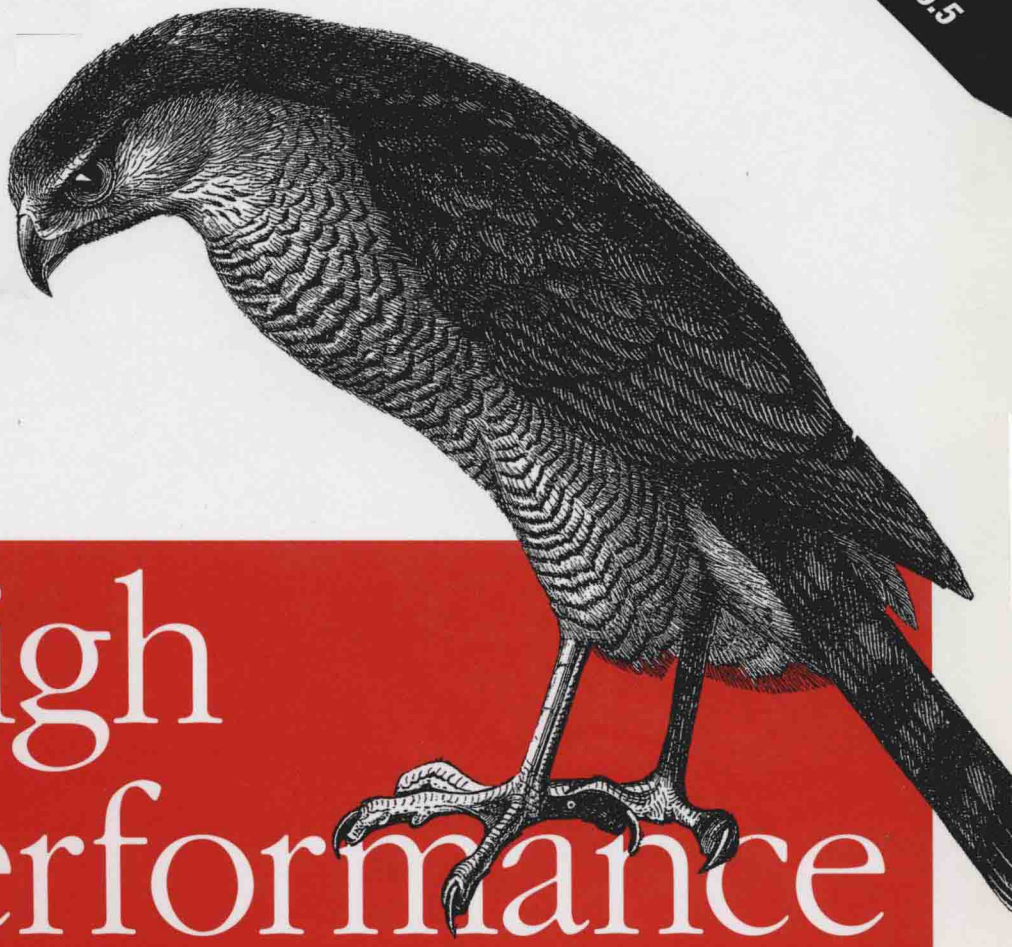
高性能MySQL（影印版）

第三版
涵蓋Version 5.5



# High Performance MySQL

*Baron Schwartz,*
*Peter Zaitsev &*
*Vadim Tkachenko* 著

O'REILLY®
東南大學出版社

# 高性能MySQL(影印版)

如何充分发挥MySQL数据库的所有能力？在《高性能MySQL》一书中，你将学到与MySQL数据库模式、索引和查询设计相关的所有高级技巧。通过使用这些技巧，你将能够对MySQL数据库服务器、操作系统和硬件进行调优以发挥它们的最大潜力。这本指南同时也提供了通过复制、负载均衡、高可用性及故障转移等技术对应用进行扩展的安全实用方法。

第三版中更新了MySQL数据库和InnoDB存储引擎在性能、特性和工具等相关领域取得的最新进展，不仅通过大量的特定示例讲解了MySQL数据库如何工作，同时也围绕MySQL数据库的设计原则，采用生动的故事和案例研究的形式解释了其对应的工作原理。本书教给读者如何采用MySQL的方式进行思考。

- 学习MySQL 5.5版提供的新特性，包括存储过程、数据库分区、触发器和视图
- 实现在复制、高可用性和集群上的改进
- 实现MySQL在云环境中运行的高性能
- 优化高级查询特性，如全文检索
- 充分利用多核处理器和固态硬盘等硬件带来的优势
- 探索备份和恢复策略——包括新的在线热备份工具

**"同本书的前几版相比，第三版是一本更加棒的书。这几位作者是唯一具有资格写这本书的人。我将继续向他们学习，并希望你也能抽出时间这样做。"**

——Mark Callaghan
Facebook软件工程师

Baron Schwartz是Percona公司的首席性能架构师。他的主要工作是通过创建一系列工具和技术来提高MySQL数据库的易用性和可靠性。

Peter Zaitsev是Percona公司的首席执行官和联合创始人之一。作为一名数据库内核、计算机硬件和应用扩展技术领域的专家，他管理着MySQL团队内部的高性能技术组直到2006年。

Vadim Tkachenko是Percona公司的首席技术官和联合创始人之一。他领导着公司的开发团队先后开发了Percona服务器、Percona XtraDB集群和Percona XtraBackup工具等产品。

**Strata**
Making Data Work

Strata是新兴的人员、工具和技术的生态系统，它使用海量数据来支持智能化决策。

责任编辑：张烨
封面设计：Karen Montgomery，张健

O'REILLY®
oreilly.com.cn

ISBN 978-7-5641-3445-7
9 787564 134457
定价：98.00元

O'Reilly Media, Inc.授权东南大学出版社出版

第三版

# 高性能MySQL（影印版）

Schwartz, Zaitsev & Tkachenko 著

O'REILLY®
东南大学出版社

（第3版）

# 高性能MySQL（影印版）

# High Performance MySQL

*Baron Schwartz, Peter Zaitsev, Vadim Tkachenko*著

O'REILLY®

Beijing · Cambridge · Farnham · Köln · Sebastopol · Tokyo

O'Reilly Media, Inc.授权东南大学出版社出版

东南大学出版社

**图书在版编目（CIP）数据**

高性能MySQL:第三版:英文/(美)施瓦兹(Schwartz, B.),(美)扎伊采夫(Zaitsev, P.)，(美)特卡申科（Tkachenko, V.)著.—影印本.—南京：东南大学出版社，2012.6

书名原文：High Performance MySQL, 3E

ISBN 978-7-5641-3445-7

I. ①高… II. ①施… ②扎… ③特… III. ①关系数据库系统－数据库管理系统－英文 IV. ①TP311.138

中国版本图书馆CIP数据核字（2012）第088915号

江苏省版权局著作权合同登记

图字：10-2012-165号



高性能MySQL 第三版（影印版）

出版发行：东南大学出版社
地　　址：南京四牌楼2号　　邮编：210096
出 版 人：江建中
网　　址：http://www.seupress.com
电子邮件：press@seupress.com
印　　刷：扬中市印刷有限公司
开　　本：787毫米×980毫米　16开本
印　　张：51.75
字　　数：1013千字
版　　次：2012年6月第1版
印　　次：2012年6月第1次印刷
书　　号：ISBN 978-7-5641-3445-7
定　　价：98.00元（册）

本社图书若有印装质量问题，请直接与营销部联系。电话（传真）：025-83791830

# Foreword

I've been a fan of this book for years, and the third edition makes a great book even better. Not only do world-class experts share that expertise, but they have taken the time to update and add chapters with high-quality writing. While the book has many details on getting high performance from MySQL, the focus of the book is on the process of improvement rather than facts and trivia. This book will help you figure out how to make things better, regardless of changes in MySQL's behavior over time.

The authors are uniquely qualified to write this book, based on their experience, principled approach, focus on efficiency, and commitment to improvement. By *experience*, I mean that the authors have been working on MySQL performance from the days when it didn't scale and had no instrumentation to the current period where things are much better. By *principled approach*, I mean that they treat this like a science, first defining problems to be solved and then using reason and measurement to solve those problems.

I am most impressed by their focus on *efficiency*. As consultants, they don't have the luxury of time. Clients getting billed by the hour want problems solved quickly. So the authors have defined processes and built tools to get things done correctly and efficiently. They describe the processes in this book and publish source code for the tools.

Finally, they continue to get better at what they do. This includes a shift in concern from throughput to response time, a commitment to understanding the performance of MySQL on new hardware, and a pursuit of new skills like queueing theory that can be used to understand performance.

I believe this book augurs a bright future for MySQL. As MySQL has evolved to support demanding workloads, the authors have led a similar effort to improve the understanding of MySQL performance within the community. They have also contributed directly to that improvement via XtraDB and XtraBackup. I continue to learn from them and hope you take the time to do so as well.

—Mark Callaghan, Software Engineer, Facebook

# Preface

We wrote this book to serve the needs of not just the MySQL application developer but also the MySQL database administrator. We assume that you are already relatively experienced with MySQL. We also assume some experience with general system administration, networking, and Unix-like operating systems.

The second edition of this book presented a lot of information to readers, but no book can provide complete coverage of a topic. Between the second and third editions, we took notes on literally thousands of interesting problems we'd solved or seen others solve. When we started to outline the third edition, it became clear that not only would full coverage of these topics require three to five thousand pages, but the book *still* wouldn't be complete. After reflecting on this problem, we realized that the second edition's emphasis on deep coverage was actually self-limiting, in the sense that it often didn't teach readers *how to think* about MySQL.

As a result, this third edition has a different focus from the second edition. We still convey a lot of information, and we still emphasize the same goals, such as reliability and correctness. But we've also tried to imbue the book with a deeper purpose: we want to teach the principles of why MySQL works as it does, not just the facts about how it works. We've included more illustrative stories and case studies, which demonstrate the principles in action. We build on these to try to answer questions such as "Given MySQL's internal architecture and operation, what practical effects arise in real usage? Why do those effects matter? How do they make MySQL well suited (or not well suited) for particular needs?"

Ultimately, we hope that your knowledge of MySQL's internals will help you in situations beyond the scope of this book. And we hope that your newfound insight will help you to learn and practice a methodical approach to designing, maintaining, and troubleshooting systems that are built on MySQL.

## How This Book Is Organized

We fit a lot of complicated topics into this book. Here, we explain how we put them together in an order that makes them easier to learn.

## A Broad Overview

Chapter 1, *MySQL Architecture and History* is dedicated to the basics—things you'll need to be familiar with before you dig in deeply. You need to understand how MySQL is organized before you'll be able to use it effectively. This chapter explains MySQL's architecture and key facts about its storage engines. It helps you get up to speed if you aren't familiar with some of the fundamentals of a relational database, including transactions. This chapter will also be useful if this book is your introduction to MySQL but you're already familiar with another database, such as Oracle. We also include a bit of historical context: the changes to MySQL over time, recent ownership changes, and where we think it's headed.

## Building a Solid Foundation

The early chapters cover material we hope you'll reference over and over as you use MySQL.

Chapter 2, *Benchmarking MySQL* discusses the basics of benchmarking—that is, determining what sort of workload your server can handle, how fast it can perform certain tasks, and so on. Benchmarking is an essential skill for evaluating how the server behaves under load, but it's also important to know when it's not useful.

Chapter 3, *Profiling Server Performance* introduces you to the response time–oriented approach we take to troubleshooting and diagnosing server performance problems. This framework has proven essential to solving some of the most puzzling cases we've seen. Although you might choose to modify our approach (we developed it by modifying Cary Millsap's approach, after all), we hope you'll avoid the pitfalls of not having any method at all.

In Chapters 4 through 6, we introduce three topics that together form the foundation for a good logical and physical database design. In Chapter 4, *Optimizing Schema and Data Types*, we cover the various nuances of data types and table design. Chapter 5, *Indexing for High Performance* extends the discussion to indexes—that is, physical database design. A firm understanding of indexes and how to use them well is essential for using MySQL effectively, so you'll probably find yourself returning to this chapter repeatedly. And Chapter 6, *Query Performance Optimization* wraps the topics together by explaining how MySQL executes queries and how you can take advantage of its query optimizer's strengths. This chapter also presents specific examples of many common classes of queries, illustrating where MySQL does a good job and how to transform queries into forms that use its strengths.

Up to this point, we've covered the basic topics that apply to any database: tables, indexes, data, and queries. Chapter 7, *Advanced MySQL Features* goes beyond the basics and shows you how MySQL's advanced features work. We examine topics such as partitioning, stored procedures, triggers, and character sets. MySQL's implementation of these features is different from other databases, and a good understanding of

them can open up new opportunities for performance gains that you might not have thought about otherwise.

## Configuring Your Application

The next two chapters discuss how to make MySQL, your application, and your hardware work well together. In Chapter 8, *Optimizing Server Settings*, we discuss how you can configure MySQL to make the most of your hardware and to be reliable and robust. Chapter 9, *Operating System and Hardware Optimization* explains how to get the most out of your operating system and hardware. We discuss solid-state storage in depth, and we suggest hardware configurations that might provide better performance for larger-scale applications.

Both chapters explore MySQL internals to some degree. This is a recurring theme that continues all the way through the appendixes: learn how it works internally, and you'll be empowered to understand and reason about the consequences.

## MySQL as an Infrastructure Component

MySQL doesn't exist in a vacuum. It's part of an overall application stack, and you'll need to build a robust overall architecture for your application. The next set of chapters is about how to do that.

In Chapter 10, *Replication*, we discuss MySQL's killer feature: the ability to set up multiple servers that all stay in sync with a master server's changes. Unfortunately, replication is perhaps MySQL's most troublesome feature for some people. This doesn't have to be the case, and we show you how to ensure that it keeps running well.

Chapter 11, *Scaling MySQL* discusses what scalability is (it's not the same thing as performance), why applications and systems don't scale, and what to do about it. If you do it right, you can scale MySQL to suit nearly any purpose. Chapter 12, *High Availability* delves into a related-but-distinct topic: how to ensure that MySQL stays up and functions smoothly. In Chapter 13, *MySQL in the Cloud*, you'll learn about what's different when you run MySQL in cloud computing environments.

In Chapter 14, *Application-Level Optimization*, we explain what we call *full-stack optimization*—optimization from the frontend to the backend, all the way from the user's experience to the database.

The best-designed, most scalable architecture in the world is no good if it can't survive power outages, malicious attacks, application bugs or programmer mistakes, and other disasters. That's why Chapter 15, *Backup and Recovery* discusses various backup and recovery strategies for your MySQL databases. These strategies will help minimize your downtime in the event of inevitable hardware failure and ensure that your data survives such catastrophes.

### Miscellaneous Useful Topics

In the last chapter and the book's appendixes, we delve into several topics that either don't fit well into any of the earlier chapters, or are referenced often enough in multiple chapters that they deserve a bit of special attention.

Chapter 16, *Tools for MySQL Users* explores some of the open source and commercial tools that can help you manage and monitor your MySQL servers more efficiently.

Appendix A introduces the three major unofficial versions of MySQL that have arisen over the last few years, including the one that our company maintains. It's worth knowing what else is available; many problems that are difficult or intractable with MySQL are solved elegantly by one of the variants. Two of the three (Percona Server and MariaDB) are drop-in replacements, so the effort involved in trying them out is not large. However, we hasten to add that we think most users are well served by sticking with the official MySQL distribution from Oracle.

Appendix B shows you how to inspect your MySQL server. Knowing how to get status information from the server is important; knowing what that information means is even more important. We cover `SHOW INNODB STATUS` in particular detail, because it provides deep insight into the operations of the InnoDB transactional storage engine. There is a lot of discussion of InnoDB's internals in this appendix.

Appendix C shows you how to copy very large files from place to place efficiently—a must if you are going to manage large volumes of data. Appendix D shows you how to really use and understand the all-important `EXPLAIN` command. Appendix E shows you how to decipher what's going on when queries are requesting locks that interfere with each other. And finally, Appendix F is an introduction to Sphinx, a high-performance, full-text indexing system that can complement MySQL's own abilities.

## Software Versions and Availability

MySQL is a moving target. In the years since Jeremy wrote the outline for the first edition of this book, numerous releases of MySQL have appeared. MySQL 4.1 and 5.0 were available only as alpha versions when the first edition went to press, but today MySQL 5.1 and 5.5 are the backbone of many large online applications. As we completed this third edition, MySQL 5.6 was the unreleased bleeding edge.

We didn't rely on a single version of MySQL for this book. Instead, we drew on our extensive collective knowledge of MySQL in the real world. The core of the book is focused on MySQL 5.1 and MySQL 5.5, because those are what we consider the "current" versions. Most of our examples assume you're running some reasonably mature version of MySQL 5.1, such as MySQL 5.1.50 or newer or newer. We have made an effort to note features or functionalities that might not exist in older releases or that might exist only in the upcoming 5.6 series. However, the definitive reference for mapping features to specific versions is the MySQL documentation itself. We expect that

you'll find yourself visiting the annotated online documentation (*http://dev.mysql.com/doc/*) from time to time as you read this book.

Another great aspect of MySQL is that it runs on all of today's popular platforms: Mac OS X, Windows, GNU/Linux, Solaris, FreeBSD, you name it! However, we are biased toward GNU/Linux[1] and other Unix-like operating systems. Windows users are likely to encounter some differences. For example, file paths are completely different on Windows. We also refer to standard Unix command-line utilities; we assume you know the corresponding commands in Windows.[2]

Perl is the other rough spot when dealing with MySQL on Windows. MySQL comes with several useful utilities that are written in Perl, and certain chapters in this book present example Perl scripts that form the basis of more complex tools you'll build. Percona Toolkit—which is indispensable for administering MySQL—is also written in Perl. However, Perl isn't included with Windows. In order to use these scripts, you'll need to download a Windows version of Perl from ActiveState and install the necessary add-on modules (`DBI` and `DBD::mysql`) for MySQL access.

## Conventions Used in This Book

The following typographical conventions are used in this book:

*Italic*
: Used for new terms, URLs, email addresses, usernames, hostnames, filenames, file extensions, pathnames, directories, and Unix commands and utilities.

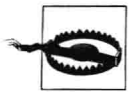
`Constant width`
: Indicates elements of code, configuration options, database and table names, variables and their values, functions, modules, the contents of files, or the output from commands.

**`Constant width bold`**
: Shows commands or other text that should be typed literally by the user. Also used for emphasis in command output.

*`Constant width italic`*
: Shows text that should be replaced with user-supplied values.

This icon signifies a tip, suggestion, or general note.

1. To avoid confusion, we refer to Linux when we are writing about the kernel, and GNU/Linux when we are writing about the whole operating system infrastructure that supports applications.
2. You can get Windows-compatible versions of Unix utilities at *http://unxutils.sourceforge.net* or *http://gnuwin32.sourceforge.net*.



This icon indicates a warning or caution.

## Using Code Examples

This book is here to help you get your job done. In general, you may use the code in this book in your programs and documentation. You don't need to contact us for permission unless you're reproducing a significant portion of the code. For example, writing a program that uses several chunks of code from this book doesn't require permission. Selling or distributing a CD-ROM of examples from O'Reilly books *does* require permission. Answering a question by citing this book and quoting example code doesn't require permission. Incorporating a significant amount of example code from this book into your product's documentation *does* require permission.

Examples are maintained on the site *http://www.highperfmysql.com* and will be updated there from time to time. We cannot commit, however, to updating and testing the code for every minor release of MySQL.

We appreciate, but don't require, attribution. An attribution usually includes the title, author, publisher, and ISBN. For example: "*High Performance MySQL, Third Edition*, by Baron Schwartz et al. (O'Reilly). Copyright 2012 Baron Schwartz, Peter Zaitsev, and Vadim Tkachenko, 978-1-449-31428-6."

If you feel your use of code examples falls outside fair use or the permission given above, feel free to contact us at *permissions@oreilly.com*.

## Safari® Books Online

Safari Books Online (*www.safaribooksonline.com*) is an on-demand digital library that delivers expert content in both book and video form from the world's leading authors in technology and business. Technology professionals, software developers, web designers, and business and creative professionals use Safari Books Online as their primary resource for research, problem solving, learning, and certification training.

Safari Books Online offers a range of product mixes and pricing programs for organizations, government agencies, and individuals. Subscribers have access to thousands of books, training videos, and prepublication manuscripts in one fully searchable database from publishers like O'Reilly Media, Prentice Hall Professional, Addison-Wesley Professional, Microsoft Press, Sams, Que, Peachpit Press, Focal Press, Cisco Press, John Wiley & Sons, Syngress, Morgan Kaufmann, IBM Redbooks, Packt, Adobe Press, FT Press, Apress, Manning, New Riders, McGraw-Hill, Jones & Bartlett, Course Technology, and dozens more. For more information about Safari Books Online, please visit us online.

## How to Contact Us

Please address comments and questions concerning this book to the publisher:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
800-998-9938 (in the United States or Canada)
707-829-0515 (international or local)
707-829-0104 (fax)

We have a web page for this book, where we list errata, examples, and any additional information. You can access this page at:

*http://shop.oreilly.com/product/0636920022343.do*

To comment or ask technical questions about this book, send email to:

*bookquestions@oreilly.com*

For more information about our books, conferences, Resource Centers, and the O'Reilly Network, see our website at:

*http://www.oreilly.com*

Find us on Facebook: *http://facebook.com/oreilly*

Follow us on Twitter: *http://twitter.com/oreillymedia*

Watch us on YouTube: *http://www.youtube.com/oreillymedia*

You can also get in touch with the authors directly. You can use the contact form on our company's website at *http://www.percona.com*. We'd be delighted to hear from you.

## Acknowledgments for the Third Edition

Thanks to the following people who helped in various ways: Brian Aker, Johan Andersson, Espen Braekken, Mark Callaghan, James Day, Maciej Dobrzanski, Ewen Fortune, Dave Hildebrandt, Fernando Ipar, Haidong Ji, Giuseppe Maxia, Aurimas Mikalauskas, Istvan Podor, Yves Trudeau, Matt Yonkovit, and Alex Yurchenko. Thanks to everyone at Percona for helping in dozens of ways over the years. Thanks to the many great bloggers[3] and speakers who gave us a great deal of food for thought, especially Yoshinori Matsunobu. Thanks also to the authors of the previous editions: Jeremy D. Zawodny, Derek J. Balling, and Arjen Lentz. Thanks to Andy Oram, Rachel Head, and the whole O'Reilly staff who do such a classy job of publishing books and running conferences. And much gratitude to the brilliant and dedicated MySQL team inside

3. You can find a wealth of great technical blogging on *http://planet.mysql.com*.

Oracle, as well as all of the ex-MySQLers, wherever you are, and especially to SkySQL and Monty Program.

Baron thanks his wife Lynn, his mother, Connie, and his parents-in-law, Jane and Roger, for helping and supporting this project in many ways, but most especially for their encouragement and help with chores and taking care of the family. Thanks also to Peter and Vadim for being such great teachers and colleagues. Baron dedicates this edition to the memory of Alan Rimm-Kaufman, whose great love and encouragement are never forgotten.

## Acknowledgments for the Second Edition

Sphinx developer Andrew Aksyonoff wrote Appendix F. We'd like to thank him first for his in-depth discussion.

We have received invaluable help from many people while writing this book. It's impossible to list everyone who gave us help—we really owe thanks to the entire MySQL community and everyone at MySQL AB. However, here's a list of people who contributed directly, with apologies if we've missed anyone: Tobias Asplund, Igor Babaev, Pascal Borghino, Roland Bouman, Ronald Bradford, Mark Callaghan, Jeremy Cole, Britt Crawford and the HiveDB Project, Vasil Dimov, Harrison Fisk, Florian Haas, Dmitri Joukovski and Zmanda (thanks for the diagram explaining LVM snapshots), Alan Kasindorf, Sheeri Kritzer Cabral, Marko Makela, Giuseppe Maxia, Paul McCullagh, B. Keith Murphy, Dhiren Patel, Sergey Petrunia, Alexander Rubin, Paul Tuckfield, Heikki Tuuri, and Michael "Monty" Widenius.

A special thanks to Andy Oram and Isabel Kunkle, our editor and assistant editor at O'Reilly, and to Rachel Wheeler, the copyeditor. Thanks also to the rest of the O'Reilly staff.

### From Baron

I would like to thank my wife, Lynn Rainville, and our dog, Carbon. If you've written a book, I'm sure you know how grateful I am to them. I also owe a huge debt of gratitude to Alan Rimm-Kaufman and my colleagues at the Rimm-Kaufman Group for their support and encouragement during this project. Thanks to Peter, Vadim, and Arjen for giving me the opportunity to make this dream come true. And thanks to Jeremy and Derek for breaking the trail for us.

### From Peter

I've been doing MySQL performance and scaling presentations, training, and consulting for years, and I've always wanted to reach a wider audience, so I was very excited when Andy Oram approached me to work on this book. I have not written a book before, so I wasn't prepared for how much time and effort it required. We first started

talking about updating the first edition to cover recent versions of MySQL, but we wanted to add so much material that we ended up rewriting most of the book.

This book is truly a team effort. Because I was very busy bootstrapping Percona, Vadim's and my consulting company, and because English is not my first language, we all had different roles. I provided the outline and technical content, then I reviewed the material, revising and extending it as we wrote. When Arjen (the former head of the MySQL documentation team) joined the project, we began to fill out the outline. Things really started to roll once we brought in Baron, who can write high-quality book content at insane speeds. Vadim was a great help with in-depth MySQL source code checks and when we needed to back our claims with benchmarks and other research.

As we worked on the book, we found more and more areas we wanted to explore in more detail. Many of the book's topics, such as replication, query optimization, InnoDB, architecture, and design could easily fill their own books, so we had to stop somewhere and leave some material for a possible future edition or for our blogs, presentations, and articles.

We got great help from our reviewers, who are the top MySQL experts in the world, from both inside and outside of MySQL AB. These include MySQL's founder, Michael Widenius; InnoDB's founder, Heikki Tuuri; Igor Babaev, the head of the MySQL optimizer team; and many others.

I would also like to thank my wife, Katya Zaytseva, and my children, Ivan and Nadezhda, for allowing me to spend time on the book that should have been Family Time. I'm also grateful to Percona's employees for handling things when I disappeared to work on the book, and of course to Andy Oram and O'Reilly for making things happen.

## From Vadim

I would like to thank Peter, who I am excited to have worked with on this book and look forward to working with on other projects; Baron, who was instrumental in getting this book done; and Arjen, who was a lot of fun to work with. Thanks also to our editor Andy Oram, who had enough patience to work with us; the MySQL team that created great software; and our clients who provide me the opportunities to fine-tune my MySQL understanding. And finally a special thank you to my wife, Valerie, and our sons, Myroslav and Timur, who always support me and help me to move forward.

## From Arjen

I would like to thank Andy for his wisdom, guidance, and patience. Thanks to Baron for hopping on the second edition train while it was already in motion, and to Peter and Vadim for solid background information and benchmarks. Thanks also to Jeremy and Derek for the foundation with the first edition; as you wrote in my copy, Derek: "Keep 'em honest, that's all I ask."

Also thanks to all my former colleagues (and present friends) at MySQL AB, where I acquired most of what I know about the topic; and in this context a special mention for Monty, whom I continue to regard as the proud parent of MySQL, even though his company now lives on as part of Sun Microsystems. I would also like to thank everyone else in the global MySQL community.

And last but not least, thanks to my daughter Phoebe, who at this stage in her young life does not care about this thing called "MySQL," nor indeed has she any idea which of The Wiggles it might refer to! For some, ignorance is truly bliss, and they provide us with a refreshing perspective on what is really important in life; for the rest of you, may you find this book a useful addition on your reference bookshelf. And don't forget your life.

## Acknowledgments for the First Edition

A book like this doesn't come into being without help from literally dozens of people. Without their assistance, the book you hold in your hands would probably still be a bunch of sticky notes on the sides of our monitors. This is the part of the book where we get to say whatever we like about the folks who helped us out, and we don't have to worry about music playing in the background telling us to shut up and go away, as you might see on TV during an awards show.

We couldn't have completed this project without the constant prodding, begging, pleading, and support from our editor, Andy Oram. If there is one person most responsible for the book in your hands, it's Andy. We really do appreciate the weekly nag sessions.

Andy isn't alone, though. At O'Reilly there are a bunch of other folks who had some part in getting those sticky notes converted to a cohesive book that you'd be willing to read, so we also have to thank the production, illustration, and marketing folks for helping to pull this book together. And, of course, thanks to Tim O'Reilly for his continued commitment to producing some of the industry's finest documentation for popular open source software.

Finally, we'd both like to give a big thanks to the folks who agreed to look over the various drafts of the book and tell us all the things we were doing wrong: our reviewers. They spent part of their 2003 holiday break looking over roughly formatted versions of this text, full of typos, misleading statements, and outright mathematical errors. In no particular order, thanks to Brian "Krow" Aker, Mark "JDBC" Matthews, Jeremy "the other Jeremy" Cole, Mike "VBMySQL.com (*http://vbmysql.com*)" Hillyer, Raymond "Rainman" De Roo, Jeffrey "Regex Master" Friedl, Jason DeHaan, Dan Nelson, Steve "Unix Wiz" Friedl, and, last but not least, Kasia "Unix Girl" Trapszo.