Agile Software Architecture [Aligning Agile Processes and Software Architectures /

Ali Babar, Muhammad,
Brown, Alan W. (
Wolpé, Sholeh (
Wolpé, Sholeh (
Wolpé, Sholeh (
1962-)
Mistrík, Iván

Elsevier/Morgan Kaufmann,
[2014]

Monografía

Focuses on principles of Agile software development and gaps in the requirements of applying architecture-centric approaches. Readers will learn how Agile and architectural cultures can co-exist and support each other according to the context

Título: Agile Software Architecture Recurso electrónico] Aligning Agile Processes and Software Architectures Edited by Muhammad Ali Babar, Alan W. Brown, Kai Koskimies, Ivan Mistrik

Editorial: Amsterdam Boston Elsevier/Morgan Kaufmann [2014]

Descripción física: xl, 392 p. il

Mención de serie: EBSCO Academic eBook Collection Complete

Bibliografía: Incluye referencias bibliográficas e índice

Contenido: Machine generated contents note: ch. 1 Making Software Architecture and Agile Approaches Work Together: Foundations and Approaches -- 1.1.Introduction -- 1.2.Software Architecture -- 1.2.1.Software Architecture Process and Architecture Lifecycle -- 1.2.2.Architecturally Significant Requirements -- 1.2.3.Software Architecture Design Methods -- 1.2.4.Documenting Software Architecture -- 1.2.5.Software Architecture Evaluation -- 1.3.Agile Software Development and Architecture -- 1.3.1.Scrum -- 1.3.2.Extreme Programming -- 1.4.Making Architectural and Agile Approaches Work -- Acknowledgments -- References -- pt. 1 FUNDAMENTALS OF AGILE ARCHITECTING -- ch. 2 The DCI Paradigm: Taking Object Orientation into the Architecture World -- 2.1.Introduction -- 2.1.1.Agile Apologia -- 2.1.2.Architecture and DCI -- 2.2.The Vision: What is Architecture? -- 2.2.1.Why do we do Architecture? -- 2.2.2.Into Software -- 2.2.3.Why Software Architecture? -- 2.2.4.Architecture and the Agile Agenda -- 2.2.5.DCI as an Integrative View of the Architecture

Metaphor -- 2.3.Form and Function in Architectural History -- 2.3.1.Historic Movements and Ideologies -- 2.3.2. Enter Postmodernism -- 2.3.3.Architecture Finds an Object Foothold -- 2.3.4.Software Engineering and Architecture Today -- 2.3.5.Measures of the Vision -- 2.4.What is Object Orientation? Achieving the Vision -- 2.4.1. The Kay Model -- 2.4.2.Mental System Models -- 2.4.3.Model-View-Controller -- 2.4.4.Patterns -- 2.4.5.Use Cases -- 2.4.6.Many Views of Objects and the Boundaries of MVC -- 2.5.Shortcomings of the Models -- 2.5.1.The Network Paradigm -- 2.5.2.Model-View-Controller -- 2.5.3.Patterns -- 2.5.4.Use Cases -- 2.5.5.The Object Canon -- 2.6.DCI as a New Paradigm -- 2.6.1.A DCI Overview -- 2.6.2.Relating DCI to the Original OO Vision -- 2.6.3.DCI and the Agile Agenda -- 2.7.DCI and Architecture -- 2.7.1.DCI and the Postmodem View -- 2.7.2.Patterns and DCI -- 2.7.3.DCI and the Network Computation View -- 2.7.4.Firmitas, Utilitas, and Venustas -- 2.8.Conclusion -- References -- Further Reading -- ch. 3 Refactoring Software Architectures -- 3.1.Introduction -- 3.2.Dealing with Design Flaws -- 3.3.Evolution and Styles of Refactoring -- Code Refactoring -- 3.4.Evolution and Styles of Refactoring -- Refactoring to Patterns -- 3.5.The Motivation for Software Architecture Refactoring -- 3.6. Architectural Smells -- 3.7.A Real-World Example -- 3.8.Quality Improvement -- 3.9.The Process of Continuous Architecture Improvement -- 3.10.Shallow and Deep Refactoring -- 3.11.Additional Examples of Architecture Refactoring Patterns -- 3.11.1.Breaking Dependency Cycles -- 3.11.2.Splitting Subsystems -- 3.12.Known Obstacles to Architecture Refactoring -- 3.13.Comparing Refactoring, Reengineering, and Rewriting -- 3.14. Summary -- References -- ch. 4 Driving Architectural Design and Preservation From a Persona Perspective in Agile Projects -- 4.1.Introduction -- 4.2.Personas in the Design Space -- 4.3.Discovering ASRs -- 4.3.1.From Features to Architectural Concerns -- 4.3.2.Embedding Architectural Concerns Into Personas -- 4.4.Personas for Driving Architectural Design -- 4.4.1.Goal Analysis -- 4.4.2.Generating and Evaluating Architectural Solutions -- 4.4.3. Examples -- 4.5.Personas and Architectural Preservation -- 4.5.1.Trace By Subscription -- 4.5.2.Generating Persona-centric Perspectives -- 4.5.3.Examples -- 4.6.ASPs in Other Project Domains -- 4.6.1.Mechatronics Traceability -- 4.6.2.Online Trading -- 4.6.3.Bond, James Bond -- 4.7.Conclusions -- Acknowledgments -- References -- ch. 5 Architecture Decisions: Who, How, and When? -- 5.1.Introduction -- 5.2.Research Methodology -- 5.3.The Agile Architecture Axes Framework -- 5.3.1.Who Makes the Architectural Decisions? -- 5.3.2.What Artifacts Are Used to Document the Decision? -- 5.3.3.What Is the Feedback Loop of An Architectural Decision? -- 5.3.4.Summary of the Axes -- 5.4.Industrial Cases -- 5.4.1.Case Alpha -- 5.4.2.Case Beta -- 5.4.3.Case Gamma -- 5.4.4.Case Delta -- 5.4.5.Case Epsilon -- 5.4.6.Overview -- 5.5.Analysis -- 5.5.1.Mapping the Cases to the Triple-A Framework -- 5.5.2. Identified Problems -- 5.5.3.Summary -- 5.6.Reflection -- 5.6.1.Findings -- 5.6.2.Questions of Validity -- 5.7. Related and Future Work -- 5.8.Conclusions -- Appendix -- A Visual Representation of the Case Studies Mapped on the Triple-A Framework -- References -- pt. 2 MANAGING SOFTWARE ARCHITECTURE IN AGILE PROJECTS -- ch. 6 Supporting Variability Through Agility to Achieve Adaptable Architectures -- 6.1.Introduction -- 6.2.Background -- 6.2.1.Variability -- 6.2.2.Agility -- 6.3.Related Work -- 6.4.Challenges when Combining Variability and Agility -- 6.5.Arguments for Combining Variability and Agility -- 6.6.Agile-Inspired Variability Handling -- 6.6.1.Industrial Context: Dutch e-Government -- 6.6.2.Step 1: Conduct Initial Variability Analysis -- 6.6.3.Step 2: Create Initial Architecture Variability Profile -- 6.6.4.Step 10 Architecture-Centric Testing for Security: An Agile Perspective -- 10.1.Introduction -- 10.2.Research Motivation -- 10.3.Overview of Limitations in Current Post-implementation Methods -- 10.3.1.Functional Testing of Security Apparatuses -- 10.3.2.Penetration Testing -- 10.3.3.Threat Modeling -- 10.3.4.Discussion -- 10.4.Introducing Implied Scenarios -- 10.4.1.Detecting Implied Scenarios -- 10.5.Approach -- 10.5.1.Stage 1: Implied Scenario Detection -- 10.5.2.Stage Note continued: 12.5.1.Accelerate the Delivery Pipeline By Incorporating Multiple Perspectives -- 12.5.2.Accelerate Delivery By Maximizing Capacity -- 12.5.3.Accelerate Delivery Through Early Integration -- 12.5.4.Accelerate Delivery Via Early and Continuous Testing -- 12.5.5.Accelerate Delivery Via An Automated Deployment Pipeline -- 12.6. Conclusion -- References -- ch. 13 Building a Platform for Innovation: Architecture and Agile as Key Enablers -- 13.1.Introduction -- 13.2.Worlds Collide -- 13.3.An Architecture Heritage -- 13.4.Iterative Development -- 13.5. Along Came Agile -- 13.6.Agile With Discipline -- 13.7.Beyond Architecture and Agile -- 13.7.1.Define a Project Lifecycle Selection Framework -- 13.7.2.Tailor the Method -- 13.7.3.Consider All Elements of a Development Environment -- 13.7.4.Adopt Change Incrementally -- 13.7.5.Implement a Center of Excellence -- 13.8.Summary -- References -- ch. 14 Opportunities, Threats, and Limitations of Emergent Architecture -- 14.1.Introduction -- 14.1.1. A Brief Definition of Emergence -- 14.1.2.The Idea of Emergent Architecture -- 14.2.Purpose, Activities, and Objectives of Architecture -- 14.2.1.Purpose -- the Why of Architecture -- 14.2.2.Activities -- the How of Architecture -- 14.2.3.Objectives -- the What of Architecture -- 14.3.Analysis of Emergent Architecture -- 14.3.1. Alignment -- 14.3.2.Structuring -- 14.3.3.Implementation of Nonfunctional Requirements -- 14.3.4.Design for

Understandability -- 14.3.5.Design for Change -- 14.4.Discussion -- 14.4.1.Comparison of Explicit and Emergent Architecture -- 14.4.2.A Joint Approach -- 14.5.Conclusion -- References -- ch. 15 Architecture as a Key Driver for Agile Success: Experiences At Aviva UK -- 15.1.Introduction -- 15.2.Challenges to Agile Adoption At Aviva UK -- 15.3.The Key Role of Architecture in Driving Agile Success -- 15.3.1.Sufficient Up-front Architecture and Design -- 15.3.2.Layered Architecture Enabling Independent Change Agility -- 15.3.3."Change-time" Architecture and "run-time" Architecture -- 15.4.Incremental Agile and Architecture Transformation -- 15.5.Conclusions -- References

Detalles del sistema: Forma de acceso: World Wide Web

ISBN: 9780124078857 0124078850 9781306176132 1306176131 9780124077720 0124077722

Autores: Ali Babar, Muhammad, Brown, Alan W. (Wolpé, Sholeh (Wolpé, Sholeh (Wolpé, Sholeh (1962-) Mistrík, Iván

---