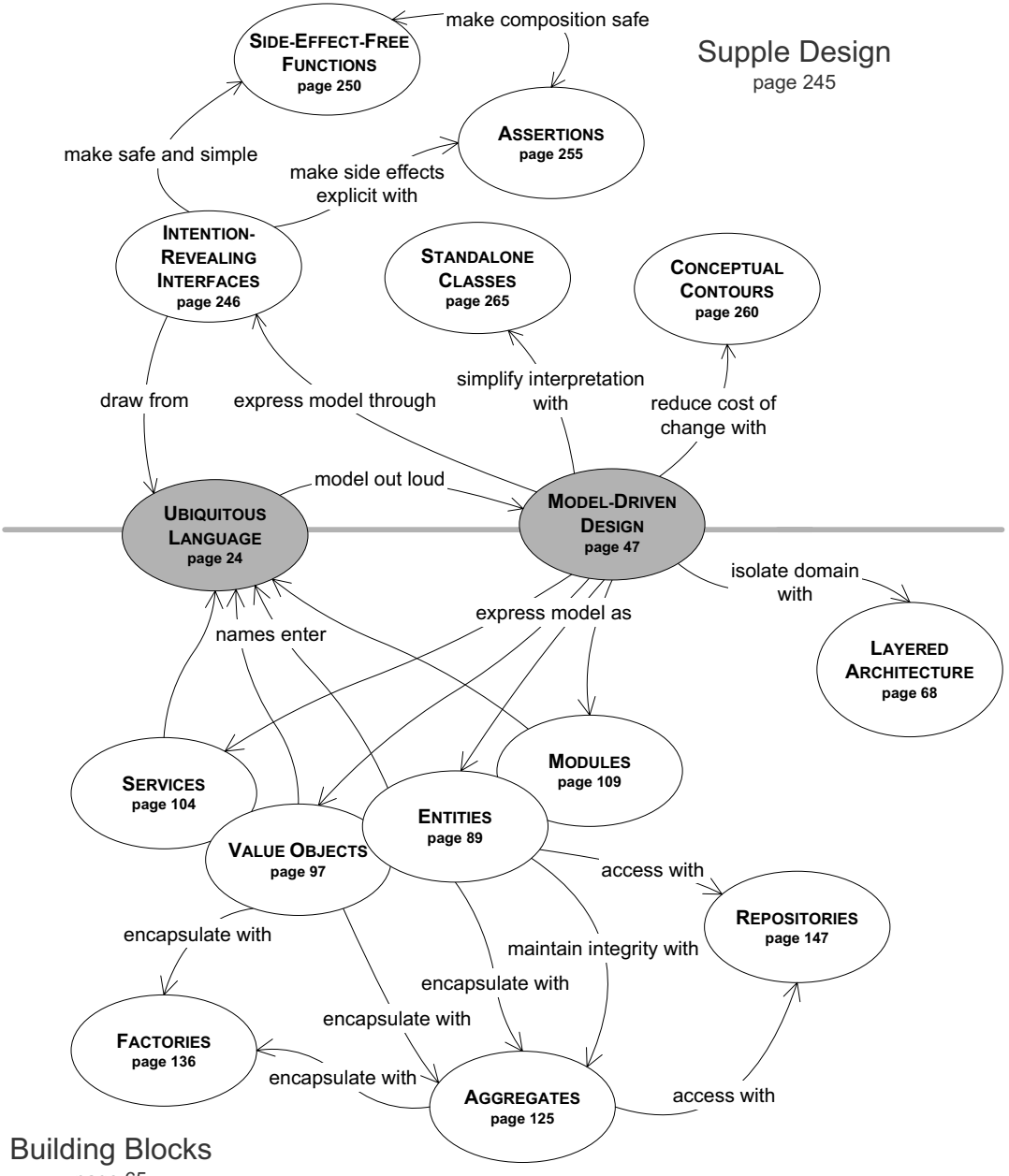
Domain-Driven DESIGN

Tackling Complexity in the Heart of Software

Eric Evans

Foreword by Martin Fowler


SIDE-EFFECT-FREE FUNCTIONS
page 250
make composition safe
Supple Design
page 245
ASSERTIONS
page 255
make safe and simple
make side effects explicit with
INTENTION-REVEALING INTERFACES
page 246
STANDALONE CLASSES
page 265
CONCEPTUAL CONTOURS
page 260
draw from
express model through
simplify interpretation with
reduce cost of change with
model out loud
UBIQUITOUS LANGUAGE
page 24
MODEL-DRIVEN DESIGN
page 47
isolate domain with
LAYERED ARCHITECTURE
page 68
express model as
names enter
SERVICES
page 104
MODULES
page 109
ENTITIES
page 89
VALUE OBJECTS
page 97
access with
REPOSITORIES
page 147
encapsulate with
maintain integrity with
encapsulate with
encapsulate with
FACTORIES
page 136
encapsulate with
AGGREGATES
page 125
access with
Building Blocks
page 65

Domain-Driven Design: Tackling Complexity in the Heart of Software

Table of Contents

Contents

Foreword

Preface

Acknowledgments

Part I: Putting the Domain Model to Work

- Chapter 1: Crunching Knowledge
  - Ingredients of Effective Modeling
  - Knowledge Crunching
  - Continuous Learning
  - Knowledge-Rich Design
  - Deep Models
- Chapter 2: Communication and the Use of Language
  - UBIQUITOUS LANGUAGE
  - Modeling Out Loud
  - One Team, One Language
  - Documents and Diagrams
  - Explanatory Models
- Chapter 3: Binding Model and Implementation
  - MODEL-DRIVEN DESIGN
  - Modeling Paradigms and Tool Support
  - Letting the Bones Show: Why Models Matter to Users

Pearson

Table of Contents

HANDS-ON MODELERS

Part II: The Building Blocks of a Model-Driven Design

Chapter 4: Isolating the Domain

LAYERED ARCHITECTURE

The Domain Layer Is Where the Model Lives

THE SMART UI ANTI-PATTERN

Other Kinds of Isolation

Chapter 5: A Model Expressed in Software

Associations

ENTITIES (A.K.A. REFERENCE OBJECTS)

VALUE OBJECTS

SERVICES

MODULES (A.K.A. PACKAGES)

Modeling Paradigms

Chapter 6: The Life Cycle of a Domain Object

AGGREGATES

FACTORIES

REPOSITORIES

Designing Objects for Relational Databases

Chapter 7: Using the Language: An Extended Example

Introducing the Cargo Shipping System

Isolating the Domain: Introducing the Applications

Distinguishing ENTITIES and VALUE OBJECTS

Designing Associations in the Shipping Domain

AGGREGATE Boundaries

Selecting REPOSITORIES

Walking Through Scenarios

Pearson

Table of Contents

Object Creation

Pause for Refactoring: An Alternative Design of the Cargo AGGREGATE

MODULES in the Shipping Model

Introducing a New Feature: Allocation Checking

A Final Look

Part III: Refactoring Toward Deeper Insight

Chapter 8: Breakthrough

Story of a Breakthrough

Opportunities

Focus on Basics

Epilogue: A Cascade of New Insights

Chapter 9: Making Implicit Concepts Explicit 205

Digging Out Concepts

How to Model Less Obvious Kinds of Concepts

Chapter 10: Supple Design

INTENTION-REVEALING INTERFACES

SIDE-EFFECT-FREE FUNCTIONS

ASSERTIONS

CONCEPTUAL CONTOURS

STANDALONE CLASSES

CLOSURE OF OPERATIONS

Declarative Design

A Declarative Style of Design

Angles of Attack

Chapter 11: Applying Analysis Patterns

Chapter 12: Relating Design Patterns to the Model

STRATEGY (A.K.A. POLICY)

Pearson

Table of Contents

COMPOSITE

Why Not FLYWEIGHT?

Chapter 13: Refactoring Toward Deeper Insight

Initiation

Exploration Teams

Prior Art

A Design for Developers

Timing

Crisis as Opportunity

Part IV: Strategic Design

Chapter 14: Maintaining Model Integrity

BOUNDED CONTEXT

CONTINUOUS INTEGRATION

CONTEXT MAP

Relationships Between BOUNDED CONTEXTS

SHARED KERNEL

CUSTOMER/SUPPLIER DEVELOPMENT TEAMS

CONFORMIST

ANTICORRUPTION LAYER

SEPARATE WAYS

OPEN HOST SERVICE

PUBLISHED LANGUAGE

Unifying an Elephant

Choosing Your Model Context Strategy

Transformations

Chapter 15: Distillation

CORE DOMAIN

Table of Contents

An Escalation of Distillations
GENERIC SUBDOMAINS
DOMAIN VISION STATEMENT
HIGHLIGHTED CORE
COHESIVE MECHANISMS
Distilling to a Declarative Style
SEGREGATED CORE
ABSTRACT CORE
Deep Models Distill
Choosing Refactoring Targets

Chapter 16: Large-Scale Structure

EVOLVING ORDER
SYSTEM METAPHOR
RESPONSIBILITY LAYERS
KNOWLEDGE LEVEL
PLUGGABLE COMPONENT FRAMEWORK
How Restrictive Should a Structure Be?
Refactoring Toward a Fitting Structure

Chapter 17: Bringing the Strategy Together

Combining Large-Scale Structures and BOUNDED CONTEXTS
Combining Large-Scale Structures and Distillation
Assessment First
Who Sets the Strategy?
Six Essentials for Strategic Design Decision Making

Conclusion

Appendix: The Use of Patterns in This Book

Glossary

Pearson

Table of Contents

- A
- B
- C
- D
- E
- F
- I
- M
- N
- R
- S
- U
- V
- W

References

Photo Credits

Index