cues for difficult concepts.	See above.
E.2.3	Enable frequent users to use shortcuts and allow them to skip introductions and instructions that they already know. Make sure children are aware of these options.	SBW allows users to skip the introductions. Include audio cues and an audio help facility, but make it clear that this can be switched off.

E.2.4	Allow children to apply their real-world or other computer-based knowledge when interacting with a new system.	SBW's recording function uses icons that resemble the buttons on a recording device. They can also apply their real-world knowledge when choosing scenes, sounds and objects. When choosing a menu item the mouse button must be held down while moving down an option list. Change this to be similar to the Windows standard. Currently the print, font and save dialogs are internally consistent with the appearance of other SBW functions but it differs from the standard Windows dialogs. Advisable to make them resemble the standard format.
E.2.5	Use appropriate context cues to activate prior knowledge that children can apply to make sense of what they perceive. Keep in mind how the context may influence their interpretation and response.	Icons can become animated when the mouse moves over them to provide cues for pre-readers.
E.2.9	Activities must help the child to fit the information presented into existing knowledge schemes, adapt existing schemes to incorporate the new information, or to combine existing schemes to form more complex schemes.	Currently SBW doesn't really support the development of storytelling skills in the sense that it teaches the child about narrative structure. Can include more examples and point out the different narrative elements to a child (e.g. beginning, climax, ending).
E.2.11	When designing screen-based manipulatives, consider how children with different backgrounds and levels of experience will react to these. Children with scripts that support the interpretation of the visual representation of the manipulative, will have an advantage.	E.g. a child who has never seen or used a recording device will probably not understand the metaphor. Can solve this by including an audio Help icon. If the user clicks on that the currently selected function is explained.
E.2.12	In products designed for classroom use, allow teachers to customise activities according to the children's needs and abilities.	Include a 'Set up' function where the following can be switched on or off: Audio cues. Availability of TTS facility. Availability of recording facility.

F. INTERACTION ENVIRONMENTS AND DEVICES

F.1 Input and output devices		
The Mouse		
F.1.1	Children aged five to eight can use the standard mouse as well as a smaller mouse successfully – the size is not an obstacle.	SBW uses the mouse as input device. The only other input device is a microphone for the recording function.
F.1.2	Marquee-type selection is hard for young children as they find it difficult to select the initial corner correctly. A more suitable way to implement this is to allow the child to select a group of objects by circling the objects with the mouse.	In the object editor users can select part of an image using marquee-type selection. This can be replaced with circling the objects, but children do not seem to struggle with the marquee-type selection required in SBW as this is the standard way of doing it.
F.1.3	When designing for children younger than six, give both mouse buttons the same functionality. They find it difficult to distinguish between left and right and will therefore find it difficult to click the left mouse button consistently.	Currently SBW does not use the right mouse button for anything, so it can just as well have the same functionality as the left one.
F.1.4	Help children to stop mouse movement on target by: • using large hotspots (clickable areas) that are widely spaced, • placing frequently used hotspots in a corner where it is easy to stop the mouse.	SBW icons are large enough – children do not seem to have a problem using them.
F.1.5	Point-and-click (or click-and-carry) is a quicker and	To move objects around on a page, users currently

	more accurate way for children to move objects on the screen than drag-and-drop (with the mouse button held down).	drag-and-drop them. This seems natural and I haven't observed them trying to use click-and-carry instead. When re-designing, experiment with both methods to see which children prefer. This may be a case where there is a kinaesthetic connection between holding the mouse button down and holding on the object to move it.
F.1.6	Drag-and-drop may be better for tasks where the kinaesthetic connection between holding the mouse button down and 'holding on' to the object involved contributes to successful performance of the task.	See above.
Speech recognition and speech output		
F.1.21	Users prefer voices that are similar to their own. Extrovert users prefer extrovert voices and introvert users prefer introvert voices. Adapting a synthesized voice to the user's personality can make the interaction more pleasurable.	Give children a choice a voices for speech feedback, audio cues and TTS.
F.1.23	Children's perceptions about technology are strongly influenced by the emotional tone of speech output.	See above.
F.1.24	Include natural variation in speech output.	SBW's current TTS voice is a monotone adult male voice. If there are different characters children should ideally be able to choose a TTS voice for each different character. The choice should include male, female, adult, child, US or British pronunciation, etc.
F.1.25	If characters are based on familiar characters the voices must be consistent with the known voices.	SBW does not use any familiar cartoon characters so this is not relevant. To include familiar characters such as Barney or Hannah Montana will have copyright issues that will complicate the re-design. I have not come across a child who was specifically searching for familiar characters. If they have access to digital photos of such characters they can easily import them into the story, but the voice issue is then not relevant.

F.2 User control vs. system control		
F.2.1	Children are attracted to activities where they can be active participants. They want to design their own activity patterns. Let them decide what they want to do, how fast they want to do it and when they want to end.	SBW is completely open-ended and children can decide in all respects what to do.
F.2.2	Giving the child control over what happens on screen will promote engagement and self-directed exploration.	SBW is proof of the fact that they are engaged by being in control.
F.2.3	User pre-emptiveness is preferable to system pre-emptiveness in children's interfaces. In other words, the user should have freedom to initiate any action. A system-pre-emptive dialogue is appropriate if children have to perform specific actions at specific times or places in the program.	Although SBW keep children engaged, they often do not complete their stories or do not keep to the original story line. It would be a good idea to build in a way to judge the completeness or logical flow to keep the child focused.
F.2.4	Unless a specific sequence of actions is necessary, allow children to go directly to their favourite parts of the system. Children like to play their favourite games over and over and will find it frustrating if they cannot reach them easily.	SBW allows the child to go directly to any function.

F.3 Engagement		
F.3.1	Use the computational capacity of the computer to enhance learning and engagement – that is, visual displays, animated graphics, speech, recording progress, detection of and adaptation to individuals.	SBW does this to a large extent (graphics, voice recording, TTS) but it could use more – e.g. recording progress so that children can be given advice on narrative structure.
F.3.3	Adults and children have different ideas about what is boring or exciting and designers should avoid using their own definitions of such concepts to guide their design decisions.	SBW shows some evidence of choices made by adults instead of children. The icons on the opening screen can be interpreted by adults or older children but not necessarily by younger children. If all functions were tested with child users the designers would have realised that the Color function has problems (i.e. Zone 1 and Zone 2 – which will mean nothing to even an adult user).
F.3.4	Make interfaces more engaging by letting them evoke emotional responses from users.	Adding sounds to pages and objects and successfully recording their own voices lead to lots of excitement.
F.3.5	Forestall situations where children can repeatedly fail at a task as this will cause them to lose interest. There must, however, be a balance between the challenge provided and the child's ability to perform the activity.	SBW activities are easy to perform – I've seen children struggling to edit an object, but they remain relatively patient. When it becomes an impossible task they will discard the object and use something else instead.
F.3.7	Offer children a variety of paths of interaction.	SBW has no set path for story creation. Children can progress according to their own preferences.
F.3.10	Provide children with opportunities to construct things that perform some kind of behaviour, as this causes high levels of interest and excitement.	Adding sounds or voices to the story objects and characters provides lots of excitement. Suggestion: Can build in a way to let objects move, jump, dance etc.
F.3.11	Activities should concentrate on the pleasure of performing the activity and not on successfully reaching the end product – if the end result is more important than the activity, children lose interest.	In SBW the process is definitely more important than the end product.
F.3.12	Include humour, warmth and spontaneity in an interface to increase children's motivation to use it.	To bring in humour, SBW can suggest storylines where boys wear dresses or where adults call children Sir and Madam, or where the role of children and teachers are reversed, etc.

F.7 Interface design		
Interface elements		
F.7.2	Interface tools that reflect how often they have been used may help children to see the relationship between actions and outcomes. This can be achieved through the use of colour, size, shape or pop-up messages.	Can have a section for frequently used objects where children can get quick access to recently or often used objects.
F.7.3	Use static and dynamic defaults to support interaction. Static defaults are defined within the system or acquired at initialization. Dynamic defaults evolve during the interactive session.	See suggestion for frequently used objects above. Author field can already contain the regular user's name when a new story is opened.
F.7.4	Successful interpretation of an icon depends on its caption (what it is meant to communicate), the context in which it will appear, and the image.	SBW has many icons that may be difficult to interpret. Will discuss all of these in detail and give suggestions for improvement.
F.7.5	When designing icons for children's interfaces, keep in mind that: • The design of icons must draw on children's existing knowledge. • Children prefer icons in colour and find black and white icons more difficult to recognize. • They prefer boxed icons.	Some of the problem icons are: The arrow for 'Getting started'. The book for 'Reading a story'. A bookshelf for 'Opening or editing an existing story'. The feather pen for 'writing a story'.

	• Children recognize animated icons more easily than static ones and they like it when the icons come alive when the mouse moves over them. • Icons with linguistic cues are not suitable for five to eight-year olds. • Icons should not be culturally specific.	The 'Undo' icon in the object editor. The traffic light to return to the main menu.
F.7.6	Always involve users when selecting images for icons. Adults cannot predict what children will think or like. Give them options to choose from.	Do this extensively when re-designing.
F.7.7	Children expect an interface to be interactive and animated.	Can use animated icons – especially on the opening page – to make the functions clear.
F.7.8	There should be a clear mapping between interface elements and their effect on the system.	Use of the same green checkmark icon for different purposes is obviously problematic.
F.7.11	Children have different preferences for fantasies, so it is advisable to provide them with a choice of fantasy in which to embed activities. Designers can also give children the opportunity to help create the fantasy by, for example, letting them choose names for characters of places.	SBW does this in abundance.
Humour		
F.7.13	Keep in mind that preschoolers' sense of humour is still unsophisticated – they think that mispronouncing words or putting clothes on the wrong way is funny (even when they have done it repeatedly) and they do not understand humour that involves irony or satire.	Can include more potentially humorous characters.
Visibility		
F.7.17	Only operations that are available should be visible, or it should be very clear which operations are not available. Use age appropriate means to make available operations known.	Users can add objects to the title page, but these cannot be edited. The editing buttons should be greyed out. Much of the functionality can only be reached via the menu. Children will not necessarily explore the menu options (especially if they cannot read) and will miss functions such as the spellchecker and the TTS facility.
F.7.18	Make it very clear to the user what the next required action is.	In SBW the next action is usually completely up to the user. One problem in this regard is when selecting objects from the object selector the user has to click + to add an object and √ to close the window. Can cause confusion about what to do.
F.7.19	Do not expect children to use browsing as a way to get a clear picture of the system's current state. If the application involves reaching a series of sub goals to achieve some central goal, it may be necessary to give them (or an adult assisting them) the option to view their current progress.	Currently SBW does not allow children to view their stories with more than one page on the screen. The only way to get an overview is to page through the story from beginning to end. Provide children with an option to view a number of reduced size pages at a time to give them an overview of their story.
F.7.20	Convey available functionality through highly visual interface components and not through textual representations, so that children who cannot yet read can use the system.	Already discussed above (F.7.17).
F.7.21	Create a transparent interface that enables children to focus on what must be done and not on how they should use the interface.	SBW succeeds in this. The interface is very easy to use and learn, although children may initially need help to interpret the icons correctly or to get access to functions that are not made visible on the main screen. Suggestions:

		Colour dialog includes radio buttons called Zone 1 and Zone 2. User has to find out by trial-and-error what the purpose of these are.
Consistency and familiarity		
F.7.22	Children should be able to determine the effect of future action based on past interaction history. Make changes to the internal state of the system visible so that users can associate them with the operations that caused them.	Inconsistency in the use of the x, + and √ buttons. Save, Font and Colour dialogues differ substantially from the Windows standards for these functions.
F.7.23	Interaction and input-output behaviour should be consistent within a system as well as across systems. The user should be able to extend knowledge of specific interaction within and across applications to other similar situations.	Save, Font, Colour dialogs are not consistent with Windows standard. Can still give them a SBW look, but make its appearance and operation comparable with the standard.
F.7.24	Familiarity can be achieved through metaphors and through affective use of affordances that exist for interface objects. The appearance of the object should promote familiarity with its behaviour or function.	Not always done in SBW. Icons (e.g. those on the opening page) do not all have a clear connection with their functions.
F.7.25	Metaphors should draw on children's existing knowledge so that they can easily see what to do and predict the outcomes of their actions.	SBW makes use of the paper-based book metaphor in some of its icons. The connection can be made clearer through the use of animated icons on the opening page. The voice recording function uses the metaphor of a manual recorder – the buttons for record, replay, pause, etc. follow the standard on most recording devices.
F.7.26	Surprise is often a desirable element in children's games and can increase the experience of fun and engagement. However, when it comes to learning how to use a system and navigating through the available functions and activities, predictability is very important. If they performed an action before, they will expect the system to behave similarly when they perform that action again.	In this application children should create their own surprises in the stories. See F.7.22 w.r.t. predictability.
F.7.27	Follow real-world conventions so that information appear natural and in logical order. Familiarity has a different meaning for children than for adults – they 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.	Test the design with children and ask them what they find strange or not expected.
Error prevention		
F.7.28	Help users to recognise, diagnose and recover from errors. Give error message in language that children can understand. Describe the problem precisely and suggest a solution.	SBW handles any situation as a non-error.
F.7.29	Make it easy to reverse an action. Young children should not be expected to know how to use Undo/Redo commands – the system should help them recover from an error through help that fits their level of understanding.	SBW's current undo button on the editing page is not really interpretable as that. Children can easily remove objects that they have placed on a page. Not that easy to remove a sound attached to an object or page – this is done through the menu.
F.7.30	Use constraints that restrict the actions a user can take at a specific point during the interaction as error prevention mechanisms.	SBW does not always disable buttons that represent functions that should not be available (e.g. editing objects placed on the title page).
F.7.31	Design to prevent errors rather than to help users	SBW is good in this regard, but the warning that a file

	recover from errors. Require users to confirm potentially erroneous actions before performing them.	may be overwritten when saving a story should also appear in audio format.
Flexibility and adaptability		
F.7.33	Allow equivalent values of input and output to be arbitrarily substituted for each other, or provide multi-modal input and output mechanisms. Children have varying skill levels and preferences that will influence the type of input or output that is suitable for a specific user. Different modalities (channels of communication) can be combined to improve articulation of input or output or to make the system accessible to more users.	In SBW children can provide story text or they can record a voice over of the story. Include simultaneous audio and text cues to make the system accessible to reading as well as pre-reading children.
F.7.34	The user interface should be: • adaptable (allowing user-initiated modification to adjust the form of input and output) • adaptive (allowing system-initiated modification to customise the user interface automatically).	The Preferences function of SBW is very limited. It only gives a choice between two languages and it lets the child pick a format for the printed version of their story (if they choose to print it). It is not clear at all what the user is setting when selecting between the print options because there is no indication that this has to do with printing. This is explained in the pdf manual that accompanies the software. Suggestions for setting preferences: Audio cues on icons on/off Language Default TTS voice Disabling certain functions (e.g. voice recording if a microphone is not available).
General interface guidelines		
F.7.35	Children's products rarely focus on productivity, therefore efficiency is not as important as in systems designed for adults.	SBW does not focus on productivity.
F.7.36	Create products that are safe to use. Safety in children's products involves: • how children are affected physically by using the system • how they can be affected by accessing material that is not appropriate such as pornography or images of violence • how they are psychologically influenced by the content.	SBW is safe in the sense that it does not expose children to violence or pornography or any other harmful material. It holds no psychological threat. Children may want to play for extended periods so it is advisable that a timer is built into the system to suggest to a child when it is time to take a break.
F.7.37	To determine whether a product has adequate utility, designers should ask whether it allows children to carry out tasks in the way that they would like to do them. For the sake of utility, it is therefore important that designers do not make assumptions about children's preferences.	In terms of utility SBW can provide children with more functionality. I have observed children trying to show the back view of people or objects (e.g. when a family is sitting around a table some members will have their backs to the users). SBW only shows characters and objects from the front or sometimes the side and children can flip them to face left or right or to be upside down or upright. Can however not change the rotation three-dimensionally.
F.7.38	There is a trade-off between user experience goals and usability goals. An action that requires more effort may contribute towards making a product more enjoyable and engaging. Designers should be aware of the consequences of combining user experience and usability goals and make sure that they address the needs of the user.	Despite many obvious usability problems, SBW is still a very successful and engaging application. When re-designing I will be careful, when adding functionality, support facilities, etc., not to detract any elements that contribute to SBW's success. E.g. the fact that it is so open-ended and all control lies with the user, contributes a lot to the user's experience. An interface agent may help to make the user aware of more functions, but it may also make the user feel less in control.

Appendix 2: Miles and Huberman's Standards for the Quality of Conclusions

Below are the lists of relevant queries suggested by Miles and Huberman [1994: 278-280] that make up practical standards that can be used to judge the quality of research findings based on qualitative research. They offer these as useful questions and not as a complete and comprehensive framework that where every questions applies to all qualitative studies.

These questions are used in Chapter 10 of this thesis to evaluate the research process and the consequent findings.

Confirmability (or Objectivity)

1. Are the study's general methods and procedures described explicitly and in detail: Do we feel that we have a complete picture, including "backstage" information?
2. Can we follow the actual sequence of how data were collected, processed, condensed/transformed, and displayed for specific conclusion drawing?
3. Are the conclusions explicitly linked with exhibits of condensed/displayed data?
4. Is there a record of the study's methods and procedures, detailed enough to be followed as an "audit trail"?
5. Has the researcher been explicit and as self-aware as possible about personal assumptions, values, biases, affective states – and how they may have come into play during the study?
6. Were competing hypotheses or rival conclusions really considered? At what point in the study? Do other rival conclusions seem plausible?
7. Are study data retained and available for reanalysis by others?

Dependability (or Reliability or Auditability)

1. Are the research questions clear, and are the features of the study design congruent with them?
2. Is the researcher's role and status within the site explicitly described?
3. Do findings show meaningful parallelism across data sources (informants, contexts, times)?
4. Are basic paradigms and analytic constructs clearly specified? (Reliability depends, in part, on its connectedness to theory.)
5. Were data collected across the full range of appropriate settings, times, respondents, and so on suggested by the research questions?
6. If multiple field-workers are involved, do they have comparable data collection protocols?
7. Were coding checks made, and did they show adequate agreement?
8. Were data quality checks made (e.g., for bias, deceit, informant knowledgeability?)

9. Do multiple observers' accounts converge, in instances, settings, or times when they might be expected to?
10. Were any forms or peer or colleague review in place?

Credibility (or Internal Validity or Authenticity)

1. How context rich and meaningful ("thick") are the descriptions?
2. Does the account "ring true", make sense, seem convincing or plausible, enable a "vicarious presence" for the reader?
3. Is the account rendered in a comprehensive one, respecting the configuration and temporal arrangement of elements in the local context?
4. Did triangulation among complementary methods and data sources produce generally converging conclusions? If not, is there a coherent explanation for this?
5. Are the presented data well linked to the categories of prior or emerging theory? Do the measures reflect the constructs in play?
6. Are the findings internally coherent; are concepts systematically related?
7. Were rules used for confirmation of propositions, hypotheses, and so on made explicit?
8. Are the areas of uncertainty identified?
9. Was negative evidence sought for?
10. Have rival explanations been actively considered? What happened to them?
11. Have findings been replicated in other parts of the database than the one they arose from?
12. Were the conclusions considered to be accurate by original informants? If not, is there a coherent explanation for this?
13. Were any predictions made in the study, and how accurate were they?

Transferability (or External or Validity Fittingness)

1. Are the characteristics of the original sample of persons, settings, processes (etc.) fully described enough to permit adequate comparisons with other samples?
2. Does the report examine possible threats to generalizability? Have limiting effects of sample selection, the setting, history and constructs used been discussed?
3. Does the researcher define the scope and the boundaries of reasonable generalization from the study?
4. Is the sampling theoretically diverse enough to encourage broader applicability?
5. Do the findings include enough "thick descriptions" for readers to assess the potential transferability, appropriateness for their own settings?
6. Does a range of readers report the findings to be consistent with their own experience?
7. Are the findings congruent with, connected to, or confirmatory of prior theory?
8. Are the processes and outcomes described in conclusions generic enough to be applicable in other settings, even ones of a different nature?
9. Is the transferable theory from the study made explicit?

10. Have narrative sequences (plots, histories, stories) been preserved unobscured? Has a general cross-case theory using the sequences been developed?
11. Does the report suggest settings where the findings could fruitfully be tested further?
12. Have the findings been replicated in other studies to assess their robustness? If not, could replication efforts be mounted easily?

Application (or Utilization or Action Orientation)

1. Are the findings intellectually and physically accessible to potential users?
2. Do the findings stimulate "working hypotheses" on the part of the reader as guidance for future action?
3. What is the level of usable knowledge offered? It may range from consciousness-raising and the development of insight or self-understanding to broader considerations: a theory to guide action, or policy advice. Or it may be local and specific: corrective recommendations, specific action images.
4. Do the findings have catalyzing effect leading to specific actions?
5. Do the actions taken actually help solve the local problem?
6. Have users of the findings experienced any sense of empowerment, of increased control over their lives?
7. Have users of the findings learned, or developed new capacities?
8. Are value-based or ethical concerns raised explicitly in the report? If not, do some exist implicitly that the researcher is not attending to?