Relating cognitive dimensions to expert software development

Marian Petre, 14 Sep 2005

This paper will reflect on how empirical studies of expert software development have informed the development of cognitive dimensions – and how such studies continue to raise questions.

The paper will consider in turn:

The identification of individual CDs, for example:

- *juxtaposability*: The notion of 'juxtaposability' was influenced by both the LabView experiment and observations of expert practice which emphasised the need to juxtapose different representations, as well as different parts of a given representation.
- *secondary notation*: the notion of 'secondary notation' took form during our analysis of the LabView experiment, which made evident experts' use of layout and perceptual cues which are not formally part of the notation to inform reading and comprehension.

Discussions about tradeoffs and context of use, for example:

- *viscosity*: The notion of 'viscosity' was adopted easily into professional programmer communities, where it was recognised immediately. One group augmented it with the notion of a 'congealing point': the point at which a design needs to stabilise, when change should become more demanding. This brought the dimension into a broader perspective reflecting the context of use 'in the large'.
- *closeness of mapping / overheads of abstraction*: Again, observation of professional practice prompted intensive discussion about the nature of tradeoffs between dimensions – and about shifting values for dimensions depending on the context. The desirability of 'closeness of mapping' contrasted with the notion exposed by experts of 'useful awkwardness' and with experts' deliberate shifts among paradigms and reasoning models in order to promote insight into a problem or solution.

The paper will go beyond this retrospective view, for example:

- fruitful images and mental imagery, where it's not at all clear that CDs are the answer but it's also not clear what would be better.
- contrast to Khazaei and Roast's 2001 PPIG paper: considering the utility of cognitive dimensions analysis in revealing *user perspectives* on representations, rather than just qualities of representations themselves.

I hope to use these reflections and evidence from observations of expert programmers to drive a consideration of underlying cognitive issues, which may lie deeper than the present cognitive dimensions reach. This is probably the 'meatiest' part of the paper, but it's also the part I am least able to articulate at present.