A case study on the design of learning interfaces

Gabriela Trindade Perry\textsuperscript{a,}*, Fernando Schnaid\textsuperscript{b}

\textsuperscript{a} DEG/UFRGS – Departamento de Design e Expressão Gráfica/Universidade Federal do Rio Grande do Sul, Av. Sarmento Leite, 320, CEP 90040-060 Porto Alegre, Brazil
\textsuperscript{b} PPGEC/UFRGS – Programa de Pós Graduação em Engenharia Civil/Universidade Federal do Rio Grande do Sul, Av. Osvaldo Aranha, 99, 3° andar, CEP 90035-190 Porto Alegre, Brazil

\textbf{ARTICLE INFO}

\textbf{Article history:}
Received 3 November 2011
Received in revised form 31 February 2012
Accepted 6 March 2012

\textbf{Keywords:}
Human–computer interface
Interdisciplinary projects
Multimedia/hypermedia systems

\textbf{ABSTRACT}

The design of educational software interfaces is a complex task, given its high domain dependency and multidisciplinary nature. It requires that teachers’ knowledge and pedagogical beliefs be incorporated into the interface, posing a challenge to both teachers and designers, as they have to act as partners from the earliest phases of the process, sharing their knowledge. The present work investigates the strategies designers used when paired with experienced teachers, to design two interfaces on chemistry, evaluating how designers work with subjects they know little about, in the initial phases of the design process. Our observations demonstrate that although experienced and non-experienced designers use different strategies to couple with the design task, both approached the task in a depth-first manner. These results should not be generalized, because few subjects were investigated, but point to the importance of being familiar with the knowledge domain – which poses a challenge for designers.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

The prevalence of Information and Communication Technologies (ICT) for educational environments justifies the urge to understand how to better design educational artifacts. Despite accumulated experience and knowledge, it still poses a great challenge. Winters and Mor (2008) state that the methodological weakness in the development [design] of such artifacts might be the reason they did not have the desired engaging effect [in educational settings]. However, the fact that it is difficult to design, and that design itself is difficult to teach (Schön, 1983), might be an explanation for this scenario. Lawson and Dorst (2009) report several declarations of professional designers and teachers describing their practice as a mixture of passion and uncertainty. From a theoretical point of view, this “feeling of uncertainty” could be explained by the assumption that design requires a different kind of thinking – a cognitive process that cannot be described by the symbolic information-processing framework developed by Newell and Simon (1972), as it deals with “fluid” states and representations (Goel, 1995). Defining design as a problem-solving activity – for which Newell and Simon’s problem-solving concepts are sufficient – opposes two of the main theories on design cognition: Simon’s (1996) and Schön’s (1983). These authors described design in opposite ways, but both hold that solving a design problem [or facing a design situation, according to Schön’s terminology] is very different from approaching a “well-defined problem”, because designers do not have algorithmic paths to follow from problem to solution.

2. Control strategies for approaching design problems

The class of problems designers face has implications on their strategy to approach the design task (also called control strategy): it could be breadth-first, depth-first or an adaptive combination of both (Visser, 1994). According to Ball and Ormerod (1995), a breadth-first strategy has many characteristics, such as a top-level design goal reduced into a number of sub-goals, which are decomposed until a requisite level of design detail is reached – no sub-goal is explored in depth. Breadth-first strategies also have the benefits of optimizing the search for solutions and reducing the commitment to early solutions. In contrast, the depth-first strategy entails taking one top-level sub-goal at a time, and developing it in considerable detail before moving on to the next sub-goal.

Ball and Ormerod (1995) argued that the difference between expert and novice designers can be characterized in terms of favoring the use of one approach over another. Expert designers prefer primarily breadth-first behavior (with occasional depth-first incursions), whereas...
novice designers primarily tend toward depth-first behavior. However, as Akin (1986) notes, a mix of these strategies is expected, because a designer will not use only one of these strategies all the time. Ball, Onarheim, and Christensen (2010) reported that designers tend to begin a task with a breadth-first strategy, and then switch to a depth-first strategy after major requirements [of the artifact being designed] are satisfied.

Deviations from these strategies have been labeled “opportunistic” shortcuts from a top-down structured approach to the design task. Ball and Ormerod (1995) argued that this might have been a “blanket” term used to describe situations that deviate from those control strategies, such as those which might be caused by memory and design failures, information unavailability, boredom and serendipitous events. This review of design strategies will ground our conclusions about the strategies employed by the subjects of this research.

At this point, it is important to stress that only one of the studies we found in our review investigated the way in which expert designers approached a task in unfamiliar domains. Adelson and Soloway (1985) found that, in these cases, experts would pursue a depth-first mode of solution development, and that the mental models of the artifact were less detailed – this did not happen when they worked in familiar domains. Regarding the design of educational software interfaces however, we would argue that designers generally work with unfamiliar domains. To support this hypothesis, characteristics of the design of educational artifacts will be described, and similarities and possible differences with the prototypical design areas – architecture and engineering – from which most of the design cognition knowledge has emerged, will be explored in the next section.

3. What is so special about educational software interfaces?

As a genuine design problem, educational software interfaces share a relevant characteristic with all design problems: they belong to the same class (be it ill-defined, wicked or indeterministic). Besides, there are two relevant characteristics: (1) they focus on teaching and learning, which are complex, multifaceted endeavors, and (2) they cannot be designed, even in the conceptual phase, by a single person, nor by a team with a single area of expertise.

To say that educational software is focused on teaching and learning is more than to merely state its purpose. It also conveys the commitment to translate the theoretical interpretation of these processes into an instructional design and into a user interface. This is not an easy goal to meet, as there are no direct and objective means to translate any theoretical interpretation into software features.

The second characteristic that we think makes educational software so special is its inherent interdisciplinarity. In interdisciplinary design teams, people have to integrate information, techniques, tools, concepts and theories from more than one body of knowledge, to solve problems whose solutions are beyond the scope of a single discipline (Committee on Science, Engineering and Public Policy, 2004). Squires (1999, p. 463) pointed out that “workers in these areas (design, programming and teaching) rarely speak to each other or take note of each others’ work”. In research on usability evaluation for educational software, Zaharias and Poylymenakou (2009, p, 76) mention an “ellipsis of research validated usability evaluation methods that address the user as a learner in a holistic way, which includes the consideration of cognitive and affective learning factors”. Hinostroza and Mellor (2001, p. 27) assert that in addition to having knowledge of learning theories, educational software designers should take teaching practices into consideration. The aforementioned authors point to a scenario where all members of the design team should share a large, multidisciplinary base of knowledge.

4. How is educational software designed?

The ideal educational software design team has several agents. These include, for example, teachers and instructional designers, analysts and programmers with many backgrounds, graphic, motion and interaction designers, writers and text-reviewers. Ideally, these team members would also share knowledge on relevant issues such as technology limitations and possibilities, design practices, classroom routines, and teaching practice and learning theories.

Siozos, Palaiogeorgiou, Triantafyllakos, and Despotakis (2009) achieved this through the adoption of participatory design practices, which besides encouraging participation of all team members in every phase of the development, brings users to the design team. Triantafyllakos, Palaiogeorgiou, and Tsoukalas (2011), by their turn, used a “design game” framework, called “We!Design&Play”, for designing educational software. The goal of design games is to focus on a set of skills, which are important in approaching design tasks. Perhaps the most famous example of a design game is the one you play with marshmallows and spaghetti, trying to build a stable structure, as tall as possible.

Winters and Mor (2008), for example, approached the integration of this multifaceted knowledge through the adoption of design patterns. During a one-year project, they elaborated and connected 120 patterns that were used in the design of educational environments, with the aid of a web toolkit. Crosier et al. (2002) proposed a less structured, yet more flexible method, based on user centered design and focusing on the integration of teachers with the development team.

In addition to methodological proposals, there are also research studies on guidelines for designing learning interfaces. The aim of these studies is to propose and validate a set of guidelines that, if followed, would help to design better interfaces. Regarding the design of Collaborative Learning Environments, Rubens, Emans, Leinonen, Skarmeta, and Simons (2005) proposed seven design principles, based on an analysis of practices in European countries. Jones (2008) advises the use of storyboards as a means to bootstrap the design of learning tasks. Ariga and Watanabe (2008) designed a web design course to enhance the “visual expression” of students who are not in art or design courses.

The point of this brief presentation is to show that, as in all areas of design practice, there is an effort to create better ways to manage the process of learning interfaces development. These methodological efforts represent the crystallised knowledge of a community and, as such, are of great value. Regarding our research, they provided insights about how the collaboration of actors of several different backgrounds might occur.

5. Material and methods: observations of a learning interface design session

The subjects of our observations were a digital/graphical designer with eight years of experience (subject ED), a design student, who was in the middle of the graduation course (subject ND), a teacher who has a master degree in chemistry education (teacher TA) and a professor...
with a PhD in chemistry education (professor TB). Both chemists were co-workers for the past two years. The group was divided into the following pairs: ED + TA and ND + TB. We chose to pair the teachers with designer with different levels of experience, to investigate differences in the way designers would approach the task.

We stress that having only four subjects does not undermine the research, as this can be described as a case study (Yin, 2009). Examples of close investigation of few subjects can be found in Suwa, Gero, and Purcell (2000), Tan and Melles (2010) and Bilda and Demirkan (2003). The reason there are few pairs in this research is that it was difficult to find chemistry teachers who would fit the profile (having at least a master’s degree in science education). We chose not to pair the designers with the same educator (TA or TB) because pairing the designers with the same educator would not create an independent variable, as the educator’s ideas about the interface could change during the design process. On the other hand, if we had paired the designers with the same educator it would have been easier to form more pairs — as there are more designers that fit our research profile (experienced and non-experienced) than educators — but then we would not have seen how they collaboratively design the interface. This reflects our position relative to the designer’s allocation on the team, in that we feel he/she should be a part of the process from the beginning.

Each pair was asked to design two interfaces about chemistry, the first on states of matter and the second on solubility in water. These topics were chosen because (1) both are usually taught with the support of several types of representations, for example, molecular models, macroscopic visualisations, symbols, tables, graphics, etc., and (2) they have different levels of complexity (it was reasonable to speculate that designers could have some understanding of the concept of states of matter, but they would not have this basic understanding of solubility).

The pairs were informed of the topics only at the beginning of the sessions, which had an average duration of 2 h per session. These sessions were conducted 24 h apart from each other. The pairs and the experimenter were alone in a studio specially set up for the experiment. Each member was rewarded at the end of the second session. They were asked to think-out-loud of whatever came to their minds and to ask questions related to the briefing to the experimenter, who played the role of the client.

The briefing stated that the audience was secondary school students and that the software could be used either in distance or presence environments. The teams were asked to provide drawings that could be used as a starting point for the development. A4 paper and pencils of different types and colors were available, but books or computers could not be used, to prevent searches for images, texts or other types of data (as this would take the focus off the interface design).

The use of the think-aloud method with more than one subject, although not the rule in design research, is not that uncommon. One can find analyses of team work tracing back to the 1980s (Cross & Cross, 1995; Valkenburg & Dorst, 1988). Austin, Steele, Macmillan, Kirby, and Spence (2001) designed a fairly similar setting to investigate multidisciplinary teams of 5 designers, whose assignment was related to architecture. Other examples of analysis of software design teams, describing settings that are similar to the one described in the present paper, are reported by Christiaans and Almendra (2010), Petre, van der Hoek, and Baker (2010), and Tang, Aleti, Burge, and van Vliet (2010).

6. The method: verbal analysis

The Verbal Analysis method (Chi, 1997) was chosen as a methodological framework because it focuses on the interpretation of representations such as verbalisations, drawings and gestures. There are other approaches to design research, but as this debate is outside the scope of this article, we refer the reader to Craig (2001). The non-optional method steps, as found in Chi, are: (1) to segment the protocols (video records of the design session); (2) to develop or choose a coding scheme or formalism; (3) to define how the coded protocols map the chosen formalism (rules of coding); (4) to seek pattern(s) in the mapped formalism and (5) to interpret the pattern(s). We consider the first and second topics as pertaining to this section. The others we consider as results, and will be presented in the 6th section.

6.1. Segmenting the protocols

The concept of design move, ‘an act of reasoning that presents a coherent proposition pertaining to an entity that is being designed’ Goldschmidt (1992, p. 125), was the rule for segmenting the protocol. Although its definition is clear, it is up to the researcher to determine when a “move” starts and ends. There are other criteria, like sentence identification, but the concept of “moves” was considered more relevant. When segmenting the protocol, the sketches were connected to the verbalisations. The thickness of the pen and the position and resolution of the camera allowed us to see the sketches clearly on video.

6.2. Developing or choosing a coding scheme or formalism

The coding scheme is a vital part of the method, as it will provide the data for analysis. Normally these coding schemes are developed for a single design experiment, for example, Gero and McNeill (1998) and Suwa and Tversky (1997), but there are examples that could be applied to several design protocols, for example, Gero’s FBS model (Gero, 1990) and Akin’s processing information model for design (Akin, 1986).

In our case, we coded the protocol – segmented it in several design moves – using a classification scheme proposed by Goel (1995), which considers two phases of a design session:

- Problem structuring (S), ‘the process of drawing on knowledge from various sources to compensate for missing information in the problem statement, and using this knowledge to construct the problem space’ (Goel, 1995, p. 114).
- Problem solving, which is divided into three phases:
  - Preliminary design (P), whose statements are related to ‘initial generation and exploration of ideas’ (Goel, 1995, p. 118);
  - Refinement (R), when ideas are further elaborated and
  - Detailing (D), when the final form is specified.

Visser (2009) states that problem-structuring and problem-solving stages can be distinguished as distinct activities only in theory. However, Goel (1995) provides details and examples on how to differentiate these four phases. We applied this set of rules to classify the segmented protocol.
The coding was carried out three times, by the same researcher, at intervals of two weeks. Although the method of having a single person coding the protocol may not be considered as the best alternative from a statistical viewpoint (as the errors associated with the measurer are embedded in the measure), this procedure has been introduced because it is fundamental that the coder possesses a deep familiarity with the protocols. This familiarity is needed because the rules in our research are relational: a move is classified (using Goel’s scheme) according to its comparison with others. It reflects our understanding of the design process: it is not stochastic. The acquisition of this familiarity was the longest step in this research. We also refer the reader to the works of Gero and McNeill (1998) and Baker and van der Hoek (2010), who also used one coder.

7. Results

Following Chi’s methodological proposal (Chi, 1997), before presenting quantitative data, significant episodes that happened during the design session are presented and discussed. This presentation has the aim to provide an overview of the main observations we made on the sessions.

7.1. The overall picture: description of significant episodes relating to the design strategies

Designers ED and ND showed different approaches to the assignment. Although we are not yet ready to identify the causes of these differences, we suppose they might be attributed to designer ED’s longer experience and professional background in graphical and digital design. However, there could be other related effects, such as the teacher’s role in the team and chemistry knowledge.

7.1.1. The use of the same design solution on all design sessions

There was an observed similarity in the main structure of the designed interfaces chosen by both designers: a web-page-like solution. Designer ED adopted this solution in the early minutes of the first session (Fig. 1a) and started the second session by suggesting to use the same structure (Fig. 1b), while designer ND committed this idea to paper after 80 min in the first design session (Fig. 2a) and after 60 min for the second session (Fig. 2b).

The tree structure offers the benefit, exploited by both designers, of easy integration of features in the interface. This initial approach to the design task was expected because we knew in advance that both designers had no experience with educational assignments and no familiarity with learning theories. Given the complexity of the topics and didactical strategy, it is plausible that both designers turned to a familiar solution (the “tree”). Visser (2009) stated that reuse of knowledge (from previous design projects) has been observed in many cognitive design studies as a central approach in design, but stressed that this reutilization is subjected to a number of factors, including the familiarity with the domain. In this case, both designers reused the structure of a hyperlinked document – although the briefing did not mention this. According to Visser (2009), designers tend to generate, at the very start of a project, a few simple objectives in order to establish an initial solution kernel which will become their global design solution. This fixation with the initial solution might persist, even if it proves inadequate later in the process (Newstetter & McCracken, 2001).

7.1.2. Effects of the interaction of the ED + TA pair

Besides this fixation effect – which perhaps was amplified by both designers’ unfamiliarity with the domain – the differences in the roles played by the teachers also might have had an influence. For example, teacher TA and designer ED shared the design task in all contexts:

![Fig. 1. Sketches representing the “tree structure”, designed by designer ED. The structure was reused across the sections.](image-url)
interface design, instructional design and chemistry. Although some of the design decisions were unilateral (as the tree-like-structure), as were some chemistry decisions (not to talk about “physical versus chemical changes”), most of them were shared and consensual. However, the lack of knowledge of learning theories on the part of designer ED was a major source of debate. For instance, teacher TA suggested a conceptual map (Fig. 3a), which designer ED agreed to display as an interactive conceptual map near the main menu (Fig. 3b), but designer ED was afraid that students could use it to reach the end of the unit without “going through everything”, i.e., in a non-linear way. In the end of this episode, the conceptual map turned into a feature known as a “breadcrumb”: a sequence of words that shows the navigation history (Fig. 3c). For a “cognitivist” educator (like teacher TA and professor TB), navigating through the content in a non-linear way is not a problem, as the map would favor the elaboration of a super-organised structure, where all sub-topics would be connected in a network.

There were other similar cases with the pair ED + TA: designer ED did not accept to allow easy access to a forum (“it could be kind of hidden”, designer ED said) because it would break the linearity of the presentation. Again, for a “cognitivist” education, a forum is a communicational aid, and it is important to allow access to it. The only occasion in which teacher TA was inflexible was when designer ED suggested that a given exercise should be corrected by the software.

7.1.3. Misunderstanding because of designer’s ED lack of chemistry knowledge

There were also some episodes when the lack of knowledge on chemistry was a source of minor misunderstandings, but they were not obstacles to the design process in either session. Some examples: differences among chemical and physical transformations; the fact that state changes do not occur instantly when a given temperature is reached; the importance of microscopic representations and the difficulty of finding example substances that are presented in the three states. These misunderstandings were overcome not because designer ED heard lengthy explanations about these topics, but precisely because he/she did not try to understand chemistry in depth. Designer ED found a way to manage how much it was needed to know about each topic, and based on this evaluation, decided to stop or proceed with
questioning the teacher. In the case of microscopic representations, a very important topic, as noted before, they had a long conversation about how it can influence students’ interpretation of chemical (and physical) phenomena. Our [subjective] evaluation is that the pair’s (ED + TA) interaction in both sessions was smooth and satisfactory.

7.1.4. Effects of the interaction within the ND + TB pair

This role-sharing characteristic was not observed in the pair formed by designer ND and professor TB. We speculate that this is related to designer ND, who in addition to having no experience with educational interface design or educational theories, had little design experience at that time. Professor TB, on the other hand, had more than ten years of experience, and managed the development of five educational software systems. Whether or not it is a function of this difference of experience, we observed that designer ND left all education and chemistry related decisions to professor TB: he/she spent most of the time listening, occasionally asking questions about chemistry. In both sessions, he/she tried to get a big picture of the problem first, and only then decide on how to approach the design. The downside of this strategy is that the problem is too big and complex. The consequence was that designer ND did not present the sketches with the level of detail required by the briefing document (failed to manage the session). Nonetheless, the solution he/she started to draft (at the middle of the sessions) has a similar structure to that one designer ED proposed: a web page with several gadgets, which were integrated as professor TB mentioned them (for example, a video player, interactive tables and an interactive conceptual map).

7.1.5. Professor’s TB design suggestions

If designer ND tried not to interfere with questions he/she considered as non-design, professor TB, in his turn, gave suggestions regarding how the software could be structured, i.e. “design” suggestions. Some were rather abstract, and others were very concrete. For example: when talking about solubility (second session), professor TB came up with the wordplay “solution: to solve a problem; to make it disappear”. When talking about states of matter, professor TB suggested using the concepts of “organised and disorganised states” (as, according to him, there are far more than five states) because what determines the physical state of a given substance is its molecular organisation. Professor TB also commented about an interview conducted with colleagues asking “which are the core topics in chemistry?” from which he/she drew a conceptual map depicting not only the network of topics but also the strength of the relationships using the length of the connection lines (Fig. 4).

Another suggestion professor TB gave was to use a classification scheme where all chemical phenomena would be described by the coordination of three axes: semantical, representational and phenomenological (Fig. 5). However, designer ND did not use them as macro structures. It should also be noted that the drawings shown in Figs. 4 and 5 were among the few drawings professor TB did during both design sessions.

We concluded that neither designer used the teachers’ suggestions as integrative approaches to interface design, although they adopted almost every suggestion both teachers made. Their lack of knowledge and experience with learning theories (including Science Education) and educational interfaces could be one of the factors behind this, as pointed out by Hinostroza and Mellar (2001). But we would like to stress that designer ED had a very proactive posture regarding instructional design and science education, and that his suggestions enriched teacher TA’s ideas.

At this point, the reader has an overall picture of the strategies that the designers used and the way they interacted with teachers. The next step is to present the quantitative data that support this report.

7.2. Analysis of design phases: two strategies

This second part of the data analysis shows how the sessions proceeded. The data is shown in the same format used by Goel (1995) to allow us to contrast our findings with his. The percentage of moves associated with each of the four phases (see Section 6.2) is depicted in Figs. 6–9. These figures show the distribution of design phases for the sessions SM – States of Matter and SW – Solubility in Water, for subjects ED (Figs. 6 and 7) and ND (Figs. 8 and 9).

Fig. 6 shows that, for designer ED, these four phases do not follow a steady sequence, such as structuring the problem first, then doing a preliminary design, then refining and detailing the solution. We emphasize that in the SM session, designer ED went through two distinct cycles, one from the start to the 80th minute, the other from the 80th minute to the end. In the first cycle, designer ED spent the time

---

**Fig. 4.** Conceptual map, drawn by professor TB, depicting the strength between chemistry concepts.
structuring the problem, doing preliminary design and refining, while doing almost no detailing. In the second cycle, after the 80th minute of the SM session, designer ED did almost no problem structuring or preliminary design. This coincides with the start of a revision process, in which all interfaces were drew again, with greater detail.

In designer ED’s second session (SW) there was no such revision cycle, and the time was spent in a more uniform way, with designer ED going through all design phases in 5 min spans – as shown in Fig. 7. Throughout session SW, designer ED spent more time doing preliminary design. Through his/hers words, it was clear that designer ED was refining and detailing each idea from beginning to the end, as in a depth-first strategy. This is regarded as evidence that designer ED treated the task as a matter of integrating every piece of information into the tree structure as quickly as possible. Adelson and Soloway (1985) also suggest that experts working in unfamiliar domains will approach the task in a depth-first manner. We called this strategy “integrate as fast as you can”.

We identified that “integrate as fast as you can” is a form of the depth-first control strategy because designer ED went through all four design phases in short spans of time. If designer ED had been following a breadth-first strategy, the graph would show predominance of each of the four phases during the design session, starting with problem structure, preliminary design, refinement and detail design.

The distribution of design phases for designer ND shows a different picture. In both sessions (Figs. 8 and 9), there is no detailed design and very few instances of design refinement, which corroborate our observation that the task was not finished.

**Fig. 5.** A classification scheme for chemistry topics.

**Fig. 6.** Distribution of design phases for designer ED, at session SM.
The amount of time devoted to problem structure in Fig. 8 and preliminary design in Fig. 9 reinforces the perception that designer ND tried to first acquire some familiarity with the domain and start the problem-solving phases later – precisely what designer ED did not do. The trouble with this strategy is that the problem space is too large and complex. We called this strategy “structure then design”. Designer ND did not approach the task in a breadth-first manner either, because there was not an attempt to subdivide the problem space. Designer’s ND approach resembles a depth-first approach – the sub-goals were developed in depth. As Figs. 8 and 9 show a very short amount of time dedicated to refinement and detail design, we conclude that designer ND needed more time and information to complete the assignment.

The fact that both designers reused the structure in the second session may be a trace of learning; they could be developing a way to approach the task of designing in unfamiliar domains. Perhaps designers ED and ND were developing what Cross (1990) called “first principles”, which could reduce the time and effort for developing solutions and could enable them to approach idea generation through recognition (Liikkanen & Perttula, 2009).

To close this article, we present our conclusions, where we will attempt to connect our results with those of Goel (1995) and others.
8. Conclusions

The results from observations and quantitative results support the conclusion that designers ED and ND followed two different strategies to design educational software interfaces: “integrate as fast as you can” (designer ED) and “structure then design” (designer ND). Although designer ED is an expert in his field, he/she did not proceed in a breadth-first manner, as both strategies are similar to depth-first approaches. Designer ED’s strategy could not be labeled “opportunistic” either, as it would imply mixing breadth-first and depth-first strategies. However, designer ED achieved a better performance than designer ND: he/she finished the task and actively generated ideas with teacher TA, while designer ND did not finish the task and did not engage in idea generation with professor TB.

- Designer ED is more experienced than designer ND. Designer ED also has professional experience with graphical and digital design, two areas that are closely related to interface design.
- Teacher TA and designer ED reached consensus in all decisions. If there were any disagreements, misunderstandings or unilateral decisions, they were of minor importance.
- Professor TB and designer ND, on the other hand, did not share decisions: designer ND accepted to be “directed” by professor TB. This does not mean that professor TB did not make suggestions related to design.

Even using different methods, both designers approached the task in a depth-first way, which might be a consequence of their unfamiliarity with the domain, as Adelson and Soloway (1985) had suggested. We also reproduced Visser (2009) and Newstetter and McCracken’s (2001) findings, relative to fixation on the first idea – as both designers reused the “tree” solution on the second design session.
These are the conclusions of this study in isolation, but it is also possible to compare these results with those of Goel (1995), which was our frame of reference after all. Goel asked an engineer, an architect and an instructional designer to accomplish different design tasks (according to each designer’s expertise). The instructional designer had to design “a self-containing instructional package to teach laypeople a reasonably complicated computational environment” (Goel, 1995, p. 96). Fig. 10 shows the results for the instructional designer.

The distribution on Fig. 10 is very different from designer ED’s and designer ND’s distributions (Figs. 6–9). Subject S-I did not overlap all the phases in short spans as designers ED and ND did – he/she did almost exclusively one thing at a time, which is consistent with a breadth-first approach. There is one interval when subject S-I went back and forth between three phases, from 40th to 55th minute. At that time, subject S-I could have been approaching the task in a depth-first manner. We emphasise that Goel’s experiment and ours were different: subject S-I was a professional instructional designer, he/she worked alone, and (we assume) he/she knew the problem in all relevant contexts – he/she was familiar with the domain.

Our conclusion is that both designers’ unfamiliarity with the domain was a major reason for adopting variations on depth-first approaches to the task. We also believe that this was the reason our results and Goel’s are different. The topics of the briefings – states of matter and solubility in water – were chosen to achieve this effect, because, when designing educational software, not being familiar with the domain will be the rule, not the exception.

Acknowledgment

The authors wish to express their gratitude to the Brazilian Research Agency CAPES for its financial support of the research group and to the National Computation Centre CESUP/UFRGS for the use of their recording studio and laboratory.

References

Arija, & Watanabe. (2008). Teaching materials to enhance the visual expression of web pages for students not in art or design majors.