

Title	形式仕様の記述スタイルに関する研究
Author(s)	杉山, 智倫
Citation	
Issue Date	1997-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1058
Rights	
Description	Supervisor:二木 厚吉, 情報科学研究科, 修士

Compare about Description Style of Formal Specification

Sugiyama Tomonori

School of Information Science,
Japan Advanced Institute of Science and Technology

February 14, 1997

Keywords: Formal Specification, Z Notation, CafeOBJ, Transition System,
Rewrite Rule.

In this paper, we have a comparative study about various description style of formal specification, pay attention to its clearness and extension easily. Our research consists of

- Survey properties of mathematical models and specification language.
- A comparative study of specification style of transition system.
- A suggestion of extensible and modifiable specification style on CafeOBJ.

Introduction

For describing specifications of software systems, we have used a colloquial text, figure and table etc, based on natural(informal) language. But, in recent day, it increases the demand of formal specification which can describe with no-conflict, completeness and no-ambiguous, so we have many research related specification language and its description styles. Formal specification is a part of formal method, written by specification language based on some mathematical model and logic exactly. Formal specification have many advantage to develop software systems, for example, it has high reliability and can mechanical verification. But actually, it has some disadvantage too, when specifier describe specification using specification language. For example, its modeling(from request specifications to mathematical model) is very difficult and it has a lot of restriction of describing it.

In this paper, we discuss these problem, take the position of user's side, using algebraic specification language CafeOBJ and model-oriented specification language Z notation. Our approach is to describe the same examples both of these two languages.

Approach

In this paper, we use two specification language, Z notation and CafeOBJ. Z notation is based on ZF set theory and first-order predicate logic. CafeOBJ is based on order-sorted algebra and rewrite logic, which is an extension to each order-sorted algebra and equational logic. A Comparative study between Z notation and algebraic specification language has been written by Heisel, it models Unix File system using Z notation and PLUS. It is similar to our research, but our approach has different concept and new algebraic techniques, like Rewrite Rule and Hidden Algebra. Our concept is, stand by user's side, to research useful and plainness about specification language. Rewrite Rule is based on Rewriting Logic, which was introduced by Meseguer as a new unifying semantics theory for concurrency. Hidden Algebra was introduced by Goguen and Malcolm, as an algebraic semantics for object paradigm capturing its main features, has designed yet.

In our research, At first, we compare simple specification using Z and CafeOBJ, to investigate description properties of each specification language. Specially, transition system's specification, which describes difficult using algebraic specification language, detailed most carefully. We described our examples Employment Agency using next five specification styles and analyzed these style by its comparison.

- Z specification style using Z schema.
- Explicit state approach style: (based on Initial Algebra)
- Implicit state approach style: (Baumeister's approach)
- Rewrite Rule style: (based on Rewrite Logic)
- Behavioural specification style: (based on Hidden Algebra)

Finally, as these research result, we have some suggestion to design extensible and modifiable specification styles on CafeOBJ. One of these is the specification style using Rewrite Rule and Class Declaration. When we use this style, we can describe transition system formally to make difference between pre-state and post-state clear. The other is based on Hidden algebraic approach. When we want to extend our behavioural state specification, we had better change our specification more concrete version.

Conclusion

In this research, we verify possibility of describing transition system based on relation between pre-state and post-state on CafeOBJ. As result, using Rewrite Rule, we can describe it easily like Z schema. Furthermore, using Class Declaration, we can describe extensible and more formally. And explain the fact with example that Behavioural specification using some hidden sort is more extensible than one hidden sort version. In addition, this paper contains many other considerations on our experience.