Title	CafeOBJ Java
Author(s)	,
Citation	
Issue Date	1999-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1230
Rights	
Description	Supervisor: , ,

The Conversion from CafeOBJ to Java, and synchronous description

Kane Hideki

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

15th Feb. 1999

Keywords: CafeOBJ, Java, Object composition , Transition System, Synchronous description.

1 Introduction

Formal specification languages are expected to improve software developments practice in several ways. One important way of using a formal specification language is for designing systems. That is, for analyzing system behavior based of formal specifications without developing the systems in a programming language.

   In CafeOBJ formal specification language, it is shown that system behaviors can be specified and analyzed very nicely based on (a) behavioral object-oriented modeling, and (b) composition/decomposition using projection operations.. The specifications in this style have the following good properties.

1. The specifications are composed from simple primitive modules in well structured hierarchical fashion.

2. The analysis of specification is easy and can be helped by automatic tools like CafeOBJ interpreter.

This paper reports the research on developing a transformation method from CafeOBJ specifications written in this style to programs written in Java programming language. It is intended to develop the transformation with the following properties:

1. It preserves the good structure of the source CafeOBJ specifications.

2. The target Java programs run with reasonable time and space efficiency

2   Object composition

Reusing specification is done by the projection operators. Projection operators are defined for each composing objects to get their states from the state of a composed object. All methods of the composed object are related to the methods of the composing objects using these projection operators. Projection operator to be able to reuse the behavioural equivalences in the composing objects, i.e. the behavioural equivalence of composed object is a conjunction of all the behavioural equivalences of the composing objects.

3   Approach

The proposal of the rule that The specification described by using CafeOBJ is changed into the efficient Java program is done.

First, basic data structure was used as the easy example for CafeOBJ specification,and a consideration about a change process to Java and changed Java satisfying a specification was done. As a system becomes bigger, verification becomes more difficult though it can be verified in one of the advantages which formal specification is used for. In the method of this thesis. Time in the verification is greatly saved because proof in composition object can be used again proved in the composed object .

Next, the proposal of the change rule about hidden sort and projection was done so that it could change it with the same change process to make it change it with the same change process even in the case as the specification description that an object is composed.

When there is concurrency in CafeOBJ specification , The method which was actually changed into concurrency code was devised by using multi-thread of Java . A consideration was done though it was related to exclusive control. By this,the description of the parallel dispersion system can be done easily by CafeOBJ.

At the end, It was investigated which range the change of CafeOBJ specification had an influence to, and it was taken into consideration that a specification could be changed in the low cost as a related research . Generally When the requirement of the software changed, the specification is changed and you must prove that a requirement is satisfied again in the changed specification . But, a big cost reduction can be expected if it is understood what kind of property you have only to verified on the whole when a change in the requirement influenced which component and component was moved with other components. About the influence range and this influence range of changed Java program was compared, and a consideration was done about the figure which could be changed in the low cost.

4   Conclusion

In this paper,
with keeping the structure of the specification from the CafeOBJ specification, te rule changed into the efficient Java program was shown by using the example of ATM with keeping the structure of the specification from the CafeOBJ specification .

As for the changed Java program, one concrete model is expressed in the model of the plural expressed by the specification of CafeOBJ.

Next, it was stated about the synchronous description to change it into the model which actually paralleled it when it is changed into Java and which could operate. and ATM was described to with CafeOBJ ascertained what was actually carried out with Java in parallel.

At last, the range that a change was necessary it was shown when the module of the extreme high rank and the module of the lowest rank were changed, by using the example that the addition of the functional existent combination and interface is done. When any module was changed, it was found out that the place to change the upper module and a lower module could be limited from this result.