

Lean Architecture For Agile Software Development

More and more Agile projects are seeking architectural roots as they struggle with complexity and scale - and they're seeking lightweight ways to do it Still seeking? In this book the authors help you to find your own path Taking cues from Lean development, they can help steer your project toward practices with longstanding track records Up-front architecture? Sure. You can deliver an architecture as code that compiles and that concretely guides development without bogging it down in a mass of documents and guesses about the implementation Documentation? Even a whiteboard diagram, or a CRC card, is documentation: the goal isn't to avoid documentation, but to document just the right things in just the right amount Process? This all works within the frameworks of Scrum, XP, and other Agile approaches

Agile techniques have demonstrated immense potential for developing more effective, higher-quality software. However, scaling these techniques to the enterprise presents many challenges. The solution is to integrate the principles and practices of Lean Software Development with Agile's ideology and methods. By doing so, software organizations leverage Lean's powerful capabilities for "optimizing the whole" and managing complex enterprise projects. A combined "Lean-Agile" approach can dramatically improve both developer productivity and the software's business value. In this book, three expert Lean software consultants draw from their unparalleled experience to gather all the insights, knowledge, and new skills you need to succeed with Lean-Agile development. Lean-Agile Software Development shows how to extend Scrum processes with an Enterprise view based on Lean principles. The authors present crucial technical insight into emergent design, and demonstrate how to apply it to make iterative development more effective. They also identify several common development "anti-patterns" that can work against your goals, and they offer actionable, proven alternatives. Lean-Agile Software Development shows how to Transition to Lean Software Development quickly and successfully Manage the initiation of product enhancements Help project managers work together to manage product portfolios more effectively Manage dependencies across the software development organization and with its partners and colleagues Integrate development and QA roles to improve quality and eliminate waste Determine best practices for different software development teams The book's companion Web site, www.netobjectives.com/lasd, provides updates, links to related materials, and support for discussions of the book's content.

For courses in Advanced Software Engineering or Object-Oriented Design. This book covers the human and organizational dimension of the software improvement process and software project management - whether based on the CMM or ISO 9000 or the Rational Unified Process. Drawn from a decade of research, it emphasizes common-sense practices. Its principles are general but concrete; every pattern is its own built-in example. Historical supporting material from other disciplines is provided. Though even pattern experts will appreciate the depth and currency of the material, it is self-contained and well-suited for the layperson.

Lean Development and Agile Methods for Large-Scale Products: Key Thinking and Organizational Tools for Sustainable Competitive Success Increasingly, large product-development organizations are turning to lean thinking, agile principles and practices, and large-scale Scrum to sustainably and quickly deliver value and innovation. However, many groups have floundered in their practice-oriented adoptions. Why? Because without a deeper understanding of the thinking tools and profound organizational redesign needed, it is as though casting seeds on to an infertile field. Now, drawing on their long experience leading and guiding large-scale lean and agile adoptions for large, multisite, and offshore product development, and drawing on the best research for great team-based agile organizations, internationally recognized consultant and best-selling author Craig Larman and former leader of the agile transformation at Nokia Networks Bas Vodde share the key thinking and organizational tools needed to plant the seeds of product development success in a fertile lean and agile enterprise. Coverage includes Lean thinking and development combined with agile practices and methods Systems thinking Queuing theory and large-scale development processes Moving from single-function and component teams to stable cross-functional cross-component Scrum feature teams with end-to-end responsibility for features Organizational redesign to a lean and agile enterprise that delivers value fast Large-scale Scrum for multi-hundred-person product groups In a competitive environment that demands ever-faster cycle times and greater innovation, applied lean thinking and agile principles are becoming an urgent priority. Scaling Lean & Agile Development will help leaders create the foundation for their lean enterprise—and deliver on the significant benefits of agility. In addition to the foundation tools in this text, see the companion book Practices for Scaling Lean & Agile Development: Large, Multisite, and Offshore Product Development with Large-Scale Scrum for complementary action tools.

Building on their breakthrough bestsellers Lean Software Development and Implementing Lean Software Development, Mary and Tom Poppendieck's latest book shows software leaders and team members exactly how to drive high-value change throughout a software organization—and make it stick. They go far beyond generic implementation guidelines, demonstrating exactly how to make lean work in real projects, environments, and companies. The Poppendiecks organize this book around the crucial concept of frames, the unspoken mental constructs that shape our perspectives and control our behavior in ways we rarely notice. For software leaders and team members, some frames lead to long-term failure, while others offer a strong foundation for success. Drawing on decades of experience, the authors present twenty-four frames that offer a coherent, complete framework for leading lean software development. You'll discover powerful new ways to act as competency leader, product champion, improvement mentor, front-line leader, and even visionary. Systems thinking: focusing on customers, bringing predictability to demand, and revamping policies that cause inefficiency Technical excellence: implementing low-dependency architectures, TDD, and evolutionary development processes, and promoting deeper developer expertise Reliable delivery: managing your biggest risks more effectively, and optimizing both workflow and schedules Relentless improvement: seeing problems, solving problems, sharing the knowledge Great people: finding and growing professionals with purpose, passion, persistence, and pride Aligned leaders: getting your entire leadership team on the same page From the world's number one experts in Lean software development, Leading Lean Software Development will be indispensable to everyone who wants to transform the promise of lean into reality—in enterprise IT and software companies alike.

A resource for individuals responsible for siting decisions, this guidelines book covers siting and layout of process plants, including both new and expanding facilities. This book provides comprehensive guidelines in selecting a site, recognizing and assessing long-term risks, and the optimal lay out of equipment facilities needed within a site. The information presented is

applicable to US and international locations. Note: CD-ROM/DVD and other supplementary materials are not included as part of eBook file.

In *Large-Scale Scrum*, Craig Larman and Bas Vodde offer the most direct, concise, actionable guide to reaping the full benefits of agile in distributed, global enterprises. Larman and Vodde have distilled their immense experience helping geographically distributed development organizations move to agile. Going beyond their previous books, they offer today's fastest, most focused guidance: "brass tacks" advice and field-proven best practices for achieving value fast, and achieving even more value as you move forward. Targeted to enterprise project participants and stakeholders, *Large-Scale Scrum* offers straight-to-the-point insights for scaling Scrum across the entire project lifecycle, from sprint planning to retrospective. Larman and Vodde help you: Implement proven Scrum frameworks for large-scale developments Scale requirements, planning, and product management Scale design and architecture Effectively manage defects and interruptions Integrate Scrum into multisite and offshore projects Choose the right adoption strategies and organizational designs This will be the go-to resource for enterprise stakeholders at all levels: everyone who wants to maximize the value of Scrum in large, complex projects.

This book constitutes the refereed proceedings of the 10th IEEE International Conference Beyond Databases, Architectures, and Structures, BDAS 2014, held in Ustron, Poland, in May 2014.

This book consists of 56 carefully revised selected papers that are assigned to 11 thematic groups: query languages, transactions and query optimization; data warehousing and big data; ontologies and semantic web; computational intelligence and data mining; collective intelligence, scheduling, and parallel processing; bioinformatics and biological data analysis; image analysis and multimedia mining; security of database systems; spatial data analysis; applications of database systems; Web and XML in database systems.

Apply lean principles to your next architectural project and improve your bottom line with the help of this practical volume *Lean Architecture: Excellence in Project Delivery* shows readers a path to consistently achieve excellence in project delivery via the application of lean concepts and process management. Authors Michael Czap and Gregory Buchanan challenge readers to reexamine their approach to architectural practice and projects by presenting a unique and compelling alternative. *Lean Architecture* details the crucial metrics and implementation strategies that combine to improve the efficiency and profitability of projects taken on by firms of all sizes. Readers of the book will learn to: Maximize the use of their resources to deliver superior results in less time Minimize waste, cost, and inefficiency in their firm's operations Move between radically different project scales while retaining efficient and effective processes *Lean Architecture* is perfect for firm leaders, project managers, and project architects who seek to improve their ability to deliver better results while reducing their cost base.

The Must-have Reference Guide for SAFe® Professionals "There are a lot of methods of scale out there, but the Scaled Agile Framework is the one lighting up the world." –Steve Elliot, Founder/CEO AgileCraft "Since beginning our Lean-Agile journey with SAFe, Vantiv has focused its strategic efforts and its execution. We have improved the predictability of product delivery while maintaining high quality, and have become even more responsive to customers—resulting in higher customer satisfaction. And just as important, employee engagement went up over the past year." –Dave Kent, Enterprise Agile Coach, Vantiv Fully updated to include the new innovations in SAFe 4.5, the SAFe® 4.5 Reference Guide is ideal for anyone serious about learning and implementing the world's leading framework for enterprise agility. Inside, you'll find complete coverage of the scaledagileframework.com knowledge base, the website that thousands of the world's largest brands turn to for building better software and systems. SAFe was developed from real-world field experience and provides proven success patterns for implementing Lean-Agile software and systems development at enterprise scale. This book provides comprehensive guidance for work at the enterprise Portfolio, Large Solution, Program, and Team levels, including the various roles, activities, and artifacts that constitute the Framework. Education & Training Key to Success The practice of SAFe is spreading rapidly throughout the world. The majority of Fortune 100 companies have certified SAFe professionals and consultants, as do an increasing percentage of the Global 2000. Case study results—visit scaledagileframework.com/case-studies—typically include: 30 — 75% faster time-to-market 25 — 75% increase in productivity 20 — 50% improvements in quality 10 — 50% increased employee engagement Successful implementations may vary in context but share a common attribute: a workforce well trained and educated in SAFe practices. This book—along with authorized training and certification—will help you understand how to maximize the value of your role within a SAFe organization. The result is greater alignment and visibility, improved performance throughout the enterprise, and ultimately better outcomes for the business.

Today, even the largest development organizations are turning to agile methodologies, seeking major productivity and quality improvements. However, large-scale agile development is difficult, and publicly available case studies have been scarce. Now, three agile pioneers at Hewlett-Packard present a candid, start-to-finish insider's look at how they've succeeded with agile in one of the company's most mission-critical software environments: firmware for HP LaserJet printers. This book tells the story of an extraordinary experiment and journey. Could agile principles be applied to re-architect an enormous legacy code base? Could agile enable both timely delivery and ongoing innovation? Could it really be applied to 400+ developers distributed across four states, three continents, and four business units? Could it go beyond delivering incremental gains, to meet the stretch goal of 10x developer productivity improvements? It could, and it did—but getting there was not easy. Writing for both managers and technologists, the authors candidly discuss both their successes and failures, presenting actionable lessons for other development organizations, as well as approaches that have proven themselves repeatedly in HP's challenging environment. They not only illuminate the potential benefits of agile in large-scale development, they also systematically show how these benefits can actually be achieved. Coverage includes: • Tightly linking agile methods and enterprise architecture with business objectives • Focusing agile practices on your worst development pain points to get the most bang for your buck • Abandoning classic agile methods that don't work at the largest scale • Employing agile methods to establish a new architecture • Using metrics as a "conversation starter" around agile process improvements • Leveraging continuous integration and quality systems to reduce costs, accelerate schedules, and automate the delivery pipeline • Taming the planning beast with "light-touch" agile planning and lightweight long-range forecasting • Implementing effective project management and ensuring accountability in large agile projects • Managing tradeoffs associated with key decisions about organizational structure • Overcoming U.S./India cultural differences that can complicate offshore development • Selecting tools to support quantum leaps in productivity in your organization • Using change management disciplines to support greater enterprise agility

Agile Values and Principles for a New Generation "In the journey to all things Agile, Uncle Bob has been there, done that, and has the both the t-shirt and the scars to show for it. This

delightful book is part history, part personal stories, and all wisdom. If you want to understand what Agile is and how it came to be, this is the book for you.” –Grady Booch “Bob’s frustration colors every sentence of Clean Agile, but it’s a justified frustration. What is in the world of Agile development is nothing compared to what could be. This book is Bob’s perspective on what to focus on to get to that ‘what could be.’ And he’s been there, so it’s worth listening.” –Kent Beck “It’s good to read Uncle Bob’s take on Agile. Whether just beginning, or a seasoned Agilista, you would do well to read this book. I agree with almost all of it. It’s just some of the parts make me realize my own shortcomings, dammit. It made me double-check our code coverage (85.09%).” –Jon Kern Nearly twenty years after the Agile Manifesto was first presented, the legendary Robert C. Martin (“Uncle Bob”) reintroduces Agile values and principles for a new generation—programmers and nonprogrammers alike. Martin, author of Clean Code and other highly influential software development guides, was there at Agile’s founding. Now, in Clean Agile: Back to Basics, he strips away misunderstandings and distractions that over the years have made it harder to use Agile than was originally intended. Martin describes what Agile is in no uncertain terms: a small discipline that helps small teams manage small projects . . . with huge implications because every big project is comprised of many small projects. Drawing on his fifty years’ experience with projects of every conceivable type, he shows how Agile can help you bring true professionalism to software development. Get back to the basics—what Agile is, was, and should always be Understand the origins, and proper practice, of SCRUM Master essential business-facing Agile practices, from small releases and acceptance tests to whole-team communication Explore Agile team members’ relationships with each other, and with their product Rediscover indispensable Agile technical practices: TDD, refactoring, simple design, and pair programming Understand the central roles values and craftsmanship play in your Agile team’s success If you want Agile’s true benefits, there are no shortcuts: You need to do Agile right. Clean Agile: Back to Basics will show you how, whether you’re a developer, tester, manager, project manager, or customer. Register your book for convenient access to downloads, updates, and/or corrections as they become available. See inside book for details.

Utilize a new layers-based development model for embedded systems using Agile techniques for software architecture and management. Firmware is comprised of both hardware and software, but the applicability of Agile in embedded systems development is new. This book provides a step-by-step process showing how this is possible. The book details how the moving parts in embedded systems development affect one another and shows how to properly use both engineering tools and new tools and methods to reduce waste, rework, and product time-to-market. Software is seen not as a commodity but a conduit to facilitate valuable product knowledge flow across the company into the hands of the customer. Embedded Systems Architecture for Agile Development starts off by reviewing the Layers model used in other engineering disciplines, as well as its advantages and applicability to embedded systems development. It outlines development models from project-based methodologies (e.g., collaborative product development) to the newer modern development visions (e.g., Agile) in software and various tools and methods that can help with a Layers model implementation. The book covers requirement modeling for embedded systems (Hatley-Pirbhai Method) and how adapting the HP Method with the help of the tools discussed in this book can be seen as a practical example for a complete embedded system. What You’ll Learn Identify the major software parts involved in building a typical modern firmware Assign a layer to each software part so each layer can be separate from another and there won’t be interdependencies between them Systematically and logically create these layers based on the customer requirements Use Model-Based Design (MBD) to create an active system architecture that is more accepting of changes Who This Book Is For Firmware engineers; systems architects; hardware and software managers, developers, designers, and architects; program managers; project managers; Agile practitioners; and manufacturing engineers and managers. The secondary audience includes research engineers and managers, and engineering and manufacturing managers.

Lean Software Development: An Agile Toolkit Adapting agile practices to your development organization Uncovering and eradicating waste throughout the software development lifecycle Practical techniques for every development manager, project manager, and technical leader Lean software development: applying agile principles to your organization In Lean Software Development, Mary and Tom Poppendieck identify seven fundamental "lean" principles, adapt them for the world of software development, and show how they can serve as the foundation for agile development approaches that work. Along the way, they introduce 22 "thinking tools" that can help you customize the right agile practices for any environment. Better, cheaper, faster software development. You can have all three—if you adopt the same lean principles that have already revolutionized manufacturing, logistics and product development. Iterating towards excellence: software development as an exercise in discovery Managing uncertainty: "decide as late as possible" by building change into the system. Compressing the value stream: rapid development, feedback, and improvement Empowering teams and individuals without compromising coordination Software with integrity: promoting coherence, usability, fitness, maintainability, and adaptability How to "see the whole"—even when your developers are scattered across multiple locations and contractors Simply put, Lean Software Development helps you refocus development on value, flow, and people—so you can achieve breakthrough quality, savings, speed, and business alignment.

“Companies have been implementing large agile projects for a number of years, but the ‘stigma’ of ‘agile only works for small projects’ continues to be a frequent barrier for newcomers and a rallying cry for agile critics. What has been missing from the agile literature is a solid, practical book on the specifics of developing large projects in an agile way. Dean Leffingwell’s book Scaling Software Agility fills this gap admirably. It offers a practical guide to large project issues such as architecture, requirements development, multi-level release planning, and team organization. Leffingwell’s book is a necessary guide for large projects and large organizations making the transition to agile development.” —Jim Highsmith, director, Agile Practice, Cutter Consortium, author of Agile Project Management “There’s tension between building software fast and delivering software that lasts, between being ultra-responsive to changes in the market and maintaining a degree of stability. In his latest work, Scaling Software Agility, Dean Leffingwell shows how to achieve a pragmatic balance among these forces. Leffingwell’s observations of the problem, his advice on the solution, and his description of the resulting best practices come from experience: he’s been there, done that, and has seen what’s worked.” —Grady Booch, IBM Fellow Agile development practices, while still controversial in some circles, offer undeniable benefits: faster time to market, better responsiveness to changing customer requirements, and higher quality. However, agile practices have been defined and recommended primarily to small teams. In Scaling Software Agility, Dean Leffingwell describes how agile methods can be applied to enterprise-class development. Part I provides an overview of the most common and effective agile methods. Part II describes seven best practices of agility that natively scale to the enterprise level. Part III describes an additional set of seven organizational capabilities that companies can master to achieve the full benefits of software agility on an enterprise scale. This

book is invaluable to software developers, testers and QA personnel, managers and team leads, as well as to executives of software organizations whose objective is to increase the quality and productivity of the software development process but who are faced with all the challenges of developing software on an enterprise scale.

Open Agile Architecture™, a standard of The Open Group, offers an approach to architect at scale with agility. It provides guidance and best practices for Enterprise Architects seeking to transition into Agile and Digital contexts. Empowering an Enterprise to Succeed with its Digital-Agile Transformation Agile teams drive the enterprise's Digital Transformation by inventing new business models, delivering superior customer experiences, developing digital products, and architecting highly-automated operating systems. The Open Agile Architecture Standard was designed keeping the needs of all business stakeholders in mind: Business Leaders – to drive the enterprise's Digital and Agile change journey Enterprise Architects – to extend their scope of influence in an Agile at scale world Product Managers – to help transform customer experience, innovate products, and generate growth Product Owners – to accelerate their transformation from managing feature backlogs to steering value delivery Operations Managers – to enable them to leverage Lean and automation to generate sustainable competitive advantages Software Engineers – to leverage the power of digital technologies to co-innovate with the business The more Agile the enterprise, the faster the learning cycles, and faster learning cycles translate to shorter time-to-market resulting in more agility. By adopting an Open Agile Architecture approach, your organization can capitalize on this accelerated learning cycle, meaning your Agile and Digital capabilities continuously and simultaneously co-create one another.

A software architecture manifests the major early design decisions, which determine the system's development, deployment and evolution. Thus, making better architectural decisions is one of the large challenges in software engineering. Software architecture knowledge management is about capturing practical experience and translating it into generalized architectural knowledge, and using this knowledge in the communication with stakeholders during all phases of the software lifecycle. This book presents a concise description of knowledge management in the software architecture discipline. It explains the importance of sound knowledge management practices for improving software architecture processes and products, and makes clear the role of knowledge management in software architecture and software development processes. It presents many approaches that are in use in software companies today, approaches that have been used in other domains, and approaches under development in academia. After an initial introduction by the editors, the contributions are grouped in three parts on "Architecture Knowledge Management", "Strategies and Approaches for Managing Architectural Knowledge", and "Tools and Techniques for Managing Architectural Knowledge". The presentation aims at information technology and software engineering professionals, in particular software architects and software architecture researchers. For the industrial audience, the book gives a broad and concise understanding of the importance of knowledge management for improving software architecture process and building capabilities in designing and evaluating better architectures for their mission- and business-critical systems. For researchers, the book will help to understand the applications of various knowledge management approaches in an industrial setting and to identify research challenges and opportunities.

For software to consistently deliver promised results, software development must mature into a true profession. Emergent Design points the way. As software continues to evolve and mature, software development processes become more complicated, relying on a variety of methodologies and approaches. This book illuminates the path to building the next generation of software. Author Scott L. Bain integrates the best of today's most important development disciplines into a unified, streamlined, realistic, and fully actionable approach to developing software. Drawing on patterns, refactoring, and test-driven development, Bain offers a blueprint for moving efficiently through the entire software lifecycle, smoothly managing change, and consistently delivering systems that are robust, reliable, and cost-effective. Reflecting a deep understanding of the natural flow of system development, Emergent Design helps developers work with the flow, instead of against it. Bain introduces the principles and practices of emergent design one step at a time, showing how to promote the natural evolution of software systems over time, making systems work better and provide greater value. To illuminate his approach, Bain presents code examples wherever necessary and concludes with a complete project case study. This book provides developers, project leads, and testers powerful new ways to collaborate, achieve immediate goals, and build systems that improve in quality with each iteration. Coverage includes How to design software in a more natural, evolutionary, and professional way How to use the "open-closed" principle to mitigate risks and eliminate waste How and when to test your design throughout the development process How to translate design principles into practices that actually lead to better code How to determine how much design is enough How refactoring can help you reduce over-design and manage change more effectively The book's companion Web site, www.netobjectives.com/resources, provides updates, links to related materials, and support for discussions of the book's content.

"We need better approaches to understanding and managing software requirements, and Dean provides them in this book. He draws ideas from three very useful intellectual pools: classical management practices, Agile methods, and lean product development. By combining the strengths of these three approaches, he has produced something that works better than any one in isolation." –From the Foreword by Don Reinertsen, President of Reinertsen & Associates; author of *Managing the Design Factory*; and leading expert on rapid product development Effective requirements discovery and analysis is a critical best practice for serious application development. Until now, however, requirements and Agile methods have rarely coexisted peacefully. For many enterprises considering Agile approaches, the absence of effective and scalable Agile requirements processes has been a showstopper for Agile adoption. In *Agile Software Requirements*, Dean Leffingwell shows exactly how to create effective requirements in Agile environments. Part I presents the "big picture" of Agile requirements in the enterprise, and describes an overall process model for Agile requirements at the project team, program, and portfolio levels Part II describes a simple and lightweight, yet comprehensive model that Agile project teams can use to manage requirements Part III shows how to develop Agile requirements for complex systems that require the cooperation of multiple teams Part IV guides enterprises in developing Agile requirements for ever-larger "systems of systems," application suites, and product portfolios This book will help you leverage the benefits of Agile without sacrificing the value of effective requirements discovery and analysis. You'll find proven solutions you can apply right now—whether you're a software developer or tester, executive, project/program manager, architect, or team leader.

Shipping imperfect software is like going into debt. When you incur debt, the illusion of doing things faster can lead to exponential growth in the cost of maintaining software. Software debt

takes five major forms: technical, quality, configuration management, design, and platform experience. In today's rush to market, software debt is inevitable. And that's okay—if you're careful about the debt you incur, and if you quickly pay it back. In *Managing Software Debt*, leading Agile expert Chris Sterling shows how understanding software debt can help you move products to market faster, with a realistic plan for refactoring them based on experience. Writing for all Agile software professionals, Sterling explains why you're going into software debt whether you know it or not—and why the interest on that debt can bring projects to a standstill. Next, he thoroughly explains each form of software debt, showing how to plan for it intelligently and repay it successfully. You'll learn why accepting software debt is not the same as deliberate sloppiness, and you'll learn how to use the software debt concept to systematically improve architectural agility. Coverage includes Managing tensions between speed and perfection and recognizing that you'll inevitably ship some "not quite right" code Planning to minimize interest payments by paying debts quickly Building architectures that respond to change and help enterprises run more smoothly Incorporating emergent architecture concepts into daily activities, using Agile collaboration and refactoring techniques Delivering code and other software internals that reduce the friction of future change Using early, automated testing to move past the "break/fix" mentality Scripting and streamlining both deployment and rollback Implementing team configuration patterns and knowledge sharing approaches that make software debt easier to repay Clearing away technical impediments in existing architectures Using the YAGNI ("you ain't gonna need it") approach to strip away unnecessary complexity Using this book's techniques, senior software leadership can deliver more business value; managers can organize and support development teams more effectively; and teams and team members can improve their performance throughout the development lifecycle.

The Must-have Reference Guide for SAFe® Practitioners "There are a lot of methods of scale out there, but the Scaled Agile Framework is the one lighting up the world." –Steve Elliot, Founder/CEO AgileCraft "You don't have to be perfect to start SAFe because you learn as you go—learning is built in. Before SAFe, I would not know how to help my teams but now I have many tools to enable the teams. My job is really fun and the bottom line is I have never enjoyed my job more!" –Product Manager, Fortune 500 Enterprise Captured for the first time in print, the SAFe body of knowledge is now available as a handy desktop reference to help you accomplish your mission of building better software and systems. Inside, you'll find complete coverage of what has, until now, only been available online at scaledagileframework.com. The SAFe knowledge base was developed from real-world field experience and provides proven success patterns for implementing Lean-Agile software and systems development at enterprise scale. This book provides comprehensive guidance for work at the enterprise Portfolio, Value Stream, Program, and Team levels, including the various roles, activities, and artifacts that constitute the Framework, along with the foundational elements of values, mindset, principles, and practices. Education & Training Key to Success The practice of SAFe is spreading rapidly throughout the world. The majority of Fortune 100 U.S. companies have certified SAFe practitioners and consultants, as do an increasing percentage of the Global 1000 enterprises. Case study results—visit scaledagileframework.com/case-studies—typically include: 20—50% increase in productivity 50%+ increases in quality 30—75% faster time to market Measurable increases in employee engagement and job satisfaction With results like these, the demand from enterprises seeking SAFe expertise is accelerating at a dramatic rate. Successful implementations may vary in context, but share a common attribute: a workforce well trained and educated in SAFe practices. This book—along with authorized training and certification—will help you understand how to maximize the value of your role within a SAFe organization. The result is greater alignment, visibility, improved performance throughout the enterprise, and ultimately better outcomes for the business.

This open access book constitutes the proceedings of the 19th International Conference on Agile Software Development, XP 2018, held in Porto, Portugal, in May 2018. XP is the premier agile software development conference combining research and practice, and XP 2018 provided a playful and informal environment to learn and trigger discussions around its main theme – make, inspect, adapt. The 21 papers presented in this volume were carefully reviewed and selected from 62 submissions. They were organized in topical sections named: agile requirements; agile testing; agile transformation; scaling agile; human-centric agile; and continuous experimentation.

Object technology pioneer Wirfs-Brock teams with expert McKean to present a thoroughly updated, modern, and proven method for the design of software. The book is packed with practical design techniques that enable the practitioner to get the job done.

A guide to successfully operating in a lean-agile organization for solutions architects and enterprise architects Key Features Develop the right combination of processes and technical excellence to address architectural challenges Explore a range of architectural techniques to modernize legacy systems Discover how to design and continuously improve well-architected sustainable software Book Description Many organizations have embraced Agile methodologies to transform their ability to rapidly respond to constantly changing customer demands. However, in this melee, many enterprises often neglect to invest in architects by presuming architecture is not an intrinsic element of Agile software development. Since the role of an architect is not pre-defined in Agile, many organizations struggle to position architects, often resulting in friction with other roles or a failure to provide a clear learning path for architects to be productive. This book guides architects and organizations through new Agile ways of incrementally developing the architecture for delivering an uninterrupted, continuous flow of values that meets customer needs. You'll explore various aspects of Agile architecture and how it differs from traditional architecture. The book later covers Agile architects' responsibilities and how architects can add significant value by positioning themselves appropriately in the Agile flow of work. Through examples, you'll also learn concepts such as architectural decision backlog, the last responsible moment, value delivery, architecting for change, DevOps, and evolutionary collaboration. By the end of this Agile book, you'll be able to operate as an architect in Agile development initiatives and successfully architect reliable software systems. What you will learn Acquire clarity on the duties of architects in Agile development Understand architectural styles such as domain-driven design and microservices Identify the pitfalls of traditional architecture and learn how to develop solutions Understand the principles of value and data-driven architecture Discover DevOps and continuous delivery from an architect's perspective Adopt Lean-Agile documentation and governance Develop a set of personal and interpersonal qualities Find out how to lead the transformation to achieve organization-wide agility Who this book is for This agile study guide is for architects currently working on agile development projects or aspiring to work on agile software delivery, irrespective of the methodology they are using. You will also find this book useful if you're a senior developer or a budding architect looking to understand an agile architect's role by embracing agile architecture strategies and a lean-agile mindset. To understand the concepts covered in this book easily, you need to have prior

knowledge of basic agile development practices.

User experience (UX) design has traditionally been a deliverables-based practice, with wireframes, site maps, flow diagrams, and mockups. But in today's web-driven reality, orchestrating the entire design from the get-go no longer works. This hands-on book demonstrates Lean UX, a deeply collaborative and cross-functional process that lets you strip away heavy deliverables in favor of building shared understanding with the rest of the product team. Lean UX is the evolution of product design; refined through the real-world experiences of companies large and small, these practices and principles help you maintain daily, continuous engagement with your teammates, rather than work in isolation. This book shows you how to use Lean UX on your own projects. Get a tactical understanding of Lean UX—and how it changes the way teams work together Frame a vision of the problem you're solving and focus your team on the right outcomes Bring the designer's tool kit to the rest of your product team Break down the silos created by job titles and learn to trust your teammates Improve the quality and productivity of your teams, and focus on validated experiences as opposed to deliverables/documents Learn how Lean UX integrates with Agile UX

This is the eagerly-anticipated revision to one of the seminal books in the field of software architecture which clearly defines and explains the topic.

Continuous Architecture provides a broad architectural perspective for continuous delivery, and describes a new architectural approach that supports and enables it. As the pace of innovation and software releases increases, IT departments are tasked to deliver value quickly and inexpensively to their business partners. With a focus on getting software into end-users hands faster, the ultimate goal of daily software updates is in sight to allow teams to ensure that they can release every change to the system simply and efficiently. This book presents an architectural approach to support modern application delivery methods and provide a broader architectural perspective, taking architectural concerns into account when deploying agile or continuous delivery approaches. The authors explain how to solve the challenges of implementing continuous delivery at the project and enterprise level, and the impact on IT processes including application testing, software deployment and software architecture. Covering the application of enterprise and software architecture concepts to the Agile and Continuous Delivery models Explains how to create an architecture that can evolve with applications Incorporates techniques including refactoring, architectural analysis, testing, and feedback-driven development Provides insight into incorporating modern software development when structuring teams and organizations

Ever-changing business needs have prompted large companies to rethink their enterprise IT. Today, businesses must allow interaction with their customers, partners, and employees at more touch points and at a depth never thought previously. At the same time, rapid advances in information technologies, like business digitization, cloud computing, and Web 2.0, demand fundamental changes in the enterprises' management practices. These changes have a drastic effect not only on IT and business, but also on policies, processes, and people. Many companies therefore embark on enterprise-wide transformation initiatives. The role of Enterprise Architecture (EA) is to architect and supervise this transformational journey. Unfortunately, today's EA is often a ponderous and detached exercise, with most of the EA initiatives failing to create visible impact. The enterprises need an EA that is agile and responsive to business dynamics. Collaborative Enterprise Architecture provides the innovative solutions today's enterprises require, informed by real-world experiences and experts' insights. This book, in its first part, provides a systematic compendium of the current best practices in EA, analyzes current ways of doing EA, and identifies its constraints and shortcomings. In the second part, it leaves the beaten tracks of EA by introducing Lean, Agile, and Enterprise 2.0 concepts to the traditional EA methods. This blended approach to EA focuses on practical aspects, with recommendations derived from real-world experiences. A truly thought provoking and pragmatic guide to manage EA, Collaborative Enterprise Architecture effectively merges the long-term oriented top-down approach with pragmatic bottom-up thinking, and that way offers real solutions to businesses undergoing enterprise-wide change. Covers the latest emerging technologies affecting business practice, including digitization, cloud computing, agile software development, and Web 2.0 Focuses on the practical implementation of EAM rather than theory, with recommendations based on real-world case studies Addresses changing business demands and practices, including Enterprise 2.0, open source, global sourcing, and more Takes an innovative approach to EAM, merging standard top-down and pragmatic, bottom-up strategies, offering real solutions to businesses undergoing enterprise-wide changes

"This remarkable book combines practical advice, ready-to-use techniques, and a deep understanding of why this is the right way to develop software. I have seen software teams transformed by the ideas in this book." --Mike Cohn, author of Agile Estimating and Planning "As a lean practitioner myself, I have loved and used their first book for years. When this second book came out, I was delighted that it was even better. If you are interested in how lean principles can be useful for software development organizations, this is the book you are looking for. The Poppendiecks offer a beautiful blend of history, theory, and practice." --Alan Shalloway, coauthor of Design Patterns Explained "I've enjoyed reading the book very much. I feel it might even be better than the first lean book by Tom and Mary, while that one was already exceptionally good! Mary especially has a lot of knowledge related to lean techniques in product development and manufacturing. It's rare that these techniques are actually translated to software. This is something no other book does well (except their first book)." --Bas Vodde "The new book by Mary and Tom Poppendieck provides a well-written and comprehensive introduction to lean principles and selected practices for software managers and engineers. It illustrates the application of the values and practices with well-suited success stories. I enjoyed reading it." --Roman Pichler "In Implementing Lean Software Development, the Poppendiecks explore more deeply the themes they introduced in Lean Software Development. They begin with a compelling history of lean thinking, then move to key areas such as value, waste, and people. Each chapter includes exercises to help you apply key points. If you want a better

understanding of how lean ideas can work with software, this book is for you." --Bill Wake, independent consultant In 2003, Mary and Tom Poppendieck's Lean Software Development introduced breakthrough development techniques that leverage Lean principles to deliver unprecedented agility and value. Now their widely anticipated sequel and companion guide shows exactly how to implement Lean software development, hands-on. This new book draws on the Poppendiecks' unparalleled experience helping development organizations optimize the entire software value stream. You'll discover the right questions to ask, the key issues to focus on, and techniques proven to work. The authors present case studies from leading-edge software organizations, and offer practical exercises for jumpstarting your own Lean initiatives. Managing to extend, nourish, and leverage agile practices Building true development teams, not just groups Driving quality through rapid feedback and detailed discipline Making decisions Just-in-Time, but no later Delivering fast: How PatientKeeper delivers 45 rock-solid releases per year Making tradeoffs that really satisfy customers Implementing Lean Software Development is indispensable to anyone who wants more effective development processes--managers, project leaders, senior developers, and architects in enterprise IT and software companies alike.

This book constitutes the proceedings of the 5th International Conference on Lean and Agile Software Development, LASD 2021, which was held online on January 23, 2021. The conference received a total of 32 submissions, of which 10 full and 2 short papers are included in this volume. In addition, one keynote paper is also included. To live the agile mindset, the LASD conference focuses on highly relevant research outcomes and fosters their way into practice. Topics discussed in this volume range from teams under COVID-19 through women in Agile, to product road-mapping and non-functional requirements.

Lean Architecture for Agile Software Development John Wiley & Sons

Agile software development approaches have had significant impact on industrial software development practices. Today, agile software development has penetrated to most IT companies across the globe, with an intention to increase quality, productivity, and profitability. Comprehensive knowledge is needed to understand the architectural challenges involved in adopting and using agile approaches and industrial practices to deal with the development of large, architecturally challenging systems in an agile way. Agile Software Architecture focuses on gaps in the requirements of applying architecture-centric approaches and principles of agile software development and demystifies the agile architecture paradox. Readers will learn how agile and architectural cultures can co-exist and support each other according to the context. Moreover, this book will also provide useful leads for future research in architecture and agile to bridge such gaps by developing appropriate approaches that incorporate architecturally sound practices in agile methods. Presents a consolidated view of the state-of-art and state-of-practice as well as the newest research findings Identifies gaps in the requirements of applying architecture-centric approaches and principles of agile software development and demystifies the agile architecture paradox Explains whether or not and how agile and architectural cultures can co-exist and support each other depending upon the context Provides useful leads for future research in both architecture and agile to bridge such gaps by developing appropriate approaches, which incorporate architecturally sound practices in agile methods

Methods for managing complex software construction following the practices, principles and patterns of Domain-Driven Design with code examples in C# This book presents the philosophy of Domain-Driven Design (DDD) in a down-to-earth and practical manner for experienced developers building applications for complex domains. A focus is placed on the principles and practices of decomposing a complex problem space as well as the implementation patterns and best practices for shaping a maintainable solution space. You will learn how to build effective domain models through the use of tactical patterns and how to retain their integrity by applying the strategic patterns of DDD. Full end-to-end coding examples demonstrate techniques for integrating a decomposed and distributed solution space while coding best practices and patterns advise you on how to architect applications for maintenance and scale. Offers a thorough introduction to the philosophy of DDD for professional developers Includes masses of code and examples of concept in action that other books have only covered theoretically Covers the patterns of CQRS, Messaging, REST, Event Sourcing and Event-Driven Architectures Also ideal for Java developers who want to better understand the implementation of DDD

The software development ecosystem is constantly changing, providing a constant stream of new tools, frameworks, techniques, and paradigms. Over the past few years, incremental developments in core engineering practices for software development have created the foundations for rethinking how architecture changes over time, along with ways to protect important architectural characteristics as it evolves. This practical guide ties those parts together with a new way to think about architecture and time.

This business parable reviews two different systems development projects. One project was an abject, expensive failure, while the other succeeded in creating a major new revenue stream, bringing in new customers. By reviewing the tales of these two systems, readers will develop a better understanding of what works and what doesn't when it comes to the leadership and action steps required to reinvent a company's procedures to get in step with the times. CEO Evan Nogelmeyer discovers to his dismay that in today's business world, technology is not just for technologists. But does he discover this soon enough and once he does, does he have the tools and the business savvy he needs to stave off disaster? Evan and his team are all well-intentioned, successful business leaders with advanced degrees and backgrounds in marketing and business. But, without technical backgrounds, do they have what it takes to manage the technology overhaul so critical to the very survival of their company and the future of their own careers? A Tale of Two Systems: Lean and Agile Software Development for Business Leaders reviews two fictional systems development projects: Cremins United and Troubled Real Estate Information Management, both launched at the imaginary Cremins Corporation. Cremins is a venerable printing company that must transform itself to survive in the Internet age. One project proves to be an abject and expensive failure, while the other succeeds in creating a major new revenue stream and solving important customer needs. Contrasting the methods employed in a traditional, process-centric 'waterfall' approach, with a lean and agile-inspired approach, this book provides business leaders with a tangible understanding of why lean thinking is so well-suited to contemporary environments requiring flexibility, speed, and the input of specialized

knowledge. At the conclusion of the two tales, author Michael Levine articulates a series of conclusions and principles based on Lean Product Development, Agile, and his 25 years of experience in business systems development. While the tales told and the companies and employees that inhabit them are pure fiction, the lessons to be learned are very real and very applicable in today's highly competitive market, where victory goes time and time again to the lean and the agile.

Xanpan is... a cross between XP and Kanban... is an example of a roll-your-own method... is distilled from Allan Kelly's own experiences running development teams and then helping multiple teams adopt Agile working methods and practices. Xanpan draws ideas from Kanban and Lean, XP and Scrum, product management and business analysis, and many other places. Allan tells the Xanpan story through a series of boards which tell the story of different teams. In between he fills in the principles, practices and thinking which together constitutes Xanpan. Each printed copy contains a code entitling the buyer to a free copy of the electronic version and subsequent updates.

Building a successful product usually involves teams of people, and many choose the Scrum approach to aid in creating products that deliver the highest possible value. Implementing Scrum gives teams a collection of powerful ideas they can assemble to fit their needs and meet their goals. The ninety-four patterns contained within are elaborated nuggets of insight into Scrum's building blocks, how they work, and how to use them. They offer novices a roadmap for starting from scratch, yet they help intermediate practitioners fine-tune or fortify their Scrum implementations. Experienced practitioners can use the patterns and supporting explanations to get a better understanding of how the parts of Scrum complement each other to solve common problems in product development. The patterns are written in the well-known Alexandrian form, whose roots in architecture and design have enjoyed broad application in the software world. The form organizes each pattern so you can navigate directly to organizational design tradeoffs or jump to the solution or rationale that makes the solution work. The patterns flow together naturally through the context sections at their beginning and end. Learn everything you need to know to master and implement Scrum one step at a time—the agile way.

The book covers the best practices and approaches for software architects to follow when developing .NET and C# solutions, along with the most up to date cloud environments and tools to enable effective app development, delivery, and deployment.

Master IBM's Breakthrough DAD Process Framework for Succeeding with Agile in Large, Complex, Mission-Critical IT Projects It is widely recognized that moving from traditional to agile approaches to build software solutions is a critical source of competitive advantage. Mainstream agile approaches that are indeed suitable for small projects require significant tailoring for larger, complex enterprise projects. In Disciplined Agile Delivery, Scott W. Ambler and Mark Lines introduce IBM's breakthrough Disciplined Agile Delivery (DAD) process framework, which describes how to do this tailoring. DAD applies a more disciplined approach to agile development by acknowledging and dealing with the realities and complexities of a portfolio of interdependent program initiatives. Ambler and Lines show how to extend Scrum with supplementary agile and lean strategies from Agile Modeling (AM), Extreme Programming (XP), Kanban, Unified Process (UP), and other proven methods to provide a hybrid approach that is adaptable to your organization's unique needs. They candidly describe what practices work best, why they work, what the trade-offs are, and when to consider alternatives, all within the context of your situation. Disciplined Agile Delivery addresses agile practices across the entire lifecycle, from requirements, architecture, and development to delivery and governance. The authors show how these best-practice techniques fit together in an end-to-end process for successfully delivering large, complex systems--from project initiation through delivery. Coverage includes Scaling agile for mission-critical enterprise endeavors Avoiding mistakes that drive poorly run agile projects to chaos Effectively initiating an agile project Transitioning as an individual to agile Incrementally building consumable solutions Deploying agile solutions into complex production environments Leveraging DevOps, architecture, and other enterprise disciplines Adapting your governance strategy for agile projects Based on facts, research, and extensive experience, this book will be an indispensable resource for every enterprise software leader and practitioner--whether they're seeking to optimize their existing agile/Scrum process or improve the agility of an iterative process.

Part I: Process design -- Introduction to design -- Process flowsheet development -- Utilities and energy efficient design -- Process simulation -- Instrumentation and process control -- Materials of construction -- Capital cost estimating -- Estimating revenues and production costs -- Economic evaluation of projects -- Safety and loss prevention -- General site considerations -- Optimization in design -- Part II: Plant design -- Equipment selection, specification and design -- Design of pressure vessels -- Design of reactors and mixers -- Separation of fluids -- Separation columns (distillation, absorption and extraction) -- Specification and design of solids-handling equipment -- Heat transfer equipment -- Transport and storage of fluids.

[Copyright: 9282835f23f52851446ab73e2da08830](https://www.amazon.com/dp/9282835f23f52851446ab73e2da08830)