

## Refactoring Databases: Evolutionary Database Design (Addison Wesley Signature)

Describes Agile Modeling Driven Design (AMDD) and Test-Driven Design (TDD) approaches, database refactoring, database encapsulation strategies, and tools that support evolutionary techniques Agile software developers often use object and relational database (RDB) technology together and as a result must overcome the impedance mismatch The author covers techniques for mapping objects to RDBs and for implementing concurrency control, referential integrity, shared business logic, security access control, reports, and XML An agile foundation describes fundamental skills that all agile software developers require, particularly Agile DBAs Includes object modeling, UML data modeling, data normalization, class normalization, and how to deal with legacy databases Scott W. Ambler is author of Agile Modeling (0471202827), a contributing editor with Software Development (www.sdmagazine.com), and a featured speaker at software conferences worldwide

Automated testing is a cornerstone of agile development. An effective testing strategy will deliver new functionality more aggressively, accelerate user feedback, and improve quality. However, for many developers, creating effective automated tests is a unique and unfamiliar challenge. xUnit Test Patterns is the definitive guide to writing automated tests using xUnit, the most popular unit testing framework today. Agile software development expert Gerard Meszaros describes 88 proven patterns for making tests easier to write, understand, and maintain. He then shows you how to make them more robust and repeatable—and far more cost-effective. Loaded with information, this book feels like a three books in one. The first part is a detailed tutorial on test automation that covers everything from test strategy to in-depth test coding. The second part, a catalog of 18 frequently encountered "test smells," provides trouble-shooting guidelines to help you determine the root cause of problems and the most applicable patterns. The third part contains detailed descriptions of each pattern, including refactoring instructions illustrated by extensive code samples in multiple programming languages.

The only complete, proven, start-to-finish blueprint for successful "just-in-time" agile database development! " Knowledge virtually every agile shop needs, because nearly all of them must build and run databases "New agile approaches to ensuring that databases are consistent and stable in fast-changing environments, and test-driving designs to identify problems upfront, when they're cheaper to fix "Based on author Max Guernsey III's pioneering NetObjectives course in database agility. Design and build truly agile databases that can be changed frequently, safely, and painlessly, no matter how much existing data they must manage! With this book, you'll finally get past old-fashioned "batch-and-queue" database development, and construct a truly agile database development environment that will pioneer agile database expert Max Guernsey III combines a complete foundation of theoretical knowledge with concrete examples and real solutions to the impediments that have prevented database developers from going agile. Guernsey especially shows how to adapt agile principles to handle massive amounts of existing data that makes database change more difficult. Test-Driven Database Development is based on the training curricula for the author's pioneering NetObjectives course, Database Agility Online Training, which has helped hundreds of database professionals master critical technical skills for designing databases that can be changed frequently, safely, and painlessly. Reflecting his immense experience with agile database development, Guernsey helps you make sure all databases and data remain consistent in agile environments; ensure stability no matter how fast databases change; and test-drive designs to find and fix errors before they're "baked into" the system. This book will be an invaluable resource for virtually every database analyst and DBA in agile organizations; for many database team, project, and group managers; and for even more agile development team members in organizations that rely on large and complex databases.

Organizations invest immense amounts of time, resources, and attention in their software projects. But all too often, when it's time to transfer the finished project to new "owners," they settle for the most superficial classroom training, documentation, and code walkthroughs. These conventional approaches to knowledge transfer often fail, dramatically reducing the value of new systems in production. The authors of *Knowledge Transfer for the Agile World* show how to elevate knowledge transfer from "necessary evil" to an activity full of agility and innovation, and bring together multiple organizations and cultures to make ownership transfer work. Sankaranarayanan explains why mere documentation of error reports and processes isn't enough, and shows how to successfully craft a knowledge transfer program that's more substantive and effective. Along the way, he offers guidance on overcoming the commercial compromises and personal tensions often associated with transferring systems to new ownership; and on transforming mere "knowledge transfer" into something much better: "taking ownership."

Like any other software system, Web sites gradually accumulate "cr" over time. They slow down. Links break. Security and compatibility problems mysteriously appear. New features don't integrate seamlessly. Things just don't work as well. In an ideal world, you'd rebuild from scratch. But you can't: there's no time or money for that. Fortunately, there's a solution: You can refactor your Web code using easy, proven techniques, tools, and scripts adapted from the world of software development. In Refactoring HTML, Elliott Rusty Harold explains how to use refactoring to improve virtually any Web site or application. Writing for programmers and non-programmers alike, Harold shows how to refactor for better reliability, performance, usability, security, accessibility, compatibility, and even search engine placement. Step by step, he shows how to migrate obsolete code to today's stable Web standards, including XHTML, CSS, and REST—and eliminate chronic problems like presentation-based markup, stateful applications, and "tag soup." The book's extensive catalog of detailed refactorings and practical "recipes for success" are organized to help you find specific solutions fast, and get maximum benefit for minimum effort. Using this book, you can quickly improve site performance now—and make your site far easier to enhance, maintain, and scale for years to come. Topics covered include "Recognizing the "smells" of Web code that should be refactored "Transforming old HTML into well-formed, valid XHTML, one step at a time "Modernizing existing layouts with CSS "Updating old Web applications: replacing POST with GET, replacing old contact forms, and refactoring JavaScript "Systematically refactoring content and links "Restructuring sites without changing the URLs you users rely upon This book will be an indispensable resource for Web designers, developers, project managers, and anyone who maintains or updates existing sites. It will be

especially helpful to Web professionals who learned HTML years ago, and want to refresh their knowledge with today's standards-compliant best practices.

Testing the Transformation is an executive guide, providing a clear framework for improving development and delivery. Instead of the traditional Agile and DevOps approaches that focus on improving the effectiveness of teams, this book targets the coordination of work across teams in large organizations—an improvement that executives are uniquely positioned to lead.

The practice of enterprise application development has benefited from the emergence of many new enabling technologies. Multi-tiered object-oriented platforms, such as Java and .NET, have become commonplace. These new tools and technologies are capable of building powerful applications, but they are not easily implemented. Common failures in enterprise applications often occur because their developers do not understand the architectural lessons that experienced object developers have learned. Patterns of Enterprise Application Architecture is written in direct response to the stiff challenges that face enterprise application developers. The author, noted object-oriented designer Martin Fowler, noticed that despite changes in technology—from Smalltalk to CORBA to Java to .NET—the same basic design ideas can be adapted and applied to solve common problems. With the help of an expert group of contributors, Martin distills over forty recurring solutions into patterns. The result is an indispensable handbook of solutions that are applicable to any enterprise application platform. This book is actually two books in one. The first section is a short tutorial on developing enterprise applications, which you can read from start to finish to understand the scope of the book's lessons. The next section, the bulk of the book, is a detailed reference to the patterns themselves. Each pattern provides usage and implementation information, as well as detailed code examples in Java or C#. The entire book is also richly illustrated with UML diagrams to further explain the concepts. Armed with this book, you will have the knowledge necessary to make important architectural decisions about building an enterprise application and the proven patterns for use when building them. The topics covered include - Dividing an enterprise application into layers - The major approaches to organizing business logic - An in-depth treatment of mapping between objects and relational databases - Using Model-View-Controller to organize a Web presentation - Handling concurrency for data that spans multiple transactions - Designing distributed object interfaces

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.

[Fowler](#)

[Evolving Knowledge Transfer for the Agile World](#)

[Monolith to Microservices](#)

[Fundamental Design Solutions for SOAP/WSDL and RESTful Web Services](#)

[Redefining the Architect's Role in the Digital Enterprise](#)

[The Art of Agile Development](#)

[Test-Driven Database Development](#)

[Oracle@ PL/SQL](#)

[Introduction to Databases and Data Warehouses](#)

[Conventions of Thought](#)

[Learning Winning Solutions](#)

[The Application Developer's Guide to Object-Orientation and the UML](#)

[Choose your WoW](#)

[Effective Strategies for the Agile Software Developer](#)

"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."--Jacket.

This book, first published in 2000, illustrates rules of Java code-writing with parallel examples of correct and incorrect usage.

distinct set of design solutions. This is the first book to systematically identify and explain best practice patterns for domain engineers. Rob Daigneau expands on the Service Layer concept (covered expertly by Fowler's Patterns of Enterprise Application Architecture) domain services can be used with Enterprise Integration Patterns (made famous by Holops and Woolf). Daigneau begins by reviewing SOA concepts, illuminating the distinctions between enterprise and domain services, and identifying key relationships between domain services and other pattern groups. Next, he introduces each essential pattern for creating and delivering domain services, providing a vocabulary of design solutions that architects and other IT professionals can implement by referencing and adapting the concrete examples he supplies.

How do you detangle a monolithic system and migrate it to a microservice architecture? How do you do it while maintaining business-as-usual? As a companion to Sam Newman's extremely popular Building Microservices, this new book details a proven method for transitioning an existing monolithic system to a microservice architecture. With many illustrative examples, insightful migration patterns, and a bevy of practical advice to transition your monolith enterprise into a microservice operation, this practical guide covers multiple scenarios and strategies for a successful migration, from initial planning all the way through application and database decomposition. You'll learn several tried and tested patterns and techniques that you can use as you migrate your existing architecture. Ideal for organizations looking to transition to microservices, rather than rebuild Helps companies determine whether to migrate, when to migrate, and where to begin Addresses communication, integration, and the migration of legacy systems Discusses multiple migration patterns and where they apply Provides database migration examples, along with synchronization strategies Explores application decomposition, including several architectural refactoring patterns Delves into details of database decomposition, including the impact of breaking referential and transactional integrity, new failure modes, and more

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.

[Evolutionary Database Design](#)

[Data Model Patterns](#)

[The Software Architect Elevator](#)

[A Disciplined Agile Delivery Handbook for Optimizing Your Way of Working](#)

[A Brief Guide to the Emerging World of Polyglot Persistence](#)

[Pattern Enterpr Applic Arch](#)

[The Big Ideas Behind Reliable, Scalable, and Maintainable Systems](#)

[Agile Data Warehousing for the Enterprise](#)

[Evolutionary Patterns to Transform Your Monolith](#)

[Effective Practices for eXtreme Programming and the Unified Process](#)

[xUnit Test Patterns](#)

[A Hands-On Guide to Data Manipulation in SQL](#)

[Database Design for Mere Mortals](#)

[NoSQL Distilled](#)

The acclaimed beginner's book on object technology now presents UML 2.0, Agile Modeling, and the latest in object development techniques. Various topics of data mining techniques are identified and described throughout, including clustering, association rules, rough set theory, probability theory, neural networks, classification, and fuzzy logic. Each of these techniques is explored with a theoretical introduction and its effectiveness is demonstrated with various chapter examples. Hundreds of organizations around the world have already benefited from DisciplinedAgile 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.&#t 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 &#t depends&# questions.&#t 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. Discover how graph databases can help you manage and query highly connected data. With this practical book, you'll learn how to design and implement a graph database that brings the power of graphs to bear on a broad range of problem domains. Whether you want to speed up your response to user queries or build a database that can adapt as your business evolves, this book shows you how to apply the schema-free graph model to real-world problems. Learn how different organizations are using graph databases to outperform their competitors. With this book's data modeling, query, and code examples, you'll quickly be able to implement your own solution. Model data with the Cypher query language and propher graph model Learn best practices and common pitfalls when modeling with graphs Plan and implement a graph database solution in test-driven fashion Explore real-world examples to learn how and why organizations use a graph database Understand common patterns and components of graph database architecture Use analytical techniques and algorithms to mine graph database information Using Agile methods, you can bring far greater innovation, value, and quality to any data warehousing (DW), business intelligence (BI), or analytics project. However, conventional Agile methods must be carefully adapted to address the unique characteristics of DW/BI projects. In Agile Analytics, Agile pioneer Ken Collier shows how to do just that. Collier introduces platform-agnostic Agile solutions for integrating infrastructures consisting of diverse operational, legacy, and specialty systems that mix commercial and custom code. Using working examples, he shows how to manage analytics development teams with widely diverse skill sets and how to support enormous and fast-growing data volumes. Collier's techniques offer optimal value whether your projects involve "back-end" data management, "front-end" business analysis, or both. Part I focuses on Agile project management techniques and delivery team coordination, introducing core practices that shape the way your Agile DW/BI project community can collaborate toward success Part II presents technical methods for enabling continuous delivery of business value at production-quality levels, including evolving superior designs; test-driven DW development; version control; and project automation Collier brings together proven solutions you can apply right now—whether you're an IT decision-maker, data warehouse professional, database administrator, business intelligence specialist, or database developer. With his help, you can mitigate project risk, improve business alignment, achieve better results—and have fun along the way. The first book to cover Agile Modeling, a new modeling techniquecreated specifically for XP projects extreme Programming (XP) hascreated a buzz in the software development community-much likeDesign Patterns did several years ago. Although XP presents amethodology for faster software development, many developers findthat XP does not allow for modeling time, which is critical toensure that a project meets its proposed requirements. They havealso found that standard modeling techniques that use the UnifiedModeling Language (UML) often do not work with this methodology. Inthis innovative book, Software Development columnist Scott Amblerpresents Agile Modeling (AM)—a technique that he created formodeling XP projects using pieces of the UML and Rational's UnifiedProcess (RUP). Ambler clearly explains AM, and shows readers how toincorporate AM, UML, and RUP into their development projects withthe help of numerous case studies integrated throughout the book. AM was created by the author for modeling XP projects-anelement lacking in the original XP design. The XP community and its creator have embraced AM, which shouldgive this book strong market acceptance Companion Web site at www.agilemodeling.com features updates,links to XP and AM resources, and ongoing case studies about applyingmodeling.

Developers using PL/SQL 9i as an environment for corporate applications will find detailed technical information and practical tips in Lakshman's book.

A book for Tech Leads, from Tech Leads. Discover how more than 35 Tech Leads find the delicate balance between the technical and non-technical worlds. Discover the challenges a Tech Lead faces and how to overcome them. You may be surprised by the lessons they have to share.

[The Elements of Java\(TM\) Style](#)

[Recipes for Continuous Database Integration](#)

[Database Systems](#)

[Refactoring HTML](#)

[Refactoring Databases](#)

[Practical Use Cases with the Leading NoSQL Database](#)

[Refactoring Test Code](#)

[Advanced Principles for Improving Database Design, Systems Modeling, and Software Development](#)

[Leading the Transformation](#)

[Evolutionary Database Development \(Digital Short Cut\)](#)

[A Value-driven Approach to Business Intelligence and Data Warehousing](#)

[Software Ownership Transfer](#)

[Agile Model-Driven Development with UML 2.0](#)

The need to handle increasingly larger data volumes is one factor driving the adoption of a new class of nonrelational "NoSQL" databases. Advocates of NoSQL databases claim they can be used to build systems that are more performant, scale better, and are easier to program. NoSQL Distilled is a concise but thorough introduction to this rapidly emerging technology. Pramod J. Sadalage and Martin Fowler explain how NoSQL databases work and the ways that they may be a superior alternative to a traditional RDBMS. The authors provide a fast-paced guide to the concepts you need to know in order to evaluate whether NoSQL databases are right for your needs and, if so, which technologies you should explore further. The first part of the book concentrates on core concepts, including schemaless data models, aggregates, new distribution models, the CAP theorem, and map-reduce. In the second part, the authors explore architectural and design issues associated with implementing NoSQL. They also present realistic use cases that demonstrate NoSQL databases at work and feature representative examples using Riak, MongoDB, Cassandra, and Neo4j. In addition, by drawing on Pramod Sadalage's pioneering work, NoSQL Distilled shows how to implement evolutionary design with schema migration: an essential technique for applying NoSQL databases. The book concludes by describing how NoSQL is ushering in a new age of Polyglot Persistence, where multiple data-storage worlds coexist, and architects can choose the technology best optimized for each type of data access.

"This book takes the somewhat daunting process of database design and breaks it into completely manageable and understandable components. Mike's approach whilst simple is completely professional, and I can recommend this book to any novice database designer." --Sandra Barker, Lecturer, University of South Australia, Australia "Databases are a critical infrastructure technology for information systems and today's business. Mike Hernandez has written a literate explanation of database technology--a topic that is intricate and often obscure. If you design databases yourself, this book will educate you about pitfalls and show you what to do. If you purchase products that use a database, the book explains the technology so that you can understand what the vendor is doing and assess their products better." --Michael Blaha, consultant and trainer, author of A Manager's Guide to Database Technology "If you told me that Mike Hernandez could improve on the first edition of Database Design for Mere Mortals I wouldn't have believed you, but he did! The second edition is packed with more real-world examples, detailed explanations, and even includes database-design tools on the CD-ROM! This is a must-read for anyone who is even remotely interested in relational database design, from the individual who is called upon occasionally to create a useful tool at work, to the seasoned professional who wants to brush up on the fundamentals. Simply put, if you want to do it right, read this book!" --Matt Greer, Process Control Development, The Dow Chemical Company "Mike's approach to database design is totally common-sense based, yet he's adhered to all the rules of good relational database design. I use Mike's books in my starter database-design class, and I recommend his books to anyone who's interested in learning how to design databases or how to write SQL queries." --Michelle Poole, President, MWDS, Inc. "Slapping together sophisticated applications with poorly designed data will hurt you just as much now as it did when you first edition, perhaps even more. Whether you're just getting started developing with data or are a seasoned pro; whether you've read Mike's previous book or this is your first; whether you're happier letting someone else design your data or you love doing it yourself--this is the book for you. Mike's ability to explain these concepts in a way that's not only clear, but fun, continues to amaze me." --From the Foreword by Ken Getz, MCV Technology, coauthor ASP.NET Developer's JumpStart "The first edition of Mike Hernandez's book Database Design for Mere Mortals was one of the few books that survived the cut when I moved my office to smaller quarters. The second edition expands and improves on the original in so many ways. It is not only a good, clear read, but contains a remarkable quantity of clear, concise thinking on a very complex subject. It's a must for anyone interested in the subject of database design." --Matcoln C. Rubel, Performance Dynamics Associates "Mike's excellent guide to relational database design deserves a second edition. His book is an essential tool for fledgling Microsoft Access and other desktop database developers, as well as for client/server pros. I recommend it highly to all my readers." --Roger Jennings, author of Special Edition Using Access 2002 "There are no silver bullets! Database technology has advanced dramatically, the newest crop of database servers perform operations faster than anyone could have imagined six years ago, but none of these technological advances will help fix a bad database design, or capture data that you forgot to include! Database Design for Mere Mortals(TM), Second Edition, helps you design your database right in the first place!" --Matt Nunn, Product Manager, SQL Server, Microsoft Corporation "When my brother started his professional career as a developer, I gave him Mike's book to help him understand database concepts and make real-world application of database technology. When I need a refresher on the finer points of database design, this is the book I pick up. I do not think that there is a better testimony to the value of a book than that it gets used. For this reason I have wholeheartedly recommended to my peers and students that they utilize this book in their day-to-day development tasks."

--Chris Kunicnik, Senior Consultant, OfficeZealot.com "Mike has always had an incredible knack for taking the most complex topics, breaking them down, and explaining them so that anyone can 'get it.' He has honed and polished his first very, very good edition and made it even better. If you're just starting out building database applications, this book is a must-read cover to cover. Expert designers will find Mike's approach fresh and enlightening and a source of great material for training others." --John Viescas, President, Viescas Consulting, Inc., author of Running Microsoft Access and coauthor of SQL Queries for Mere Mortals "Whether you need to learn about relational database design in general, design a relational database, understand relational database terminology, or learn best practices for implementing a relational database, Database Design for Mere Mortals(TM), Second Edition, is an indispensable book that you'll refer to often. With his many years of real-world experience designing relational databases, Michael shows you how to analyze and improve existing databases, implement keys, define table relationships and business rules, and create data views, resulting in data integrity, uniform access to data, and reduced data-entry errors." --Paul Cornell, Site Editor, MSDN Office Developer Center Sound database design can save hours of development time and ensure functionality and reliability. Database Design for Mere Mortals(TM), Second Edition, is a straightforward, platform-independent tutorial on the basic principles of relational database design. It provides a commonsense design methodology for developing databases that work. Database design expert Michael J. Hernandez has expanded his best-selling database design process. This book will give you the knowledge and tools you need to create efficient and effective relational databases.

"This book presents cutting-edge research and analysis of the most recent advancements in the fields of database systems and software development"--Provided by publisher.

As the digital economy changes the rules of the game for enterprises, the role of software and IT architects is also transforming. Rather than focus on technical decisions alone, architects and senior technologists need to combine organizational and technical knowledge to effect change in their company's structure and processes. To accomplish that, they need to connect the IT engine room to the penthouse, where the business strategy is defined. In this guide, author Gregor Hohpe shares real-world advice and hard-learned lessons from actual IT transformations. His anecdotes help architect senior developers, and other IT professionals prepare for a more complex but rewarding role in the enterprise. This book is ideal for: Software architects and senior developers looking to shape the company's technology direction or assist in an organizational transformation Enterprise architects and senior technologists searching for practical advice on how to navigate technical and organizational topics CTOs and senior technical architects who are devising an IT strategy that impacts the way the organization works IT managers who want to learn what's worked and what hasn't in large-scale transformations

The #1 Easy, Common-Sense Guide to SQL Queries--Updated for Today's Databases, Standards, and Challenges SQL Queries for Mere Mortals® has earned worldwide praise as the clearest, simplest tutorial on writing effective SQL queries. The authors have updated this hands-on classic to reflect new SQL standards and database applications and teach valuable new techniques. Step by step, John L. Viescas and Michael J. Hernandez guide you through creating reliable queries for virtually any modern SQL-based database. They demystify all aspects of SQL query writing, from simple data selection and filtering to joining multiple tables and modifying sets of data. Three brand-new chapters teach you how to solve a wide range of challenging SQL problems. You'll learn how to write queries that apply multiple complex conditions on one table, perform sophisticated logical evaluations, and think "outside the box" using unlinked tables. Coverage includes -- Getting started: understanding what relational databases are, and ensuring that your database structures are sound -- SQL basics: using SELECT statements, creating expressions, sorting information with ORDER BY, and filtering data using WHERE -- Summarizing and grouping data with GROUP BY and HAVING clauses -- Drawing data from multiple tables: using INNER JOIN, OUTER JOIN, and UNION operators, and working with subqueries -- Modifying data sets with UPDATE, INSERT, and DELETE statements Advanced queries: complex NOT and AND, conditions, if-then-else using CASE, unlinked tables, driver tables, and more Practice all you want with downloadable sample databases for today's versions of Microsoft Office Access, Microsoft SQL Server, and the open source MySQL database. Whether you're a DBA,

developer, user, or student, there's no better way to master SQL. Informit.com/aw/forMereMortals.com

This title contains standards and guidelines for creating UML diagrams that are concise and easy to understand.

This is the digital version of the printed book (Copyright © 1996). Learning the basics of a modeling technique is not the same as learning how to use and apply it. To develop a data model of an organization is to gain insights into its nature that do not come easily. Indeed, analysts are often expected to understand subtleties of an organization's structure that may have evaded people who have worked there for years. Here's help for those analysts who have learned the basics of data modeling (or "entity/relationship modeling") but who need to obtain the insights required to prepare a good model of a real business. Structures common to many types of business are analyzed in areas such as accounting, material requirements planning, process manufacturing, contracts, laboratories, and documents. In each chapter, high-level data models are drawn from the following business areas: The Enterprise and Its World The Things of the Enterprise Procedures and Activities Contracts Accounting The Laboratory Material Requirements Planning Process Manufacturing Documents Lower-Level Conventions

This text aims to help all members of the development team make the correct nuts-and-bolts architecture decisions that ensure project success.

[Applying Agile and DevOps Principles at Scale](#)

[Beyond Software Architecture](#)

[Evolve the Monolith to Microservices with Java and Node](#)

[Practical Applications of Data Mining](#)

[Improving the Design of Existing Web Applications](#)

[From Novices to Practitioners](#)

[Continuous Integration](#)

[Evolutionary Database Design \(paperback\)](#)

[Service Design Patterns](#)

[Support Constant Change](#)

[Agile Analytics](#)

[A Developer's Guide](#)

[MongoDB Applied Design Patterns](#)

[The Elements of UML 2.0 Style](#)

Microservices is an agile in which large, complex software applications are composed of one or more smaller services. Each of these microservices focuses on completing one task that represents a small business capability. These microservices can be developed in any programming language. This IBM Redbooks® publication shows how to break out a traditional Java EE application into separate microservices and provides a set of code projects that illustrate the various steps along the way. These code projects use the IBM WebSphere® Application Server Liberty, IBM APT ConnectTM, IBM Bluemix®, and other Open Source Frameworks in the microservices ecosystem. The sample projects highlight the evolution of monoliths to microservices with Java and Node.

Building upon his earlier book that detailed agile data warehousing programming techniques for the Scrum master, Ralph's latest work illustrates the agile interpretations of the remaining software engineering disciplines: Requirements management benefits from streamlined templates that not only define projects quickly, but ensure nothing essential is overlooked. Data engineering receives two new "hyper modeling" techniques, yielding data warehouses that can be easily adapted when requirements change without having to invest in ruinously expensive data-conversion programs. Quality assurance advances with not only a stereoscopic top-down and bottom-up planning method, but also the incorporation of the latest in automated test engines. Use this step-by-step guide to deepen your own application development skills through self-study, show your teammates the world's fastest and most reliable techniques for creating business critical systems, or ensure that the IT department working for you is building your next decision support system the right way. Learn how to quickly define scope and architecture before programming starts Includes techniques of process and data engineering that enable iterative and incremental delivery

Demonstrates how to plan and execute quality assurance plans and includes a guide to continuous integration and automated regression testing Presents program management strategies for coordinating multiple agile data mart projects so that over time an enterprise data warehouse emerges Use the provided 120-day road map to establish a robust, agile data warehousing program

An introductory, yet comprehensive, database textbook intended for use in undergraduate and graduate information systems database courses. This text also provides practical content to current and aspiring information systems, business data analysis, and decision support industry professionals. Database Systems: Introduction to Databases and Data Warehouses covers both analytical and operations database as knowledge of both is integral to being successful in today's business environment. It also provides a solid theoretical foundation and hands-on practice using an integrated web-based data-modeling suite.

This is the new, revised, and updated edition of the classic book, The practice of software development. These methods embrace change in requirements even late in the project. The ability to change software is because of certain practices that are followed within teams, such as Test Driven Development, Pair Programming, and Continuous Integration. Continuous Integration provides a way for software teams to integrate their work more than once a day, and promotes confidence in the software that is being developed by the team. It is thought that this practice is difficult to apply when continuously integrating the database with application code; hence, Evolutionary Database Development is considered a mismatch with agile methods. Pramod Sadalage shows that this is not necessarily true. Continuous Integration changed the way software is written. Why not extend and make the database part of the same

Continuous Integration cycle so that you can see integrated results of your application as well as your database? Delivered in PDF format for quick and easy access, *Recipes for Continuous Database Integration* shows how the database can be brought under the preview of Continuous Integration, allowing all teams to integrate not only their application code, but also their database. This Short Cut presents a recipe for each task that needs to be done. Each recipe starts with a statement of a problem, followed by an explanation and solution. It provides concrete ways and examples to implement ideas in Refactoring Databases: Evolutionary Database Design by Scott W Ambler and Pramod Sadalage. Table of Contents What This Short Cut Covers Introduction Recipe 1 Continuously Integrating? Recipe 2 Extracting Your Database in Scripts Recipe 3 Using Version Control for Your Database Recipe 4 Automating Database or Schema Creation Recipe 5 Creating Objects in Your Database Recipe 6 Removing Database Objects Recipe 7 Removing Your Database Recipe 8 Using the Build Property Files Recipe 9 Re-creating Your Application Database for Any Build Recipe 10 Making It Easy for New Developers to Join the Team Recipe 11 Integrating on Every Check-In Recipe 12 Naming Upgrade Scripts Recipe 13 Automating Database Change Script Creation Recipe 14 Implementing Database Version Checking Recipe 15 Sending Upgrades to Customers Sample Code Further Reading About the Author What's in the Companion Book Related Publication

Whether you're building a social media site or an internal-use enterprise application, this hands-on guide shows you the connection between MongoDB and the business problems it's designed to solve. You'll learn how to apply MongoDB design patterns to several challenging domains, such as ecommerce, content management, and online gaming. Using Python and JavaScript code examples, you'll discover how MongoDB lets you scale your data model while simplifying the development process. Many businesses launch NoSQL databases without understanding the techniques for using their features most effectively. This book demonstrates the benefits of document embedding, polymorphic schemas, and other MongoDB patterns for tackling specific big data use cases, including: Operational intelligence: Perform real-time analytics of business data Ecommerce: Use MongoDB as a product catalog master or inventory management system Content management: Learn methods for storing content nodes, binary assets, and discussions Online advertising networks: Apply techniques for frequency capping ad impressions, and keyword targeting and bidding Social networking: Learn how to store a complex social graph, modeled after Google+ Online gaming: Provide concurrent access to character and world data for a multiplayer role-playing game

Refactoring Databases Evolutionary Database Design (paperback) Pearson Education

Scott Ambler, author of *Building Object Applications that Work*, *Process Patterns*, and *More Process Patterns*, has revised his acclaimed first book, *The Object Primer*. Long prized in its original edition by both students and professionals as the best introduction to object-oriented technology, now this book is completely up-to-date with new material in every chapter. There are also new chapters on good OO programming techniques and OO software testing. All modeling notation has been rewritten in UML notation. Review questions at the end of each chapter allow readers to test their newly acquired knowledge. In addition, the author takes time to reflect on the lessons learned over the past few years by discussing the proven benefits and drawbacks of the technology. This is the perfect book for any software development professional or student seeking an introduction to the concepts and terminology of object technology.

[Unlocking Agility](#)  
[Agile Modeling](#)  
[Building Evolutionary Architectures](#)  
[Designing Data-Intensive Applications](#)  
[Agile Database Techniques](#)  
[SQL Queries for Mere Mortals](#)  
[Thinking with Tech Leads](#)  
[Refactoring to Patterns](#)  
[The Object Primer](#)  
[A Hands-on Guide to Relational Database Design](#)  
[Graph Databases](#)  
[A Guide for Solution Architects and Project Leaders](#)