

Cambridge University Press  
0521616107 - Predicate Transformer Semantics  
Ernest G. Manes  
Frontmatter  
[More information](#)

---

**PREDICATE TRANSFORMER SEMANTICS**

## Cambridge Tracts in Theoretical Computer Science

*Managing Editor* Professor C.J. van Rijsbergen,  
 Department of Computing Science, University of Glasgow

### Editorial Board

S. Abramsky, Department of Computing Science, Imperial College of Science  
 and Technology  
 P.H. Aczel, Department of Computer Science, University of Manchester  
 J.W. de Bakker, Centrum voor Wiskunde en Informatica, Amsterdam  
 J.A. Goguen, Programming Research Group, University of Oxford  
 J.V. Tucker, Department of Mathematics and Computer Science,  
 University College of Swansea

### Titles in the series

1. G. Chaitin *Algorithmic Information Theory*
2. L.C. Paulson *Logic and Computation*
3. M. Spivey *Understanding Z*
4. G. Revesz *Lambda Calculus, Combinators and Functional Programming*
5. A. Ramsay *Formal Methods in Artificial Intelligence*
6. S. Vickers *Topology via Logic*
7. J.-Y. Girard, Y. Lafont & P. Taylor *Proofs and Types*
8. J. Clifford *Formal Semantics & Pragmatics for Natural Language Processing*
9. M. Winslett *Updating Logical Databases*
10. K. McEvoy & J.V. Tucker (eds) *Theoretical Foundations of VLSI Design*
11. T.H. Tse *A Unifying Framework for Structured Analysis and Design Models*
12. G. Brewka *Nonmonotonic Reasoning*
14. S.G. Hoggar *Mathematics for Computer Graphics*
15. S. Dasgupta *Design Theory and Computer Science*
17. J.C.M. Baeten (ed) *Applications of Process Algebra*
18. J.C.M. Baeten & W. P. Weijland *Process Algebra*
23. E.-R. Olderog *Nets, Terms and Formulas*
27. W.H. Hesselink *Programs, Recursion and Unbounded Choice*
29. P. Gärdenfors (ed) *Belief Revision*
30. M. Anthony & N. Biggs *Computational Learning Theory*
33. E.G. Manes *Predicate Transformer Semantics*
34. F. Nielson & H.R. Nielson *Two Level Functional Languages*

Cambridge University Press  
0521616107 - Predicate Transformer Semantics  
Ernest G. Manes  
Frontmatter  
[More information](#)

---

# PREDICATE TRANSFORMER SEMANTICS

---

ERNEST G. MANES  
*Department of Mathematics*  
*University of Massachusetts at Amherst*



Cambridge University Press  
0521616107 - Predicate Transformer Semantics  
Ernest G. Manes  
Frontmatter  
[More information](#)

---

PUBLISHED BY THE PRESS SYNDICATE OF THE UNIVERSITY OF CAMBRIDGE  
The Pitt Building, Trumpington Street, Cambridge, United Kingdom

CAMBRIDGE UNIVERSITY PRESS  
The Edinburgh Building, Cambridge CB2 2RU, UK  
40 West 20th Street, New York NY 10011-4211, USA  
477 Williamstown Road, Port Melbourne, VIC 3207, Australia  
Ruiz de Alarcón 13, 28014 Madrid, Spain  
Dock House, The Waterfront, Cape Town 8001, South Africa  
<http://www.cambridge.org>

© Cambridge University Press 1992

This book is in copyright. Subject to statutory exception  
and to the provisions of relevant collective licensing agreements,  
no reproduction of any part may take place without  
the written permission of Cambridge University Press.

First published 1992  
First paperback edition 2004

*A catalogue record for this book is available from the British Library*

ISBN 0 521 42036 9 hardback  
ISBN 0 521 61610 7 paperback

**“...I view Elgot’s work as but a first chapter of a theory of iteration and fixed points of which we will see many more future chapters from many hands, alas, without Cal’s criticism and guidance.”**

**Dana S. Scott**

## TABLE OF CONTENTS

Introduction .....	1
<b>Part I. Finite Coproducts</b>	
1. Fundamental Frameworks .....	23
2. Natural Choice Operators .....	31
3. Polymorphic Iterates .....	46
<b>Part II. Structure of a Boolean Category</b>	
4. The Axioms .....	71
5. The Boolean Algebra of Summands .....	80
6. Null and Total Morphisms .....	90
7. Preadditive Categories .....	98
8. Projections and Guards .....	106
9. Hoare Assertions and while-do .....	118
10. Predicate Transformers .....	131
<b>Part III. Metatheory</b>	
11. Representations by Relations .....	146
12. Deterministic Morphisms .....	161
13. Completion and Duality .....	171
14. Equations for 3-valued if-then-else .....	182
<b>Part IV. Distributive Categories</b>	
15. Atomic Morphisms .....	190
16. Boolean Monads .....	199
Bibliography .....	216
Index .....	221

## PREFACE

The founding paper [Pratt 1976] on dynamic logic begins as follows:

“This paper deals with logics of programs. The objective is to formalize a notion of program description and to give both plausible (semantic) and effective (syntactic) criteria for the notion of truth of a description. A novel feature of this treatment is the development of the mathematics underlying Floyd-Hoare axiom systems independently of such systems.”

This book continues study of such mathematics with particular emphasis on semantic frameworks. We intend for these frameworks to be flexible, relying on no particular concept of state. Ultimately, extensions of the theory are to address at least program semantics, operating systems, concurrent processes and distributed networks; but the accomplishments of the foundational core herein are modest.

We shall be concerned with a category-theoretic foundation. One possible paradigm is that a morphism is the behaviour of a program. Composition of morphisms models program-chaining. An implementation of a programming language must provide a definite category in which to assign morphisms to programs. We shall also require that high-level specifications about programs map, as well, to true-false assertions about the corresponding interpreted programs.

Our semantic frameworks are categories satisfying certain axioms, that is, are models of the first-order theory of categories. Composition is the only primitive operation. Such models are strongly typed in that two morphisms cannot be composed unless the target of the first coincides exactly with the source of the second. Other program constructs must be expressed in terms of composition; that this can be done is tribute to the unusual expressiveness of category theory as has been increasingly documented since the seminal founding paper by Sammy Eilenberg and Saunders Mac Lane in 1945. This greatly minimizes the assumptions about programming languages which need to be made up front. Whether the end justifies the means we must leave for you the reader to judge.