Funcons in Rascal - report

Davy Landman

November 16, 2011

1 The architecture

1.1 ASF-SDF

In ASF-SDF the funcons are implemented in a two stage approach. The first stage is 'compiling' the CSF (funcons specifications) into SDF definitions. These definitions can then be used to describe the semantics of a programming language. These funcon semantics are connected to syntax of the programming language using the ASF equations.

Current limitations: At the time of writing, not all funcons are specified in CSF, which means that not all the SDF defined funcons were generated from the CSF specifications. Moreover, the CSF specifications contain the descriptions of how to 'interpret' a funcon, but the generation of an interpreter was not yet implemented.

1.2 Rascal

The Rascal funcons implementation aims to closely match the ASF-SDF implementation. Rascal however, has more features suited for this domain and we have chosen to use those features to showcase the possible improvements.

The first stage is similiar, Rascal also uses CSF to generate the funcon specifications. However, since Rascal features functions and ADTs we do not generate funcon as a language specification but we generate Rascal files containing the funcon ADT structure. These funcon ADTs can than be used to define the semantics of a programming language in a similar fashion as in ASF-SDF.

Current limitations: Same as in ASF-SDF.

2 Manual implementation

3 CSF generated implementation

4 Limitations of rascal implementation

5 Advantages of rascal implementaton