Algebraic methods for the study of module constructs in Logic Programming

Fernando Orejas
Departament Llenguatges i Sistemas Informatics
Univ. Polit. Catalunya
Barcelona

1 Abstract

In contrast with the original emphasis in equational logic, in recent years, much of the work conducted in the area of algebraic specification has dealt with modularity and structuring issues for "general" specification formalisms, i.e. the aim is that the results obtained should be applicable independently of the specific logic used for specification. This kind of work essentially started with the introduction of the notion of "institution" by Goguen and Burstall [1, 2]. In our opinion, there is much to gain in applying general "institution-independent" results to the area of logic programming. Some previous work already done in this direction is [3, 7].

In this tutorial we will present and "instantiate" some of these general results in order to study a number of aspects concerning modularity in Logic Programming that, in our opinion, have often been neglected or not always adequately approached. Being specific, following [6], first we will see how one can approach, in a systematic manner, the semantic definition of modular constructions. The idea is that, if we want to define the semantics for a given kind of modular unit and to prove that this definition satisfies certain properties (for instance, compositionality), then we should first work at an abstract level, by identifying the intended semantics with adequate algebraic constructions, and by proving the given properties making use of the structural properties of the underlying formalism. Afterwards, if needed, we may provide a concrete representation of these algebraic constructions.

We will also analyse several issues concerning modular system development. In particular, following [4, 5], we will study what are the adequate notions of module specification and implementation and how to ensure the satisfaction of the property of modular correctness, i.e. that correctness of a modular system is a consequence of the correctness of all the modules in the system. Moreover, we will see what is the impact of these results in the design of specific Logic Programming module constructs.

References

[1] J.A. Goguen, R.M. Burstall. *Introducing institutions.* Proc. Logics of Programming Workshop, Carnegie-Mellon. Springer-Verlag LNCS 164, 221-256 (1984).

[2] J.A. Goguen, R.M. Burstall. *Institutions: Abstract model theory for specification and programming* Journal of the ACM 39, 1 (1992) 95-146.

[3] J. A. Goguen, J. Meseguer. *Models and Equality for Logical Programming.* In Proc. TAPSOFT'87 vol.2, Springer LNCS 250 (1987), pp. 1-22.

[4] M. Navarro, F. Orejas, A. Sanchez. *On the correctness of modular systems.* Th. Comp. Sc. 140 (1995) 139 - 177.

[5] F. Orejas, M. Navarro, A. Sanchez. *Algebraic Implementation of abstract data types: a survey of concepts and new compositionality results.* Math. Struct. in Comp. Science 6 (1996) 33-67.

[6] F. Orejas, E. Pino, H. Ehrig. *Institutions for Logic Programming.* To appear in Th. Comp. Sc. Short version in Mathematical Foundations of Computer Science 94 (J. Privara, B. Rovan, P. Ruzicka, eds.) Springer-Verlag LNCS 841 pp. 112-126.

[7] D.T. Sannella, L.A. Wallen. *A calculus for the construction of modular Prolog programs.* In Proc. 1987 IEEE Symp. on Logic Programming (1987), pp. 368-378.