The Scalability Problem
of Denotational Semantics

Abstract

Denotational semantics is a very powerful and elegant method to describe the semantics of programming languages. However, denotational semantics descriptions of full-scale programming languages are very difficult to read and understand, and even harder to maintain or modify. All the elegance of the method succumbs under the complexity of large semantics descriptions. Denotational semantics definitions simply do not scale.

In Software Engineering there are powerful mechanisms to improve software readability, reusability and extensibility, and, therefore, to keep software complexity under the designer control, which is essential to achieve scalability. The most important mechanisms are visibility control and encapsulation.

Although it is widely accepted that in the field of software development high degree of modularization can be achieved nowadays, formal programming languages semantics definitions are still hard to be efficiently modularized, even using the most advanced and currently available abstraction mechanisms. Perhaps this is the main reason why denotational semantics is often considered as only of research interest, and stands little chance of becoming mainstream practices.

In this talk we will take the denotational semantics method as a model to highlight the principal causes that make modularization of formal semantics inheritly difficult, and to identify reseach directions.