CARSS: A Framework for Learner-Centred Design with Children

Judith Good, IDEAS Lab, Department of Informatics, University of Sussex, Brighton, BN1 9QH, UK
J.Good@sussex.ac.uk

Judy Robertson, School of Mathematical and Computer Sciences, Heriot-Watt University, Edinburgh, EH14 4AS, UK
Judy.Robertson@hw.ac.uk

Abstract. Learner-centred design (LCD) is a nebulous concept. It can range from attempts to design with the needs of the learner at the forefront, to involving the learner at various stages of the design process, sometimes throughout the whole process. In addition, learner-centred design involving children implies additional issues which do not present themselves when using an LCD approach with adults. In this paper, we argue that current LCD frameworks do not consider the full range of issues necessary for successful design. We propose CARSS (Context, Activities, Roles, Stakeholders, Skills), a learner-centred design framework specifically for child learners, and describe two design case studies which demonstrate the framework in use.

Keywords. Learner-centred participatory design, design frameworks, software design with children

INTRODUCTION

Learner-centred design (LCD) is becoming an increasingly pervasive approach in the design of interactive learning environments, and justifiably so. Given the desire to implement systems which encourage and support rich forms of learning, the needs of the learner must necessarily take centre-stage. However, the term appears to mean many different things to many different people. For some, the focus is on the design product, i.e. the educational software being produced (Soloway et al., 1994; 1996). For others, the focus is on the design process, and how to incorporate the learner’s perspective in that process in some way (Druin, 1999; 2002; Scife & Rogers, 1999).

Complicating the issue further, in many cases the learners we are targeting are children. The question then arises as to the extent to which we, as adult designers, are able to understand the needs, skills and motivations of child learners in such a way that the system is truly learner-centred. What children want and expect from educational software is likely to be different from what adults want and expect. In addition, what children want and expect may well be different from what adults think children want and expect. Finally, our experiences from field work and teacher training workshops suggest that adults often have an inaccurate picture of children’s level of software competence, typically underestimating their capabilities. In this paper, we review the current literature on learner-centred design and its variants, particularly as it relates to children as learners. We suggest that while many of the current methodologies are important steps toward fostering a learner-centred design process, they rarely consider all of the potential parameters involved in learner-centred design.
Finally, we present CARSS (Context, Activities, Roles, Stakeholders, Skills), a new framework for learner-centred design with children, and describe two case studies which show the use of the framework. In choosing the case studies, we selected one involving the design of an intelligent learning environment, and one involving a web-based learning environment. Our work with CARSS suggests that the framework is applicable to both intelligent and “non-intelligent” learning environments.

LEARNER-CENTRED DESIGN: A BRIEF OVERVIEW

Soloway, Guzdial et al. (1994) draw a parallel between the development of general computer systems, where design moved from a focus on the system itself to a focus on the user of the system (user-centred design), and the development of learning environments, suggesting that a similar change in focus should make the learner the focus of the design process (learner-centred design).

They posit that the three concepts of import in a user-centred design approach (tasks, tools and interfaces) hold for learner-centred design, but that the model requires the inclusion of a fourth concept: the learner’s needs, leading to the TILT model (tasks, interfaces, learner’s needs, tools). The authors then describe how, in an educational system, the relationship between tasks, tools and the interface, on the one hand, and the learner’s needs, on the other, can be mediated through scaffolding. As an example, the scaffolding which links the learner’s needs to a particular task may take the form of coaching, i.e. providing suggestions and guidance as the learner performs a task.

This view of learner-centred design is very much product oriented: it describes the necessary components of a system designed according to a learner-centred approach, namely scaffolding of various types, but does not describe the process of learner-centred design and, as such, does not provide explicit guidance for designers wishing to implement a learner-centred design process.

Conlon and Pain’s (1996) Persistent Collaboration Methodology (PCM) provides a thorough account of the process of designing intelligent educational systems (or “applied AIED”, in their terms). They view their methodology as hailing from a combination of user-centred design and action research. Although they do not explicitly state that they are targeting child learners, their methodology involves teachers in the design process, and their example describes the use of an expert system shell in schools. One of the hallmarks of this methodology is that it involves teachers, researchers and technologists in an ongoing cycle of observation, reflection, design and action. Conlon and Pain provide a detailed case study which shows the extent of collaboration of all parties, and gives the sense that each member of the design team felt themselves to be an integral part of the project. Their involvement in each phase was deliberate, methodical, and well-reasoned. However, the Persistent Collaboration Methodology could be seen to be more school-centred than learner-centred, given that learners were not members of the design team. This suggests an implicit assumption that teachers can somehow act as spokespersons for the learner, and that the learner’s needs are in fact subsumed by the requirements of the curriculum in question.

Many researchers have argued that rather than second-guessing learners, and making assumptions about their needs, the learner must be involved in some way in the design process in order to create a truly learner-centred product. Although an obvious stage for learner involvement is at the prototype testing stage, other methodologies have taken this further, and suggested that the learner should be involved in the design process from the very start.
Druin (1999, 2002) in particular has taken the general notion of participatory design, where the end user is involved in the design process, and narrowed the focus to children as end users, resulting in a methodology which she terms cooperative inquiry. Cooperative inquiry at its outset involved children in the design of new technologies for children using a three step process. Firstly, contextual inquiry involves the collection of data in the user’s own environment, typically children’s homes. The data, taken in the form of notes, is then synthesized and analyzed, with design guidelines resulting from the process. A participatory design phase typically follows from contextual inquiry. In this phase, the design team, which includes children, takes part in low-tech prototyping sessions. A final technique in cooperative inquiry is technology immersion. This involves creating a space where children are able to access and use a wide variety of existing technologies over a sustained period of time with researchers observing children’s activity patterns in an unconstrained setting.

Druin’s inclusion of various types of activities within the overall conceptual framework of cooperative inquiry has changed over the years, with a phase of “sticky note critiquing” inserted between the phases of contextual inquiry and participatory design (Druin et al. 2001). In this activity, children and adults critique an existing piece of technology and, using sticky note pads, record their likes, dislikes and a third category, for example, surprises. They are asked to write approximately three of these per category. All of the sticky notes are then organized graphically by topic in a summary of the teams’ opinions of the software. In addition to the inclusion of new techniques, the existing techniques have been adapted for work with younger children (Guha et al., 2004).

From a learner-centred design perspective, Druin’s work offers a number of ideas that can be reused. At the same time however, LCD comprises a number of assumptions that cooperative inquiry does not address. The focus in cooperative inquiry is on children as technology users, whereas in LCD, the focus must necessarily be more constrained, and focus on children as technology learners, i.e. children who use the technology as a vehicle for learning. This means, for a start, that children who are involved in the design of interactive learning environments for their target age group will, by definition, not have attained the learning objectives which the software aims to impart in the first place. Additionally, Druin’s work has a laudable aim in that it is looking at the development of new forms of technology, and helping children to imagine what does not yet exist. In the case of designing interactive learning environments, this push for new forms of technology must be tempered by constraints relating to best practice in pedagogical theory, best fit with the existing curriculum, and likelihood of use in classrooms as they are currently set up. This does not mean that we can’t, or shouldn’t, involve children in the design of educational technologies which push the boundaries of existing practice, simply that this is not always possible or appropriate.

Of course, involving children in the design of learning environments does not necessarily require them to be design partners, i.e. to be involved in the design process from the outset. Druin (2002) describes other forms of involvement in the design process, beginning with the least involvement, the technology user, and moving to increasing levels of involvement, with children acting as testers of prototype software, informants, who give input into the design process at various times, through to fully fledged design partners, acting as equal stakeholders throughout the design process.

A number of researchers have contributed with guidelines for all of these roles. For the “child as tester” role, Hanna, Risden et al. (1997) have developed guidelines which deal with issues specific to conducting usability testing with children. Similarly, Markopoulos and Bekker (2002) describe a number of parameters relative to usability testing methods with children which can help to distinguish and compare usability tests. Rode, Stringer et al. (2003) have looked at how to perform testing in a
way that fits within the constraints of the English National Curriculum so that testing can be carried out in schools during normal lessons.

Increasing the level of participation to informant, Scaife, Rogers, Aldrich and Davies (1997) provide an ‘informant design framework’ in which stakeholders including children, teachers, software designers and psychologists contribute to the design process for interactive learning environments. The contributors are involved in specifying the learning goals and teaching practices within the domain, and translating the specification into low-tech, and later high-tech designs. In this approach, the stakeholders do not work as an integrated team at all stages of the project. Rather, expertise from different stakeholder groups is invited on specific aspects of the learning environment throughout the design process. Scaife et al. (1997) provide suggestions of design methods which can be employed by different stakeholder groups such as storyboarding, interviews, high-tech prototyping in multimedia environments and cognitive and interactivity analysis. However, although the authors allude to some of the difficulties they encountered in working with children as informants, they do not discuss in detail the skills required by child and adult members of a design team. We believe that an understanding of the necessary skills for participants in a design process is required before embarking on a learner-centred design project in order to ensure that the learning environment will be deliverable within the funding constraints. Although some research projects have invested heavily in the skills development of child designers, this is often not possible in smaller research projects, or those in which commercial partners are involved. An insight into the skills required of child designers is useful in deciding the extent to which children can be involved in a project, and in recruiting both child and adult members of design teams.

In summary, we have looked at the definition of learner-centred design, and the ways in which it can be applied to the design of educational software specifically for children. In so doing, we have considered methodologies which look at the process of designing educational software (the Persistent Collaboration Model, (Conlon & Pain, 1996)), at methods which involve children in the design process (Cooperative Inquiry (Druin, 2002)), and the Child as Informant framework (Scaife & Rogers, 1999).

Although there are lessons to be learned from all of these approaches, we feel that no single existing model fully captures the process of involving children, and other relevant stakeholders, in the process of designing learning environments, intelligent or otherwise. This is because they either 1) do not consider the process of design, 2) do not fully consider the roles of all stakeholders, 3) don’t focus on the particularities of designing educational software, or 4) don’t take into account the skills required by the design team members. To this end, we present CARSS (Context, Activities, Roles, Stakeholders, Skills), a framework for the participatory design of interactive learning environments (ILEs), which we describe in detail in the following section.

**CARSS FRAMEWORK**

CARSS is a framework for participatory, learner-centred design involving children. It attempts to provide a fully inclusive design framework which can be used for the design of ILEs, intelligent or otherwise. CARSS is comprised of five primary components:

*Context* concerns an awareness of the broader context in which the design activities take place: although learner-centred design with children can sometimes be carried out in the rarefied atmosphere of a lab designed specifically to accommodate LCD activities, in many cases, the design sessions will
take place within the constraints imposed by a number of sources, either physical, such as the school environment, or virtual, such as the constraints inherent in producing commercial quality software.

*Activities* describes the sequence of events which occur in the typical educational software design cycle, highlighting those that are specifically relevant to work with children.

The *Roles* category describes the various functions which a design team member can fulfil within the team. Often, an adult team member may fulfil more than one role.

The *Stakeholders* category covers all of the individuals who have a vested interest in the design process: although balancing the conflicting needs and desires of the various stakeholder groups is sometimes a difficult process, we believe that ensuring the largest possible number of stakeholders are represented is likely to lead to the most successful outcome.

*Skills* cover the personal attributes and dispositions necessary to conduct successful design sessions. We feel that the skills needed by children to be successful design partners have not been fully described in previous frameworks. In addition, we believe that the skills of the adult design partners are equally as important, but have, to date, received no attention.

From a theoretical perspective, the CARSS framework has its roots in distributed cognition (Hutchins, 1995; Hollan, Hutchins, & Kirsh, 2000). The learner-centred design context can be seen as a cognitive system, with cognitive processes distributed across that system. Hollan et al. (2000) note that human activity which occurs “in the wild” involves the distribution of cognitive processes across group members, “coordination between internal and external (material or environmental) structure” (p. 176), and the distribution of processes across time such that “the products of earlier events can transform the nature of later events” (p. 176). These processes are evident in the work of our learner-centred design teams: novel ideas arise through the interactions between team members, involve the coordination of internal and external structure (in the form of cognitive off-loading, and the use of numerous external artefacts, tools and representations), and the design, and indeed the design activities, evolve over time as a partial function of the team’s earlier activities.

From a practical perspective, our use of the term framework is consistent with Rogers and Muller (2006) who suggest that “within HCI, it is commonly used to describe a form of guidance that is explicated in a particular way to inform design and analysis” (p. 3). They further suggest that frameworks fulfil various functions, ranging from explanatory to predictive. The position of a framework along this continuum will determine whether it consists of “a series of steps or principles to be followed” (p. 3), in the case of a predictive framework, or “a set of concepts or dimensions to be considered” (p. 3), in the case of an explanatory framework. We feel that CARSS sits in the middle of the predictive-explanatory continuum: on the one hand, it aims to allow researchers and/or project managers to plan and lead a learner-centred project with an awareness of the numerous parameters which constitute a design context (predictive). At the same time, it can be useful in determining the extent to which the design process was successful (explanatory). Nonetheless, it would be unwise to view the framework as a sort of recipe which somehow guarantees a successful design project. Instead, it should be used in much the same way as Malone and Lepper’s (1987) taxonomy of intrinsic motivations for learning, about which the authors state that is a “checklist of heuristics for designing instructional activities” (p. 247). In the same way, when using CARSS, simply including every possible stakeholder into the mix will not guarantee a successful design outcome. Instead, the list of potential stakeholders should be weighed up within the context of the project and its overall aims, in order to determine the most relevant stakeholders.

The CARSS framework is shown in Figure 1, and described in detail below.
Learner-centred design for children, which involves children throughout the design process, is often subject to a number of constraints. Although in some large, generously funded research projects some of these constraints can be bypassed, in most cases, researchers will need to be aware that various constraints exist, and to plan their research around them in a way that ensures that the research can occur as harmoniously as possible. The five constraints which we have identified are described in the following subsections.

**Curriculum constraints**

In designing an ILE, it is important to be aware of the curriculum constraints that exist in the particular country, or indeed region, in which one is doing research. As described above, this differs from the cooperative inquiry approach, where the goal is to imagine technologies for children which don’t yet exist. Here, if one hopes to deploy the ILE being designed in schools in the future, then it should be designed so that it can support the standard curriculum, either because the curriculum itself is explicitly designed into the ILE, or because the tool is sufficiently open-ended that it can be adapted into the curriculum (in which case, support materials should be provided to allow teachers to do this). In terms of the stakeholders, described below, this area is the one in which teachers will likely have the most input.
In the design of Webkit, a tangible user interface for teaching rhetorical skills (Stringer et al., 2004), researchers selected their topic focus such that it fit well within the English National Curriculum (Rode et al., 2003). Doing so had the added advantage of allowing researchers to gain access to a local school to try out their prototypes, given that they weren’t taking time away from the material that the teacher was required to cover.

**Timetable constraints**

Unless the design sessions can take place in a setting away from school (where there will nonetheless be some constraints on time), there are a number of constraints which must be taken into consideration when planning design sessions. Children may need to be taken out of ordinary lessons, the sessions may have to be relatively short, and school holidays may occur at points which are inconvenient in terms of the design process. It is therefore crucial to plan and prepare for the sessions very carefully in order to extract the maximum benefit. A number of accounts of research projects describe these constraints, and ways in which the researchers tried to work within them while at the same time maximising their pay-off in terms of the knowledge gained from their child design partners (see, for example, (Rode et al., 2003), and (Goolnik et al., 2006)).

**Environmental constraints**

The environment for the design sessions is also a factor in the success of the project. At a practical level, the facilitator should negotiate with the school to reserve a regular venue for the sessions. It is helpful to have a quiet room with space for the children to write and draw, and where sessions will not be interrupted by other pupils. On occasions where access is required to computers, a self-contained computer suite is preferable to an open plan area because high levels of background noise can make it difficult for the children to concentrate.

Again, in research projects with substantial budgets for learner-centred design, or in commercial usability settings, environments may exist which are optimal for the purpose. In other cases however, the constraints of the environment may be such that they compromise the design process (Knudtzon et al., 2003).

**Commercial constraints**

Some projects, which are either unfunded, or funded by research councils, may have relatively few commercial constraints. On the other hand, when a project involves an industrial partner, it becomes subject to the constraints that commonly operate in the commercial world. These include working to short time scales, and ensuring that the product is both feasible to develop and profitable. Children are unlikely to have had experience with these constraints, and care must be taken to ensure that researchers are both aware of these constraints, and manage them in the best possible way so that the project accomplishes its goals, and children do not end up feeling frustrated, rushed, or devalued.

In some cases, the role of the child designers may not be as extensive as one would like. Antle describes the development of an online collaborative storytelling environment for the Canadian Broadcasting Corporation thusly: “The challenge was how to include children in a fast-paced, low budget development cycle in ways that would result in informative and usable input and feedback” (Antle, 2003, p. 61). In the end, the project developers were forced to scale back their child
involvement, resulting in uncertainty as to whether they had actually met the needs of the target audience.

Legal and ethical constraints

When working with children in the context of a design project, a number of formal approvals may be required. Given that this will differ from country to country and, in some cases, from institution to institution, it is difficult to provide specific guidelines in this matter.

At the very least however, working with a child design team will involve written approval from the university (if involved), for example, via the institution’s review board or ethics committee. If the design work is conducted in a school, it will obviously require approval from the school’s head teacher. Additionally, parental agreement will need to be obtained, as will agreement from the children. As the UN Convention on the Rights of the Child (UNICEF, 1989) states that children have a right to express their views in matters concerning them, and equally have the right to be listened to, child team members should be informed of their role in the project before they agree to take part and should have the option to drop out of the team if they so desire.

Adults working with children face to face need to be aware of the relevant child protection laws, and should have undergone the required background checks. For example, in Scotland, every adult who works with children or vulnerable adults must undergo a Disclosure Scotland police background check.

Finally, once the design team is in place and functioning, adults need to continue to be aware of issues of child protection, and seek advice from the school and local education authority for best practice in the area. These issues may arise, for example, when considering children’s participation in online discussions, or the use of images and/or films of the design team.

Roles

In the CARSS framework, we distinguish between roles and stakeholders. Roles designates the various functions which the members of the design team can fulfil. In the latter, we focus more specifically on the groups which have a vested interest in the design. Ideally, the relationship between roles and stakeholders will be such that the members of the design team take on roles which allow them to be representative of the stakeholders.

Below we consider the roles which members of the design team may take on. Note that there may be some overlap in roles and in the case of a small project, a single person may play more than one role. For example, the researcher on the design team may often be a learning scientist and, in addition, may play the role of collaboration facilitator.

Design partners

In the CARSS framework, children function first and foremost as design partners. Within that role, they contribute to the full range of design tasks, from requirements gathering, through to evaluation. In addition, they act as representatives of the child learners for whom the software is being designed. As noted in the introduction, it would be imprudent for adults to fulfil this role, as it would require them to second guess children’s capabilities, and their preferences.
**Project manager**

In order to oversee the learner-centred design process, one person must act as project manager. The project manager will ensure that the project progresses according to schedule, and that the various deadlines are met. The role of project manager is all the more crucial in the case where children are involved in the design of a system, as they will typically not have had experience of working to external deadlines, etc. One of the roles of the project manager will be to tread a fine line between ensuring that the children feel that their contributions are valued, and that the project advances as necessary.

**Technology specialists**

It goes without saying that in projects devoted to the design of educational technology, the role of the technology specialist is central. The technologist provides expertise in current and emerging technologies which may be appropriate to the educational design problem. Her advice will be crucial in determining the technical feasibility of ideas raised by child and non-technical adult team members, and forecasting the time scale required to develop software features. Furthermore, her software development skills will be required to implement a series of high-tech prototypes throughout the design process, which will then be the subject of discussion and evaluation by the other team members. In later stages of the design project a larger team of software developers are likely to be needed to turn the prototype into a product.

**Researchers**

In projects which take place in a university context and which may be funded by research councils, the role of researcher is a central one. Typically, the software being developed will be innovative in some way, thus necessitating a researcher to provide the necessary theoretical knowledge and/or guidance to ensure the project fulfils its aims. Alternatively, or in addition, the innovation may lie in the project’s methodology and again, the researcher will play a central role in ensuring that the methodology is followed and appropriately documented.

**Subject matter experts**

When designing software for a particular subject matter, it may be appropriate to bring in a subject matter expert, who can assist in developing the material. In some cases, a teacher may be able to fill this role. However, teachers sometimes teach outside their area of speciality and teachers required to teach across a broad spectrum of subjects may not have sufficiently deep knowledge of a particular domain. In this case, it may be necessary to include expert teachers and/or domain experts in the design team.

**Child development experts**

Another potentially important role in the design team is that of child development expert, particularly where new technologies for younger children are concerned. When designing technology, it is important to ensure that the subject matter is presented in an age appropriate manner, and that activities are planned in a way that takes into account attention spans, ability to concentrate, etc.
Again, teachers may in fact possess this knowledge and understanding, but the support of an expert in child development could also be useful.

**Learning scientists**

When designing educational technologies, it is important that they are grounded in, and consistent with, theories of learning and our current understanding of how learning occurs. In the case of projects which are funded by research councils, the principal investigator or research fellow often assumes this role. In commercial projects, it is important to identify a team member who can fulfil this role.

**Collaboration facilitator**

In projects which involve a number of design team members, a “collaboration facilitator” is almost always necessary. In most cases, this will be a role played by the researcher and may involve a number of different tasks. For example, the collaboration facilitator will need to manage the collaboration (face to face or remote) between industrial partners and children, as this is likely to be a very new experience for each party. Similarly, when requirements gathering with children and teachers, part of the collaboration facilitator’s role will be to integrate the child and teacher perspectives in a harmonious and useful way. The greater the number of stakeholders, the more important this role will become.

**Stakeholders**

In the CARSS framework, we consider the broadest range of potential stakeholders in projects which aim to design interactive learning environments. Many models focus on a particular stakeholder, e.g. the teacher, or the child, to the exclusion of others. This is not to suggest that the models are somehow wrong. However, we feel that when the software design process takes into consideration the needs of as many stakeholders as possible, the software is more likely to be successful in both implementation and deployment, even if the process of doing so is not as straightforward. The stakeholders we have identified are described in the following paragraphs.

**Children**

It goes without saying that children are the major stakeholders in our framework. They are included throughout the design process, and act in various capacities, from co-designer to expert evaluator. We feel that models which use other persons (e.g. teachers) as spokespersons for children’s needs, desires and skill levels run the risk of getting the design seriously wrong. We also feel it is important to mention that children should not be seen as some sort of homogeneous whole, and that simply selecting a few at random to participate in the design process will ensure its success: in the section on *Skills*, we describe personal characteristics which are important to possess, either before the design sessions begin, or which can be fostered during the design process.

Given the increasing acceptance of including children in the design process, it would be impossible to enumerate each project which does so, nonetheless, Druin (2005) provides a good example of a large-scale project which involves children in the design of the International Children’s Digital Library.
**Teachers**

In many models which focus on children as the primary stakeholders in the design process, the teacher’s role may be diminished or even ignored. We feel it is important for the teacher to be considered as a primary stakeholder: after all, a teacher can provide much needed pedagogical expertise which children do not, and cannot be expected to, possess. In addition, he can bring to bear knowledge of the curriculum in which he and fellow teachers operate, and provide feedback on the likely success and uptake of any given software product *in situ*.

Conlon and Pain (1996) describe the development of *Primex 2.5*, a tool for knowledge-based modelling, and the extensive role which teachers played in ensuring the software met their needs, and could be used in the classroom. Although we feel that the inclusion of children into the design process would have been useful, this project nonetheless contrasts with many in which software is developed without a real regard for the context of use.

**Parents**

Many models don’t consider parents as stakeholders in the design of educational software, and in some cases, it may not be relevant to do so. However, given that parents play an indisputably central role in their children’s lives and educations, they should also be involved in the design of educational software when appropriate (e.g. when the software in question is likely to be used in and/or marketed to a home environment).

The involvement of parents also seems to be most appropriate when younger age groups are being considered, for example, the design of a communication device to allow children to tell their parents about their day at play group (Hoenderdos et al., 2002). Similarly, the Homework project (Underwood et al., 2005), aimed at children aged 6-7, involves parents extensively in the design process via diaries, questionnaires, and semi-structured interviews.

**Industrial partners**

Projects which are commercially funded must consider the role of industrial partners and/or commercial development teams in the design process. The inclusion of industrial partners as stakeholders brings with it the need to consider possible pressures from the commercial world, such as the need for the product to be profitable, and the need to conduct work according to tight schedules, as described in the section on *commercial constraints*. As Antle’s (2003) experience shows, the presence of industrial partners on educational software projects may in turn influence the scope of children’s involvement.

**Academic funders**

In any project funded either by a research council, such as the Engineering and Physical Sciences Research Council (EPSRC) in the UK, or bodies such as the National Science Foundation (NSF) in the US, the funder will be a key stakeholder in the project. For a start, their involvement will likely determine the nature of the project itself, as the project will have had to fulfil certain criteria in order to be funded in the first place. Additionally, there will be an onus on the project to produce innovative results, either in terms of the software developed, or the methodology used (or both). Given that research council applications are generally peer reviewed, in a sense the wider academic community
can be considered as a stakeholder. The project findings will be disseminated to the relevant academic fields for critique and comment, therefore requiring the research to be carried out according to standards of best practice in the domain.

Activities

The various stakeholders in the design of educational software can be involved at different stages of the design process, taking on different roles. In this framework, we assume that the interactive learning environment is developed using a rapid prototyping approach (Pressman, 1992). The ILE design process starts with a requirements gathering phase in which the stakeholders are consulted about the features the software should contain. This is followed by collaborative design and low-tech prototyping in which the design team creates non-digital mock ups of the ILE. After a period of hi-tech prototyping, in which the design team focuses on digital prototypes of some aspect of the system, the team evaluates a working prototype. This process can be iterated several times to progressively refine the prototypes until an alpha release is ready. At this point, it is beneficial to introduce the ILE to another group of target users to get a fresh perspective. In some projects it may not be appropriate or feasible to involve stakeholders at each of these phases, but it is valuable to ensure that they are represented at the requirements gathering and prototype evaluation phases.

Within the software design process, activities to which team members contribute have different aims. Researchers on the KidStory project (Taxen et al., 2001) note that team members may be involved in educational, evaluation and brainstorming activities in which the aims are, respectively: to educate the team members in design practices and skills; evaluate existing software and new prototypes; and generate new ideas for products. They report that the highest proportion of the activities in early KidStory design sessions were evaluation oriented, a large proportion involved educational activities and that a lower proportion of sessions (6%) were intended to generate new ideas. However, as one would expect, as the design team became more experienced in the second year of the project, the proportion of educational sessions decreased and the ideas generation activities increased (to 42% of sessions). It seems that early investment in educational activities builds team members’ skills so that they can contribute new ideas at a later stage in the design. However, this training process has a heavy resource implication which may not be feasible on projects with small budgets; indeed Taxen et al. (2001) note that they found it difficult to decide whether to focus resources on refining the design process itself or to invest those resources into the technology development.

Requirements gathering

It is beneficial to consult with a wide range of stakeholders at the requirements gathering stage, although later stages of the design process may focus closely on a particular stakeholder group.

Broad consultation with groups of stakeholders: It is useful first to get a broad overview of stakeholders’ viewpoints about the proposed ILE. Consultation with learners can provide initial information about the aspects of the domain with which they have difficulty, and the software support features that they would appreciate. This can be accomplished by issuing questionnaires to classes of pupils in the target age group, taking care to ensure that the vocabulary is appropriate for children of that age. Similarly, teachers can be asked to fill in questionnaires about their normal approach to teaching in the target domain, the present challenges which pupils face, and initial suggestions of how the software could address current problems.
Based on the data gathered from the questionnaires, it is often useful to follow up particular issues in more depth by conducting a series of focus groups or semi-structured interviews with pupils and teachers. In an early stage of the Homework project, the researchers led a workshop for thirty local teachers to explore their attitudes to the adoption of technology which would integrate home and school learning (Luckin et al., 2004). It is useful to consult groups whose members are likely to have contrasting perspectives or specific needs. For example, in the BBC Geography project, described in the Case Study section, we consulted with teachers and pupils in a number of schools, including the Royal Blind School to help the designers arrive at an inclusive design.

**Observation of current practice:** Interactive learning environments are often designed to facilitate a learning activity which currently takes place in another form. Learning and instruction in that domain may be delivered in another medium, using textbooks, videos, or oral presentation; or it may be the case that an existing software product is used. It is important, therefore, to understand the learning processes in which the pupils currently engage before attempting to enhance learning with a new environment. Observation of current classroom practice is beneficial in this respect; the researcher can gain an understanding of the challenges which teachers and learners face, both conceptual and practical. With this deeper understanding of existing practice, the team can begin to design a technology intervention which is appropriate to the stakeholders’ needs. For example, before introducing new technologies to support early literacy learning through social interaction, researchers in the NIMIS project spent time observing, recording and analysing interactions between teachers and pupils (Cooper & Brna, 2000). Note that we are not suggesting that the software developed will be designed simply to act as a technological counterpart to existing classroom practice. Rather, we feel that as learning scientists, it would be misguided to develop educational software, however innovative, without a thorough understanding of its intended context of use.

**Evaluation of existing software:** After a design team is formed, a fruitful activity can be to ask team members to evaluate existing software which is related to the ILE, either because it supports learning in a similar domain, it supports a similar type of learning, or it employs a similar interface representation. The researcher identifies appropriate software titles and gives the designers an opportunity to explore them. If more than one package is used, it is helpful to ask the designers to accomplish the same task in various packages to scaffold the comparison process. During the case study projects described below, we have adopted an approach where adult team members observe the child team members using the software. Afterwards, the adults facilitate discussion with the children about the positive and negative aspects of the software, and any difficulties they encountered.

By contrast, in the contextual inquiry approach (Druin, 2002), adult and child “researchers” observe the activity patterns and roles which children adopt when using existing technology. Druin believes that the children under observation may feel more comfortable when observed, and particularly filmed, by other children. If such an approach is adopted, the child observers will require training in using recording equipment, and in appropriate methods for structuring note-taking. We believe that it is more effective to employ experienced adult researchers in the role of observer, as they will be better able to decide which phenomenon to record and to articulate their observations. Of course, the trade-off inherent in this decision is that the child designers will not have gained experience in this particular aspect of design.

**Analysis of curriculum guidelines:** Educational software which will be used in a classroom context often must support curricular objectives required by policy makers. Certainly, this is an important issue in the UK where teachers are constrained by national curricula and find it difficult to justify spending class time on activities which are not specified in a curriculum. In addition, local
education authorities and individual schools may have their own policies for teaching in particular domains. For example, the school in which the StoryStation software was designed had recently adopted a rather prescriptive yet successful approach to writing instruction which had implications for the activities in which the software would be used (North Lanarkshire Council, 2002). The researcher may require assistance from a teacher in interpreting the curriculum assessment guidelines and understanding how they would be used in practice. From this exercise, the researcher will learn the language used to describe the domain, which will be of assistance when explaining the ILE to other teachers and policy makers at a later date.

The researcher will of course need to consult academic literature for theoretical insight and empirical studies of teaching and learning in the domain, and consider which theories have been employed in the policy development. ILEs which are based on theories that are inconsistent with those embodied in the policy may end up not being used in the classroom, regardless of their merit, because teachers will find their use difficult to justify.

**Analysis of learners’ work and teachers’ assessment:** While it is necessary to be familiar with curricula specified by policy makers, the guidelines are unlikely to contain sufficient detail to establish requirements for the feedback which the ILE should provide for the learner. In some cases, knowledge acquisition techniques can be used to capture teachers’ procedures for assessing pupils’ work based on the curriculum guidelines. Samples of children’s corrected work can be gathered to identify common misconceptions and errors, establish the criteria which a teacher uses to assign grades, and identify the teacher’s priorities and techniques for giving feedback to the learner. Additionally, the researcher and learning scientist in the design team will be able to provide insight into misconceptions and appropriate types of feedback and remediation based on theory.

**Design**

An important contribution of the Cooperative Inquiry design approach is the belief that children can contribute to new technology, not only by evaluating software, but also by generating new design ideas (Druin, 1999). Creative ideas produced by children will be general concepts of how technology can be used and suggestions about the ways in which the user can interact with the technology, rather than innovative technical contributions at an implementation level.

It is generally accepted that design requires creativity – but what does creativity mean in this context? What can we realistically expect from our child designers?

It is a commonly held belief that children are more creative than adults, and that formal schooling suppresses natural creativity (Sawyer, 2003). However, researchers studying the development of creativity argue that although children often demonstrate expressive spontaneity which adults find pleasing (Cropley, 2001), they lack the knowledge and understanding which would enable them to make enduring contributions to a domain (Feldman, 2003). Boden (1992), in her discussion of the nature of creativity, makes the point that to generate creative ideas within a domain, the creator must have a highly developed mental map of the conceptual space of that domain. A conceptual space is an accepted way of thinking about a topic, containing a set of generative principles which define the possibilities within the domain. In the domain of music, for example, tonal harmony is a conceptual space containing rules from which it is possible to generate new compositions. Creative ideas can be generated from either an exploration of parts of the conceptual space which are not usually considered, or from a transformation of the conceptual space which changes the generative principles. In either case, the creator must have a deep understanding of the conceptual space before its structure can be exploited in the generation of creative ideas. In contrast to the popular notion that creativity comes
from freedom, an understanding of the constraints on the conceptual space is necessary before the constraints can be relaxed to produce creative concepts. For this reason, adult facilitators need to scaffold the child designers as they form mental maps of the domain before embarking on design tasks which require the generation of new ideas. Evaluation of existing software, followed by analysis of the features of that software, can enable the children to form mental representations of the design possibilities.

Consider also the notions of P-creativity and H-creativity, proposed by Boden (1992). An idea is considered to be P-creative if it is original from the point of view of the creator, even if it has previously been generated by someone else at some point in history. For an idea to be considered H-creative, it must meet a stricter criterion – it must be wholly novel within the culture, possibly involving the transformation of a conceptual space. The point here is that even in the most successful design project, to which children have contributed a number of creative ideas, a high proportion of these ideas will be P-creative. That is, they represent a significant personal achievement on the part of the child, but may not make an original contribution to the project as a whole simply because the more experienced adults on the project have previously considered it. On the other hand, assuming the viewpoint that children are “an entirely different user population with their own culture, norms, and complexities” (Druin, 2002. p. 1), children’s unique perspective may result in a different mental map of the domain to the one an adult might produce, thus opening up the possibility of exploring new parts of the creative space. In any case, when planning a project which includes child designers, it is important to have realistic expectations about the children’s contributions to avoid pressuring the children with unfeasible hopes. Furthermore, in keeping with a view of creativity as emerging in a context of distributed cognition (see, for example, (Fischer, 2005) for a discussion of the notion of social creativity), it is important to nurture in the children a sense of contributing to the creativity of the group, and to imagining joint solutions. Fostering such an atmosphere takes conscious effort, as much of children’s schooling will have reinforced the notion of individual performance.

The design team can engage in a variety of brainstorming activities to generate new ideas to incorporate into the design. For best results however, brainstorming should be structured around practical tasks, as most people find it extremely difficult to be creative without stimulus. Effective activities include drawing designs on paper, working with an adult artist to sketch interface designs, and low-tech prototyping, a design method employed in Cooperative Inquiry. Low-tech prototyping involves the production of conceptual prototypes in a non-digital medium such as plasticine or cardboard. This technique is also employed by adult designers – the designers of the computer game Metal Gear Solid 2 used Lego bricks to design the spatial environments in which the game would take place (Konami, 2001).

Having established a design, perhaps using a low-tech prototyping process, the design team can help to refine the ideas by focussing on specific aspects in more detail. A pen and paper walkthrough of how the different components interact can be useful to get an overview of how the system will work as a whole. Another technique for refining design is high-tech prototyping, where the team members use simple authoring tools to modify aspects of the interface design such as colour scheme and layout.

**Evaluation of prototypes**

When a pilot implementation of the design is ready, evaluation can result in valuable feedback for the software developers. Depending on the robustness of the software, children can use it to accomplish a task, or watch a software developer demonstrate it. The latter is recommended in situations where the software is at such an early stage that there has been insufficient time for adequate software testing.
Under such circumstances it is useful for the developer to get feedback on a pilot version by demonstrating particular features of the software. It is a waste of time at this stage to employ children as software testers – the developer can find bugs in a more effective and less stressful fashion without consulting the children.

The Wizard of Oz (WOZ) technique can be useful for evaluating the user interface of an ITS before the underlying tutoring system is complete. WOZ evaluation is commonly used in the development of speech and dialogue systems, but can be adapted to explore learners’ interactions with a prototype tutoring system. In this methodology, a learner uses the interface to the software, but the output displayed on the interface is generated not by the software, but by a human who monitors the user’s input. The role of the wizard can be played either by an adult, a teacher for example, or a child. In some cases, the user is aware that the computer’s behaviour is simulated by a human wizard; if this is not the case, there are ethical issues to be considered. Hammerton and Luckin (2001) describe a WOZ study to investigate patterns of help seeking behaviour in which pairs of child learners adopt the role of computer and user. A WOZ study was also used in the StoryStation project to ascertain appropriate points in the writing process to provide feedback, and the style of feedback which is most suitable (see Case Study section).

The following list provides a summary of the various stages of the design process, and the activities which are most appropriate at each stage:

Requirements gathering:
- Broad consultation with groups of stakeholders;
- Observation of current practice;
- Evaluation of existing software;
- Analysis of curriculum guidelines;
- Analysis of learners’ work and teachers’ assessment.

Design:
- Drawing designs on paper;
- Sketching interface designs (with artist);
- Low-tech prototyping;
- Pen and paper walkthrough;
- High-tech prototyping.

Evaluation of prototypes:
- Participant evaluation;
- Software demonstration;
- WOZ evaluation.

Skills

In order to successfully function as a member of the design team, team members require a set of core skills and attitudes. Sternberg (2003) discusses the resources required by an individual undertaking a creative project, in this case collaborative educational software design. Three key intellectual skills are 1) the synthetic skill to see problems in a new way and propose new solutions; 2) the analytic skill to recognise potentially successful ideas, and 3) the practical contextual skill which enables the originator of an idea to convince others of its merit. As noted in the previous section, knowledge of the domain is also required in order to generate new ideas within the design space. Personality traits and
attitudes are also important: Sternberg identifies personal attributes of successful creative people, which include perseverance in the face of obstacles, the conviction to oppose convention by introducing new ideas, willingness to take sensible risks, and strong intrinsic motivation to focus on the creative task.

The specific skills required by children and adult facilitators in educational software design projects are discussed in turn below.

In discussing these skills, it is important to note that, strictly speaking, there are no firm prerequisite skills for a design project, such that not possessing those skills would preclude one from taking part in the project. Nonetheless, it is important that team members who do not possess the necessary skills have the capacity and willingness to acquire them. In this case, it is also important that one of the team members has the ability to foster these skills in the others.

For children, being a member of a design team, and acquiring a number of new skills is a rich educational experience in and of itself. It offers them an authentic experience to take on a role and discover what it means to “be a designer”, as opposed to simply learning about what designers do (see Gee (2003) for a description of the sort of learning which can take place when learners take on realistic roles within affinity groups in a semiotic domain).

In the projects we have worked on, children are very articulate in describing the benefits they reaped in taking part in a design team. In this article however, we have chosen to focus on the skills related to the design process rather than the general educational benefits which child participants may gain from being involved in such a project.

**Skills for child team members**

Working as part of a design team can be very challenging for children. It places them in an unfamiliar social situation in which they are asked to collaborate with unfamiliar adults over a long period of time on a project which may have some significance in the real world. Contrast a software design project of this sort with normal classroom activities. Firstly, in a school setting children generally work with the same group of familiar adults (teachers), with whom their relationship is well defined. When working in a design team, the children will be introduced to new adults who may wish to have a less formal relationship with the team than is usual between pupil and teachers. Secondly, the project is likely to last over a period of weeks and months, whereas the children may be more used to less sustained activities which may span only days. Lastly, many projects which children are asked to do in school are educational exercises only – they do not have importance outside the school environment and are not subject to external deadlines, nor to external critique.

Given that taking part in a design team will be challenging for child members, it is the responsibility of the collaboration facilitator to ensure that the children have the social and cognitive skills to carry out the project without becoming frustrated or confused. As some of the skills are developmentally linked, it may be the case that members of the target demographic are simply too young to function effectively in a team.

The most important skill is the ability to take part in a discussion by being able to contribute one’s own point of view at a relevant juncture, and by listening without interruption to other group members. This level of group discussion skills would be expected in pupils working at Level B in English Language Curriculum guidelines in Scotland i.e. around 6-8 years old. Researchers on the Classroom of the Future project discuss the issue of how young design partners can be (Farber et al., 2002). They describe design sessions conducted with four to six year old children, mentioning that the
children found it difficult to work in groups, and to listen to each other’s ideas. They report that when working with children of this age group, the adults need to offer more ideas, start discussions and propose more design ideas.

Note that communication in the context of a design team may go beyond the traditional face to face communication often assumed in child-centred design sessions. Given issues such as time constraints, it may be useful to teach the team how to use communication software to keep in touch outside of face to face meetings.

Another skill is the maturity to take part in sustained work at two levels: working on the same task for an hour or more, and working on the same project over a period of weeks. The first is desirable on the grounds that many design tasks are complex and cannot be completed in a short period of time. Additionally, from a practical perspective, it may not be cost effective for researchers to visit a school for very short periods of time. It is advisable to consult the team’s class teacher to discover how long the children typically spend on a task. Activities can be broken up into shorter tasks with breaks between them to counter fatigue. Similarly, variety in the activities at consecutive sessions can be introduced to keep the children interested in the project as it progresses.

In some projects, team members may need to have a certain degree of proficiency in an aspect of the target domain in order to understand the attendant design problems. In most projects, it is desirable for the team members to have a basic degree of literacy so that they can read instructions and record their ideas. This is particularly the case when the team is evaluating existing software, given that school computer systems often require logins and passwords. While it is possible to gain useful ideas from illiterate users, in this case researchers should budget for more time and resources for one on one support for team members.

As the design team learns to work together, team members will need to develop further skills and attitudes to enable them to function successfully in the group. One such skill is the ability to clearly present design ideas to other group members, either in spoken or written form. The adult facilitators may need to help child team members express their ideas concisely and precisely by helping them to structure their presentation, for example by completing a worksheet with answers to specific questions, or by asking the children to draw a diagram or interface sketch and verbally questioning the designer about features of the drawing. It is necessary for the adult facilitators to gather as much information about the specifics of the design to relay to the adult designers (if they are not present or, indeed, if the facilitator and designer roles are taken on by different persons) so that they are able to interpret the child’s intentions later (note that the issue of adult interpretation of children’s intentions is considered in depth in (Scase & Rogers, 1999)).

A related skill is the ability to give and receive critical feedback, something which many people find difficult regardless of their age. As children work collaboratively in the design team, and present ideas to others, it can be demoralising for them to hear criticism of their ideas, even if it is intended constructively. In previous projects, we have first trained children in giving positive feedback to their peers, for example by asking the group to comment on their favourite features of an idea. The next step is to ask the child if he would like the group to give him suggestions for improvements. If at that stage, he doesn’t feel ready to hear the criticism, he can opt out and perhaps discuss improvements in private with an adult facilitator. If the child feels confident about hearing improvements from his peers, the facilitator moderates the discussion and makes sure that everyone understands that any criticism is of the idea, not the child. With training of this kind, and a supportive environment, children can learn to use constructive criticism to improve the group’s work with considerable success. As with any design project, there are constraints on the number of ideas which can be included in the
final product. Group members may have conflicting ideas which cannot all be incorporated into the design. For this reason, it is important that the team members accept that their ideas may never be used, even if they are very good.

Important cognitive skills for team members to learn are the analysis of the strengths and weaknesses of software prototypes, and comparison of related software packages. The children may at first be inclined to give an overall assessment of software without much detail, for example “it’s brilliant!” or “It’s rubbish”. The adult facilitator may have to help team members to think of specific features which they liked and disliked. An example of a structured task to facilitate this is as follows. The children use the software package under consideration for a period of time while the adults observe any difficulties they encounter. The children are then asked to write down at least three good points, three bad points and three improvements for the software. Group members are asked in turn to elaborate on their contributions, and these explanations are recorded for later transcription by the researcher.

The design team members may also need to develop project specific skills, often learning how to use new software packages.

**Skills for adult team members**

Much has been made of the equal status of child and adult team members in Druin’s work (Druin, 2002). However, in most cases, time and budgetary constraints will ensure that control of the project rests with an experienced adult manager. A number of adults may be involved in a large project, including the manager, technical experts, education experts and artists, and not all may be able to afford the time to attend sessions with the child design team.

One of the adults who works face to face with children may have responsibility for facilitating the child design team. The facilitator faces a number of challenges in liaising between the child designers and the rest of the team, and should possess a set of skills for dealing with these challenges.

Firstly, some of the adults on the project may not be convinced of the value of working with children in this way, and may be reluctant to commit time to the approach. Other adult team members may not be comfortable working directly with children, and may be inexperienced in doing so. It is important that the facilitator ensures that the other adults know that their expertise is valued, and that they do not feel that the facilitator wishes to replace their experienced insight with children’s ideas. From the start, then, all team members should be realistic about the scope of the children’s contribution. It is counter productive, and possibly insulting to the adult experts, to imply that the children will be the source of creativity in the project. After all, adults on an educational software design team are likely to be employed in that role because they are highly trained, creative individuals. There is no good reason to suppose that the children selected for a design team are more likely to be creative simply because they are younger. In our experience, only some children are highly creative designers; this is unsurprising in that one would expect only some members of the general population to have an aptitude for design.

For these reasons, the facilitator should be tactful when liaising with the adult team members and the child designers. In one direction, he is responsible for accurately conveying the adult designers’ concepts to the children in a form which they can understand, and relaying the adults’ feedback on the children’s designs in a constructive manner. In the other direction, he must explain and interpret the children’s design ideas to the adult designers, as well as giving details of the children’s suggestions for improving the software. The facilitator’s observations of children using early prototypes will also be
The processes is listen which precedence. It may be necessary for the facilitator to explain to the children why their ideas have not been adopted by the designers, or that the project plans have changed due to management constraints. The children will not be used to working on real world projects with strict deadlines, changing objectives and the possibility of cancellation. The facilitator may have to reassure the children that changes of this sort are relatively common, and that they are not related to the quality of the children’s work.

The facilitator should project confidence in the team members’ abilities and affirm how valuable their contribution is to the project. At the start of the project, the facilitator should work with the team to decide on the boundaries for acceptable behaviour. Some groups of children will implicitly pick up what is considered acceptable by observing the adults’ behaviour, but other groups may require an explicit discussion of these norms as they may find it difficult to judge what degree of formality is appropriate. An approach where the group members suggest and agree on a set of group rules can work well, and group members will remind each other when the rules are broken. For example, one group we worked with came up with these rules: “Listen to the adults and to each other”, “Join in as best as you can” and “Treat others with respect”. The children may need guidance in simple conventions such as whether the adults should be addressed by their first names and whether they are expected to raise their hands to give an answer.

One aspect of the group interactions which should be considered carefully is how decisions should be made. As an example, a design team might be faced with the task of prioritising a set of features which should be included in an early prototype. Firstly, the group might find it difficult to accept that not all of the features that they recommend can be included due to time constraints, and secondly they are likely to need help from the facilitator in deciding as a group which features take precedence.

As mentioned previously, one role of the facilitator is to model the positive attitudes and skills which will be required for group members. A central aspect of this is listening – the facilitator should listen carefully to each child’s contribution and give non-verbal feedback to reassure the child that she is attending. Given that the children are apprentice designers, the facilitator should model the thought processes and specialist vocabulary used by designers in order to introduce the children to this world. The facilitator should make sure that definitions of specialist vocabulary are given as the terms are valuable to the adult designers. An issue which the collaboration facilitator commonly encounters is balancing the child designers’ preference for fun features with the educationalists’ requirements for good pedagogical design. Incorporating engaging child friendly features is likely to be motivating for learners, but resource constraints and stakeholder expectations often require educational features to be prioritised. A way to approach this is to ensure that core educational aspects of the software are presented in a way which children find accessible, but to de-prioritise components which are purely entertaining.

The team will work best when the facilitator is enthusiastic, patient, flexible and confident - attitudes which are difficult to maintain under the pressures of working with prototype software in a school computer suite! It can be tiring to stay enthusiastic throughout a session, but the facilitator is a role model for the children and they will take cues from his attitude. Remaining patient can also be difficult in the face of frustrating technical difficulties, but the children may need reassurance that the technical problems are not their fault. Working in a busy school environment requires flexibility on the facilitator’s part and the ability to quickly re-plan a session. Unexpected timetable clashes and failing hardware are common problems which may disrupt carefully laid session plans, and as the facilitator is usually a guest in the school, it is important to retain the goodwill of the school staff by being adaptable. Lastly, the facilitator must project confidence in the team members’ abilities and affirm how valuable their contribution is to the project.

Some groups of children will implicitly pick up what is considered acceptable by observing the adults’ behaviour, but other groups may require an explicit discussion of these norms as they may find it difficult to judge what degree of formality is appropriate. An approach where the group members suggest and agree on a set of group rules can work well, and group members will remind each other when the rules are broken. For example, one group we worked with came up with these rules: “Listen to the adults and to each other”, “Join in as best as you can” and “Treat others with respect”. The children may need guidance in simple conventions such as whether the adults should be addressed by their first names and whether they are expected to raise their hands to give an answer. One aspect of the group interactions which should be considered carefully is how decisions should be made. As an example, a design team might be faced with the task of prioritising a set of features which should be included in an early prototype. Firstly, the group might find it difficult to accept that not all of the features that they recommend can be included due to time constraints, and secondly they are likely to need help from the facilitator in deciding as a group which features take precedence.

As mentioned previously, one role of the facilitator is to model the positive attitudes and skills which will be required for group members. A central aspect of this is listening – the facilitator should listen carefully to each child’s contribution and give non-verbal feedback to reassure the child that she is attending. Given that the children are apprentice designers, the facilitator should model the thought processes and specialist vocabulary used by designers in order to introduce the children to this world. The facilitator should make sure that definitions of specialist vocabulary are given as the terms are
introduced, and should be aware of the normal working vocabulary of children in the design team. It may be necessary to paraphrase some of the language used by expert adults on the design team to ensure that the children understand.

When planning sessions, the facilitator should structure them in such a way as to give children breaks if necessary. During a session, it may be advisable to change the pace of activities to prevent the children getting tired or bored. For example, children have a limited appetite for long group discussions, so it is prudent to intersperse discussion with practical activities such as drawing or working on the computer.

As a final reflection on the role of the facilitator, we believe that the facilitator should make a personal commitment to ensure that the children feel their contributions have been valued. The children are investing a good deal of time into the design process, so it is important that they feel they have achieved something worthwhile during the project. Project closure is therefore important – perhaps in the form of a party, or an occasion where the children’s parents and teachers can admire their work.

The following table summarises the skills needed by the child and adult design team members:

<table>
<thead>
<tr>
<th>Skills – Child Team Members</th>
<th>Skills – Adult Team Members</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Ability to take part in a discussion by contributing viewpoint, and listening without interruption;</td>
<td>• Ability to deal tactfully with adult and child design team members, and stakeholders;</td>
</tr>
<tr>
<td>• Ability to communicate via electronic means;</td>
<td>• Ability to interpret children’s ideas for adult designers and vice versa;</td>
</tr>
<tr>
<td>• Maturity to take part in sustained work;</td>
<td>• Flexibility, enthusiasm and confidence to work in a potentially frustrating, unpredictable school environment;</td>
</tr>
<tr>
<td>• Proficiency in some aspect of target domain;</td>
<td>• Ability to act as a role model for child designers to illustrate group work and design skills;</td>
</tr>
<tr>
<td>• Basic degree of literacy;</td>
<td>• Motivation to make a personal commitment to ensure that the children feel their contributions are valued.</td>
</tr>
<tr>
<td>• Ability to clearly present ideas to other group members (spoken or written);</td>
<td>• Ability to give and receive critical feedback;</td>
</tr>
<tr>
<td>• Ability to analyse and compare software packages;</td>
<td>• Willingness to acquire project skills (e.g. using new software packages).</td>
</tr>
</tbody>
</table>

**CASE STUDIES**

The techniques and activities in this article are illustrated with examples from two contrasting learner-centred design projects in which the authors were involved. The first is a small academic research project which aimed to explore the effectiveness of animated pedagogical agents in an intelligent tutoring system. The second project is the initial stage in the development of an extensive public service online resource for UK classroom teachers and pupils.
StoryStation

StoryStation is an intelligent tutoring system designed to give feedback and advice on story writing skills (Robertson et al., 2004; Robertson & Cross, 2004). The software was developed as part of a research project in which the objectives were to: a) explore the effects of animated pedagogical agents on learning and motivation; and b) to use language technology techniques to provide tailored feedback on specific writing skills, such as spelling and vocabulary usage. Children and teachers were closely involved in the design process, as described below and further documented in (Robertson, 2002).

Stakeholders

The stakeholders in the project were, firstly, children of the target age range. Parents were seen as less crucial in terms of representation, given that the software was designed for use in a school context.

Teachers were also considered to be important stakeholders in the project, given that StoryStation was designed to be used in Scottish schools to support the objectives of the English Language curriculum guidelines in Scotland.

The academic funders of the project were the EPSRC, a UK government funding body whose remit includes the support of technology innovation. This placed a constraint on the project in the sense that the development of innovative technology was expected. Furthermore, the funding was provided to support the specific objectives of researching animated agents in the context of an intelligent tutoring system for story writing.

Roles

The project team consisted of the second author in the roles of project manager, researcher and technology specialist; two teachers, one of whom was the children’s class teacher, and the other an experienced retired teacher; and eight children aged between ten and twelve, who were pupils in a state funded primary school in a small Scottish town. The children were chosen by the researcher and teachers to form a group with a range of ability levels and personalities. In addition, the teachers chose children whom they felt would benefit particularly from working on the project. The original team was comprised of four girls and four boys.

An advantage of the researcher’s multiple roles was that there was no overhead for liaison between technology specialists and the facilitator of the children’s design team, but the inevitable disadvantage was the lack of resources for software development and design work.

Context

StoryStation was designed to be used in Scottish schools to support the objectives of the English Language curriculum guidelines in Scotland. These guidelines are basically consistent with the curriculum used in the rest of the UK, although different terminology is employed. The software is intended to support children who have reached a basic competency in story writing (Scottish levels C and above, around eight years and above), although it has been used with older pupils with learning difficulties at the request of their teacher.

The design team met in the school computer lab, and the pupils were released from their normal classroom lessons to take part. The project budget was unfortunately not sufficient to pay to release the class teacher from her teaching duties. However, one of the teachers in the team is retired, and was
therefore able to contribute her time either to participate in design sessions, or occasionally to stand in for the class teacher to enable her to take part in the team. The budget for design sessions is an important consideration – teachers are professionals whose time is extremely valuable. Even if a school is committed in principle to working collaboratively with a research team, it may not be in a financial position to do so.

The StoryStation design team worked on the project over a period of 18 months, with the children contributing frequently over 15 months. In the early stages of the project, the team met weekly for around two hours. This intensity reduced in the implementation stage, and increased again in the pilot testing phase. The children remained motivated throughout because the sessions were structured to avoid repetition, and because there were natural gaps between sessions during implementation phases in the software lifecycle. Thus the children had “holidays” from the project which ensured that they were excited when the project resumed.

Activities

The children in the design team were involved throughout the life cycle of the project, in the requirements gathering, design, implementation and evaluation phases.

During the requirements gathering stage, the researcher administered questionnaires to a class of pupils in the target age range and to their teachers to discover what pupils find difficult about writing on one hand, and what teachers find difficult about writing instruction on the other hand. In addition, a corpus of corrected stories written by children working at each curriculum level was gathered, which was useful both to help the researcher understand the range of problems which children at different development levels exhibit, and also later as a data resource for testing language analysis techniques in the implementation phase.

More in depth information from discussions with the design team was used to supplement the broad range of opinions gleaned from the pupil questionnaire. The pupils told the researcher the resources they use when writing stories (e.g. dictionary, thesaurus and word banks) and the processes they follow in the classroom (e.g. starting off by mind-mapping some ideas). In these early sessions with the design team, the teachers and the researcher also worked to foster group skills: discussing with the children what it meant to be designers and encouraging the team to listen effectively and provide constructive criticism to the other team members.

The next stage of the process was to work with the design team to critically evaluate existing software which can be used to support story writing, including Writer’s ToolKit and Kidspiration. This allowed the researcher to observe the level of interface complexity with which the children were comfortable, and also gave children the opportunity to think more carefully about interface design issues.

Much of the children’s contribution to the design stage was focussed on the animated pedagogical agents and the feedback which they could offer. Using cards with pictures of sample animated agents, the children wrote speech bubbles to indicate what the agents would say to help a pupil as they wrote a story. They also drew pictures of animated agents which they thought would appeal to other children. Finally, the children took part in a Wizard of Oz study to ascertain appropriate styles of language to be used by the agents when giving feedback. Classmates of the designers wrote stories using a standard word processor to which a set of animated agents had been added. The child designers played the role of “wizard” by monitoring the pupil’s progress and typing feedback to the pupil at what seemed like appropriate points. The feedback was relayed to the pupil via the animated agents on his screen. The
teacher also took a turn at providing feedback to the pupils, which was considerably more useful for two reasons. As might be expected, the child designers found it difficult to diagnose errors in the writing of their classmates because they were learner writers themselves. In addition, some of the child designers were not particularly tactful when phrasing feedback, which had a demoralising effect on the other pupils. Analysis of the log file from the teacher’s interventions allowed us to ascertain the teacher’s priorities in drawing the pupil’s attention to mistakes, the points at which she found it appropriate to intervene and the sort of language which she used when correcting errors. It was interesting that some of the children provided such critical feedback in the role of teacher – they were mimicking their internalisation of a “teacherly” attitude based on their own experiences. Previous studies have in fact found that pupils find writing feedback offered by teachers to be demoralising and difficult to interpret (Dunsbee & Ford, 1980; Coupe, 1986). As a general point, pupils are likely to partially base their design ideas for intelligent tutoring systems on their own interactions with teachers which may not always embody sound educational and motivational principles.

At the implementation stage, the child designers used a graphics package called XaraX to create the animated pedagogical agents which are used in StoryStation. They also created a set of interface icons and an animation for the splash screen, and chose the voices to be used in the speech synthesis. They each wrote a section about themselves to be displayed in the “about StoryStation” menu and contributed text to the story writing tips in the application help facility. As an interim evaluation of the language technology techniques used to generate feedback, a class of pupils in another school wrote stories by hand which were typed in and analysed by the pilot software. The feedback was printed out and returned to the children who used it to improve their stories at the next draft. This process was invaluable in fine tuning the tutoring rules and feedback based on real examples of children’s writing. It was also an interesting opportunity to observe classroom practice for story writing lessons in another school.

In the evaluation phase, the child designers piloted and tested the software, and their suggestions were incorporated into the final version. The researcher also led a workshop with eighteen teachers to gauge their opinions about the usefulness of the system. Finally, a full-scale field study took place in a different state funded primary school in Edinburgh and involved 57 10-12 year old pupils. Focus group interviews and questionnaires were used to establish the children’s attitudes to the animated agents, and to ascertain how they perceived the feedback offered by the agents in comparison to that offered by teachers. The class teachers and head teacher of the school were also interviewed. Results from these studies are described in full in (Robertson & Cross, 2004; Robertson et al., 2004).

Skills

The StoryStation child designers gained a number of skills through working in the design team. Firstly, the pupils learned team working and group discussion skills throughout the course of the project, as noted by the class teacher, who felt that their new collaboration skills had transferred to other class work. Nonetheless, one member of the design team was extremely disruptive to group discussions, interrupting her peers with irrelevant comments. Although she was 11 years old, her social skills were underdeveloped in comparison to other children in her age group. Originally she was included in the design team because it was thought that her writing difficulties might give useful insight to inform the design of the intelligent tutoring system, but this proved not to be the case because of her difficulties in expressing her ideas orally.
In terms of technical skills, the children learned generic skills which translate to other software packages (such as cut and paste, and grouping and ordering objects) and general problem solving and exploration behaviours which are required when learning to use new software. The graphics package which they used to animate the agents is complex, but they were able to learn how to use it to successfully perform the required task. The teachers were impressed by the children’s computing skills, noting that their skills outstripped their own (a relatively common occurrence). This highlights a problem with relying on teachers’ perceptions of their pupils’ capabilities, as in Conlon and Pain’s Persistent Collaboration Model (1996).

A notable aspect of the StoryStation team’s productive collaboration was that team members often helped each other learn new technical skills. After working with the design team during a session, one of the teachers commented that there was a “real workshop atmosphere”. She was impressed with the way the children worked with each other and the researcher by setting and meeting their own goals.

One of the pupils provided a summary of what she felt the team had learned for an academic poster published at AIED 2001 (Wiemer-Hastings & Robertson, 2001):

“The design team learned the following things:

• Team work - Working together
• Learning a decent amount about using the computer
• How professional designers work
• Thinking constantly
• Helping others
• Evaluating ideas - compromising
• Experimenting with ideas
• Working with animation
• They compared different types of writing - Kidspiration / Writers Toolkit
• Meeting the deadline”.

The researcher also learned a great deal during the project, as it was her first experience of learner-centred design with children. It was particularly challenging to work with a group of eight children – other authors have recommended a ratio no more than three children to each adult (Taxen et al., 2001). The researcher also lacked the experience to effectively deal with children with social and behavioural difficulties, and found this particularly stressful. It is, however, true that patience can be learned – in subsequent work with children the researcher has found it easier to be patient during stressful circumstances. Similarly, flexibility in planning for sessions in a busy, changeable school environment is a skill which develops over time. The researcher was fortunate in that she had the support of the experienced teacher in the design team who is an excellent role model for the requisite facilitation skills. She is particularly good at quickly establishing a rapport with children and identifying and addressing problems which they may have. It is extremely beneficial for adult members of the design team to be mentored by facilitators who are experienced in working with children. One of the most useful ways to accomplish this is to “shadow” the facilitator as she works with children. The second author learned a lot prior to the project by accompanying the teacher on her visits to classrooms as a storyteller and observing her techniques for establishing a rapport, managing discussions, making everyone feel included and facilitating creative ideas. Similarly, Taxen et al. (2001) mention that a researcher from the KidStory project spent time working on an established
design team at the University of Maryland in order to learn how to work with children in a design context.

**Reflections**

Overall, the CARSS framework was useful in determining the relevant design parameters, and also helping us to pinpoint where difficulties did occur, and how they might be remedied in future projects. As such, we see the framework as being able to evolve to capture empirical recommendations derived from the application of the framework.

The context of the project was relatively fixed from the outset, but an awareness of the various constraints allowed us to work within them effectively.

The choice of stakeholder representation, focusing on children and teachers, was appropriate given the context of use of the software (classroom only). One of the most useful contributions from these groups was at the requirements gathering stage where learners described the aspects of writing which were difficult, and the design team explained the resources and processes that they use to write stories in the classroom. The teachers’ experiences of writing skills that they find difficult to teach, and common difficulties that their pupils encounter were also valuable. Combined with example stories written by the target user group, this information helped the researcher to understand the English Language curriculum guidelines, and prioritise those aspects which appeared to be problematic in practice. The inclusion of teachers and pupils in the design process validates the software to other teachers, and to educationalists in local authorities, who believe that it is consequently more likely to meet classroom requirements.

In terms of children’s contributions to activities in the implementation phase, the animated agents offer an interesting example. The younger children (<10 years old) who took part in the evaluation enjoyed the amusing appearance of the agents and liked to receive feedback from them. Older children, however, were more inclined to find the agents childish in appearance. The original intention was that the child designers would be better able to understand what other children would like and so create more appealing agents. However, as children are used to high quality multimedia in software, the production value of the interface is an issue. A compromise would be to partner children with graphic designers who could work with them to polish their art work. This would certainly be desirable for projects with commercial partners.

The issue of the skills of the design team also gave us cause for reflection. When selecting the team, it was decided to include two children who were working at lower levels of the writing curriculum to get an insight into the problems of struggling writers. Unfortunately, the two children also had social and behavioural difficulties which at times lowered the team’s productivity. One of these children benefited greatly from inclusion in the project as he found it more engaging than his normal class work and put great effort into it. The other child successfully contributed her ideas, but was not so willing to listen to other ideas. Her lack of social skills was disruptive and difficult to manage. While it is a good idea to include struggling learners in the design of the software, it is also worth bearing in mind that additional resources might be required to work with them productively.

**BBC Scotland Geography Project**

BBC Scotland recently employed a learner-centred design approach to the design of an interactive geography resource. A key part of this project was to develop a set of general purpose learning support tools which could be deployed in other curriculum areas. The developers were keen to consult closely
with the target user group on the design of these tools in order to ensure that they were both fun and fit for the learning purpose. A central theme of the support tools is that they are learner centred – the learner can author and manage his own learning material, organised around curriculum areas. In addition, the support tools correspond to constructivist and constructionist perspectives on learning, with the aim that learning would arise through situated activities involving the production of artefacts on the part of child learners. The domain area for this project was geography, and the project involved designing an online geography resource, and a webzine authoring tool to be used in that resource.

**Context**

The geography resource is aimed at 7-11 year old pupils, and is to be used at home and in the classroom. It was developed in collaboration with a child design team in a school setting. The design team met for an initial session in June 2004, and was re-established in September 2004, when it ran for ten sessions until December 2004. The sessions each lasted for around two hours, and took place weekly when timetable constraints permitted. The design team met in the school library, and had access to an open plan computer suite with internet access. Each session was videotaped in order to document the design process, and the adult team members wrote notes immediately after each session. The content created by the children using the software packages they trialled was also collected.

The children were released from other classroom activities in order to take part in design sessions. This represents a commitment from the head teacher and class teachers who, after initial discussion, decided that the skills and knowledge which the children would gain as designers would justify them missing class lessons. It also represents a commitment from the children who were required to catch up on class work which they missed during the sessions. This particular school is an excellent partner for research projects because of the enthusiasm and flexibility of the staff, and was chosen because the authors had worked with them successfully on previous occasions.

**Stakeholders**

The BBC is a public service provider, which in this case has undertaken to provide resources which support the curriculum framework devised by governmental educational policy makers. This initial phase of the Geography project lasted for a year, and the design and prototyping aspects, in which the children were involved, for six months. The project was not set up to involve an open-ended research-driven design process where the main focus is on developing children’s ideas. Rather, it was to develop a suite of prototype learning tools relating to human geography and local environments in close consultation with learners, therefore, delivering the prototypes on time was of a higher priority than developing the children in the role of designers.

The stakeholders also comprised children and teachers. If time had permitted, it would have been appropriate to include parents as stakeholders, given that the resources were also targeted at home use.

**Roles**

The production team was comprised of a senior producer (a former teacher) in the role of project manager, two assistant producers who provided educational expertise (one of whom is also a former teacher), and two technical experts who implemented prototype designs. The second author was employed as a consultant to act as the collaboration facilitator. The younger members of the design
team were six 11 year old pupils (three boys and three girls) from a state funded primary school in Edinburgh. The second author and the two assistant producers took part in the design sessions regularly, while the technical experts visited only occasionally as time would allow.

Activities

The first stage in the project was requirements gathering, particularly understanding the curricular objectives which should be delivered in this domain. At this stage, the adult design team studied the curriculum specifications and guidelines and consulted with subject expert teachers. Before establishing a regular child design team, adult team members visited a number of schools to solicit teachers’ and pupils’ general opinions of what should be included in a geography resource. In particular, we consulted with visually impaired users at the Royal Blind School, Edinburgh, with pupils at a school for pupils with behavioural difficulties, and teachers at a school for pupils with physical disabilities. The viewpoints of users with specific requirements were useful to the adult designers, and enabled us to explain to the children in the design team more about the needs of a wide spectrum of learners. The activities during each session with the child design team were generally a mixture of pen and paper design tasks, piloting of software packages and group discussions. As part of the design sessions, children were asked to create an online newsletter (webzine) about their local area. This encompassed activities such as collecting material for articles about the school and its environment, designing the layout for a paper version of the newsletter, evaluating existing software which contained components which would be required in a webzine authoring tool and evaluating the newly created webzine authoring tool on which the adult designers were working. The children also contributed to the design of a video editing tool which enables learners to structure an argument from a series of video clips in which interviewees express their opinions about a controversial topic. Throughout the project, there was a strong element of group discussion, either face to face during design sessions or via an online discussion tool. As not all adult team members were able to visit the school, they kept in touch with the children via a private discussion board. The children particularly enjoyed using this shared space to make suggestions to the adult designer and were pleased to receive replies from the rest of the team. The adult designers were able to use the online discussion tool to ask clarification questions between sessions.

Towards the end of the project the team members were interviewed on camera explaining why they think it is important for children to contribute to the design of children’s software, and what they thought they had learned from the process. This video was shown to the pupils’ teachers and head teacher during an end of project party. The children also demonstrated the software that they had helped to design and the new skills they had learned. The teachers were impressed with the quality of the children’s contributions, and the children were pleased to explain the project to their teachers. A highlight of the party was the opportunity for the children to meet the designer who created the prototype, because although they had used the prototype he created, and given him suggestions for improving it via the other adults, they had not met him face to face.

Skills

The school places a strong emphasis on the development of collaborative skills, supporting this in class work and in after school clubs. The children in the design team started with a good grounding in discussion skills, and the teachers valued the opportunity for the children to develop these further.
Nonetheless, one of the children thought that the best way for the group to make a decision was for everyone to adopt her ideas consistently.

The design team valued the technical skills they had developed, including using Slideroll, KidPix and WebCT. The online discussion tool on WebCT was particularly popular because they liked to read each others’ ideas and keep in touch outside school hours. The teachers were particularly impressed by the children’s new IT skills and intend to ask the team members to share their new skills with other class members.

The adult team members enjoyed the experience of working closely with the children and intend to adopt this approach on other educational projects. We encountered a number of challenges throughout the project, largely centred on the difficulty of using the school computers. We learned to plan sessions flexibly, devising alternative activities which could be undertaken if the computing facilities or internet connection were unavailable. It was challenging to establish an informal relationship with the children, while keeping them focussed on the task at hand – often our patience was tested by the teams’ fascination with the cartoon character SpongeBob SquarePants, who tended to creep into most designs! It was also time consuming to prepare sessions and liaise with the other adults on the design team, but we felt that this investment of time was worth it to validate the prototype design with members of the target user group.

Reflections

The adult members of the design team took a reflective approach to their interactions with the children – after each session they considered which of the children’s suggestions were useful, and which activities were most fruitful. To some extent they were evaluating the efficacy of the learner-centred design approach, weighing the cost of the staff time against the children’s impact on the design. Experiences from working closely with learners on the Geography project will give insight to the BBC developers who work on other curriculum areas, and so it was important to extract some heuristics for good practice in the future. These are discussed below.

In terms of the activities comprising the design sessions, the most fruitful involved the children as evaluators of the designer’s early prototypes. The evaluations comprised either the demonstration of the prototype on a laptop by an adult, interspersed with discussion and hands-on opportunities for the children; or an exploration session of the prototype by the children on individual computers followed by a group discussion. In both cases, the children’s direct experiences of the prototype were the basis for specific, relevant and helpful feedback to the designer. As an example, one of the most useful sessions was the evaluation of an interactive map of a Scottish landscape which is annotated with information about various aspects of the local area. The children were very enthusiastic about the method of navigating between sites on the map (via an animated helicopter). They were able to identify glitches in interactions with the helicopter, such as the difficulties in control which arose when the helicopter reached the top of the screen, and to propose solutions, such as a “deflector shield” to prevent the helicopter flying out of view. They also suggested useful features, such as a compass, and the facility to land the helicopter at specific sites to find out more detailed information about a topic. The children also considered the sound design carefully and were able to identify missing or misleading sounds. In summary, this was an extremely useful evaluation of an early prototype with members of the target user group.

In terms of evaluation activities, we also noted that the most useful feedback came from interactions with a high-tech prototype rather than paper interface sketches. On one occasion, the
designer had a new design for the interface to the video tool. He had created a new method for manipulating the video timeline and had created mock-up screen shots of his original and new ideas to seek the children’s opinion on the change. Unfortunately, most children in the group could not comprehend from the static diagram how the dynamics of the interface would work in practice, so were unable to help the designer much in this regard. This experience suggests that it is worth investing some time in creating a digital prototype to illustrate how an interface will change over time, as a static representation may be too difficult for children to use.

We also found activities involving the analysis of existing software packages to be useful. For example, we asked the children to evaluate Slideroll, a web based application which enables the user to create short picture presentations. The children listed the good and bad points about the application, suggested improvements and then synthesised a prioritised list of features to be included in the webzine authoring tool. We found that the children’s contributions were generally sensible and useful, and confirmed that the designer’s work so far was on the right track. As part of our reflective practice, we wanted to establish whether the child team members were coming up with new ideas or confirming what we already knew. The six children suggested thirteen improvements to Slideroll. After the session the adult members of the BBC design team were asked to consider each suggestion and note whether it was an issue of which they were already aware. The adults had not previously considered eight of the suggestions, although they had considered the remaining ideas but had not yet implemented them. The useful suggestions were ideas such as including a spell checker for textual annotations of slides, including example presentations, including a full screen mode for presenting slides or including background music. It could be argued that the adults would have thought of these features given more time, as they are common features in commercial software and are not particularly innovative. One of the suggestions as stated was unfeasible in our time scale – “making a search engine with a powerful mega drive to search for animation clips”. In other sessions where we asked the children to suggest improvements, the education experts were unconvinced of the educational value of some ideas, for example including cartoon clips in the video editing tool. On balance, the children’s opinions were worth seeking and although the adults possibly could have come up with similar ideas, the children’s input has added value because they are members of the target user group. It is worth pointing out, however, that the classroom is a key target environment for use of the resources, which is why the education experts decided not to use some suggestions which seemed fun but which did not add to the educational value of the resource.

We found the early activities to be least productive. This is possibly because the group was not yet well established, but we also found that the children had conceptual difficulties in understanding the design task. There were two layers to the geography project – designing an online geography resource, and designing a webzine authoring tool to be used in that resource. The children could easily understand that they were going to help us design a web site, but they were confused about the webzine authoring tool because they were not used to software in that genre. Their ideas about this aspect of the design were therefore rather vague until they saw examples of software for related tasks. This suggests sequencing activities such as the analysis of existing software in the requirements analysis phase, before design activities take place, a recommendation we have incorporated into CARSS.
CONCLUSION

In this paper, we have presented CARSS, a framework for carrying out learner-centred design with children. CARSS can be used in the design of intelligent and “non-intelligent” learning environments alike. The framework comprises five components: context, activities, roles, stakeholders, and skills. As such, it offers a comprehensive set of issues to consider when planning to use a child-centred design approach. As suggested by a number of the examples of research we have discussed, a fully fledged participatory design approach with children may not always be possible due to constraints such as time and budget. However, CARSS offers a way of specifying the initial parameters and constraints of the project in such a way as to determine the level of child involvement which is suitable. Once this has been decided, CARSS offers a number of ideas for appropriate activities at each stage of the design process, and a consideration of the skills necessary to ensure a successful design outcome.

ACKNOWLEDGEMENTS

The authors would like to thank Philip Dundas for his input to this paper.

REFERENCES


