Software Architecture in Practice

Introduction

1: What Is Software Architecture?

- What Software Architecture Is and What It Isn’t
- Architectural Structures and Views
- Architectural Patterns
- What Makes a “Good” Architecture?

2: Why Is Software Architecture Important?

- Inhibiting or Enabling a System’s Quality Attributes
- Reasoning About and Managing Change
- Predicting System Qualities
- Enhancing Communication among Stakeholders
- Carrying Early Design Decisions
- Defining Constraints on an Implementation
- Influencing the Organizational Structure
- Enabling Evolutionary Prototyping
- Improving Cost and Schedule Estimates
- Supplying a Transferable, Reusable Model
- Allowing Incorporation of Independently Developed Components
- Restricting the Vocabulary of Design Alternatives

3: The Many Contexts of Software Architecture

- Architecture in a Technical Context
- Architecture in a Project Life-Cycle Context
- Architecture in a Business Context
- Architecture in a Professional Context
- Stakeholders
- How Is Architecture Influenced?
- What Do Architectures Influence?

Quality Attributes

4: Understanding Quality Attributes
• Architecture and Requirements
• Functionality
• Quality Attribute Considerations
• Specifying Quality Attribute Requirements
• Achieving Quality Attributes through Tactics
• Guiding Quality Design Decisions

5: Availability
• Availability General Scenario
• Tactics for Availability
• A Design Checklist for Availability

6: Interoperability
• Interoperability General Scenario
• Tactics for Interoperability
• A Design Checklist for Interoperability

7: Modifiability
• Modifiability General Scenario
• Tactics for Modifiability
• A Design Checklist for Modifiability

8: Performance
• Performance General Scenario
• Tactics for Performance
• A Design Checklist for Performance

9: Security
• Security General Scenario
• Tactics for Security
• A Design Checklist for Security

10: Testability
- Testability General Scenario
- Tactics for Testability
- A Design Checklist for Testability

11: Usability

- Usability General Scenario
- Tactics for Usability
- A Design Checklist for Usability

12: Other Quality Attributes

- Other Important Quality Attributes
- Other Categories of Quality Attributes
- Software Quality Attributes and System Quality Attributes
- Using Standard Lists of Quality Attributes—or Not
- Dealing with “X-ability”: Bringing a New Quality Attribute into the Fold

13: Architectural Tactics and Patterns

- Architectural Patterns
- Overview of the Patterns Catalog
- Relationships between Tactics and Patterns
- Using Tactics Together

14: Quality Attribute Modeling and Analysis

- Modeling Architectures to Enable Quality Attribute Analysis
- Quality Attribute Checklists
- Thought Experiments and Back-of-the-Envelope Analysis
- Experiments, Simulations, and Prototypes
- Analysis at Different Stages of the Life Cycle

Architecture in the Life Cycle

15: Architecture in Agile Projects

- How Much Architecture?
• Agility and Architecture Methods
• A Brief Example of Agile Architecting
• Guidelines for the Agile Architect

16: Architecture and Requirements

• Gathering ASRs from Requirements Documents
• Gathering ASRs by Interviewing Stakeholders
• Gathering ASRs by Understanding the Business Goals
• Capturing ASRs in a Utility Tree
• Tying the Methods Together

17: Designing an Architecture

• Design Strategy
• The Attribute-Driven Design Method
• The Steps of ADD

18: Documenting Software Architectures

• Uses and Audiences for Architecture Documentation
• Notations for Architecture Documentation
• Views
• Choosing the Views
• Combining Views
• Building the Documentation Package
• Documenting Behavior
• Architecture Documentation and Quality Attributes
• Documenting Architectures That Change Faster Than You Can Document Them
• Documenting Architecture in an Agile Development Project

19: Architecture, Implementation, and Testing

• Architecture and Implementation
• Architecture and Testing

20: Architecture Reconstruction and Conformance

• Architecture Reconstruction Process
21: Architecture Evaluation

- Evaluation Factors
- The Architecture Tradeoff Analysis Method
- Lightweight Architecture Evaluation

22: Management and Governance

- Planning
- Organizing
- Implementing
- Measuring
- Governance

Architecture and Business

23: Economic Analysis of Architectures

- Decision-Making Context
- The Basis for the Economic Analyses
- Putting Theory into Practice: The CBAM
- Case Study: The NASA ECS Project

24: Architecture Competence

- Competence of Individuals: Duties, Skills, and Knowledge of Architects
- Competence of a Software Architecture Organization

25: Architecture and Software Product Lines

- An Example of Product Line Variability
- What Makes a Software Product Line Work?
- Product Line Scope
- The Quality Attribute of Variability
- The Role of a Product Line Architecture
- Variation Mechanisms
- Evaluating a Product Line Architecture
- Key Software Product Line Issues

**The Brave New World**

26: Architecture in the Cloud

- Basic Cloud Definitions
- Service Models and Deployment Options
- Economic Justification
- Base Mechanisms
- Sample Technologies
- Architecting in a Cloud Environment

27: Architectures for the Edge

- The Ecosystem of Edge-Dominant Systems
- Changes to the Software Development Life Cycle
- Implications for Architecture
- Implications of the Metropolis Model