ARL and PR2, reversible languages

Jurriaan Hage

1993-1994
Abstract

This paper is devoted to the description of the languages ARL and PR2. ARL can be viewed as a cross-fertilization of term rewriting systems and functional languages. The key concept in the language is reversibility, which means informally, that under certain circumstances, we can reconstruct the question from the answer. PR2 is an extension to ARL to make it more suitable for our domain of application: the translation between programming languages.

Keywords:
reversibility, analysis, generation, controlled continued analysis, pairing, rules, strategy, driver, matching, interlingua.
Acknowledgements

No-one pulls it off without the help of others, so I guess that includes me: first I'd like to thank the ones, with whom I talked 'professionally': Paul Jansen for our discussion over the phone of 42 and its relation with PR2, Gert-Jan Kamsteeg for our discussion on the matching of patterns and his suggestion to use empty rules in concatenations, Henk Goeman for his help with the book of Tennent and his advise on the use of \LaTeX, prof. dr. J. Landsbergen for answering my questions about Rosetta, but most of my thanks go out to my supervisor Alexander Ollongren for listening to all of my ideas and helping me find my way when there were so many roads that could be followed.

On the non-professional side I'd like to thank the friends with whom I talked about my project, although they might or might not have understood a word of it. Most importantly my thanks go out to my parents, who have provided me with a home in which I had the room and time to study, and more specifically to work on my project and to Nikë and my parents for their moral support in times of hardship.
# Contents

1 Introduction .......................... 5  
  1.1 The Rosetta project .................. 5  
  1.2 Our problem ......................... 6  
  1.3 The user view ....................... 6  
  1.4 What to expect ...................... 7  

2 The syntax of ARL ..................... 8  
  2.1 Defining the syntax and terminology .......... 8  
  2.2 Restricting the language of patterns .......... 10  

3 The semantics of ARL ................. 12  
  3.1 Design choices ..................... 12  
  3.1.1 Guards do not contain subrules .......... 12  
  3.1.2 Eagerness ........................ 13  
  3.1.3 Nested patterns within subrules .......... 13  
  3.2 Strict equality ..................... 13  
  3.3 Substitutions ....................... 15  
  3.3.1 Introductory definitions ............... 15  
  3.3.2 Composing substitutions ............... 17  
  3.3.3 Inverting substitutions ............... 19  
  3.4 Morphological similarity ............... 20  
  3.5 Generality ......................... 22  
  3.6 Unifiability ....................... 25  
  3.7 Isomorphism ....................... 26  
  3.8 Partial orders ..................... 27  
  3.8.1 Definitions ...................... 27  
  3.8.2 Least common upper bounds .......... 28  
  3.8.3 Greatest common lower bounds .......... 30  
  3.9 History .......................... 32  
  3.9.1 Unique reconstruction of an argument .... 32  
  3.9.2 Weakly bijective functions .......... 34  
  3.9.3 The implementation of a history ........ 34  
  3.9.4 History and its merits ............. 35  
  3.10 Forward and backward application ........ 36  
  3.10.1 Forward application ............. 36