Lecture Notes in Computer Science 562
Edited by G. Goos and J. Hartmanis

Advisory Board: W. Brauer D. Gries J. Stoer

R. Breu

Algebraic Specification Techniques in Object Oriented Programming Environments

Springer-Verlag
Berlin Heidelberg New York
London Paris Tokyo
Hong Kong Barcelona
Budapest

Series Editors

Gerhard Goos
Universität Karlsruhe
Postfach 69 80
Vincenz-Priessnitz-Straße 1
W-7500 Karlsruhe, FRG

Juris Hartmanis
Department of Computer Science
Cornell University
5148 Upson Hall
Ithaca, NY 14853, USA

Author

Ruth Breu
TU München, Institut für Informatik
Postfach 20 24 20, W-8000 München 2, FRG

CR Subject Classification (1991): D.1.5, F.3.1, D.2.1, D.3.3, D.2.4, F.3.2, D.3.1

ISBN 3-540-54972-2 Springer-Verlag Berlin Heidelberg New York
ISBN 0-387-54972-2 Springer-Verlag New York Berlin Heidelberg

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

© Springer-Verlag Berlin Heidelberg 1991
Printed in Germany

Typesetting: Camera ready by author
Printing and binding: Druckhaus Beltz, Hemsbach/Bergstr.
45/3140-543210 - Printed on acid-free paper

Foreword

The professional development of large correct software systems in a systematic, structured and modular way is still a challenge for research and practice in software engineering. In recent years many software engineers had expressed hope that object oriented techniques may improve the technical and economic standards in software engineering by providing better structuring techniques supporting abstraction and reusability. However, until now a foundational theoretical framework for object oriented program development was missing.

Abstract data types as main concept of algebraic specifications have been proved to be among the most fruitful contributions to the foundations of software engineering. Algebraic specifications support an axiomatic logical style of treating data structures and program development concepts such as data refinement. Roughly speaking, algebraic specifications and object oriented ideas can be traced back to common roots: classes of objects and "prefixing" were the central concepts of Simula 67 that proved to be the main sources leading to object oriented programming languages and to the theory of algebraic specifications.

Taking this into account it seems rather obvious to bring together algebraic specifications and object oriented concepts. As B. Meyer had pointed out, object oriented software systems can be understood as structured collections of abstract data type implementations. Thus one may use the algebraic approach as a foundation for explaining and formally defining concepts of object orientation, such as classes, objects and inheritance.

In the Ph.D. thesis of Ruth Breu algebraic specifications serve as a basis for object orientation. The thesis concentrates on notions of inheritance, clientship and subtyping, on data abstractions as a structuring mechanism and finally on questions of dynamic creation of objects. It does not treat the concept of concurrent message passing between objects. In particular, formal model-theoretic definitions of inheritance, clientship, and subtyping are given and their algebraic and logical properties are studied. Moreover, a step towards a methodology for object oriented software design based on proper formal techniques is given.

The thesis is a milestone in the formal foundation of object oriented programming. It provides an encouraging step bringing together pragmatic ideas developed by practitioners and theoretical results from the theory of programming and axiomatization of data structures. This way a solid foundation is given on which an elaborated methodology for object oriented development and for tools and languages supporting such a methodology can be based.

Munich and Passau, October 1991 Manfred Broy, Martin Wirsing

Preface

The main aim of this thesis is to provide a framework for the integrated design of object oriented programs with algebraic specification techniques. Algebraic specifications allow the description of systems in a problem oriented, implementation independent way. Their application within the design and reuse of object oriented programs, therefore, is of major concern.

A central concept of the design method presented is the notion of *data types*. This is the basis for the structuring of algebraic specifications and object oriented programs. An integrated software design can thus be regarded as the step by step development of data types. Depending on the level of abstraction, data types are described by algebraic specifications or by object oriented programs. Two aspects can be observed to be crucial for this integration.

On the one hand, object oriented programs and algebraic specifications have to be related by a notion of *correctness* to model the transition from specifications to program implementations. This thesis presents a notion of correctness which relies on the idea of abstraction functions. The correctness relation is formalised in an algebraic theory and hence enables formal reasoning. This theory is based on an algebraic model of objects.

On the other hand, the object oriented approach to software construction is intimately connected with a set of powerful structuring mechanisms. In order to provide an integrated design environment, a uniform structuring concept for object oriented programs and algebraic specifications has to be developed. *Inheritance*, *subtyping* and *clientship* are three central notions of object oriented structuring. As a starting point, these concepts are described in a language-independent theory. This is the basis for an algebraic specification language and for the kernel of a typed object oriented programming language which are presented subsequently.

This thesis provides the formal foundation for a unified framework of algebraic specifications and object oriented programs. A major guideline has been the development of a design method supporting the structured step-by-step design and reuse of software in this environment.

Acknowledgements

I wish to take the opportunity to express my thanks to all those people who have helped me to produce this work.

My major thanks must go to my supervisors, Prof. Martin Wirsing and Prof. Manfred Broy. Prof. Wirsing deserves my special thanks for his advice and encouragement over the past three years. To Prof. Broy I am particularly grateful for his fruitful comments and continuous support.

I wish to thank my colleagues in the Esprit project DRAGON. In particular, I am indebted to Elena Zucca and Prof. Egidio Astesiano. It has been a pleasure to work with them. Colin Atkinson deserves a special mention for eliminating some of the worst English mistakes in this thesis.

Moreover, I am grateful to my colleagues at the Universität Passau and at the TU München who have read and commented on earlier drafts of this thesis. They are Rolf Hennicker, Heinrich Hußmann and Alfons Geser. For fruitful discussions I wish to thank Prof. Jean-Pierre Finance.

Finally no acknowledgement would be complete without thanking my family. Michael, thank you for hours of discussion with you and for your patience and moral support. A special mention is also due to my little son Korbinian who was born during this work. For their continuous support and encouragement during all my studies I wish to thank my mother and my father. I dedicate my work to them.

Munich, October 1991 Ruth Breu

Contents