

---

# The Rational Unified Process Made Easy A Practitioners Guide To The Rup A Practitioners Guide To The Rup

---

Thank you unconditionally much for downloading **The Rational Unified Process Made Easy A Practitioners Guide To The Rup A Practitioners Guide To The Rup**. Most likely you have knowledge that, people have see numerous period for their favorite books like this The Rational Unified Process Made Easy A Practitioners Guide To The Rup A Practitioners Guide To The Rup, but end stirring in harmful downloads.

Rather than enjoying a good PDF in the same way as a mug of coffee in the afternoon, instead they juggled gone some harmful virus inside their computer. **The Rational Unified Process Made Easy A Practitioners Guide To The Rup A Practitioners Guide To The Rup** is open in our digital library an online admission to it is set as public for that reason you can download it instantly. Our digital library saves in merged countries, allowing you to get the most less latency era to download any of our books subsequently this one. Merely said, the The Rational Unified Process Made Easy A Practitioners Guide To The Rup A Practitioners Guide To The Rup is universally compatible like any devices to read.

*The Rational Unified Process Made Easy A Practitioners Guide To The Rup A Practitioners Guide To The Rup*

2021-07-26

---

## **PATRICK WATSON**

---

Solution Designer (RUP)

John Wiley & Sons

A Comprehensive Process for Defining Software Architectures That Work A good software architecture is the foundation of any successful software system. Effective

architecting requires a clear understanding of organizational roles, artifacts, activities performed, and the optimal sequence for performing those activities. With The Process of Software Architecting , Peter Eeles and Peter Cripps provide guidance on these challenges by covering all aspects of architecting a software system, introducing best-practice techniques that apply in every environment, whether based on Java EE,

Microsoft .NET, or other technologies. Eeles and Cripps first illuminate concepts related to software architecture, including architecture documentation and reusable assets. Next, they present an accessible, task-focused guided tour through a typical project, focusing on the architect's role, with common issues illuminated and addressed throughout. Finally, they conclude with a set of best practices that can be applied to today's most

complex systems. You will come away from this book understanding The role of the architect in a typical software development project How to document a software architecture to satisfy the needs of different stakeholders The applicability of reusable assets in the process of architecting The role of the architect with respect to requirements definition The derivation of an architecture based on a set of requirements The relevance of architecting in creating complex systems The Process of Software Architecting will be an indispensable resource for every working and aspiring software architect—and for every project manager and other software professional who needs to understand how architecture influences their work.

Beginning Software Engineering Project Management Institute

"This book isn't just another introduction to use cases. The authors have used their wealth of experience to produce an excellent and insightful collection of detailed examples, explanations, and advice on how to work with use cases."

–Maria Ericsson The toughest challenge in

building a software system that meets the needs of your audience lies in clearly understanding the problems that the system must solve. Advanced Use Case Modeling presents a framework for discovering, identifying, and modeling the problem that the software system will ultimately solve.

Software developers often employ use cases to specify what should be performed by the system they're constructing.

Although use case-driven analysis, design, and testing of software systems has become increasingly popular, little has been written on the role of use cases in the complete software cycle.

This book fills that need by describing how to create use case models for complex software development projects, using practical examples to explain conceptual information. The authors extend the work of software visionary Ivar Jacobson, using the Unified Modeling Language (UML) as the notation to describe the book's models. Aimed primarily at software professionals, Advanced Use Case Modeling also includes information that relates use case

technique to business processes. This book presents a process for creating and maintaining use case models in a framework that can be fully customized for your organization. The authors, pioneers in the application of use cases in software development, bring their extensive experience to cover topics such as: A process model for applying a use case model How to keep your use case modeling effort on track Tips and pitfalls in use case modeling How to organize your use case model for large-system development Similarities between Advanced Use Case Modeling and the Rational Unified Process framework Effect of use cases on user interface design Guidelines for quality use case modeling Visual Modeling with Rational Rose 2000 and UML Project Management Institute

Keep your cool and get the results you want when faced with crucial conversations. This New York Times bestseller and business classic has been fully updated for a world where skilled communication is more important than ever. The book that revolutionized business communications has been updated for

today's workplace. Crucial Conversations provides powerful skills to ensure every conversation—especially difficult ones—leads to the results you want. Written in an engaging and witty style, the book teaches readers how to be persuasive rather than abrasive, how to get back to productive dialogue when others blow up or clam up, and it offers powerful skills for mastering high-stakes conversations, regardless of the topic or person. This new edition addresses issues that have arisen in recent years. You'll learn how to: Respond when someone initiates a crucial conversation with you Identify and address the lag time between identifying a problem and discussing it Communicate more effectively across digital mediums When stakes are high, opinions vary, and emotions run strong, you have three choices: Avoid a crucial conversation and suffer the consequences; handle the conversation poorly and suffer the consequences; or apply the lessons and strategies of Crucial Conversations and improve relationships and results. Whether they take place at work or at

home, with your coworkers or your spouse, crucial conversations have a profound impact on your career, your happiness, and your future. With the skills you learn in this book, you'll never have to worry about the outcome of a crucial conversation again. *Bureaucratic Democracy* Addison-Wesley Professional John Hunt's book guides you through the use of the UML and the Unified Process and their application to Java systems. Key topics focus explicitly on applying the notation and the method to Java. The book is clearly structured and written, making it ideal for practitioners. This second edition is considerably revised and extended and includes examples taken from the latest version of Rational Rose and Together. Considers how Agile Modelling fits with the Unified Process, and presents Design Patterns Self contained - covers both the Unified Process and UML in one book Includes real-world case studies Written by an experienced author and industry expert Ideal for students on Software Engineering courses **Implementing the IBM Rational Unified**

**Process and Solutions** Pearson Education The authors explain the underlying software development principles behind theRUP, and guide readers in its application in their organization. *A Policy at War With Itself* Addison-Wesley Professional Ship It! is a collection of tips that show the tools and techniques a successful project team has to use, and how to use them well. You'll get quick, easy-to-follow advice on modern practices: which to use, and when they should be applied. This book avoids current fashion trends and marketing hype; instead, readers find page after page of solid advice, all tried and tested in the real world. Aimed at beginning to intermediate programmers, Ship It! will show you: Which tools help, and which don't How to keep a project moving Approaches to scheduling that work How to build developers as well as product What's normal on a project, and what's not How to manage managers, end-users and sponsors Danger signs and how to fix them Few of the ideas presented here are controversial or extreme; most

experienced programmers will agree that this stuff works. Yet 50 to 70 percent of all project teams in the U.S. aren't able to use even these simple, well-accepted practices effectively. This book will help you get started. Ship It! begins by introducing the common technical infrastructure that every project needs to get the job done. Readers can choose from a variety of recommended technologies according to their skills and budgets. The next sections outline the necessary steps to get software out the door reliably, using well-accepted, easy-to-adopt, best-of-breed practices that really work. Finally, and most importantly, Ship It! presents common problems that teams face, then offers real-world advice on how to solve them.

### **Agile Modeling** IGI

Global

Larman covers how to investigate requirements, create solutions and then translate designs into code, showing developers how to make practical use of the most significant recent developments. A summary of UML notation is included

[Software Development for Small Teams](#) Pearson

Education

Essay from the year 2018 in the subject Psychology - Work, Business, Organisational and Economic Psychology, grade: A, University College London, language: English, abstract: Each day we face numerous decisions that determine the outcome of our life to a great extent. Whether we are rich or poor, happy or sad, good or bad - it all comes down to the decisions we make. It is therefore logical to assume that human beings have evolved to be experts in decision making. For decades, Economists, Psychologists and Philosophers have struggled to answer this question by conceptualizing the way people make decision and by coming up with models that explain judgment and decision making. Over the last few centuries, models of rationality have been constantly changing and evolving. In 1654, Blaise Pascal and Pierre Fermat had a prolonged discussion about human rationale in gambling scenarios, giving birth to the rational choice theory. Its main assumption is that humans take into consideration every available information, cost and benefit associated

with a decision and proceed to select the best choice amongst all available alternatives - the one that maximizes Expected Utility. This model, however, is inherently flawed, as it ignores elements such as cognitive biases and mental shortcuts (also known as heuristics) that could lead to deviations from the assumption of perfectly rational decisions. In response to the limitations of the rational choice theory, Herbert Simon proposed the notion of bounded rationality to compensate for flaws in human thinking and to take into consideration environmental constraints when making a decision. Bounded rationality suggests that humans act as satisficers rather than maximizers, aimed at finding a decision that is good enough, taken the amount of information and time that were available for making the decision. This framework applies to any decision making process - including consumer behaviour. Therefore, this essay will argue that consumers are not perfectly rational agents, aimed at maximizing expected utility with each transaction. Rather,

humans have inherent limitations to their information processing abilities leading to the development of cognitive biases and heuristics, which help them make decision faster and more efficient. This ability, however, comes at the cost of making suboptimal decision. To illustrate the point and to compare and contrast both models of rationality, we will use a practical example of a car purchase throughout the essay.

**Project Management with the IBM Rational Unified Process** Pearson Education India  
 The Only Official RUP® Certification Prep Guide and Compact RUP Reference The IBM® Rational Unified Process® has become the de facto industry-standard process for large-scale enterprise software development. The IBM Certified Solution Designer - IBM Rational Unified Process V7.0 certification provides a powerful way for solutions developers to demonstrate their proficiency with RUP. The first and only official RUP certification guide, this book fully reflects the latest versions of the Rational Unified Process and of the IBM RUP exam. Authored by two leading

RUP implementers, it draws on extensive contributions and careful reviews by the IBM RUP process leader and RUP certification manager. This book covers every facet of RUP usage. It has been carefully organized to help you prepare for your exam quickly and efficiently--and to provide a handy, compact reference you can rely on for years to come. Coverage includes A full section on RUP exam preparation and a 52-question practice exam Core RUP concepts, the new RUP process architecture, and key principles of business-driven development RUP's architecture-centric approach to iterative development: practical issues and scenarios Patterns for successful RUP project implementation--and "anti-patterns" to avoid The Unified Method Architecture (UMA): basic content and process elements RUP content disciplines, in depth: Business Modeling, Requirements, Analysis and Design, Implementation, Test, Deployment, Project Management, Change and Configuration Management, and Environment Essential

RUP work products, roles, and tasks RUP phases, activities, and milestones RUP tailoring and tools for your organization-- including introductions to IBM Rational Method Composer (RMC) and MyRUP

**The Confessions of S. Augustine** Pearson Education

Hundreds of organizations around the world have already benefited from Disciplined Agile Delivery (DAD). Disciplined Agile (DA) is the only comprehensive tool kit available for guidance on building high-performance agile teams and optimizing your way of working(WoW). As a hybrid of all the leading agile and lean approaches, it provides hundreds of strategies to help you make better decisions within your agile teams, balancing self-organization with the realities and constraints of your unique enterprise context.The highlights of this handbook include:&•As the official source of knowledge on DAD, it includes greatly improved and enhanced strategies with a revised set of goal diagrams based upon learnings from applying DAD in the field.&•It is an essential handbook to help coaches

and teams make better decisions in their daily work, providing a wealth of ideas for experimenting with agile and lean techniques while providing specific guidance and trade-offs for those &“it depends&” questions.&•It makes a perfect study guide for Disciplined Agile certification.Why &“fail fast&” (as our industry likes to recommend) when you can learn quickly on your journey to high performance? With this handbook, you can make better decisions based upon proven, context-based strategies, leading to earlier success and better outcomes.

*Advanced Use Case Modeling* Apress

“Unified Modeling Language (UML), Unified Process (UP), and other information modeling methods are addressed in this scholarly consideration of the analysis, design, and development of web-based and enterprise applications. The most current research on conceptual, theoretical, and empirical issues of modeling for online business and static information is provided.”

**A Unified Framework**  
Prentice Hall Professional  
PMBOK® Guide is the

go-to resource for project management practitioners. The project management profession has significantly evolved due to emerging technology, new approaches and rapid market changes. Reflecting this evolution, The Standard for Project Management enumerates 12 principles of project management and the PMBOK® Guide &– Seventh Edition is structured around eight project performance domains.This edition is designed to address practitioners' current and future needs and to help them be more proactive, innovative and nimble in enabling desired project outcomes.This edition of the PMBOK® Guide:•Reflects the full range of development approaches (predictive, adaptive, hybrid, etc.);•Provides an entire section devoted to tailoring the development approach and processes;•Includes an expanded list of models, methods, and artifacts;•Focuses on not just delivering project outputs but also enabling outcomes; and• Integrates with PMIstandards+™ for information and standards application content based

on project type, development approach, and industry sector. Lean Requirements Practices for Teams, Programs, and the Enterprise 5starcooks “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 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.

An Introduction to Modern Software Engineering  
GRIN Verlag

I highly recommend this book for anyone who's ever tried to implement RUP on a small project. Pollice and company have demystified and effectively scaled the process while ensuring that its essence hasn't been compromised. A must-have for any RUPster's library! Chris Soskin, Process Engineering Consultant, Toyota Motor Sales Do you want to improve the process on your next project? Perhaps you'd like to combine the best practices from the Rational Unified Process (RUP) and from agile methodologies (such as Extreme Programming). If so, buy this book! Software Development for Small Teams describes an entire software development project, from the initial customer contact through delivery of the software. Through a case study, it describes how one small, distributed team designed and applied a successful process. But this is not a perfect case study. The story includes what worked and what didn't, and describes how the team might change its process for the next

project. The authors encourage you to assess their results and to use the lessons learned on your next project. Key topics covered include: Achieving a balance between people, process, and tools; recognizing that software development Agility and Discipline Made Easy International Technical Support Organization IBM Although everyone agrees on the need to make government work better, few understand public bureaucracy sufficiently well to offer useful suggestions, either theoretical or practical. In fact, some consider bureaucratic efficiency incompatible with democratic government. Douglas Yates places the often competing aims of efficiency and democracy in historical perspective and then presents a unique and systematic theory of the politics of bureaucracy, which he illustrates with examples from recent history and from empirical research. He argues that the United States operates under a system of "bureaucratic democracy," in which governmental decisions increasingly are made in bureaucratic settings, out of the public eye. He describes the rational,

selfinterested bureaucrat as a "minimaxer," who inches forward inconspicuously, gradually accumulating larger budgets and greater power, in an atmosphere of segmented pluralism, of conflict and competition, of silent politics. To make the policy process more competitive, democratic, and open, Yates calls for strategic debate among policymakers and bureaucrats and insists that bureaucrats should give a public accounting of their significant decisions rather than bury them in incremental changes. He offers concrete proposals, applicable to federal, state, and local governments, for simplifying the now-chaotic bureaucratic policymaking system and at the same time bolstering representation and openness. This is a book for all political scientists, policymakers, government officials, and concerned citizens. It may well become a classic statement on the workings of public bureaucracy.

Crucial Conversations: Tools for Talking When Stakes are High, Third Edition Harvard University Press

The Authoritative, Best-Practice Guide to Improving Development Processes with IBM® Rational Unified Process® (RUP®) This book delivers all the knowledge and insight you need to succeed with the IBM Rational Unified Process and Solutions. Joshua Barnes presents a start-to-finish, best-practice roadmap to the complete implementation cycle of IBM RUP—from projecting ROI and making the business case through piloting, implementation, mentoring, and beyond. Drawing on his extensive experience leading large-scale IBM RUP implementations and working with some of the industry's most recognized thought leaders in the Software Engineering Process world, Barnes brings together comprehensive "lessons learned" from both successful and failed projects. You'll learn from real-world case studies, including actual project artifacts. Whether you're an executive, software professional, or consultant, this book will help you continuously improve the maturity of your development processes—and reap the benefits: better quality, faster delivery, and more

business value. After reading this book you will be able to · Get past the myths of software process improvement to focus on what's truly practical · Identify and evaluate your best candidate process solutions · Objectively project the ROI achievable with IBM R UP and IBM Rational solutions · Develop funding models, business cases, and executive support · Recruit, staff, organize, and motivate your implementation team · Plan for effective integration, process alignment, and change management · Choose the right pilots, learn the right lessons, and develop effective adoption models · Move quickly to successful program-level implementation · Set maturity level goals for process and tool utilization · Map "End States" for both quantity and quality · Plan for training and mentoring—and understand the distinct role of each · Keep the momentum going after your implementation is complete Link to [www.upmentors.com](http://www.upmentors.com), where you can download actual sample implementation documents—not just templates!



www.ibmpressbooks.com  
 Preface xvii  
 Acknowledgments xxiii  
 About the Author xxv  
 Chapter 1: Evaluating Process Solutions 1  
 Chapter 2: Your First Steps Toward Implementing RUP and IBM Rational Solutions 17  
 Chapter 3: Assessing Your Organization and Building Your Business Case for Organizational Change 29  
 Chapter 4: Implementation Team 49  
 Chapter 5: Setting Up Pre-Integrated and Process-Aligned Tooling 67  
 Chapter 6: Implementation Approach 75  
 Chapter 7: Transitioning to a Program Approach 99  
 Chapter 8: Funding Model 117  
 Chapter 9: Training and Mentoring Models 131  
 Chapter 10: Is Your Implementation Complete? 149  
 Appendix 1: Executive ROI Overview 155  
 Appendix 2: Detailed Appendix for Executive ROI Overview 159  
 Appendix 3: Maturity Level Goals—Sample Tasks 167  
 Index 175  
**Building J2EE Applications with the Rational Unified Process** Addison-Wesley Professional  
 · Master win-win techniques for managing outsourced and offshore projects, from

procurement and risk mitigation to maintenance  
 · Use RUP to implement best-practice project management throughout the software development lifecycle  
 · Overcome key management challenges, from changing requirements to managing user expectations  
**The Hands-On, Start-to-Finish Guide to Managing Software Projects with the IBM® Rational Unified Process®**  
 This is the definitive guide to managing software development projects with the IBM Rational Unified Process (RUP®). Drawing on his extensive experience managing projects with the RUP, R. Dennis Gibbs covers the entire development lifecycle, from planning and requirements to post-mortems and system maintenance. Gibbs offers especially valuable insights into using the RUP to manage outsourced projects and any project relying on distributed development teams—outsourced, insourced, or both. This “from the trenches” guidebook is invaluable for anyone interested in best practices for managing software development: project managers, team leaders, procurement and

contracting specialists, quality assurance and software process professionals, consultants, and developers. If you’re already using the RUP, Gibbs will help you more effectively use it. Whatever your role or the RUP experience, you’ll learn ways to  
 · Simplify and streamline the management of any large-scale or outsourced project  
 · Overcome the challenges of using the RUP in software project management  
 · Optimize software procurement and supplier relationships, from Request for Proposals (RFPs) and contracts to delivery  
 · Staff high-performance project teams and project management offices  
 · Establish productive, consistent development environments  
 · Run effective project kickoffs  
 · Systematically identify and mitigate project risks  
 · Manage the technical and business challenges of changing requirements  
 · Organize iterations and testing in incremental development processes  
 · Transition new systems into service: from managing expectations to migrating data  
 · Plan system maintenance and implement effective change control  
 · Learn all you can from project post-

mortems—and put those lessons into practice

[A Guide to Improving Your Software Development Capability and Maturity](#)  
Addison-Wesley Professional  
bull; Reflects all of the changes that were integrated into RUP v2003—the latest version of the very popular product bull; Learn the key concepts, fundamentals of structure, integral content, and motivation behind the RUP bull; Covers all phases of the software development lifecycle—from concept, to delivery, to revision

**A Disciplined Agile Delivery Handbook for Optimizing Your Way of Working** Pearson Education  
"The Japanese samurai Musashi wrote: 'One can win with the long sword, and one can win with the short sword. Whatever the weapon, there is a time and situation in which it is appropriate.' "Similarly, we have the long RUP and the short RUP, and all sizes in between. RUP is not a rigid, static recipe, and it evolves with the field and the practitioners, as demonstrated in this new book full of wisdom to illustrate further the liveliness of a process adopted by so many

organizations around the world. Bravo!" --Philippe Kruchten, Professor, University of British Columbia "The Unified Process and its practices have had, and continue to have, a great impact on the software industry. This book is a refreshing new look at some of the principles underlying the Unified Process. It is full of practical guidance for people who want to start, or increase, their adoption of proven practices. No matter where you are today in terms of software maturity, you can start improving tomorrow." --Ivar Jacobson, Ivar Jacobson Consulting "Kroll and MacIsaac have written a must-have book. It is well organized with new principles for software development. I encounter many books I consider valuable; I consider this one indispensable, especially as it includes over 20 concrete best practices. If you are interested in making your software development shop a better one, read this book!" --Ricardo R. Garcia, President, Global Rational User Group Council, [www.rational-ug.org/index.php](http://www.rational-ug.org/index.php) "Agile software development is real, it works, and it's here to stay. Now is the time to

come up to speed on agile best practices for the Unified Process, and this book provides a great starting point." --Scott W. Ambler, practice leader, Agile Modeling "IBM and the global economy have become increasingly dependent on software over the last decade, and our industry has evolved some discriminating best practices. Per and Bruce have captured the principles and practices of success in this concise book; a must for executives, project managers, and practitioners. These ideas are progressive, but they strike the right balance between agility and governance and will form the foundation for successful systems and software developers for a long time." --Walker Royce, Vice President, IBM Software Services-Rational "Finally, the RUP is presented in digestible, byte-size pieces. Kroll and MacIsaac effectively describe a set of practices that can be adopted in a low-ceremony, ad hoc fashion, suited to the culture of the more agile project team, while allowing them to understand how to scale their process as needed." --Dean Leffingwell, author and software business

advisor and executive "This text fills an important gap in the knowledge-base of our industry: providing agile practices in the proven, scalable framework of the Unified Process. With each practice able to be throttled to the unique context of a development organization, Kroll and Maclsaac provide software teams with the ability to balance agility and discipline as appropriate for their specific needs." -- Brian G. Lyons, CTO, Number Six Software, Inc. In *Agility and Discipline Made Easy*, Rational Unified Process (RUP) and Open Unified Process (OpenUP) experts Per Kroll and Bruce Maclsaac share twenty well-defined best practices that you and your team can start adopting today to improve the agility, predictability, speed, and cost of software development. Kroll and Maclsaac outline proven principles for software development, and supply a number of supporting practices for each. You'll learn what problems each practice addresses and how you can best leverage RUP and OpenUP (an open-source version of the Unified Process) to make the practice work for you. You'll find proactive,

prescriptive guidance on how to adopt the practices with minimal risk and implement as much or as little of RUP or OpenUP as you want. Learn how to apply sample practices from the Unified Process so you can Execute your project in iterations Embrace and manage change Test your own code Describe requirements from the user perspective Architect with components and services Model key perspectives Whether you are interested in agile or disciplined development using RUP, OpenUP, or other agile processes, this book will help you reduce the anxiety and cost associated with software improvement by providing an easy, non-intrusive path toward improved results--without overwhelming you and your team. The Unified Software Development Process Pragmatic Bookshelf Refactoring has proven its value in a wide range of development projects--helping software professionals improve system designs, maintainability, extensibility, and performance. Now, for the first time, leading agile methodologist Scott Ambler and renowned

consultant Pramodkumar Sadalage introduce powerful refactoring techniques specifically designed for database systems. Ambler and Sadalage demonstrate how small changes to table structures, data, stored procedures, and triggers can significantly enhance virtually any database design--without changing semantics. You'll learn how to evolve database schemas in step with source code--and become far more effective in projects relying on iterative, agile methodologies. This comprehensive guide and reference helps you overcome the practical obstacles to refactoring real-world databases by covering every fundamental concept underlying database refactoring. Using start-to-finish examples, the authors walk you through refactoring simple standalone database applications as well as sophisticated multi-application scenarios. You'll master every task involved in refactoring database schemas, and discover best practices for deploying refactorings in even the most complex production environments. The second half of this book systematically

covers five major categories of database refactorings. You'll learn how to use refactoring to enhance database structure, data quality, and referential integrity; and how to refactor both architectures and

methods. This book provides an extensive set of examples built with Oracle and Java and easily adaptable for other languages, such as C#, C++, or VB.NET, and other databases, such as

DB2, SQL Server, MySQL, and Sybase. Using this book's techniques and examples, you can reduce waste, rework, risk, and cost-and build database systems capable of evolving smoothly, far into the future.