

## Do Diagrams make us Smart(ER)?

JUDITH GOOD AND RICHARD COX

Department of Artificial Intelligence, University of Edinburgh  
80 South Bridge, Edinburgh EH1 1HN, Scotland  
*e-mail: judithg@aisb.ed.ac.uk*

PAUL BRNA

Computing Department, Lancaster University  
Lancaster LA1 4YR, England

External representations (ERs) are currently attracting a great deal of interest, partly because new technologies allow the use of more sophisticated, multimedia programming environments, and partly because of the increasing interest in collaborative work. The former raises the possibility of switching between representations or presenting multiple representations simultaneously. In the latter case, external representations can act as an important communication medium for software development teams.

We examine the utility of multiple representations in reasoning and consider how these results may be applied to work in the visual programming area. This study is being undertaken partly due to the (currently) limited number of experimental studies which address the utility of graphical/diagrammatic representations in programming. Our knowledge of the utility of diagrammatic ERs in programming is relatively slight - with research results indicating effects that are not entirely favourable to the claim that diagrammatic ERs actually do make us smarter (Anjaneyulu & Anderson, 1992; Cunniff & Taylor, 1987; Green et al, 1991; Green & Petre, 1992; Pandey & Burnett, 1993). Some of these studies have gone beyond a simple attempt to prove the superiority of visual languages over their textual counterparts to look at such issues as the relationship between notation and task requirements (Green et al, 1991; Green & Petre, 1992) and the role of individual differences (Cunniff & Taylor, 1987). However, we argue that the interaction between a number of factors needs further investigation.

The findings from studies on graphical reasoning in the domains of logic (Cox et al, 1994; Oberlander et al, 1994; Stenning et al, in press) and analytical reasoning (Cox & Brna, 1994) show that users have cognitive styles which vary, and which may affect their performance using a given type of representation, and also that a single representation type may not be optimally useful across a range of problems.

The implications of the findings from these other domains support the argument that systems are required which can offer a range of representations to the user. We believe that these results have important consequences for the development of novice visual programming environments. We therefore seek to develop the argument that this is indeed the case. To this end, we examine some issues that have a bearing on the claim that diagrams effectively help programmers in significant ways and we consider the implications of the logic and analytic reasoning research for novice programmers working in visual programming environments.

Many diagrammatic notations found in visual programming are informationally equivalent abstractions (c.f. (Pennington, 1987)). However, the utility of any particular abstraction might be expected to vary for different individuals and for different tasks. We outline a preliminary investigation into the utility of presentational form in relation to informational content - the initial results will be presented at the workshop.

## References

- Anjaneyulu, K.S.R. and Anderson, J.R. (1992). The advantages of data flow diagrams for beginning programming. In Frasson, C., Gauthier, G. and McCalla, G.I., (eds.), *Intelligent tutoring systems: Second International Conference, ITS '92*, pages pp. 585-592.
- Cox, R. and Brna, P. (1994). Supporting the use of external representations in problem solving: The need for flexible learning environments. DAI Research Report, the University of Edinburgh. Submitted to the *Journal of Artificial Intelligence in Education*.
- Cox, R., Stenning, K. and Oberlander, J. (1994). Graphical effects in learning logic: Reasoning, representation and individual differences. In *Proceedings of the 16th Annual Meeting of the Cognitive Science Society*.
- Cunniff, N. and Taylor, R.P. (1987). Graphical vs. textual representation: An empirical study of novices' program comprehension. In Olson, G.M., Sheppard, S. and Soloway, E., (eds.), *Empirical Studies of Programmers: Second Workshop*, pages 114-131, New Jersey. Ablex Publishing Corporation.
- Green, T.R.G. and Petre, M. (September 1992). When visual programs are harder to read than textual programs. In *ECCE-6: Proceedings of the Sixth European Conference on Cognitive Ergonomics*.
- Green, T.R.G., Petre, M. and Bellamy, R.K.E. (1991). Comprehensibility of visual and textual programs: A test of superlativism against the 'match-mismatch' conjecture. In Koenemann, S.P., Moher, T.G. and Robertson, S.P., (eds.), *Empirical Studies of Programmers: Fourth Workshop*, pages 121-146, New Jersey. Ablex Publishing Corporation.
- Oberlander, J., Cox, R. and Stenning, K. (1994). Proof styles in multimodal reasoning. In Seligman, J. and Westerstahl, D., (eds.), *Language, Logic and Computation: The 1994 Moraga Proceedings*. CSLI Publications.
- Pandey, R.K. and Burnett, M.M. (1993). Is it easier to write matrix manipulation programs visually or textually? an empirical study. In *Proceedings of the 1993 IEEE Symposium on Visual Languages*, pages 344-351.
- Pennington, N. (1987). Stimulus Structures and Mental Representations in Expert Comprehension of Computer Programs. *Cognitive Psychology* 19:295-341.
- Stenning, K., Cox, R. and Oberlander, J. (in press). Contrasting the cognitive effects of graphical and sentential logic teaching: Reasoning, representation and individual differences. *Language and Cognitive Processes*.