Psicológica (2001), 22, 25-42.

The role of working memory on measuring mental models of physical systems

José J. Cañas*, Adoración Antolí, and José F. Quesada

Universidad de Granada, Spain

Up until now there has been no agreement on what a mental model of a physical system is and how to infer the mental model a person has. This paper describes research aimed at solving these problems by proposing that a Mental Model is a dynamic representation created in WM by combining information stored in LTM (the Conceptual Model of the system) and characteristics extracted from the environment. Three experiments tested hypotheses derived from this proposal. Implications for research on Mental Model are discussed.

Key words: Mental Model, Representation, Working Memory, Long-term Memory.

When a person learns to interact with a system it means she/he acquires knowledge about its operation and about the structural relationships between its components. Researchers have called this knowledge the 'Mental Model' of the system (Moran, 1981). The existence of Mental Models, and their importance during the interaction with the system, has been demonstrated in numerous experiments (e.g., Kieras and Bovair, 1984; Cañas, Bajo and Gonzalvo, 1994). Research on group co-operation has also acknowledged the importance of mental models. When members of a group share similar and accurate mental models of group interaction, the group interacts more efficiently and performs more effectively (Cannon-Bowers et al, 1993). The concept of Mental Model is particularly important for research on Team behaviour. Teams are groups in which the members work together on the same task to solve a common problem and there is no division of work responsibilities (Cannon-Bowers et al, 1993). To perform the task, members of teams must develop a

* Address correspondence to: José J. Cañas. Departamento de Psicología Experimental Universidad de Granada 18017 Granada, Spain delagado@goliat.ugr.es+34 958 243767

common body of knowledge that has been called Team Mental Model (Klimoski and Mohammed, 1994) .

Because of this, in the current investigations on design, learning new interfaces, group co-operation, etc., it is common practice to try to infer which mental model a person or group has. However, the investigation on mental models is blocked at present by two problems. First, a theoretical problem exists, which is reflected in the great confusion concerning the definition of a Mental Model. Though this problem has been stressed for a long time (Rouse and Morris, 1986) it still has not been solved satisfactorily. Secondly, a methodological problem exists that, in part, is a consequence of the definition problem. Although many methods have been proposed for inferring which model users have, all have been criticized as being unreliable (Sasse, 1991).

The term Mental Model has been used by researchers that work in different areas and study different tasks. Johnson-Laird (1983) has formulated his mental model definition in his attempt to explain the reasoning processes in tasks of syllogisms and language comprehension. To execute these tasks, a person forms in Working Memory a mental representation of the world, combining the information stored in Long-Term Memory with the information on the task characteristics extracted by perceptual processes. This representation is called the Mental Model in this context and is, by nature, dynamic. Though the information retrieved from LTM, the knowledge that a person has about the world, is important, Johnson-Laird (1983) gave greater importance to the information extracted by perceptual processes of the characteristics of the task (Rasmussen, 1990).

The research on the interaction with physical systems also considers that a person forms a representation in WM by combining the knowledge stored in LTM and the information extracted from the task characteristics (Gentner and Stevens, 1983). However, in this case the information stored in LTM that is relevant for these researchers is related to the knowledge of the structure and the operation of the physical system. Therefore, the emphasis is on this representation, which is called Mental Model of the Physical System, and the efforts are placed on investigating how it is acquired and extracted from LTM.

Therefore, there are two uses of the term Mental Model. For some researchers, a Mental Model is a representation stored in Working Memory, while for others it is the knowledge stored in Long Term Memory.

The definition problems arise when researchers play with these two definitions and do not make clear to which of them they refer when using the Mental Model concept.

To solve this definition problem, we could reserve the term Mental Model for the dynamic representation that is formed in WM, combining the information stored in LTM and the extracted information from the environment (task). Then, we could use some other terms for referring to the information stored in LTM like, for example, Conceptual Model (Young, 1993). Solving the definition problem in this way would make it possible to emphasise the more important characteristic of a Mental Model: the function of a Mental Model is to simulate the reality in WM.

The dynamic nature of the mental models has an important consequence when considering the methods that have been used to measure them. As Staggers and Norcio (1993) have indicated, if a Mental Model is a knowledge structure that is simulated in WM, we must speak of the Mental Model as a process and as the result of that process. When we measure the Mental Model we are measuring the result of the simulation process. This result we take as a reflection of the knowledge structure that is stored in LTM, the Conceptual Model. However, the simulation is accomplished by selecting the part of the permanent knowledge that is relevant for the task. That is to say, not all the knowledge is selected. The part that is selected will depend on the task, the context, the intentions, etc. It is also possible that the mental model, as measured, might be affected not only by selection but also by transformations performed on the knowledge in order to comply with the elicitation task.

An additional problem with the knowledge elicitation methods that have been used is that subjects are requested to accomplish a different task from the one they would accomplish in the real situation. Therefore, when we measure the mental model with a knowledge elicitation task, the person simulates the real task in her/his WM and responses are given based on this simulation. Therefore, we did not measure the knowledge stored in LTM, but rather the knowledge which is put into WM depending on the elicitation method we use.

When the dynamic nature of Mental Models is ignored, researchers run into some common and unexpected results found in Mental Model literature: experts in a field that are supposed to have good and similar knowledge seem to have different Mental Models (Cooke and Schvaneveldt, 1987; Navarro, Cañas and Bajo, 1996). For example, Cooke and Schvaneveldt (1987) found that expert computer programmers had less similar mental representations of computer concepts than novices. These results could be easily explained: even when two people share the same

Conceptual Model, they can appear to have different Mental Models because when tested individually they execute different tasks in their WM.

Take for example a knowledge elicitation task such as relationship judgements that has been widely used in interface design (Cooke, 1994). This task is similar to the classical similarity task, but instead of asking for similarity, subjects are asked to judge how related two components of the interface are.

In the similarity task, it has been shown that subjects could compare objects along different dimensions. For instance, The United States and Cuba are very related when we compare them taking their geographical location into account. However, they are unrelated when we think of their political regimes (Tversky, 1977). For the same reason, we could assume that the dimension on which two components are judged to be related would depend on the task the subjects simulate in their WM.

A judgement of relationship is the result of processing the characteristics of the two concepts or objects to be compared. Since the comparison is done in WM, the capacity limitation of WM forces the person to select one or two among all possible dimensions (Halford, Wilson and Phillips, 1998). Then, the context effects found in the similarity task (Tversky, 1977; Medin, Golstone and Gentner, 1993) could be explained by assuming that context biases the selection process of dimensions. The same contextual effects, and for the same reasons, could be expected in the relationship task.

The purpose of the three experiments reported here was to demonstrate that task characteristics would affect what a person simulates in her/his Working Memory while performing a knowledge elicitation task and, therefore, what is inferred about his/her knowledge of the system.

The basic hypothesis tested in the experiments was that a Mental Model is a dynamic representation created in Working Memory (WM) by combining information stored in Long-Term Memory (LTM) and characteristics extracted from the environment. Methods proposed to infer the Mental Model a person holds must keep this in mind. The methods that are currently used require that the persons perform a task that is different from the real task they perform when interacting with the system or co-operating with other team members. We assume that the person being tested with these methods simulates the real task in her/his WM to perform the elicitation task. Therefore, what is inferred is the result of this simulation.

The experimental rationale of the experiments was as follows: If several people are taught to interact with a system in such a way that all of them are capable of accomplishing any task we give them, we can assume that they will all have stored the same knowledge (Conceptual Model) about the components of the system and the relationships among them in their LTM. Then, when they are requested to perform a task in which we infer their Mental Model, e.g., to give us a relationship judgement between two components, they will simulate a task in their WM where those two components are implicated. Their judgement will reflect the relationship between the two components as a function of that task. However, on many occasions two components are implicated in several tasks and in each the relationship between them will be different. Therefore, judgements will be made based on the particular task that subjects simulate on their WM.

In the experiments that follow, participants learned to interact with the system until they performed without any errors. They also passed a declarative test in which they answered a set of questions regarding all aspects of the system. Then, they performed a knowledge elicitation task (relationship judgements) in which changes were introduced to affect the operation in Working Memory. If performance on the elicitation task depends on these changes, we will be able to say that it was the result of what happened in Working Memory and not of the Conceptual Model the participants have, the Conceptual Model being perfect and the same for all of them.

EXPERIMENT 1

A commonly used knowledge elicitation technique in interface design has been the relationship judgement task (Cooke, 1994). This task is similar to the classical similarity task, but instead of asking for similarity, participants are asked to judge how related two components are.

Researchers that have worked with the topic of similarity know that the context in which the similarity between two concepts is judged influences the judgements that a subject gives (Goldstone, Medin and Halberstadt, 1997). For example, Medin, Goldstone and Gentner (1993) had groups of subjects rating the similarity of *sunrise* and *sunset* and *sunrise* and *sunbeam*. In one condition of their experiment, one group of subjects rated the pair sunrise-sunset and another group rated sunrise-sunbeam. In this condition, sunrise-sunset was rated as less similar than sunrise-sunbeam. However, in other condition of the experiment, subjects rated both pairs simultaneously. In this condition, sunrise-sunset was rated as more similar than sunrise-sunbeam. Medin et al (1993) argued that since sunset and sunrise are antonyms, they are considered to be not very similar when they are judged in isolation. However, this some characteristic causes them to be considered very related when they are judged in the context of sunrise-sunbeam. Therefore, the context influences how we judge the similarity of two concepts.

In a typical relationship judgement task, item pairs are presented in a sequence. The person sees a pair and judges it, then sees other pair and judges it, etc. The sequence in which the pairs are presented can have an effect on the judgements. When a pair is presented, the person simulates in his/her Working Memory a task in which that item pair intervenes, and, depending on the task that he/she simulates, will issue his/her judgement. After issuing the judgement, a trace of the simulated task remains in Working Memory. When the following pair is presented, the person returns to simulate a task in his/her WM. However, the task simulated to judge this second pair would depend on the task simulated for the first pair. For example, let us suppose that we are evaluating the mental model of two MSWORD users. Then, we present two sequences, one to each user:

1. User One: 1. "Print-File" ; 2. "Search-Edit"; 3. "Search-File"
2. User Two: 1. "Save-File"; 2. "Open-File; 3. "Search-File"

We could predict that the similarity of "Search-File" would be rated higher by b user Two than by user One. File is a menu and an object. The first sequence would let the expert think about the relation "it is in the menu". However, the second sequence points to the relation "things you can do with a file".

In this experiment, we manipulated the presentation sequence of the system's pairs of components on which the participants were giving relationship judgements. For one group, the sequence was random and different for each participant. For the other group, all the participants were presented with only one random sequence. The hypothesis was that the group that had the same sequence would show greater within-group similarity in their judgements than the group with different sequences.

METHOD

Participants. Fifty-six students from the University of Granada participated in the experiment. After eliminating those participants that did not pass the declarative test, results from forty-three participants were analysed. Eighteen participants performed the elicitation task in the same sequence condition and twenty-five in the different sequence condition.

Apparatus. Participants learned to operate a control panel device displayed on the computer screen. The device was a modified version of the one used by Kieras and Bovair (1984) consisting of switches, pushbuttons, and indicator lights (See Figure 1). We told the participants that the device was a control panel of an electrical circuit. Their task consisted of making the current flow from panel S1 to panel S3. They were instructed on the three possible action sequences that allowed them to complete the task:

Route X: press button ON in panel S1 (light I1 turned on); switch toggle switch in panel S2 to X (light I2 turned on; press button B1 in panel S3 (light I3 turned on).

Route Y: press button ON in panel S1 (light I1 turned on); switch toggle switch in panel S2 to Y (light I2 remained off); press button B2 in panel S3 (light I3 turned on).

Route Z: press button On in panel S1 (light I1 turned on); switch toggle switch in panel S2 to Z (lights I2 and I3 turned on).

This system was sufficiently simple so that the participants could learn it easily in a short period of time.

Experimental design. The design was a One-way between-subjects design. Two groups of participants performed the elicitation task in one of two conditions: (1) Same Sequence; (2) Different Sequence.

Procedure. Participants performed three tasks during the experimental session:

The Learning Task: In the first phase of the experiments the participants learned to operate the system until they were capable of executing the three action sequences two times without making any mistakes.

The Declarative Task: Then, they passed a test in which they answered ten questions on the operation of the interface. The purpose of this test was to have a measure of the declarative knowledge that participants had and to assure us that they had actually learned to interact with the interface and that, therefore, we could suppose that they had acquired the conceptual model of the system. We eliminated from the experiment all the participants that failed on one or more questions from the questionnaire.

The Elicitation Task: Finally, participants completed a relationship judgement task concerning 11 interface items. They were to assign ratings to pairs of items presented on the computer screen according to how related they thought the items were. The scale ranged from 1 to 6. A rating of one indicated that the items were unrelated, and a rating of six indicated a high degree of relatedness. The participants were to indicate their responses by pressing the numbers corresponding to their ratings on the keyboard. The instructions emphasised that they should work fast, basing their ratings on their first impression of relatedness.

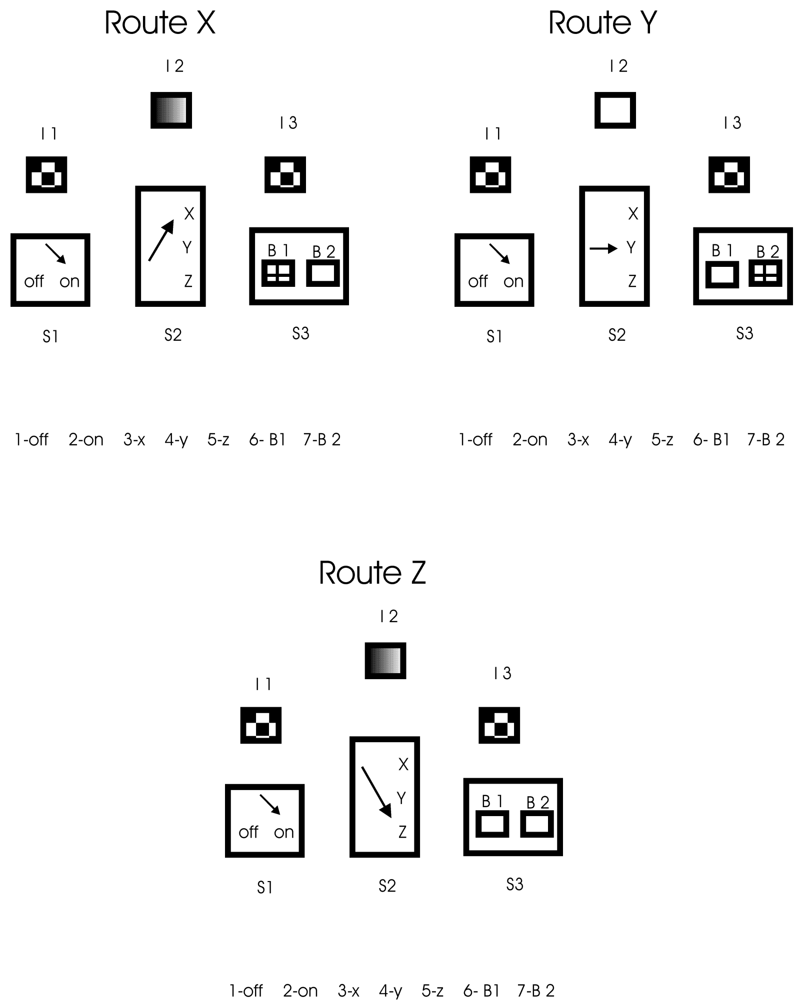

Figure 1. Device set.

For the Same Sequence group, one sequence of pairs was generated at random and all the participants in that group rated the pairs in that sequence. For the Different Sequence group, the computer generated a different random sequence for each participant.

RESULTS

The judgements matrices were transformed into network representations using the Pathfinder algorithm (Schvaneveldt, 1990). Pathfinder is a graph-theoretic technique that derives network structures from proximity data. Pathfinder algorithm takes proximity matrices and produces a network in which concepts are represented as nodes and relations between concepts are represented as links between nodes.

Pathfinder analysis provided us with a measure of the similarity between two networks called C. This value reflects the degree to which the same node in the two graphs is surrounded by a similar set of nodes. A C value of 0 corresponds to two complementary graphs and a value of 1 corresponds to equal graphs.

We calculated the network similarity between all pairs of subjects within one group. Then those C values (253 from the Same Sequence group and 300 from the Different Sequence groups) were submitted to a One-Way ANOVA.

The results of this analysis showed that when participants judged the concept pairs in the same sequence their ratings had more similar network representations (Mean C = 0,563) than when they judged them in different sequences (Mean C = 0,531), $F(1,451) = 8{,}25$, $MSe = {,}013$, $p < {,}01$. Therefore, we can say that the task simulated to judge a pair leaves a track in the Working Memory that influences the rating of the following pair.

EXPERIMENT 2

If the result of the first experiment could be explained by the trace left in Working Memory of the task simulated to judge a pair of items, introducing an interference task after a participant's response to one pair could eliminate that trace.

In this second experiment, all participants were presented with the same sequence of pairs. However, one group saw a number on the computer screen after they rated one pair of items. They had to count backward from that number for 2 seconds. The other group performed the task without counting backward.

We hypothesised that counting backward would erase the WM trace. Therefore two participants that rated the items with this interference task would have less similar network representation than two participants that performed the elicitation task without counting backward.

Participants. Forty-eight students from the University of Granada participated in the experiment. After eliminating those that did not pass the declarative test, results from forty participants were analysed. Twenty students performed the elicitation task with the retention interval and twenty without the interval.

Apparatus and experimental design. The apparatus was the same as in Experiment 1. The experimental design was also a One-way between-subjects design. One group of participants performed the elicitation task with the retention interval and the other group performed without the interval.

Procedure. As in Experiment 1, participants performed three tasks. First, they learned to operate the system. Then they answered questions about the system components and functioning. Finally, they performed the knowledge elicitation task. All participants performed the task with the same sequence. For the group with the retention interval, a number appeared on the screen after the subject's response. Subjects had to count backward from that number for 2 seconds. Then the following pair appeared on the screen.

RESULTS

As in the first experiment, we calculated the network similarity between all pairs of subjects within each group. The C values (190 from each group) were submitted to a One-Way ANOVA.

The group that performed the elicitation task without counting backward showed more within-group similarity (Mean C = 0,602) than the group that performed the task counting backward (Mean C = 0,555) , $F(1,378) = 16,28$, $MSe = ,21$, $p< ,01$. Therefore, counting backward after rating a pair of items erased the trace that the simulated task had left in WM. Then, when a new pair of items had to be rated, the probability that two participants simulated the same task decreased.

When the C values are compared with those found in Experiment 1, we could see an increase that came as a surprising result. We would expect that with more WM load (especially in the group that counted backward), the C values should be lower. Although this could be due only to individual differences, it is difficult to find an explanation based on the type of data obtained in these two experiments. After all, that data showed only how similar ratings were across participants, not what mental model they simulated in their WM. Therefore, we ran a third experiment in which we tried to deal with this problem by forcing participants to pay attention to one dimension of the physical system by manipulating the sequence of pairs. If our general hypothesis was correct, the type of mental model simulated by the participants should be affected by this manipulation of the task characteristics.

EXPERIMENT 3

In our system, the components were related in three dimensions. First, they could be in the same panel (i.e. both were in the S2 panel), be the same class of component (i.e. both were lights), or have a functional relation (i.e. X turned I2 on). In the two previous experiments, the sequence of presentation had an effect on the ratings the participants assigned to pairs of components. However, since the sequence was constructed at random, the control that we had on the role the three dimensions could play was only relative.

In this experiment, two groups of subjects performed the elicitation task with two different sequences of pairs. One group rated the pairs presented in one sequence that was constructed at random, as in the previous experiments, and was the same for all participants. The other group performed the elicitation task with a sequence of pairs in which pairs of components that could be related due to being in the same panels were presented first. With this manipulation, we tried to force participants in this second group to focus attention on the dimension 'Being in the same panel'.

Participants. Forty students from the University of Granada participated in the experiment. After eliminating those that did not pass the declarative test, results from thirty-two participants were analysed. Fifteen students performed the elicitation task with the 'Panel-relation-first sequence' and seventeen with a random sequence.

Apparatus and experimental design. The apparatus was the same as in Experiment 1. The experimental design was also a One-way between-subjects design. One group of participants performed the elicitation task with the 'Panel-relation-first' sequence and the other group with the random sequence.

Procedure. As in Experiments 1 and 2, participants performed three tasks. First, they learned to operate the system. Then they answered questions about the system components and functioning. Finally, they performed the knowledge elicitation task. All participants within each group performed the task with the same sequence.

RESULTS

Three vectors were constructed, one for each dimension. Each pair was scored in each vector depending on whether they had or did not have a relationship on that dimension. A value of one meant that the pair could be related on the dimension, otherwise a value of zero was assigned to that pair. The ratings were averaged for each group. Two regression analyses were performed on the data, one for each group. In the analysis, the vector with the ratings was the dependent variable and the three vectors with the scores on the three dimensions were the independent variables.

Panel and Function dimensions were good predictors of the ratings in the panel-relation-first group, $R^2 = 0.520$, $F(3,51) = 18.47$, $p < 0.001$. Tolerance values for the three variables were 0.996, 0.941, and 0.944, for Class, Panel and Function, respectively. For the random group, all three dimensions were significant predictors, $R^2 = 0.316$, $F(3,51) = 7.875$, $p < 0.001$. Tolerance values were 0.996, 0.941, and 0.944, for Class, Panel and Function, respectively.

The results of these analyses showed the following regression equations. For the panel-relation-first group the equation was

Ratings = 3.25 + 1.45 Panel + 1.38 Function

and for the random group it was

Rating = 3.90 – 0.50 Class + 0.64 Panel + 0.93 Function

Having a class relationship was a significant predictor for the rating of the random group only, $t(51) = 2.07$, $p < .05$. However, for both groups being in the same panel and having a functional relation between components were significant predictors of the rating, t (51) = 5.28, $t(51) = 3.69$, $t(51) = 2.57$, and $t\ (51) = 2.75$ respectively, all with $p < 0.02$. But more relevant for our hypothesis, the comparison between the Beta values for the panel dimension was more important for the 'Panel-relation-first' group ($z = 1.63$, $p = 0.05$). Therefore, focusing the attention of the participant on the panel dimension increased the probability that that dimension influenced his/her rating.

DISCUSSION

The variables that were introduced to affect the contents of WM had an effect on the participants' execution on the elicitation task. Though all participants learned perfectly to interact with the system and were capable of answering questions that were presented to them, their ratings of relationship were affected by what occurred during the elicitation task. Since the inferences that are made on the mental model are based on these judgements, those inferences would be affected by these variables, and, as we claim, what we measure with our elicitation tasks is the content of WM, the Mental Model, and not the Conceptual Model stored in LTM. Therefore, these results would have broad theoretical and methodological implications for further research.

To understand these implications, we should start considering the role that knowledge representation plays in Cognitive Science. Cognition refers to how knowledge is acquired, stored and processed. In Cognitive Science, behaviour is explained as being determined by the knowledge that a person has acquired from the environment. Therefore, since the beginning of the Cognitive Revolution, the development of methods for inferring the knowledge acquired by one person has been within the research agenda of the cognitive scientists.

The research carried out to develop these methods has been conducted after taking for granted the Cognitive Architecture broadly accepted in Cognitive Science. Then, most models in Cognitive Science (e.i. SOAR, ACT-R) take the computer metaphor to assume that there are two memory stores: a permanent store in which knowledge is represented in semantic structures, and a temporal store in which knowledge retrieved from the permanent store is combined with that acquired from the environment.

Therefore, most of the research effort has been dedicated to identifying how knowledge is represented in the permanent store. During the last 30 years there have been many proposals on the possible semantic structures (e.i. categories, concepts, mental models, plans, schemes, etc.) that have generated a lot of discussion on how the information is represented in long-term memory. However, as Anderson (1978) said some years ago, what we observed in our experiments is the result of both a knowledge structure and a process that works on it. Therefore, it is impossible to have a unique picture of the knowledge represented in Long-term Memory without considering the task subjects have to perform. Our results were consistent with this proposal: we cannot study representation independently from how knowledge is used on a particular task**.** In our experiments, when a pair of items was presented, the subject judged the degree of their relation by doing some computation based on how they were represented in Long-term Memory. However, as the results suggested, this computation was affected by the task characteristics.

Two methodological implications follow from these results. First, research on Mental Model should be conducted with a model of the elicitation task that is used for inferring it. This model should take into account the experimental results that have been obtained when the task has been used in previous research. For example, our results were related to the context effects in similarity judgements. The data found in the three experiments are another example of the diagnosticity principle (Tversky, 1997): The more salient features weigh more when judging for similarity or relatedness. In our case, the more salient features for judging one pair of components were the ones that were used for judging the previous pair.

Second, these results also implied that an elicitation task only provides a partial picture of the Conceptual Model and that what we get depends on the particular task performed by our subjects. Therefore, we suggest that when conducting research to prove hypotheses about the knowledge stored in LTM (the conceptual model), elicitation tasks should be used along with other experimental techniques in which that knowledge is used to perform a task that is relevant to the particular domain in which that knowledge is required.

A recent experiment conducted by Navarro and Cañas (2000) illustrates how knowledge elicitation techniques could be combined with semantic priming tasks to study how knowledge is stored and used in the particular domain of Psychology of Programming. These authors were interested in explaining why and under what circumstances visual programming languages would be easier to understand than textual programming languages. Towards this goal, they brought together research from Psychology of Programming and Image Processing.

When they linked these two lines of research, image processing and Psychology of Programming, it was possible to propose a hypothesis about the advantages of visual languages over textual languages. On the one hand, according to the more accepted theories of Program Comprehension (Pennington, 1987), programmers go through two phases when they are understanding a program. In the first phase, programmers develop a knowledge structure representation, 'program model', based on the control flow relationships. In later stages of program comprehension, under appropriate task conditions, programmers develop a plan knowledge representation based on the data flow of the program. This representation contains the main functions of the program, the domain model, and the key information to understand what the program does. It also includes information about the programming situation. The programmer's mental representation seems to depend on his or her experience, the task goal, the length of the program, and the programming language characteristics.

Therefore, Navarro and Cañas (2000) hypothesised that if the role of imagery is to enhance access to meaningful information, then Visual Programming Languages should allow quicker access to data flow information. Therefore visual programmers should more quickly develop a representation based on data flow relationships, even in easier tasks, in comparison with other non-visual programming languages. This hypothesis tried to reconcile these two lines of investigation, which have emphasised *either* the effects of the organisation of the programmer's knowledge representation *or* the role played by features of the notation of the task language on the emergence, development, and support of particular forms of programming strategy (Davies, 1991).

To test this hypothesis, they designed an experiment in which C and Spreadsheet programmers were assessed on their mental representations of programs, under different comprehension conditions. Their hypothesis predicted differences in the type of information represented in the programmers' mental models; differences that depend upon the programming language in which the program to be understood is written. Programmers also performed a primed recognition task.

In the experiment, programmers went through with three phases of the experimental session. First, they were asked to read or modify a program in the language of their expertise. Then, they were presented with small segments of the program and they were asked to group those segments together according to their relation in the program. This grouping data was submitted to an analysis using the Pathfinder technique to assess the programmers' mental representation of the program. Results from this task showed that C programmers grouped segments following mainly the Control Flow structure, while Spreadsheets programmers followed the Data Flow structure.

Finally, they performed a primed recognition task. In each trial, subjects were presented with a program segment (target) taken either from the program that they had read or modified, or from a different program. Their task was to decide as quickly as possible whether or not the segment was part of the program they had already seen. The target segment was preceded by another program segment (prime). The underlying assumption was that if the prime and the target were related in the mental model, the activation of the prime would facilitate the activation of the target. The critical manipulation was the prime-target relationship. To test whether the mental models developed by the subjects were based on data or Control Flow relationships, segments were selected from the two theoretical networks (one Control Flow network and another Data Flow network) to create four priming conditions:

1. Data Flow related Condition: A target segment in the test was preceded by a prime close in the theoretical Data Flow network and far in the Control Flow theoretical network.

2. Control Flow related Condition: A target segment in the test was preceded by a prime close in the theoretical Control Flow network and far in the Data Flow theoretical network.

3. Unrelated condition: the target segment was preceded by a segment from the same program, but hypothesised to be far away in both the control and Data Flow theoretical networks.

4. Non-Program condition: the target segment was from a different program than the prime segment.

The Unrelated and Non-program were the control conditions. Recognition accuracy and time were recorded. As the authors predicted, a priming effect was found. Response times to the target segment preceded by a prime close in the network structure were faster (and with better accuracy) than response time to the same target preceded by a prime which was not as close in the cognitive structure. This priming effect was observed in the control and/or Data Flow conditions depending on the mental model developed by the subject.

Thus, the results from both tasks confirmed the authors' hypothesis in showing that spreadsheet programmers developed Data Flow based mental representations in all situations, while C programmers seemed to access first Control Flow and then Data Flow based mental representations.

Therefore, the combination of primed recognition and Knowledge Elicitation Techniques could give us complementary information, which would allow us a deeper understanding of both the mental representations and the processes beneath them. In this experiment, the use of Knowledge Elicitation Techniques, such as the grouping task, allowed the authors to infer the subject's representation and test whether programmers acquired a type of information or not. On the other hand, the primed recognition task provided detailed time and accuracy data about whether these relationships were used to make decisions about the original code. Primed recognition is a task related to what programmers do when they are reading and comprehending a program. Therefore, it is a task that would reflect the mental model that programmers develop when they are performing a task that is relevant to their domain.

RESUMEN

El papel de la Memoria Operativa en la medición de los modelos mentales de los sistemas físicos. Hasta ahora no ha existido un acuerdo sobre la definición correcta de Modelo Mental de un sistema físico y sobre la forma como podemos inferir el Modelo Mental que una persona tiene del sistema con el que está interactuando. En este artículo se describe una investigación encaminada a solucionar estos problemas con la propuesta de una definición según la cual un Modelos Mental es una representación dinámica creada en la Memoria Operativa, combinando la información almacenada en la Memoria a Largo Plazo y las características extraídas del ambiente. Las hipótesis derivadas de esta propuesta se prueban en tres experimentos y se discuten las implicaciones que sus resultados tienen para la investigación futura sobre Modelos Mentales.

Palabras clave: Modelo Mental, Representación, Memoria Operativa, Memoria a Largo Plazo.

REFERENCES

Cañas, J.J., Bajo, M.T. y Gonzalvo, P. (1994). Mental Models and computer programming. *International Journal of Human-Computer Studies, 40*, 795-811.

Cannon-Bowers, J. A., Salas, E. and Converse, S.A. (1993). Shared mental models in expert team decision making. In Castellan (Ed.) *Individual and group decision making*. Hillsdale, NJ: LEA.

Cooke, N.J. (1994). Varieties of knowledge elicitation techniques. *International Journal of Human-Computer Studies*, *41*, 801-849.

Cooke, N.J. and Schvaneveldt, R.W. (1987). Effects of computer programming on network representations of abstract programming concepts. *International Journal of Man-Machine Studies*, *29*, 407-427.

Davies, S. P. (1991) The Role of Notation and Knowledge Representation in the Determination of Programming Strategy: A Framework of Integrating Models of Programming Behaviour. *Cognitive Science*, *15*, 547-572.

Gentner, D. and Stevens, A.L. (1983) *Mental Models*. Hillsdale: NJ: LEA.

Goldstone, R. L., Medin, D. L. and Halberstadt, J. (1997). Similarity in context. *Memory and Cognition*, *25*, 237-255.

Halford, G.S., Wilson, W.H., and Phillips, S. (1998). Processing capacity defined by relational complexity: Implications for comparative, developmental and cognitive psychology. *Behavioral and Brain Sciences*, *21*, 803-831.

Johnson-Laird, P.N. (1983). *Mental Models. Cambridge: Cambridge University Press.*

Kieras, D.E., and Bovair, S. (1984) The role of mental model in learning to operate a device. *Cognitive Science*, *8*, 255-273.

Klimoski, R. and Mohammed, S. (1994). Team Mental Model: Construct or Metaphor?. *Journal of Management*, *20*, 2, 403-437.

Medin, D.L., Goldstone, R.L., and Gentner, D. (1993). Respects for similarity. *Psychological Review*, *100*, 254-278.

Moran, T.P. (1981) An applied psychology of the user. *Computing Surveys*, 13, 1-11.

Navarro, R., Cañas, J.J. and Bajo, M.T. (1996). Pictorial aids in computer use. In T.R.G. Green, J.J. Cañas and C. Warren (eds). *Proceedings of the 8th European Conference on Cognitive Ergonomics*. Granada. EACE.

Navarro, R. , and Cañas, J. J. (2000). Are visual programming languages better? The role of imagery in program comprehension. [under review]

Orasanu, J. and Salas, E. (1993). Team decision making in complex environments. In G. Klein, J. Orasanu, R. Caldewood and C.E. Zambok (Eds.) *Decision making in action: Models and Methods. Norwood, NJ: Ablex.*

Pennigton, N. (1987) Stimulus Structures and Mental Representation In Expert Comprehension of Computer Programs. *Cognitive Psychology*, 19, 295-341.

Rasmussen, J. (1990). Mental models and the control of action in complex environments. D. Ackermann and M.J. Tauber (Eds.) *Mental Models and Human-Computer Interaction*, vol. 1. Amsterdam: North-Holland.

Rouse, W. B. and Morris, N.M. (1986). On looking into the black box: Prospects and limits in the search for mental models. *Psychological Review*, *3*, 349-363.

Sasse, M.A. (1991). How to t(r)ap users' mental models. En M.J. Tauber y D.Ackermann (Ed.) *Mental Models and Human-Computer Interaction*,vol. 2. Amsterdam: North-Holland.

Schvaneveldt, R.W. (1990). *Pathfinder Associative Networks: Studies in Knowledge Organization*. Norwood, NJ: Ablex Publishing.

Staggers, N. and Norcio, A.F. (1993). Mental models: concepts for human-computer interaction research. *International Journal of Man-machine studies*, *38*, 586-605.

Tversky, A. (1977). Features of similarity. *Psychological Review*, *84*, 327-352.

Young, R. M. (1983). Surrogates and mappings: two kinds of conceptual models for interactive devices. In D. Gentner and A.L. Stevens (Eds.) *Mental Models*. Hillsdale: NJ: LEA.

(Recibido10/1/00:; Aceptado:12/9/00)