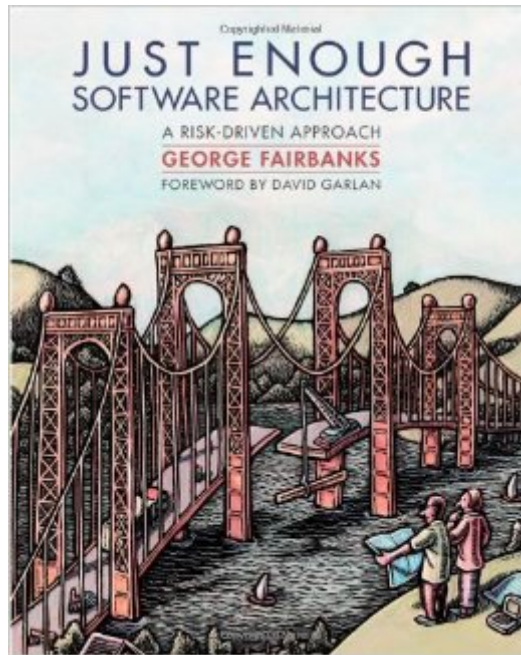


The book was found

Just Enough Software Architecture: A Risk-Driven Approach



Synopsis

This is a practical guide for software developers, and different than other software architecture books. Here's why: It teaches risk-driven architecting. There is no need for meticulous designs when risks are small, nor any excuse for sloppy designs when risks threaten your success. This book describes a way to do just enough architecture. It avoids the one-size-fits-all process tar pit with advice on how to tune your design effort based on the risks you face. It democratizes architecture. This book seeks to make architecture relevant to all software developers. Developers need to understand how to use constraints as guiderails that ensure desired outcomes, and how seemingly small changes can affect a system's properties. It cultivates declarative knowledge. There is a difference between being able to hit a ball and knowing why you are able to hit it, what psychologists refer to as procedural knowledge versus declarative knowledge. This book will make you more aware of what you have been doing and provide names for the concepts. It emphasizes the engineering. This book focuses on the technical parts of software development and what developers do to ensure the system works not job titles or processes. It shows you how to build models and analyze architectures so that you can make principled design tradeoffs. It describes the techniques software designers use to reason about medium to large sized problems and points out where you can learn specialized techniques in more detail. It provides practical advice. Software design decisions influence the architecture and vice versa. The approach in this book embraces drill-down/pop-up behavior by describing models that have various levels of abstraction, from architecture to data structure design.

Book Information

Hardcover: 376 pages

Publisher: Marshall & Brainerd; 1 edition (August 30, 2010)

Language: English

ISBN-10: 0984618104

ISBN-13: 978-0984618101

Product Dimensions: 7.5 x 1.1 x 9.3 inches

Shipping Weight: 2.1 pounds (View shipping rates and policies)

Average Customer Review: 4.3 out of 5 starsÂ Â See all reviewsÂ (31 customer reviews)

Best Sellers Rank: #196,250 in Books (See Top 100 in Books) #49 inÂ Books > Computers &

Technology > Programming > Software Design, Testing & Engineering > Tools #84 inÂ Books >

Computers & Technology > Programming > Software Design, Testing & Engineering > Testing

Customer Reviews

This is a broad and deep book on all things software architecture except the architecting process (for process, one recommended read is "Process Software Architecting" by Eeles/Cripps). Several readerships will benefit from "Just Enough Software Architecture":- As an experienced IT architect, I do not necessarily agree with everything in the book (this does not come as a surprise, as architects have opinions). That said, I certainly learned a lot that I can apply immediately on my projects and some of the more provocative statements challenge me to leave my comfort zone (or at least consider doing so).- Junior architects can use the book both as a tutorial and as a reference when/while growing in their profession.- Developers with a "who needs architects" mindset (hopefully) will understand architects and modelers much better after having read this book, and appreciate the value of architecture. Things I liked in particular:- Overall vision and message of pragmatism sent- The risk-driven approach increases chances to get accepted both in agile development and in more traditional architecture communities- There is a lot of practical advice e.g. in Chapters 10, 11 and 15- The author is in command of a large body of relevant related work (both industry and academia) and puts them in perspective adequately- Editorial quality: structure, figures, command of the English language (some words and expressions a bit hard to comprehend for non-native speaker) Some room for improvements (2nd edition?):- Not all metaphors and analogies work internationally, e.g. not all IT people are sports fans that follow baseball or know what a rookie is- The connection between parts 1 and 2 could be a bit stronger, even if loose coupling generally is a good thing; e.g., some more backward references In summary, I'd say Just Enough Software Architecture is a highly recommended read for every architect in touch with development reality and every developer with a desire to build complex systems that will stand the test of time.

I was brought up in Japan (born in China), and most of my playground was in my aunt's farm house. Many beautiful pen sketches throughout the book bring back my childhood memory. The Old "Edo" period began in 1600 and ended in 1868, but the type of things depicted in the book were seen until just 50 years ago in Japan. So-called modernization, or Americanization, has wiped out eco-friendly, waste-nothing culture completely. I hope people will realize that we can live happily without many modern amenities. I bought 2 additional copies and gave to my American friends in my neighbourhood.

There's many reasons to recommend this book, depending on your interests. The goal of this book is to analyze Edo Japan's sustainable practices and apply them to modern life. It does this by taking a fictionalized journey through the country, starting at a farming village and ending at the home of a lower-level Edo Samurai. As we travel with the narrator, the story points out various things we'd notice and what they mean. Between chapters different levels of society and locations are analyzed for useful lessons. Thus you'll read about the energy-saving virtue of pickling, the value of latrine outputs, the life of traveling city pottery repairman, and samurai who farm on their small estates. It's actually a bit dizzying, and the author packs in a lot - almost a bit too much to be frank, but he's got a lot to cover. This human-level look at a sustainable culture, why it evolved, and what it means is very intriguing and has high impact. Backed by illustrations and research, giving these fictionalized but historical examples of efficiency, good construction, food production, etc. helps one understand what we can learn and apply to our lives. This varies from ethical/personal approaches to serious thoughts about material usage and land. The book will make you think, will help you see the value of history, and will give you ideas. Despite it's many triumphs of stability, efficiency, and literacy, the book doesn't set the Edo period as something to emulate entirely. The Edo period was also a time of social immobility for most, high taxes for farmers, the practice of infanticide for some, and a samurai class whose comparative wealth were constrained by propriety and social policy. The author clearly admires the Edo period, and perhaps in one or two cases praises it more than I feel it deserved, but he also acknowledges its many flaws. He regards these issues with a kind of sad affection, blots on a period that he feels shows important virtues, but blots nonetheless. Do I consider this a book worth reading? I do - for the right audience.

Ecology/Sustainability: If you're interested in sustainability in a historical context, this book will definitely be for you.

Those Interested in Japan: If you're into Japanese food, culture, or history, then you'll probably adore this book. This might also be a great gift for anyone interested in period anime and manga because of the wealth of details.

Writers: This book's way of using fiction, illustration, and analysis is actually very interesting from a pure level of literature. As a writer or instructional designer, if this book sounds at least mildly interesting to you, you may want to get it as a study of a useful instructional writing style. So there you have it. The book's not for everyone, but it's quite a good book for the right people.

[Download to continue reading...](#)

Just Enough Software Architecture: A Risk-Driven Approach
Just Enough Requirements Management: Where Software Development Meets Marketing
How to Retire with Enough Money:
And How to Know What Enough Is
Next Door Savior: Near Enough to Touch, Strong Enough to

Trust Lessons Learned in Software Testing: A Context-Driven Approach The Feeling of Risk: New Perspectives on Risk Perception (Earthscan Risk in Society) What Customers Want: Using Outcome-Driven Innovation to Create Breakthrough Products and Services: Using Outcome-Driven Innovation to Create Breakthrough Products and Services Software Engineering Classics: Software Project Survival Guide/ Debugging the Development Process/ Dynamics of Software Development (Programming/General) Surreptitious Software: Obfuscation, Watermarking, and Tamperproofing for Software Protection: Obfuscation, Watermarking, and Tamperproofing for Software Protection Software Architecture in Practice (3rd Edition) (SEI Series in Software Engineering) Project Management Lite: Just Enough to Get the Job Done...Nothing More Just Enough Light for the Step I'm On: Trusting God in the Tough Times Software Reuse: A Holistic Approach (Wiley Series in Software-Based Systems) Domain-Driven Design: Tackling Complexity in the Heart of Software Model-Driven Software Migration: A Methodology: Reengineering, Recovery and Modernization of Legacy Systems Android 6 for Programmers: An App-Driven Approach (3rd Edition) (Deitel Developer Series) Mastering Data Modeling: A User-Driven Approach Public Relations: A Value Driven Approach (5th Edition) Public Relations: A Value Driven Approach Cost Management and Control in Government: A Proven, Practical Leadership Driven Management Approach to Fighting the Cost War in Government (Managerial Accounting Collection)

[Dmca](#)