The use of a computational tool to support the refinement of ideas

M.B. TWIDALE, T. RODDEN, I. SOMMERVILLE
Computing Department, Lancaster University, Lancaster LA1 4YR.
Email: mbt@uk.ac.lancs.comp

Introduction

It is well established that the use of concept mapping has considerable educational benefits [1]. A computational tool to support concept mapping can enhance these benefits by providing features that make the mapping activity easier for students. In particular, computer-based concept mapping can more effectively support the refinement of ideas that leads to the revision of the structure of the map. Given that mapping leads to learning it is inevitable that student's concepts will evolve and so the representation of the concepts should be easily alterable. This can be difficult and messy to do with paper-based maps, but can be made into a trivial activity for paper based maps.

We describe a computational tool which can support not only concept mapping but brainstorming and the refinement and revision of ideas. Although initially developed to support software design, we believe that its core features are equally applicable as a learning environment for students of any domain. A preliminary study of the use of the tool by second year computing undergraduates and students of other disciplines has been undertaken as part of its incremental development. We report the initial findings of this study and their implications for future development of this and related learning-support tools.

Background

The Designers' Notepad (DNP) [2] is a tool developed to support the very early stages of software design. It is part of a joint project between the departments of Computing and Sociology at Lancaster University. The project is part of a larger investigation into the contribution that sociological techniques of analysis such as ethnography can make to the systems development process [3]. In order to accommodate the stream of information that ethnography provides, a rapid prototyping approach to the development of DNP was used.

There are certain features of the activities undertaken during initial software design that are similar to the learning activities undertaken in many domains. It is frequently a cooperative activity which involve brainstorming, planning, organising, the refinement of ideas and the consideration and selection between alternative options. These features are equally present in many other activities including project management, planning, preparing talks, papers or presentations and organising learning. Therefore we decided to investigate whether the DNP could be used in other domains and to support learning. Would the tool be usable and useful, and what additional facilities should it provide for this new application?

Essential features of the DNP

Figure 1 shows the DNP interface. In order to make the learning of the system easy, there are a small set of core features which are sufficient to enable productive activity to take place in a short time. The basic unit is a design: rather a misnomer here, essentially it is a window into which idea elements can be placed. The design in the figure was used in preparing a presentation of the system.

The user creates an entity by typing in a design window. She may then move it with the mouse. Linking is done by selecting one entity and then clicking on the entity one wants to link to with the shift key down. We also provided Textnotes which are note pads based on the Post-It Note metaphor (see figure 1) and allow users to attach one or more notes to an entity. These can be used for more textual comments, ideas, opinions, paragraphs of a final document, references etc. A variety of Textnote types are provided and users may define their own (including form-like structures). An entity with Textnotes has an icon attached (eg. the entity 'Annotation of learning' in figure 1) and the notes can be examined by clicking on the icon. Designs can be saved and loaded from a file and a paper report may be created containing a screendump of the design and a list of the entities and their Textnotes. Each entity may itself be expanded to become a subdesign. A new window is opened and entities and links can be created in the normal way (see figure 3). Subdesigns may contain entities that are themselves subdesigns. A loose form of typing for entities and links is provided using colour, shape and labels. The user controls the degree to which she wants to use typing. The type of an entity or link can easily be changed at any time. Additional features enable the design to be annotated. These include a framing facility for grouping related collections of entities. Groups can then be moved en masse to assist rearrangement.

The Development Process

The bulk of the development work has gone into refining the interface to make it easy both to learn and to use, particularly the core features. As well as being easy to create, the elements are also easy to revise, to move around designs and across subdesigns. The aim was to enable the user to focus quickly on the domain issues, to be able to quickly enter many ideas as these occur and to not to have to be concerned initially with the form or type that the idea should have. This can only occur if she trusts that it will be easy to edit structures with any refinements that occur to her later. The use of subdesigns enables complex structures to be handled by data hiding: using the framing facility, a group of interlinked entities can be 'pushed down' creating a sub-design, to be replaced in the original design by a single, summarising entity. We believe it to be essential to provide an easy to use interface if the tool is ever to be used effectively. Any benefits to the student that potentially accrue from using a learning environment can be swamped by the effect of a poor interface [4]. Indeed if the tool is hard to use, no one will even bother to use it. Therefore the bulk of development resources were concentrated on the interface rather than on the provision of sophisticated additional features.

Achieving an easy to use interface is not easy. It difficult to predict what users will find easy to learn and use and also to determine the features they will require. Furthermore, user requirements co-evolve with the system: as one feature becomes better developed and easier to use, its increased used engenders a whole new set of requirements. For example, as we refined the Textnote features to make them easier to use and offering more useful functionality, some users began making far greater use of them. As a result they had some entities with tens of Textnotes attached and so now wanted better ways of searching for particular notes and transferring notes between entities. This co-evolution makes the determination of requirements inherently unknowable.

Our solution is to use rapid prototyping and continual testing of the evolving system by volunteers working on authentic tasks. That is, tasks that they would have to perform anyway that they agree to work on while using the DNP. Authentic tasks have features that are very hard to replicate in contrived tasks. These features include ambiguity, open-endedness, history and engagement of the user, who ideally will be focussing more on the problem than the tool. Our instructions to the users were to try and use the tool, asking where necessary how to achieve anything they want to do. Our assumption is that the tool will be usable up to some point when it becomes frustrating because it prevents the user doing something they want to do. We can identify that need and then can assess what sort of functionality should be added to the system. The system has been developed in Smalltalk which makes the continual revision of the interface particularly easy.

DNP as a Concept Mapping Tool

We believe that the DNP can offer a supportive learning environment for a wide variety of subject domains. The ease of entry of concepts means that the advantages claimed for conventional paper-based concept mapping are at least equally true for a computational version. However the latter offers additional advantages:

- As a structure builds up, the map can be rearranged, clarifying the emerging structure. Space can be created in particularly cluttered areas.
- Alternative structures can be experimented with quickly and easily.
- Colour and shape may be used as for pen and paper, but can subsequently easily be changed.
- Copies can be made of parts of structures to save re-entry of recurrent themes.
- The subdesign hierarchy provides a means of complexity management. This is analogous to using multiple sheets of paper but offers the advantages of moving structures rather than having to rewrite them.
- Versions can be created to try out alternative structuring.

Learning Activities DNP can support

- Concept mapping in general has already been mentioned.
- It can operate as an advance organiser [5] where the student notes down what she already knows about the domain before undertaking further study.
- It can support the brainstorming of ideas, their organisation, assessment and refinement.
- It can support the restructuring of knowledge by means of rearrangement, renaming,

elaboration and annotation.

- It can support the planning of a course of study. This could be a research project (eg term paper or PhD) or an experiment.
- It can support the linearisation of complex interlinked idea structures for such outputs as essays, reports, papers, presentations or summaries.
- It can support the articulation of plans, goals and beliefs and subsequent reflection on these by the student. This externalisation can be a powerful metacognitive activity but is frequently too inconvenient for learners to perform while problem-solving.
- It can provide a means of annotating the learning process. This is a means of explicit metacognition, where the student notes her reflections on the learning process. It can cover a range of activities including annotating concepts by how much the student is confident that she has understood the concept. Other possibilities are the creation of 'to do' lists of subsequent activities.
- It can support collaborative learning by providing an evolving record of a debate. The elements under discussion can be made concrete and externalised making their discussion more concrete.
- It can provide a rapid way of diagramming information flows, or cause and effect. Thus it can be used in developing ideas for computer programs, economic models, qualitative models in physics etc.

Relation to other work

Collins and Brown [6] have reported on the use of NoteCards by a graduate student to organise the study for a term paper. This has many similarities with the potential of the DNP. However Monty [7] and Marshall & Rogers [8] have described difficulties that users have with the NoteCards interface. Many of these difficulties stem from the requirement of the user to specify the type of the elements and links to be used. This can lead to premature commitment, forcing the user to consider elements in too great detail during the initial stages. Indeed this is also a problem with the paper-based form of concept mapping where the user must think of a suitable label for a link and perhaps a pen colour at the moment of entry. By contrast the typing in DNP is far looser and is imposed by the user by means of colour, shape and textual labelling. The types are easy to subsequently alter. The benefits that accrue from this are that the user can very rapidly enter ideas that still are not completely thought through, contain considerable ambiguity and whose relationship to other ideas is not fully worked out. The disadvantage is that the absence of strong typing precludes automatic checking for type mismatches. We believe this choice to be preferable for the early stages of idea formation and refinement. An earlier version of the DNP [9] included rules for strong typing and checking, providing an existence proof of the possibility of this approach. Continuing development of the DNP for supporting software development is investigating a way of providing a smooth transition from the open ended structure of the present system to the more rigorous structure of conventional CASE tools.

There are many similarities between the DNP and tools to support writing both by individuals and groups such Sharples' Writer's Assistant [10]. The DNP can particularly well support the early stages of considering the elements to potentially write about, elaborating and developing the relationships between the elements and then supporting linearisation. The latter is a difficult process, particularly for inexperienced writers. We find that many of our Computing undergraduates have particular difficulty in organising the structure of reports and essays. Part of the reason for the difficulty is that the structure of the ideas to be reported can be strongly interlinked so that it is difficult to decide on a sensible and logical ordering of the elements.

Conventional word-processor outliners can support a top-down approach, but rearranging ordering and hierarchies can be clumsy. With the DNP it is possible to rapidly investigate multiple alternative orderings by moving a group of section or paragraph headings around on the screen, creating new sections and deleting them as needed. It is also possible to simultaneously support a bottom-up approach. Phrases, sentences or paragraphs that occur to the student as useful to include in the report (although the user may not quite be sure where yet) may be incorporated in the planning stage by attaching them as Textnotes to the heading entities. An ASCII version of the textual elements of the design can be created which can be used as the core of the ultimate report in the word processor of choice. Naturally where the report benefits from the provision of structural diagrams, the DNP offers the additional advantage of making the construction of these diagrams easy. Indeed the diagrams constructed by the student for their own benefit while exploring the domain may be directly included in the report.

The DNP does not endeavour to 'understand' the student's actions. Rather, by avoiding the imposition of semantic structures (which would enable the system to understand the student's meaning), we provide a very open-ended interface that can be used by different users in widely varying ways. We regard this tool as complementary to our other work on structured, form-based interfaces for Intelligent Learning Environments [11].

A key difference between the DNP and other design tools (and indeed other concept mapping tools) is that most other systems have an implicit process model embedded in them. If the user's preferred model for the problem at hand matches this then all is well. However, the process model may not only vary between users but an individual user may use different models and variations depending on the problem faced. In cases where there is a mismatch of models, the tool will not be effective, it will feel awkward to use and is likely to be abandoned in favour of traditional, more flexible methods. By contrast, while developing the DNP we have endeavoured to avoid any such implicit model (or at least to minimise its effect) by maximising the flexibility of use of the tool. As a result it can be used in a variety of learning styles.

Purpose of the educational study of DNP in use

Our aim was to investigate whether the potential of DNP as a learning environment could be realised. The intention was *not* to evaluate the current DNP but rather to investigate the requirements of learners; to determine whether the interface needed to be modified, what additional features should be added to the system and to improve our understanding of the nature of activity that occurs when using the tool in an educational context.

Preliminary studies were done with a number of volunteers; various computing graduate students, and undergraduates majoring in Biology, Religious Studies and French. Students were introduced to the minimal system and appeared to have little difficulty in learning and using the system. Lack of experience with a computer appeared to be no problem, however students with poor typing skills made slow progress. Figure 2 illustrates a map constructed by a Religious Studies major. As for paper-based concept mapping, the tool does not seem to be domain-specific. The students reported that they found the tool most useful for organising ideas, particularly when they had many thoughts about the subject, such as after having read a number of books on the subject and made various notes. It was found less useful when the reading still had to be done. This may be an artifact of the study: the students felt they should be 'performing' using the system and spending large periods of time reading a book felt unproductive.

Following the preliminary study, a more detailed study has been undertaken with Computing undergraduates.

Nature of the task

As part of their assessed coursework, second year computing students are required to undertake a design project in groups of four. They were asked to develop an electronic system to support a computing department's management of lectures and tutorials. The students were to develop a list of requirements, design a solution and write it up as a report. The project is intended to teach design 'method'. In previous years it was noted that students had difficulty in grasping the creative aspects of design and little experience in working cooperatively. The project is deliberately open-ended, under-specified, intractable and contentious. We believe these features (not normally found in conventional computing exercises) mean the results are not domain specific.

Groups who volunteered to use DNP were introduced to the core features. It was emphasised that the aim of the study was to test the usefulness of the system and its interface as well as to determine additional requirements. Sessions were videotaped. They lasted as long as the participants wanted, and were generally of one hour. Printed reports of the designs were produced at the end of a session and the group was invited to return for a subsequent session if they felt this would help them with their assignment.

Results

4 groups of 4 students used the DNP. 3 of the groups chose to return, producing a usage of 1, 2, 4 and 5 sessions. Given the aims of the study and the small number of students, the results cannot be definitive. However the reaction to the tool was extremely positive. Students were able to quickly learn how to use the core features and then focus on the task (the introduction to the features took about 10 minutes). They were able to enter ideas rapidly and then rearrange and refine them. Figures 3 to 5 illustrate the evolving design of one group. There were certain features of activity using the tool that concurred with our observations of use by staff and research students in the Computing Department [2]. These include:

The Evolution of structure

The ideas entered as entities are necessarily ambiguous; they consist of a label of a few words referring to a concept. However, not only is that label capable of misinterpretation by anyone other than its creator, but even for the creator it seems that deliberately ambiguous terms are chosen (for example, entities were often called "object" or "user"). This is an instance of postponing decisions about details in order to deal with overview concepts. Later on in the design process, the meaning of the initial concepts will be refined. This can involve qualifying the entity by editing its name (including completely changing the name), creating more entities for the constituent concepts in an ambiguous description, adding more information by attaching Textnotes, creating a subdesign for the entity, or completely replacing the entity. Similarly, links have labels added quite late in a session, and often require some discussion to choose the right term.

It would seem that designers need to have a certain degree of ambiguity during design so as not to not overwhelmed by details when working on the broad features of the problem. Gradually these ambiguities are addressed and refined. This has been observed in other situations [8] and reasons proposed for why users wish to avoid formalisation [12]. In a similar manner, the meaning of a link can change over time. Initially its meaning may be 'these entities are in some way connected'. Eventually this is refined into a more precise meaning. The gradual evolution of precision is often associated with the usage of link typing; links with a similar meaning are now given the same colour or label.

Variability of use

Our studies have confirmed the great variability in activities both between users and by the same users over time and circumstance. For example, some people use very terse entity names with Textnotes to contain details, whereas others use phrase- or sentence-like names. Some use many links to indicate connectedness whilest others use two-dimensional proximity [8] (just having two entities close together implies they have some sort of connection). Some use a great deal of colour and many different shapes, whilst others use black rectangles all the time. Some have labels on most links (as is encouraged for concept mapping) while others rarely use labels. As examples of variability over time, we have observed cycles involving bouts of entity creation and rough positioning. These involve very rapid and intense activity where the minimum of options are employed. After such a bout there is a recovery period where the display is 'tidied up'. Links are created and the entities rearranged to convey additional meanings by their proximity to other entities as well as to reduce the clutter of areas of great activity.

This variability confirms the need for flexibility and ease of revision in the DNP. Much of this flexibility comes from our decision to avoid associating semantics with entities, links and subdesigns and to allow these to be used primarily as a means of expression by users. The meaning of these initial design graphs are left to the interpretation of different users. The intense 'bursts' of activity surrounding the generation of entities caused us to focus on supporting the rapid entry of entities (just typing and hitting return between entities generates entries positioned in a list format). Once entered these entities can be tidied by altering their position, colour and shape. For example, contrast the design in figure 3 showing the early stages of the design with a later stage shown in figure 5.

Integration with other documents

When using the DNP, group members would sometimes consult paper documents. These included lecture notes, textbooks, results of their requirements survey and initial diagrams produced from earlier meetings as well as printed reports from DNP produced in earlier sessions. These documents were principally used as resources, hardly anything was written on them while using the DNP. Tools to support the organisation of ideas need to allow for their being used alongside other media. In particular they need to allow for the transfer of information that can occur in many different formats.

Problems observed

The aim of the study was to discover the problems that arose from using the DNP for educational purposes in order to improve the system for subsequent testing. However rather gratifyingly there were very few difficulties or major requirements that emerged. A few minor interface difficulties have been resolved. The most apparent were cases where a user accidentally chose the wrong menu selection and then may be confused by the result or need an easy way to recover from it.

One group, however did appear to have difficulties with the DNP. This is the one group that did not request to use the tool after the initial session. In that session it transpired that they had not achieved a consensus on what they were meant to be doing. Hence there was considerable debate before all computer based activity. They attempted to use the tool to create a particular type of detailed software engineering diagram. This seemed to be based on a misapprehension of the purpose of the tool; they considered it more of a drawing tool than a brainstorming or ideas organising tool. The process of perfecting the diagram meant that very little was produced and the debate that accompanied this activity also revealed that they had not fully worked out the implications of the structure they were trying to draw. In essence they were seduced by the drawing features of the DNP to progress onto the later stages of the design process that involve structured diagrams before they had thought through their initial design sufficiently. By contrast, the groups who chose to continue using the DNP would ignore the initial appearance of the designs they produced until they had agreed on the content. They had an attitude to the designs of postponing refinement and treating them as a tool for their benefit rather than as products in their own right.

Thus we conclude that there is a danger when using the tool of form overriding content: that students will spend excessive time on attempting to achieve a precise diagrammatic format at the expense of considering the meaning of the entities they are manipulating. This danger is analogous to the problems of novice writers using word processors who put excessive focus on the presentation options at the expense of the content of the document. Hopefully, these difficulties can be tackled during the presentation of the system to subsequent users and by the provision of suitable documentation in its use.

User reaction

Users had an extremely favourable impression of the tool. The fact that three of the four groups chose to make arrangements to continue using the tool for their task is further evidence that they considered it useful (the experimenter had no connection with the coursework). However it would require a far more rigorous and controlled experiment to determine the degree of improvement that using the tool led to. The students were asked how they felt using the tool compared to conventional group working. Their experience of this consisted of sitting round a table taking notes on pieces of paper. They considered that DNP made note-taking easier, particularly when ideas were revised. Other comments included the ease of keeping a sense of perspective compared to paper-based approaches which lead to unproductive debate about side issues.

Observations of collaborative working

The evolving design on the screen is the main focus of discussion. All group members point at the screen to illustrate their ideas, leaning over other participants as necessary. With four people sitting round a screen, inevitably some are closer than others. At first glance it can seem that there is very unequal access to the system, with the fourth person often sitting some distance from the screen behind the other users. Unless encouraged to take turns, a single user often operates the tool for long periods (even an entire session). However on closer examination, they may not be dominating the conversation. Although it is certainly true that at some times the person with the keyboard directs activity in a similar manner to the way the person with the sole marker pen directs activity round a whiteboard, at other times the operator may take a much more secretarial role. This can involve minuting the discussion, doing what the other members direct, or tidying up the design while the others argue more substantive points. The ability to lean over and point at the screen enables group members positioned some way from the keyboard to interrupt and direct debate and activity. Those further away from the screen seem to keep more of an overview of the problem, spotting features that still need to be considered or refined. We suspect however that four is the maximum number who can productively use a single terminal. The opinions of the users as reported in the questionnaire support these observations of the advantages of the tool in supporting collaboration.

Conclusions and Future Work

We have discussed the potential that a computer based ideas organiser has as a supportive learning environment. Our preliminary studies support this view. However, for any tool to be effective it must be sufficiently easy to learn and use that students can focus their attention on the subject domain rather than the tool. Therefore considerable attention and resources should be devoted to the interface. Given the great difficulty of determining what it is that makes a system easy to use and also what features are needed by such a system, we believe that iterative development with particular attention to the interface is the best approach to developing a usable system.

The next stage is to continue in-depth studies of the use of the system, but across a number of subject domains and the various learning activities outlined here. We also wish to do a comparison of the use of the tool with conventional approaches to organising learning, both to determine the advantages of the existing tool, and its limitations so that we can work on features to minimise the latter.

Acknowledgements

Michael Twidale is a Science and Engineering Research Council Junior Research Fellow. The development of the Designers' Notepad was funded by the Joint Council Initiative in Cognitive Science and Human Computer Interaction.

References

1. Novak, J.D., Concept maps and Vee diagrams: two metacognitive tools to facilitate meaningful learning. *Instructional Science* **19**,19-52 (1990).
2. Twidale, M. B., Rodden, T. & Sommerville, I., The Designers' Notepad: Supporting and understanding cooperative design. Paper accepted for *ECSCW93*, Milan (September 1993).
3. Sommerville, I., Rodden, T., Sawyer, P., Bentley, R. & Twidale, M. B, Integrating ethnography into the requirements engineering process. Proceedings *1st International Conference on Requirements Engineering,* San Diego, IEEE Press (January 1993).
4. Twidale, M. B., Redressing the balance: the advantages of informal evaluation techniques for Intelligent Learning Environments. *J. AI In Educ.* (In press 1993).
5. Ausubel, D. P., *Educational psychology: a cognitive view*, Holt, Rinehart and Winston, New York (1968).
6. Collins, A. & Brown, J.S., The Computer as a Tool for Learning Through Reflection. In Mandl, H. & Lesgold, A. (Eds.) *Learning Issues for Intelligent Tutoring Systems,* Springer-Verlag, New York. 1-18 (1988).
7. Monty, M.L., *Issues for supporting notetaking and note using in the computer environment*, Dissertation, University of California, San Diego (1990).
8. Marshall, C. C., & Rogers, R. A., Two years before the mist: experiences with Aquanet. Proceedings *ECHT '92*, Milan (December 1992).
9. Sommerville, I., Haddley, N., Mariani, J.A. & Thomson, R., The designer's notepad - a hypertext system tailored for design. In R. McAleese & C. Green (Eds.), *Hypertext: state of the art* 260-266, Intellect, Oxford (1990).
10. Sharples, M., & Pemberton, L., *Starting from the writer: guidelines for the design of user-centred document processors* (Research Report No.119). School of Cognitive and Computing Sciences, University of Sussex (1990).
11. Twidale, M. B., Improving error diagnosis using intermediate representations. *Instructional Science* **20**, 359-387 (1991).
12. Shipman, F. M., & Marshall, C. C. Formality considered harmful: experiences, emerging themes and directions. Proceedings *InterCHI '93*, (1993).