

Creating a HasCASL Library

Glauber M. Cabral^{1*}, Arnaldo V. Moura^{1†}, Christian Maeder²,
Till Mossakowski², Lutz Schröder²

¹Institute of Computing , University of Campinas, Brazil

`glauber.cabral@students.ic.unicamp.br, arnaldo@ic.unicamp.br`

²DFKI-Lab Bremen and Dept. of Comput. Sci., Universität Bremen, Germany

`{Christian.Maeder, Till.Mossakowski, Lutz.Schroeder}@dfki.de`

Abstract

The practical use of a specification language depends on the availability of a set of predefined standard specifications. Although the *Common Algebraic Specification Language (CASL)* has a library with such predefined specifications, its higher order extension, named *HasCASL*, lacks predefined higher order specifications. In this paper, we describe the specification of a library for the *HasCASL* language based on the *Haskell Prelude* library providing the higher order functions and data types lacking in the *CASL* library. We use the *Hets* tool for parsing specifications, generating theorems and translating between the *HasCASL* and *HOL* languages. To verify the generated theorems, we use the *Isabelle* theorem prover. Our library comprises basic data types, such as boolean, ordering, lists, characters, strings, numbers and monadic types, and related functions. Two example specifications were developed to demonstrate the use of our library. We specified theorems to guarantee the behavior of the specifications and we verified most of them using the available tools. Our library leaves room for proofs dealing with numbers, as they depend on homomorphisms that have been developed by the *HasCASL* development team.

*Supported by CNPq grant 132039/2007-9

†Supported by CNPq grants 472504/2007-0 and 304363/2008-1