

## **The design and evaluation of hypertext structures for supporting design problem solving**

ERICA DE VRIES<sup>1</sup> and TON DE JONG<sup>2</sup>

<sup>1</sup>*Educational Science Laboratory, Pierre Mendès France University, BP 47, 38040 Grenoble cedex 9, France; Phone: (+33) 4 76 82 57 09; Fax: (+33) 4 76 82 78 11; E-mail: Erica.deVries@upmf-grenoble.fr;* <sup>2</sup>*University of Twente, Faculty of Educational Science and Technology, PO Box 217, 7500 AE Enschede, The Netherlands; Phone: (31) 53 48 93 613; Fax: (31) 53 48 92 895; E-mail: jong@edte.utwente.nl*

**Abstract.** Computer-based complex information systems are used increasingly more often, for a growing variety of purposes, in both educational and professional contexts. Since the effectiveness of information systems will largely depend on the particular purpose and the particular task context at hand, at least part of our research efforts should be directed at studying specific application areas. This paper reports a study on the use of hypertext information systems during architectural-design problem solving. Theoretical notions on design problem solving, such as distinguishing between a problem-structuring and a problem-solving phase, provide us with expectations about the changing informational needs during the design process. Specific information structures are proposed, incorporating design principles from learning research, to accommodate these informational needs. Results of an empirical study indeed showed interactions between design phase and information structure when separately inspecting the outcomes for problem structuring and problem solving. Educational implications include the use of a combination of hierarchical decomposition and cross-referencing for certain instructional goals, such as teaching complexity and abstraction.

**Key words:** architectural design, hypertext, problem-solving

### **Introduction**

The area of application of computer-based complex information systems is expanding. Information systems are no longer used for fact finding only, but are also considered a promising tool for knowledge transfer in educational contexts and other specific task contexts. Complex information systems, such as hypertext information systems, are advocated for supporting both learning and problem-solving processes in complex domains, e.g., history (Jacobson and Spiro, 1992), literature (Spiro and Jehng, 1990), and design (Case, 1990; Fischer, McCall, and Morch, 1989). This paper presents an empirical study in the latter domain. The situation studied involves students using a hypertext information system while performing an architectural-design task.

Hypertexts allow users to move through text passages by means of links or electronic references. Two features in particular are thought to be advantageous for learning and problem solving. First, hypertexts provide authors with the opportunity to express the content relations of a domain in the structure of the information, i.e. in the way of linking text passages. These references then determine what information can be chosen at any particular moment. Second, hypertexts allow users to take an individualised route through the information. The user's actions determine which parts of the text are paid attention to, and in which order.

Designing an educational hypertext application involves making use of these features in order to achieve instructional goals. Hence, as has been noticed by Duffy and Jonassen (1991), theories of learning and developments in educational technology are largely intertwined. However, in order to investigate the actual use of complex information systems in situations that involve learning and problem solving, research in educational technology has to deal with a number of issues. First, research has to be directed towards establishing the role that can be played by complex information systems in educational practice in supporting specific learning and problem-solving processes. Second, in the development of these systems, a large number of design decisions have to be taken, the outcomes of which are hard to foresee because of the complexity of learning and problem-solving processes. Third, the observation of students' actual use in educational practice, and the evaluation and assessment of the effectiveness of these complex information systems pose a number of theoretical and methodological problems. These three issues are shortly presented in the remaining part of this section.

#### *Learning and problem solving supported by hypertext*

A number of claims related to learning are based on the above mentioned features of hypertext information systems. According to cognitive flexibility theory, hypertexts offer a proper representation of knowledge through which crucial aspects of knowledge in complex domains can be conveyed (Spiro, Feltovich, Jacobson, and Coulson, 1991; Spiro and Jehng, 1990; Spiro, Vispoel, Schmitz, Samarapungavan, and Boerger, 1987). Furthermore, hypertexts are assumed to stimulate learning because they encourage the structuring and restructuring of knowledge depending on the learner's actions in exploring a specific application (Jonassen and Grabinger, 1990). Similar advantages are thought to hold for supporting problem-solving processes. An empirical investigation of the design principles from the cognitive flexibility theory can be found in Jacobson and Spiro (1992). The main results of this study revealed that while a control treatment led to higher performance on measures of memory for factual knowledge, a more hypertext-like treatment promoted superior

knowledge transfer. Yet, this type of study is scarce, and further empirical investigation on the role that can be played by hypertext information systems in learning and problem solving is needed.

#### *Hypertext development and design issues*

Three broad types of design decisions have to be considered when developing a particular hypertext application: issues regarding content (the raw material), structure (relations between parts of the material), and means of navigation (types of access). These issues have been studied using a variety of question answering tasks (Edwards and Hardman, 1989; Girill and Luk, 1992; McKnight, Dillon, and Richardson, 1990; Mohageg, 1992; Wright and Lickorish, 1990). However, substantial effort has to be put into rendering an information system efficient for a specific task to be carried out, and more importantly, for achieving its associated goals. The latter issue is especially significant since the goal to be reached furnishes the criteria for deciding whether or not the chosen content, structure, and means of navigation are appropriate (see also de Vries, 1994). Aspects of the task, such as the existence of subtasks and subgoals, the nature of the information needed, and the type of users, have to be explicitly taken into account in the design of an application. Examining the implications of a particular main goal can thus lead to specific design decisions.

#### *Evaluation of task performance*

Analysing the characteristics of a particular task context is also essential for guiding the evaluation of proposed information systems. Evaluation raises a number of methodological issues, such as the development of performance measures pertinent to the ultimate learning goals and appropriate to the specific task (de Vries and de Jong, 1997). Measurement of performance in terms of speed and efficiency in the aforementioned question answering tasks does not suffice to establish effectiveness in more complex learning and problem-solving situations. Furthermore, evaluation has to entail a meaningful comparison of an experimental information system with an alternative information system that does not possess the characteristics especially designed for a certain (sub)task. Such meaningful alternative situations permit the kind of comparison that provides necessary empirical support for claims about learning and problem solving with complex information systems.

The presented issues suggest that research into particular application areas is worthwhile. Adapting both the information system's specifications and the evaluation of its effectiveness to a specific situation requires knowledge of the task domain in terms of task phases and needed information. In this

paper, the case of hypertext information systems for design problem solving is examined. First, some theoretical background on design problem solving is given, and informational needs in different phases in the design process are identified on theoretical grounds. Then, alternative structures of information are proposed for supporting design problem solving. Finally, an empirical evaluation of the effectiveness of the proposed structures for specific goals in the design process is presented.

### **A study in the domain of architectural design**

A design process involves the creation of an external representation of something to be made, e.g., the plan of a hospital. As such, designing has been recognised as a cognitive task (Goel and Pirolli, 1992; Simon, 1981). The design process as a problem-solving process has become a subject of study in cognitive psychology.

Design problems are characterised as being both ill structured and involving a large quantity of domain knowledge. This characterisation of design problems leads to a number of statements about design problem solving. Problem solving in design involves satisfying a large set of constraints (Simon, 1973). Moreover, constraints are often implicit, i.e., they are not stated in the problem description. Therefore, a representation phase or problem-structuring phase prior to problem solving is considered to be important (Voss and Post, 1988). This means that information is needed throughout the entire design process.

#### *Design information and design phases*

Roughly three levels of abstraction can be identified in design information. For example, information on the design of a child play area may include *principles* according to which objects may be placed in space, e.g., minding safety or movement, *needs or functions* that have to be fulfilled, e.g., climbing or cleaning, and finally, *possible subsolutions*, e.g., objects like a slide or a fence. Similar distinctions in abstraction levels appear in other design areas (Goel and Pirolli, 1989; Visser, 1990). In the following, these three levels will be called abstract concepts, performance requirements, and materialisations, respectively. The three kinds of information do not have the same salience throughout the design process.

According to both prescriptive and descriptive design theories, gathering information is important throughout the entire design process. Normative theories prescribe an analysis stage in which available information is sought (Jones, 1980; Lawson, 1990). Early information gathering is supposed to

have a divergent character, whereas later information gathering has to converge in order to develop a solution. Descriptive theories situate information-gathering-processes at the beginning of the design process as well (Akin, 1986; Hamel, 1990; Rowland, 1992).

At the same time, two types of activities can be distinguished, roughly corresponding to two phases in the design process: *problem structuring* and *problem solving* (see for example Goel and Pirolli, 1992). Although problem-structuring and problem-solving activities may occur throughout the entire design process, problem structuring characterises the beginning of a design process and problem solving takes place more towards the end. At the start of the design process, a problem is stated verbally in terms of purposes or needs that have to be fulfilled, i.e., abstract concepts. Towards the end of the design process, a solution to the problem is developed. A solution consists of a description of the proposed artefact, i.e., materialisation, and of the way in which it has to be used in order to function properly, i.e., fulfill a performance requirement. Thus, problem structuring presumably involves higher abstraction levels because it has to produce the purposes and higher goals to be satisfied by an artefact. Problem solving involves materialisations and concrete examples of solutions. At the end of the solution process, a designer generally provides a justification to show that the proposed artefact constitutes a solution to the problem.

#### *Rationale of the study*

As a consequence of the shift from problem structuring to problem solving, information needs are likely to shift from abstract concepts in the beginning, to materialisations towards the end of the design process. The question arises how to adapt the hypertext structure to the demands of the design process, and, more specifically, when making a distinction between problem structuring and problem solving.

There are several ways to structure information in a hypertext aimed at supporting design activities. A traditional technique uses networks in which original text passages are linked according to content relations. In such an *integrated network* structure, the three abstraction levels, abstract concepts, performance requirements, and materialisations, are intertwined in information sections, and references permit quick navigation from section to section. Another way to structure information is through using the nature of design information to create an *abstraction hierarchy*. In an abstraction hierarchy, each screen presents an element on only one of three abstraction levels. An abstraction hierarchy allows navigation from concept to performance requirement to materialisation. Such an abstraction hierarchy embodies a hierarchical organisation, as advocated by Eylon and Reif (1984) in the context of physics

problem solving. The abstract concepts are high in the hierarchy since they are considered important in designing an artefact adapted to its intended use. Furthermore, according to the recommendations of Eylon and Reif (1984), the hierarchical organisation may be explicitly explained prior to working with the hypertext. An alternative version of the abstraction hierarchy, the *cross-referenced abstraction hierarchy*, permits navigation within abstraction levels. An abstract concept gives references to performance requirements but also references to other abstract concepts. Similarly, a performance requirement gives references to materialisations but also to related performance requirements. A cross-referenced abstraction hierarchy incorporates one of the design principles from cognitive flexibility theory (Spiro, Feltoovich, Jacobson, and Coulson, 1991). It enables criss-crossing from element to element on each abstraction level, e.g., from materialisation to materialisation.

The experiment presented was designed to investigate the use of these information structures during design problem solving. We created a situation in which subjects, involved in either a problem-structuring or a problem-solving activity, had access to one of three differently organised information systems (integrated network, abstraction hierarchy, or cross-referenced abstraction hierarchy). The problem-solving phase was studied independently from the problem-structuring phase in order to avoid an influence of individual differences in problem structuring on subsequent problem solving.

### *Expectations*

As mentioned in the introduction, traditional measures such as speed and accuracy are not appropriate for evaluating task performance in situations that involve complex problem solving. Instead, performance was assessed by looking at desired outcomes for the particular task situation at hand. The outcome of the problem-structuring phase was investigated by measuring the enlargement of the information span, i.e., of the number of issues which a subject is able to generate on a particular design problem. The produced design proposal and its justification were assessed representing the desired outcome of the problem solving phase.

For the experiment, the following expectations can be stated. In *problem structuring*, an organisation into abstraction levels is expected to be beneficial, since attention can be paid to the abstract concepts in particular. The cross-referenced abstraction hierarchy is expected to be the most appropriate since this structure permits staying at the level of abstract concepts. The attention paid to abstract concepts is expected to be expressed in the use of the hypertext, and reflected in an enlargement of the information span. In *problem solving*, however, the advantages of organisation into abstraction level may be less

pronounced. Nevertheless, the cross-referenced abstraction hierarchy again permits staying at one level. In this phase, the preferred levels are expected to be the lower levels: performance requirements and materialisations. The cross-referenced abstraction hierarchy condition may also result in visiting a lot of materialisations at the expense of performance requirements and abstract concepts. As a consequence, the justification or argumentation of the design proposals may contain a smaller number of abstract concepts in this condition.

Finally, in order to explore the impact on the product of problem solving, the actual design proposals produced were compared across hypertext structure conditions.

## **Method**

In the experiment, subjects were engaged in either a problem-structuring or a problem-solving activity during which they were invited to use a hypertext information system.

### *Domain*

The domain of the study was the design of child play areas. The hypertext contained design information in the domain of child play areas (adapted from Cohen, Hill, Lane, McGinty, and Moore, 1979).

Three structure conditions were created: integrated network, abstraction hierarchy, and cross-referenced abstraction hierarchy. The integrated network condition consisted of a network of 56 sections on child play areas. In this condition, each of the 56 sections, including title, issue, principle, recommendation, and references, occupied one screen (see Figure 1). The references at the bottom of each section pointed to titles of other sections. The integrated network (Figure 2) can be seen as an (abbreviated) electronic version of the original text.

Furthermore, two experimental structures were developed by rearranging the material in order to distinguish elements at three abstraction levels. An abstraction hierarchy (see Figure 3) was created consisting of 21 concepts, 45 performance requirements, and 69 materialisations. Each concept pointed to 1–4 performance requirements, and each performance requirement pointed to 1–4 materialisations.

The cross-referenced abstraction hierarchy was identical to the abstraction hierarchy, but one feature was added that permitted navigation on one particular abstraction level: cross-references (see Figure 3). For example, abstract concepts in this condition contained names of related abstract concepts.

**Retreat and breakaway points**

**Issue**  
Children occasionally need solitude for self-knowledge and self-control. They need to be able to get away by themselves from time to time to think, regain control, and escape from a too-intensive play situation.

**Principle**  
Sheltered, separated spaces which are large enough for one person only, located in various parts of the play area will provide necessary retreat points for children. Provision must also be made for a child to leave before completing an activity without feeling as a failure.

**Recommendation**  
Provide breakaway points within an activity, e.g. a ladder might have access to two platforms before reaching the top level, thus permitting children to stop at any stage while they are overcoming fears of height.

-> Nests of quiet play  
-> Range of social scale      <- Back = Starting Screen

Figure 1. Example of a section in the integrated network condition.

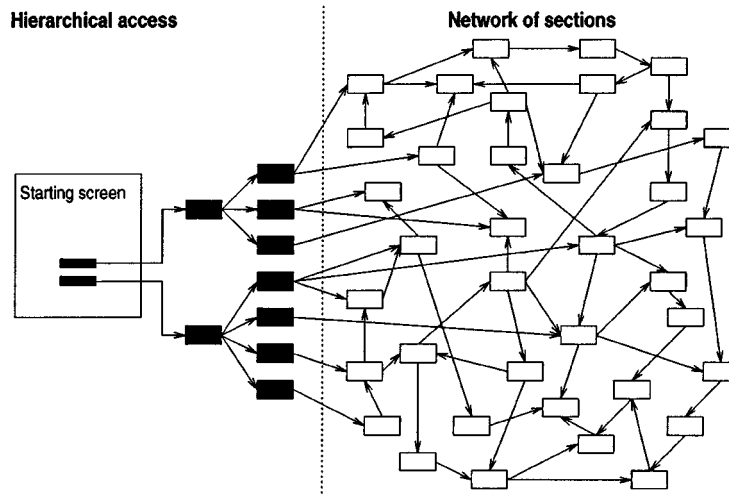


Figure 2. Diagram of the integrated network.

### Subjects

Fifty-six 3rd to 5th year students in architecture with some experience in designing participated in the study. The subjects were paid for their participation.

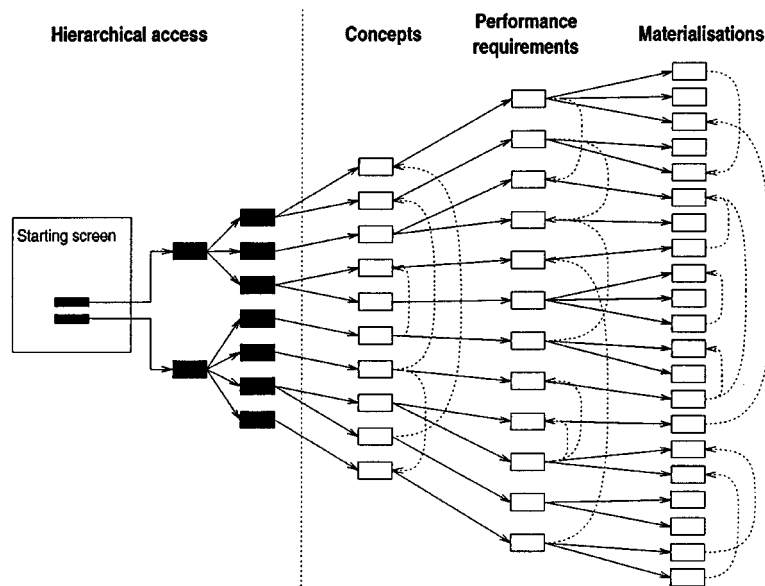


Figure 3. Diagram of an abstraction hierarchy. Links (dotted lines) within a level are only available in the cross-referenced abstraction hierarchy.

Table 1. Overview of the experimental design and number of subjects

Structure	Group	
	Problem structuring	Problem solving
Integrated network	9	9
Abstraction hierarchy	9	10
Cross-referenced abstraction hierarchy	9	10

The subjects were randomly assigned to one of three structure conditions (integrated network, abstraction hierarchy, and cross-referenced abstraction hierarchy) and to one of two design phase groups (a problem-structuring and a problem-solving group). The fifty-six subjects were distributed over the experimental conditions as displayed in Table 1.

### Procedure

The problem-structuring group completed a pre-test, an assignment with the hypertext and a post-test. The pre-test was administered to assess the subject's prior knowledge in terms of the number of issues he or she was able to generate on the design of a child play area (the initial information span). Subjects were invited to write down all the ideas they could think

of regarding the design of such a child play area. They were instructed to imagine receiving the assignment of the design of a child play area near a school, and some characteristics of the school, such as number of children, were given. This description also served the purpose of setting the framework of the problem-structuring activity. Hereafter, the subjects in this group were asked to get to know as much as possible about designing child play areas. However, it was mentioned that this did not mean that they had to look at everything in the hypertext, but that they could choose the topics of their interest. The subjects were given between 30 and 45 minutes to explore the hypertext. The post-test was analogous to the pre-test.

The problem-solving group was given a problem description for the design of a playground near a school in a park. In order to bypass the problem-structuring phase, five aspects representing the result of problem structuring were added to the problem description. The five aspects (lay-out, creativity, body movement, social behaviour and appropriate utilities) were introduced by telling the subjects that the school board, following the advice of a designer, identified them as being particularly important in the project. The subjects were told that they should bear these aspects in mind, and that the hypertext contained information about them. The time in which to make a design proposal was about 75 minutes. Finally, subjects in the problem-solving group filled in an argumentation form in which they were invited to comment on their designs.

#### *Scoring procedure*

The problem-structuring group wrote pre-test and post-test reports, in which the subjects indicated the objects and ideas they would think of when designing a child play area. The statements in the reports were coded using a classification scheme. The coding of the reports allowed assessment of the *size* of the information span (the number of elements), its *variety* (the number of different themes that were activated), and the *abstraction level* at which the elements were expressed. An example of a statement is "For safety / I would design a fence / and for body movement / I would make a high object". This statement contains four elements (separated by slashes), two themes, i.e., safety and body movement, and two abstraction levels, i.e., abstract concepts and materialisations. After a short practising period, all reports were scored separately by two raters. The codes were compared and reliability over the whole set of data was calculated in two steps: theme and abstraction level. The percentage of agreement on the themes was 74% or 820 out of 1113 decisions, where each decision could involve up to three elements. Cohen's *Kappa* was computed to correct for chance level,  $Kappa = 0.72$ . Reliability for the level of abstraction was calculated by looking at the episodes for which

there was agreement on the theme. The 820 themes concerned 1265 single elements. Percentage of agreement for the level of abstraction was 73% (924 out of 1265 elements). *Kappa* for abstraction level was 0.60. The results of the scoring by the two raters were compared, and differences were discussed until agreement was reached. The size (number of elements) and variety (number of themes) was calculated for each pre-test and post-test. In addition, the increase in size and variety as a result of hypertext use was calculated by taking the difference between pre-test and post-test. The enlargement in size and variety was also calculated as a function of abstraction level.

The argumentation reports of the problem-solving group were coded in roughly the same way. However, the phrasing was often not as clear as in the problem-structuring reports. Therefore, the two raters jointly scored the 29 argumentation reports. Size and variety of the argumentation reports were calculated.

After scoring each proposal, raters assigned a value for the degree in which attention was paid to the aspects mentioned in the problem description. Each aspect was rated on the basis of the design proposal and its argumentation. A score of 0 (no apparent concern), 1 (some concern) or 2 (sufficient concern) was given for each aspect, and for each design proposal. Individual *Kappa*'s for the 5 aspects were low (Aspect 1 through Aspect 5: *Kappa* = 0.28, 0.37, 0.67, 0.26, 0.47). Percentage agreement over all aspects was 66% (99/150), *Kappa* = 0.46. In comparing the scores, all differences were discussed until agreement between raters was reached. Moreover, a significant correlation was found between sumscores of Rater 1 and Rater 2 (Pearson's  $r = 0.64$ ,  $p < 0.001$ ). The agreed-upon scores for each aspect of each design proposal were summed up to obtain an aspect score for the degree in which the proposals paid attention to the aspects in the problem description.

## Results

The results are presented in three parts. First, general hypertext use is examined, presenting both time spent working with the hypertext and exploration of the hypertext (percentage of the available information screens visited). Second, the outcomes of the problem-structuring phase are presented by looking at the enlargement in size and variety of the information span. Finally, the argumentation reports and the design proposals, as a result of the problem-solving phase, are inspected.

*General hypertext use*

Mean time spent using the hypertext was 33 minutes ( $SD = 8$ ). Despite the differences in assignment of the problem-structuring and the problem-solving phase, subjects spent about the same amount of time using the hypertext ( $F(1, 50) = 0.18$ , ns). Furthermore, an overall effect of structure of information was found ( $F(2, 50) = 3.22$ ,  $p < 0.05$ ). Subjects in the integrated network condition spent less time ( $M = 29$  minutes) than subjects in the abstraction hierarchy and cross-referenced abstraction hierarchy condition ( $M = 34$  and  $M = 35$  minutes respectively).

Both abstraction hierarchies, but not the integrated network, allow measurement of the time spent at each abstraction level. Subjects using the abstraction hierarchies in the problem-structuring group spent slightly more time on the abstract concepts ( $M = 11$  minutes) than on the materialisations ( $M = 9$  minutes). In the problem-solving group, the opposite was observed ( $M = 8$  and  $M = 12$ , respectively). The interaction between design phase and abstraction level was significant ( $F(2, 33) = 6.07$ ,  $p < 0.01$ ).

Regarding the exploration of the hypertext, an analysis of variance showed a structure effect ( $F(2, 50) = 5.17$ ,  $p < 0.01$ ). Exploration was higher in the integrated network condition (78%) as compared to the abstraction hierarchy conditions (62% and 59%).

In the abstraction hierarchy conditions, an effect of abstraction level on exploration was found ( $F(2, 33) = 19.60$ ,  $p < 0.01$ ). Exploration in percentages was highest for abstract concepts followed by the performance requirements and materialisation levels.

*Problem structuring and hypertext organisation*

The increase in the information span in the problem-structuring group was calculated by taking the post-test – pre-test difference in size (number of elements) and variety (number of themes). In general, an enlargement took place in size ( $M = 16.9$  elements) and in variety ( $M = 4.9$  themes). This overall enlargement was present in all three structure conditions.

However, the enlargement was not equally distributed over the three abstraction levels. The contrast for testing the linear trend in enlargement across abstraction levels was significant for the enlargement in size ( $F(1, 24) = 4.27$ ,  $p < 0.05$ ), but only approached significance for variety ( $F(1, 24) = 3.45$ ,  $p < 0.10$ ). In general, the enlargement increases from abstract concepts to performance requirements, and finally to materialisations.

In the next step, structure differences in the distribution of the enlargement across abstraction levels was examined. The Structure x Abstraction Level interaction effect reached significance for variety ( $F(4, 48) = 2.81$ ,  $p < 0.05$ ),

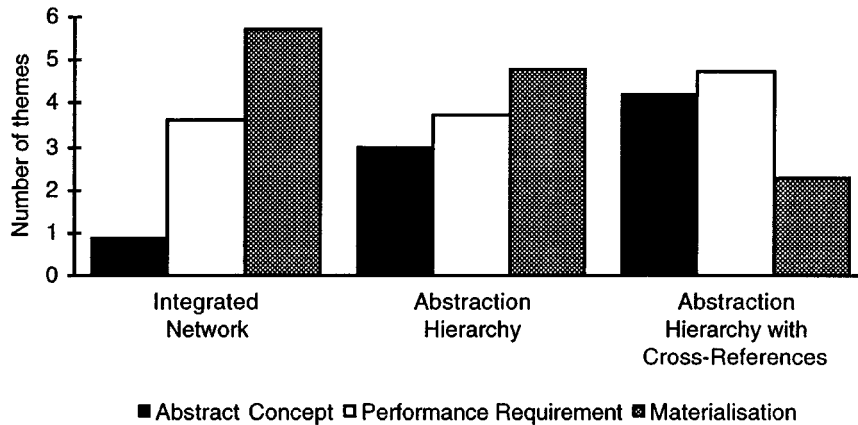


Figure 4. Problem structuring: Enlargement in variety (number of themes) as a function of information structure and abstraction level.

but did not reach significance for size ( $F(4, 48) = 1.54$ , ns). Figure 4 shows that in the integrated network condition, the enlargement in variety was low for abstract concepts, somewhat higher for performance requirements and highest for materialisations. This was also observed in the abstraction hierarchy condition, but to a lesser extent. In the cross-referenced condition, the increase in variety was high on the level of abstract concepts and low on the level of materialisations. The parameters for the interaction between abstraction level (abstract concepts vs materialisations) and structure (integrated network vs cross-referenced abstraction hierarchy) were significant for size and variety. In the integrated network, the increase of abstract concepts was low, whereas the increase of materialisations was high. In the cross-referenced abstraction hierarchy, the opposite was observed.

#### *Problem solving and hypertext organisation*

After finishing the proposal, the subjects in the problem solving group were asked to give a description of their design proposal with arguments for their choices and design decisions. The mean size of these argumentation reports was 32.8 elements (SD = 10.3). The variety of themes appearing in the reports was 11.6 different themes (SD = 1.7). No structure differences on overall size and variety were detected.

An analysis of variance revealed an abstraction level effect on size and variety ( $F(2, 25) = 57.27$ ,  $p < 0.001$ , and  $F(2, 25) = 48.25$ ,  $p < 0.001$ , respectively). The argumentation reports contained a high number of materialisations; less performance requirements and very few mentions of abstract concepts (see Figure 5). However, no Structure x Abstraction level interaction

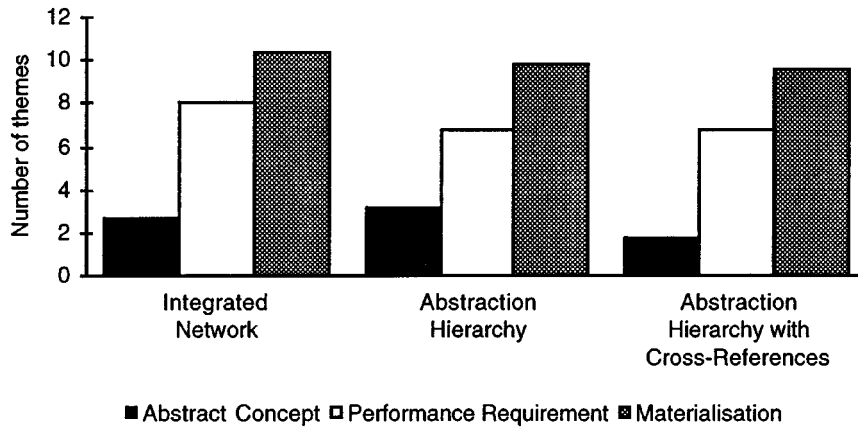


Figure 5. Problem solving: Variety (number of themes) in the argumentation reports as a function of information structure and abstraction level.

Table 2. Mean sumscore for the aspects from the problem description as a function of structure

Structure	Aspect score	
	<i>M</i>	<i>SD</i>
Integrated network	4.2	1.1
Abstraction hierarchy	4.3	1.9
Cross-referenced abstraction hierarchy	3.4	1.3

on size ( $F(4, 52) = 1.08$ , ns) or on variety ( $F(4, 52) = 0.46$ , ns) was found. This means that the argumentation reports showed similar distributions over abstraction levels, regardless of hypertext structure.

The problem description in the problem-solving phase mentioned a number of aspects. It was explicitly stated that these aspects were crucial to the design of the child play area. Design proposals were rated in order to measure the degree in which attention was paid to the five aspects (maximum = 10). Table 2 displays the mean aspect scores as a function of structure.

This table shows that aspect scores were relatively low (about 40% of the maximum score). Analysis of variance showed no effect of structure ( $F(2, 26) = 1.06$ , ns). There was no difference in structure with respect to the degree in which subjects paid attention to aspects in the problem description. Finally, no correlation was found between the exploration of the hypertext and the aspect score in any of the three structure conditions, integrated network ( $r = -0.13$ , ns), abstraction hierarchy ( $r = 0.02$ , ns), or cross-referenced abstraction hierarchy ( $r = 0.30$ , ns).

## Conclusions and discussion

In this study, interactions between design phase and information structure can be examined. The effectiveness of different information structures is investigated by looking at the desired outcome of two main phases: problem structuring and problem solving. At first sight, the integrated network structure seems to be the most efficient structure. In such an information structure, a larger percentage of the information can be covered in less time than in abstraction hierarchies with and without cross-references. The latter structures, however, permit a more selective choice of the type of information to which attention is paid in a particular phase or during a particular activity. In the case of problem structuring, this means that abstract concepts are favoured, whereas during problem solving, more time is spent on the materialisation level. This effect is strongest in the cross-referenced abstraction hierarchy because this structure permits navigation using cross-references per level. However, although in the hierarchical structures more time is spent on the materialisation level, this does not result in more exploration on that level.

The hierarchical organisation was expected to be task-adapted because abstract concepts are high in the hierarchy and because the organisation was explicitly mentioned in the instructions for using the hypertext (Eylon and Reif, 1984). The cross-references enable criss-crossing within one level. In problem structuring, this did benefit the abstract concept level in the information span, i.e., the higher purposes or goals that the artefact eventually has to fulfill. So, although the integrated network structure permits exploration of more information, an ordering into abstraction level with cross-references permits a selective focus on abstract concepts, and this in turn results in a larger increase of abstract concepts in the information span. For problem solving, the results are less clear. The argumentation of the design proposals did not show differences between structure conditions. Furthermore, no structure differences existed in the aspect scores for the design proposals. The integrated network structure can be considered the most efficient structure because the same results on argumentation and aspect scores can be obtained with a larger exploration of the hypertext in less time.

### *Adapting structure to activity*

Our study showed interactions between specific activities in a task situation and the way in which information is presented. For problem structuring directed towards higher abstraction levels (divergent processes), our study showed that the structure of information matters. However, for problem solving directed towards the lower abstraction levels (convergent processes), the outcomes remained relatively unaffected by the structure of the information.

Information structures which allow switching between topics have been recommended in other design disciplines such as software design (Visser, 1990). Whereas normative theories of software design and mechanical engineering prescribe a top-down approach of hierarchical decomposition, empirical research showed that even expert programmers and mechanical engineers showed opportunistic deviations from hierarchical plans (Guindon, 1990; Visser, 1990). So, whereas network structures support opportunism, it is clear that whenever consultation of an information system involves a particular search goal, this demands some kind of organisation. Furthermore, whereas in a real design process periods of problem structuring and problem solving may alternate, subjects in our study were engaged in one of the activities roughly corresponding to two main design phases. A combined approach to supporting design processes would therefore entail an information system which proposes an information structure depending on the particular purpose for using the information system, or on the particular activity at hand. Such an information system would require a technique for evaluating users' purposes, by for example, asking the user to formulate his or her main goal upon entering the information system (de Vries, 1994), by taking into account a designer's working style (van Bakel, 1995), or by dynamically building a model of the user's goals based on the interaction with the information system (Brusilovsky, 1996).

#### *Educational implications*

Results of this study also lead to some educational implications. Learning in complex domains is thought to be fostered by exploration of multiple cases and multiple conceptual perspectives (Spiro et al., 1991, 1987). Moreover, an abstraction hierarchy allows showing top-down relationships, e.g., from goal through function to materialisation. Such a hierarchy has to be explicitly taught for learning to take place (Eylon and Reif, 1984). These two instructional goals, teaching complexity and abstraction, may be accomplished by using a combination of hierarchical decomposition and cross-referencing. The hypertext information system then functions as a representation of knowledge in a domain which can be explored by students. Students can select a preferred mode of traversing, either along the lines of an abstraction hierarchy and means-end relationships or along lines of content relations between concepts. Alternatively, in some educational stages, students may be allowed to consult only one particular level of abstraction. In both cases, education would have to change from linearly presenting domain knowledge to supporting exploration of domain knowledge (Pieters, 1992).

Finally, this paper shows how the design of an information system can be guided by an analysis of how their use may serve in achieving learning and

problem-solving goals. Research on the use of complex information systems in general could broaden the scope of its conclusions by developing appropriate tasks and by augmenting the relevancy of assessment of performance. More attention has to be paid to the setting in which a complex information system is used, especially in cases in which information gathering is not the main task, but a subtask, as in learning and problem solving. In this way, design and evaluation of educational complex information systems may benefit both learning theory and instructional technology.

### Acknowledgements

This research was carried out at the Department of Philosophy and Social Sciences of the Eindhoven University of Technology in The Netherlands. The authors would like to express their gratitude to Joost van Andel and Jules Pieters for many helpful discussions related to the topic of this paper.

### References

- Akin, Ö. (1986). *Psychology of Architectural Design*. London: Pion.
- Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. *User Modeling and User-Adapted Interaction* 6: 87–129.
- Case, D. (1990). Using hypertext to create design programming databases. *Journal of Interior Design Education and Research* 16: 37–52.
- Cohen, U., Hill, A.B., Lane, C.G., McGinty, T. and Moore, G.T. (1979). Recommendations for child play areas (Report R79-1). Milwaukee: University of Wisconsin-Milwaukee, Center for Architecture and Urban Planning.
- De Vries, E. (1994). Structuring information for design problem solving. Doctoral dissertation, Eindhoven University of Technology, Eindhoven, The Netherlands.
- De Vries, E. and De Jong, T. (1997). Using information systems while performing complex tasks: An example from architectural design. *International Journal of Human-Computer Studies* 46: 31–54.
- Duffy, T.M. and Jonassen, D.H. (1991, May). Constructivism: New implications for instructional technology? *Educational Technology*, pp. 7–12.
- Edwards, D.M. and Hardman, L. (1989). Lost in hyperspace: Cognitive mapping and navigation in a hypertext environment. In R. McAleese, ed., *Hypertext: Theory into Practice*, pp. 105–125. Oxford: Intellect Limited.
- Eylon, B.S. and Reif, F. (1984). Effects of knowledge organization on task performance. *Cognition and Instruction* 1: 5–44.
- Fischer, G., McCall, R. and Morch, A. (1989). JANUS: Integrating hypertext with a knowledge-based design environment. In *Proceedings of Hypertext '89*, pp. 105–118. Pittsburgh, Pennsylvania, New York: ACM.
- Girill, T.R. and Luk, C.H. (1992). Hierarchical search support for hypertext on-line documentation. *International Journal of Man-Machine Studies* 36: 571–585.
- Goel, V. and Pirolli, P. (1989, Spring). Motivating the notion of generic design within information processing theory: the design problem space. *AI Magazine*, pp. 18–36.
- Goel, V. and Pirolli, P. (1992). The structure of design problem spaces. *Cognitive Science* 16: 395–429.

- Guindon, R. (1990). Designing the design process: Exploiting opportunistic thoughts. *Human-Computer Interaction* 5: 305–344.
- Hamel, R. (1990). Over het denken van de architect: Een cognitief psychologische beschrijving van het ontwerpproces bij architecten (On designing by architects: A cognitive psychological description of the architectural design process). Doctoral dissertation, University of Amsterdam, Amsterdam, The Netherlands.
- Jacobson, M.J. and Spiro, R.J. (1992). Hypertext learning environments, cognitive flexibility, and the transfer of complex knowledge: An empirical investigation. Paper presented at the Annual Meeting of the *American Educational Research Association*, San Francisco, CA.
- Jonassen, D.H. and Grabinger, R.S. (1990). Problems and issues in designing hypertext/hypermedia for learning. In D.H. Jonassen and H. Mandl, eds, *Designing Hypermedia for Learning*, pp. 3–25. Berlin Heidelberg: Springer-Verlag.
- Jones, J.C. (1980). *Design Methods: Seeds of Human Futures* (2nd edn). New York: Wiley.
- Lawson, B. (1990). *How Designers Think* (2nd edn). London: The Architectural Press.
- McKnight, C., Dillon, A. and Richardson, J. (1990). A comparison of linear and hypertext formats in information retrieval. In R. McAleese and C. Green, eds, *Hypertext: State of the Art*, pp. 10–19. Oxford: Intellect Limited.
- Mohageg, M.F. (1992). The influence of hypertext linking structures on the efficiency of information retrieval. *Human Factors* 34: 351–367.
- Pieters, J.M. (1992). Het ongekende talent: over het ontwerpen van artefacten in de instructietechnologie (The unknown talent: on the design of artefacts in instruction technology). Inaugural lecture, University of Twente, Enschede, The Netherlands.
- Rowland, G. (1992). What do instructional designers actually do? An empirical investigation of expert practice. *Performance Improvement Quarterly* 5: 65–86.
- Simon, H.A. (1973). The structure of ill-structured problems. *Artificial Intelligence* 4: 181–201.
- Simon, H.A. (1981). *The Sciences of the Artificial* (2nd edn). Cambridge MA: MIT Press.
- Spiro, R.J., Feltovich, P.J., Jacobson, M.J. and Coulson, R.L. (1991, May). Cognitive flexibility, constructivism and hypertext: Random access instruction for advanced knowledge acquisition in ill-structured domains. *Educational Technology*, pp. 24–33.
- Spiro, R.J. and Jehng, J.-C. (1990). Cognitive flexibility and hypertext: Theory and technology for the nonlinear and multidimensional traversal of complex subject matter. In D. Nix and R.J. Spiro, eds, *Cognition, Education and Multimedia: Exploring Ideas in High Technology*, pp. 163–205. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Spiro, R.J., Vispoel W.P., Schmitz, J.G., Samarapungavan, A. and Boerger, A.E. (1987). Knowledge acquisition for application. In B.K. Britton and S.M. Glynn, eds, *Executive Control Processes in Reading*, pp. 177–199. London: Erlbaum.
- Van Bakel, A.P.M (1995). Styles of architectural designing: Empirical research on working styles and personality dispositions. Doctoral dissertation, Eindhoven University of Technology, Eindhoven, The Netherlands.
- Visser, W. (1990). More or less following a plan during design: Opportunistic deviations in specification. *International Journal of Man-Machine Studies* 33: 247–278.
- Voss, J.F. and Post, T.A. (1988). On the solving of ill-structured problems. In M.T.H. Chi, R. Glaser and M.J. Farr, eds, *The Nature of Expertise*, pp. 261–285. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Wright, P. and Lickorish, A. (1990). An empirical comparison of two navigation systems for two hypertexts. In R. McAleese and C. Green, eds, *Hypertext: State of the Art*, pp. 84–93. Oxford: Intellect Limited.