Topic 11

Parallel Programming: Models, Methods, and Languages

Luc Bougé, Bill McColl, Mamoun Filali, and Henk Sips

Co-chairmen

For the last three years, Euro-Par has included two separate lines of workshops on parallel programming models, languages and methods.

- The first has been devoted to the design of parallel languages and their semantics: *Parallel languages, programming and semantics* (Ian Foster, 1996), *Parallel Languages* (Ron Perrott, 1997), *Parallel Programming Languages* (Henk Sips, 1998).
- The second has been devoted to the abstract models for parallel programming: *Theory and models of parallel computing* (Bill McColl, 1996), *Programming Models and Methods* (David Skillicorn, 1997; Christian Lengauer, 1998),

This year, we have decided to merge the two lines of workshops into a single topic, to emphasize the necessary collaboration between language designers and those working on theoretical models. We feel that these two fields have reached a stage where it is no longer appropriate to pursue these issues in isolation from one another. Developing abstract programming models in the long term makes sense only if they can be “embodied” into some sort of programming language: only then can they be tested against *real* problems. Conversely, the common experience is that it is hopeless to promote yet-another-programming-language (or library) unless it is based on some sort of abstract model which helps the users in reasoning about their programs in a structured way: parallel programming in the large is far too complex to rely on intuition only.

There has already been several examples of this converging trend in the past. For instance, papers about BSP, skeletons, or data-parallel programming can be found in both lines of workshops, depending on whether the stress is put on the use of languages derived from these models, or on the study of their mathematical properties. The decision to merge the two fields in this year’s Euro-Par recognizes the importance of this evolutionary trend, and aims to emphasize it in current and future research.

This decision to merge these two strands has resulted in a large number of papers being submitted: 33, one of the highest figures among this year’s topics. The selection was a difficult task, as the scientific level of the submissions was high, and covered a wide range of issues related to parallel programming

models. Which criteria may be used to identify a model as specific to parallel programming? How can one evaluate the practical usability of a new programming concept? We hope that this conference, and others which follow, will provide a place to discuss these and other important related questions.

We wish to thank Dominique Méry whose help and encouragement were crucial in making this merging possible.