

Contents

<i>Preface</i>	<i>page vii</i>
1 Components and the Way Ahead Clemens Szyperski	1
Part One: Frameworks and Architectures	21
2 Key Concepts in Architecture Definition Languages David C. Luckham, James Vera, and Sigurd Meldal	23
3 Acme: Architectural Description of Component-Based Systems David Garlan, Robert T. Monroe, and David Wile	47
4 A Formal Language for Composition Markus Lumpe, Franz Achermann, and Oscar Nierstrasz	69
5 A Semantic Foundation for Specification Matching Yonghao Chen and Betty H. C. Cheng	91
Part Two: Object-Based Specification and Verification	111
6 Concepts of Behavioral Subtyping and a Sketch of Their Extension to Component-Based Systems Gary T. Leavens and Krishna Kishore Dhara	113
7 Modular Specification and Verification Techniques for Object-Oriented Software Components Peter Müller and Arnd Poetzsch-Heffter	137
8 Respectful Type Converters for Mutable Types Jeannette M. Wing and John Ockerbloom	161
Part Three: Formal Methods and Semantics	187
9 A Formal Model for Componentware Klaus Bergner, Andreas Rausch, Marc Sihling, Alexander Vilbig, and Manfred Broy	189

vi	<i>Contents</i>	
10	Toward a Normative Theory for Component-Based System Design and Analysis David S. Gibson, Bruce W. Weide, Scott M. Pike, and Stephen H. Edwards	211
11	An Implementation-Oriented Semantics for Module Composition Joseph A. Goguen and Will Tracz	231
	Part Four: Reactive and Distributed Systems	265
12	Composition of Reactive System Components Kevin Lano, Juan Bicarregui, Tom Maibaum, and Jose Fiadeiro	267
13	Using I/O Automata for Developing Distributed Systems Stephen J. Garland and Nancy Lynch	285