

<<软件构架实践>>

图书基本信息

书名：<<软件构架实践>>

13位ISBN编号：9787302312932

10位ISBN编号：7302312931

出版时间：2013-2

出版时间：清华大学出版社

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<软件构架实践>>

内容概要

<<软件构架实践>>

作者简介

作者:(美)巴斯、克莱门茨、凯兹曼

## 书籍目录

Preface Reader's Guide Acknowledgments PART ONE INTRODUCTION 1 CHAPTER 1 What Is Software Architecture? 3 1.1 What Software Architecture Is and What It Isn't 4 1.2 Architectural Structures and Views 9 1.3 Architectural Patterns 18 1.4 What Makes a "Good" Architecture? 19 1.5 Summary 21 1.6 For Further Reading 22 1.7 Discussion Questions 23 CHAPTER 2 Why Is Software Architecture Important? 25 2.1 Inhibiting or Enabling a System's Quality Attributes 26 2.2 Reasoning About and Managing Change 27 2.3 Predicting System Qualities 28 2.4 Enhancing Communication among Stakeholders 29 2.5 Carrying Early Design Decisions 31 2.6 Defining Constraints on an Implementation 32 2.7 Influencing the Organizational Structure 33 2.8 Enabling Evolutionary Prototyping 33 2.9 Improving Cost and Schedule Estimates 34 2.10 Supplying a Transferable, Reusable Model 35 2.11 Allowing Incorporation of Independently Developed Components 35 2.12 Restricting the Vocabulary of Design Alternatives 36 2.13 Providing a Basis for Training 37 2.14 Summary 37 2.15 For Further Reading 38 2.16 Discussion Questions 38 CHAPTER 3 The Many Contexts of Software Architecture 39 3.1 Architecture in a Technical Context 40 3.2 Architecture in a Project Life-Cycle Context 44 3.3 Architecture in a Business Context 49 3.4 Architecture in a Professional Context 51 3.5 Stakeholders 52 3.6 How Is Architecture Influenced? 56 3.7 What Do Architectures Influence? 57 3.8 Summary 59 3.9 For Further Reading 59 3,10 Discussion Questions 60 PARTTWO QUALITY ATTRIBUTES 61 CHAPTER 4 Understanding Quality Attributes 63 4.1 Architecture and Requirements 64 4.2 Functionality 65 4.3 Quality Attribute Considerations 65 4.4 Specifying Quality Attribute Requirements 68 4.5 Achieving Quality Attributes through Tactics 70 4.6 Guiding Quality Design Decisions 72 4.7 Summary 76 4.8 For Further Reading 77 4.9 Discussion Questions 77 CHAPTER 5 Availability 79 5.1 Availability General Scenario 85 5.2 Tactics for Availability 87 5.3 A Design Checklist for Availability 96 5.4 Summary 98 5.5 For Further Reading 99 5.6 Discussion Questions 100 CHAPTER 6 Interoperability 103 6.1 Interoperability General Scenario 107 6.2 Tactics for Interoperability 110 6.3 A Design Checklist for Interoperability 114 6.4 Summary 115 6.5 For Further Reading 116 6.6 Discussion Questions 116 CHAPTER 7 Modifiability 117 7.1 Modifiability General Scenario 119 7.2 Tactics for Modifiability 121 7.3 A Design Checklist for Modifiability 125 7.4 Summary 128 7.5 For Further Reading 128 7.6 Discussion Questions 128 CHAPTER 8 Performance 131 8.1 Performance General Scenario 132 8.2 Tactics for Performance 135 8.3 A Design Checklist for Performance 142 8.4 Summary 145 8.5 For Further Reading 145 8.6 Discussion Questions 145 CHAPTER 9 Security 147 9.1 Security General Scenario 148 9.2 Tactics for Security 150 9.3 A Design Checklist for Security 154 9.4 Summary 156 9.5 For Further Reading 157 9.6 Discussion Questions 158 CHAPTER 10 Testability 159 10.1 Testability General Scenario 162 10.2 Tactics for Testability 164 10.3 A Design Checklist for Testability 169 10.4 Summary 172 10.5 For Further Reading 172 10.6 Discussion Questions 173 CHAPTER 11 Usability 175 11.1 Usability General Scenario 176 11.2 Tactics for Usability 177 11.3 A Design Checklist for Usability 181 11.4 Summary 183 11.5 For Further Reading 183 11.6 Discussion Questions 183 CHAPTER 12 Other Quality Attributes 185 12.1 Other Important Quality Attributes 185 12.2 Other Categories of Quality Attributes 189 12.3 Software Quality Attributes and System Quality Attributes 190 12.4 Using Standard Lists of Quality Attributes- or Not 193 12.5 Dealing with "X-ability": Bringing a New Quality Attribute into the Fold 196 12,6 For Further Reading 200 12.7 Discussion Questions 201 CHAPTER 13 Architectural Tactics and Patterns 203 13.1 Architectural Patterns 204 13.2 Overview of the Patterns Catalog 205 13.3 Relationships between Tactics and Patterns 238 ..... PARTTHREE ARCHITECTURE INTHE LIFE CYCLE 271 PART FOUR ARCHITECTURE AND BUSINESS 435 PART FIVE THE BRAVE NEWWORLD 501

## 章节摘录

版权页：插图： Increase Cohesion Several tactics involve moving responsibilities from one module to another. The purpose of moving a responsibility from one module to another is to reduce the likelihood of side effects affecting other responsibilities in the original module. Increase semantic coherence. If the responsibilities A and B in a module do not serve the same purpose, they should be placed in different modules. This may involve creating a new module or it may involve moving a responsibility to an existing module. One method for identifying responsibilities to be moved is to hypothesize likely changes that affect a module. If some responsibilities are not affected by these changes, then those responsibilities should probably be removed. Reduce Coupling We now turn to tactics that reduce the coupling between modules. Encapsulate. Encapsulation introduces an explicit interface to a module. This interface includes an application programming interface (API) and its associated responsibilities, such as "perform a syntactic transformation on an input parameter to an internal representation." Perhaps the most common modifiability tactic, encapsulation reduces the probability that a change to one module propagates to other modules. The strengths of coupling that previously went to the module now go to the interface for the module. These strengths are, however, reduced because the interface limits the ways in which external responsibilities can interact with the module (perhaps through a wrapper). The external responsibilities can now only directly interact with the module through the exposed interface (indirect interactions, however, such as dependence on quality of service, will likely remain unchanged). Interfaces designed to increase modifiability should be abstract with respect to the details of the module that are likely to change that is, they should hide those details.



版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>