

<<高性能MySQL>>

图书基本信息

书名：<<高性能MySQL>>

13位ISBN编号：9787121198854

10位ISBN编号：7121198851

出版时间：2013-5

出版时间：电子工业出版社

作者：(美) 施瓦茨(Schwartz,B.),(美) 扎伊采夫(Zaitsev,P.),(美) 特卡琴科(Tkachenko,V.)

译者：宁海元,周振兴,彭立勋,翟卫祥,刘辉

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

前言

我们写这本书不仅仅是为了满足MySQL应用开发者的需求，也是为了满足MySQL数据库管理员的需要。

我们假定读者已经有了一定的MySQL基础。

我们还假定读者对于系统管理、网络和类Unix的操作系统都有一些了解。

本书的第二版为读者提供了大量的信息，但没有一本书是可以涵盖一个主题的所有方面的。

在第二版和第三版之间的这段时间里，我们记录了数以千计有趣的问题，其中有些是我们解决的，也有一些是我们观察到其他人解决的。

当我们在规划第三版的时候发现，如果要把这些主题完全覆盖，可能三千页到五千页的篇幅都还不够，这样本书的完成就遥遥无期了。

在反思这个问题后，我们意识到第二版强调的广泛的覆盖度事实上有其自身的限制，从某种意义上来说也没有引导读者如何按照MySQL的方式来思考问题。

所以第三版和第二版的关注点有很大的不同。

我们虽然还是会包含很多的信息，并且会强调同样的诸如可靠性和正确性的目标，但我们也会在本书中尝试更深入的讨论：我们会指出MySQL为什么会这样做，而不是MySQL做了什么。

我们会使用更多的演示和案例学习来将上述原则落地。

通过这样的方式，我们希望能够尝试回到下面这样的问题：“给出MySQL的内部结构和操作，对于实际应用能带来什么帮助？”

为什么能有这样的帮助？

如何让MySQL适合（或者不适合）特定的需求？

”最后，我们希望关于MySQL内部原理的知识能够帮助大家解决本书没有覆盖到的一些情况。

我们更希望读者能培养发现新问题的洞察力，能学习和实践合理的方式来设计、维护和诊断基于MySQL的系统。

本书是如何组织的本书涵盖了许多复杂的主题。

在这里，我们将解释一下是如何将这些主题有序地组织在一起的，以便于阅读和学习。

概述第1章是非常基础的一章，在更深入地学习之前建议先熟悉一下这部分内容。

在有效地使用MySQL之前应当理解它是如何组织的。

本章解释了MySQL的架构及其存储引擎的关键设计。

如果读者还不太熟悉关系数据库和事务的基础知识，本章也可以带来一点帮助。

如果之前已经对其他关系数据库如Oracle比较熟悉，本章也可以帮助读者了解MySQL的入门知识。

本章还包括了一点MySQL的历史背景：MySQL随着时间的演进、最近的公司所有权更替，以及我们认为比较重要的内容。

打造坚实的基础本书前几章的内容在今后使用MySQL的过程中可能会被不断地引用到，它们是非常基础的内容。

第2章讨论了基准测试的基础，例如服务器可以处理的工作负载的类型、处理特定任务的速度等。

基准测试是一项至关重要的技能，可用于评估服务器在不同负载下的表现，但也要明白在什么情况下基准测试不能发挥作用。

第3章介绍了我们常用于故障诊断和服务器性能问题分析的一种面向响应时间的方法。

该方法已经被证明可以解决我们曾碰到过的一些极为棘手的问题。

当然也可以选择修改我们所使用的方法（实际上我们的方法也是从Cary Millsap的方法修改而来的），但无论如何，至少不能没有方法胡乱猜测。

从第4章到第6章，连续介绍了三个关于良好的数据库逻辑设计和物理设计基础的话题。

第4章涵盖了不同数据类型的细节差别以及表设计的原则。

第5章则展开讨论了索引，这是数据库的物理设计。

对于索引的深入理解和利用是高效使用MySQL的基础，相信这一章会经常需要回头翻看。

而第6章则包含了分析MySQL的查询是如何执行的，以及如何利用查询优化器的话题。

<<高性能MySQL>>

该章也包含了大量常见类型查询的例子，演示了MySQL是如何做好工作的，以及如何改写查询以利用MySQL的特性。

到此为止，已经覆盖了关于数据库的基础内容：表、索引、数据和查询。

第7章则在MySQL基础知识之外介绍了MySQL的高级特性是如何工作的。

这章的内容包括分区、存储引擎、触发器，以及字符集。

MySQL中这些特性的实现可能不同于其他数据库，可能之前读者并不清楚这些不同，因此理解它们对于性能可能会带来新的收益。

配置应用程序接下来的两章讲述的是如何让MySQL、应用程序及硬件一起很好地工作。

第8章介绍了如何配置MySQL，以便更好地利用硬件，达到更好的可靠性和鲁棒性。

第9章解释了如何让操作系统和硬件工作得更好。

另外也深入讨论了固态硬盘，为高可扩展性应用发挥更好的性能提供了硬件配置的建议。

上面两章都一定程度地涉及了MySQL的内部知识。

这将会是一个反复出现的主题，附录中也会有相关内容可以学习到MySQL的内部是如何实现的，理解了这些知识将帮助读者更好地理解某些现象背后的原理。

作为基础设施组件的MySQL不是存在于真空中的，而是应用整体的一个环节，因此需要考虑整个应用架构的鲁棒性。

下面的章节将告诉我们该如何做到这一点。

第10章讨论了MySQL的杀手级特性：能够设置多个服务器从一台主服务器同步数据。

不幸的是，复制可能也是MySQL给很多用户带来困扰的一个特性。

但实际上不应该发生这样的情况，本章将告诉你如何让复制运行得更好。

第11章讨论了什么是可扩展性（这和性能不是一回事），应用和系统为什么会无法扩展，该怎么改善扩展性。

如果能够正确地处理，MySQL的可扩展性是足以应付任何需求的。

第12章讲述的是和可扩展性相关但又完全不同的主题：如何保障MySQL稳定而正确地持续运行。

第13章将告诉你当MySQL在云计算环境中运行时会有什么不同的事情发生。

第14章解释了什么是全方位的优化（full-stack optimization），就是从前端到后端的整体优化，从用户体验开始直到数据库。

即使是世界上设计最好、最具可扩展性的架构，如果停电会导致彻底崩溃，无法抵御恶意攻击，解决不了应用的bug和程序员的错误，以及其他一些灾难场景，那就不是什么好的架构。

第15章讨论了MySQL数据库各种备份与恢复的场景。

这些策略可以帮助读者减少在各种不可抗的硬件失效时的宕机时间，保证在各种灾难下的数据最终可恢复。

其他有用的主题在本书的最后一章以及附录中，我们探讨了一些无法明确地放到前面章节的内容，以及一些被前面多个章节引用而需要特别注意的主题。

第16章探索了一些可以帮助用户更有效地管理和监控MySQL服务器的工具，有些是开源的，也有些是商业的。

附录A介绍了近年来成长迅速的三个主要的非MySQL官方版本，其中一个是我们公司在维护的产品。

知道还有其他什么是可用的选择是有价值的；很多MySQL难以解决的棘手问题在其他的变种版本中说不定就不是问题了。

这三个版本中的两个（Percona Server和MariaDB）是MySQL的完全可替换版本，所以尝试使用的成本相对来说是很低的。

当然，在这里我们也需要补充一点，Oracle提供的MySQL官方版本对于大多数用户来说都能服务得很好。

附录B演示了如何检查MySQL服务器。

知道如何从服务器获取状态信息是非常重要的；而了解这些状态代表的意义则更加重要。

这里将覆盖SHOW INNODB STATUS的输出结果，因此这里包含了InnoDB事务存储引擎的深入信息。

。

<<高性能MySQL>>

在这个附录中讨论了很多InnoDB的内部信息。

附录C 演示了如何高效地将大文件从一个地方复制到另外一个地方。

如果要管理大量的数据，这种操作是经常都会碰到的。

附录D 演示了如何真正地使用并理解EXPLAIN 命令。

附录E 演示了如何破除不同查询所请求的锁互相干扰的问题。

最后，附录F 介绍了Sphinx，一个基于MySQL 的高性能的全文索引系统。

软件版本与可用性MySQL 是一个移动靶。

从Jeremy 写作本书第一版到现在，MySQL 已经发布了好几个版本。

当本书第一版的初稿交给出版社的时候，MySQL 4.1 和5.0 还只是alpha 版本，而如今MySQL 5.1 和5.5 已经是很多在线应用的主力版本。

在我们写完这第三版的时候，MySQL 5.6 也即将发布。

本书的内容并不依赖某一个具体的版本。

相反，我们会利用自己在实际环境中获得的更广泛的知识。

本书的核心内容主要关注MySQL 5.1 和5.5 版本，因为我们认为这是“当前”的版本。

本书的大多数例子都假设运行在MySQL 5.1 的某个成熟版本上，比如MySQL 5.1.50 或者更高的版本。

对于在旧版本中可能不存在，或者只在即将到来的5.6 版本中出现的特性或者功能，我们也会特别标注出来。

然而，关于某个MySQL 版本的特性的权威指南还是要看官方文档。

在阅读本书时，建议随时访问在线官方文档的相关内容（<http://dev.mysql.com/doc/>）。

MySQL 的另外一个伟大特点是能够运行在现今流行的所有平台：Mac OS X，Windows, GNU/Linux, Solaris, FreeBSD，以及只要你能举出名字的其他平台。

然而，本书主要基于GNU/Linux和其他类Unix 系统。

Windows 的用户可能会碰到一些困难。

比如说文件路径就和Windows 完全不一样。

我们也会引用一些Unix 的命令行工具，我们假设读者能够知道Windows 上对应的工具是什么。

在Windows 上搞MySQL 的另外一个难点是Perl。

MySQL 中有很多有用的工具是用Perl 写的。

在本书的一些章节中，也有一些Perl 脚本，在此基础上可以构建更加复杂的工具。

Percona Toolkit 是不可多得的MySQL 管理工具，也是用Perl 写的。

然而，Windows 平台默认是没有Perl 环境的。

为了使用这些工具，需要从ActiveState 下载Perl的Windows 版本，以及访问MySQL 所需要的一些额外的模块（DBI 和DBD::MySQL）。

本书使用的约定下面是本书中使用的一些约定。

斜体 (Italic) 新的名字、URL、邮件地址、用户名、主机名、文件名、文件扩展名、路径名、目录，以及Unix 命令和工具都使用斜体表示。

等宽字体 (Constant width) 包括代码元素、配置选项、数据库和表名、变量和值、函数、模块、文件内容、命令输出等，使用的是等宽字体。

加粗的等宽字体 (Constant width bold) 命令或者其他需要用户输入的文本，命令输出中需要强调的某些内容，会使用加粗的等宽字体。

斜体的等宽字体 (Constant width italic) 需要用户替换的文本以斜体的等宽字体表示。

使用示例代码本书的目标是为了帮助读者更好地工作。

一般来说，你可以在程序或者文档中使用本书中的代码。

只要不是大规模地复制重要的代码，使用的时候不需要联系我们。

例如，你编写的程序中如果只是使用了本书部分的代码片段则无须取得授权，而出售或者分发O

' Reilly 书籍示例代码的CD-ROM 盘片则需要经过授权。

引用本书的代码回答问题也无须取得授权，而大量引用本书的示例代码到产品文档中则需要获取授权。

示例代码维护在<http://www.highperfmysql.com> 站点中，会及时保持更新。

但我们无法确保代码会跟随每一个MySQL的小版本进行更新和测试。

我们欢迎大家在使用了本书代码后进行反馈，但这不是一个强制要求。

反馈时请提供标题、作者、出版公司和ISBN。

例如：“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”。

如果你使用了本书的代码，但又不在上面描述的一些无须授权的范围之内，不确定是否需要获取授权时，请联系permissions@oreilly.com。

Safari 在线书店Safari 在线书店 (www.safaribooksonline.com) 是一家提供定制服务的数字图书馆，提供技术和商务领域内顶级作家的高质量内容的书籍和音像制品。

很多技术专家、软件开发者、Web 设计师、商务人士和创新专家都将Safari 在线书店作为他们研究、解决问题、学习和认证练习的首选资料来源。

Safari 在线书店为组织、政府机构和个人提供了一系列的产品组合和定价计划。

订阅者可以访问数以千计的图书、培训视频和手稿，这些存在于一个可搜索的数据库中，涵盖的出版公司有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, 等等。

如需了解更多关于Safari 在线书店的情况，请访问在线网站。

如何联系我们若有关于本书的任何评论或者问题，请和出版公司联系：O’Reilly Media, Inc.1005

Gravenstein Highway North Sebastopol, CA 95472800-998-9938 (in the United States or Canada)707-829-0515

(international or local)707-829-0104 (fax)本书有一个配套的网页，上面列出了勘误表、示例代码及其他相关信息。

下面是此网页的地址：<http://shop.oreilly.com/product/0636920022343.do>如果有关于本书的评论和技术问题，也可以通过邮件进行沟通：bookquestions@oreilly.com如果了解更多关于我们出版公司的书籍、

会议、资源中心和O’Reilly网络的信息，请访问网站：<http://www.oreilly.com>我们的Facebook

：<http://facebook.com/oreilly>我们的Twitter：<http://twitter.com/oreillymedia>我们的YouTube

：<http://www.youtube.com/oreillymedia>当然，读者也可以直接和作者取得联系，可以访问作者的公司网站<http://www.percona.com>。

我们将乐于收到大家的反馈。

本书第三版的致谢感谢以下人员给予的各种帮助：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, Alex Yurchenko。

感谢Percona公司的所有员工，多年来为本书提供了无数的支持。

感谢很多著名博主和技术大会的演讲者，他们为本书的很多思想提供了大量的素材，尤其是Yoshinori Matsunobu。

另外也要感谢本书前面两版的作者：Jeremy D.Zawodny、Derek J. Balling 和Arjen Lentz。

感谢Andy Oram、Rachel Head，以及O’Reilly的整个编辑团队，你们为本书的出版和发行做了卓有成效的工作。

非常感谢Oracle的才华横溢且专注的MySQL团队，以及所有之前的MySQL开发者，不管你现在是在SkySQL还是在Monty团队。

Baron也要感谢他的妻子Lynn、他的母亲Connie，以及他的岳父母Jane和Roger，感谢他们一如既往地支持他的工作，尤其是不断地鼓励他，并且承担了所有的家务和照顾整个家庭的重任。

也要感谢Peter和Vadim，你们是如此优秀的老师和同事。

Baron将此版本献给Alan Rimm-Kaufman，以纪念他给予的伟大的爱和鼓励，这些都将永志不忘。

本书第二版的致谢Sphinx的开发者Andrew Aksyonoff编写了附录F。

我们非常感谢他首次对此进行深入的讨论。

<<高性能MySQL>>

在编写本书的时候，我们得到了很多人的无私帮助。

在此无法一一列举——我们真的非常感谢MySQL社区和MySQL AB公司的每一个人。

下面是对本书做出了直接贡献的人，如有遗漏，还请见谅。

他们是：Tobias Asplund, Igor Babaev, Pascal Borghino, Roland Bouman, Ronald Bradford, Mark Callaghan, Jeremy Cole, Britt Crawford 和他的HiveDB项目，Vasil Dimov, Harrison Fisk, Florian Haas, Dmitri Joukovski 和他的Zmanda项目（同时感谢Dmitri为解释LVM快照提供的配图），Alan Kasindorf, Sheeri Kritzer Cabral, Marko Makela, Giuseppe Maxia, Paul McCullagh, B. Keith Murphy, Dhiren Patel, Sergey Petrunia, Alexander Rubin, Paul Tuckfield, Heikki Tuuri, 以及 Michael “Monty” Widenius。

在这里还要特别感谢O’Reilly的编辑Andy Oram和助理编辑Isabel Kunkle，以及审稿人Rachel Wheeler，同时也要感谢O’Reilly团队的其他所有成员。

来自Baron我要感谢我的妻子Lynn Rainville和小狗Carbon。

如果你也曾写过一本书，我相信你就能体会到我是如何地感谢他们。

我也非常感谢Alan Rimm-Kaufman和我在Rimm-Kaufman集团的同事，在写书的过程中，他们给了我支持和鼓励。

谢谢Peter、Vadim和Arjen，是你们给了我梦想成真的机会。

最后，我要感谢Jeremy和Derek为我们开了个好头。

来自Peter我从事MySQL性能和可扩展性方面的演讲、培训和咨询工作已经很多年了，我一直想把它们扩展到更多的受众。

因此，当Andy Oram加入到本书的编写当中时，我感到非常兴奋。

此前我没有写过书，所以我对所需要的时间和精力都毫无把握。

一开始我们谈到只对第一版做一些更新，以跟上MySQL最新的版本升级，但我们想把更多新素材加入到书中，结果几乎相当于重写了整本书。

这本书是真正的团队合作的结晶。

因为我忙于Percona公司的事情——这是我和Vadim的咨询公司，而且英语并非我的第一语言，所以我们有不同的角色分工。

我负责提供大纲和技术性内容，评审所有的材料，在写作的时候再进行修订和扩展。

当Arjen（MySQL文档团队的前负责人）加入之后，我们就开始勾画出整个提纲。

在Baron加入后，一切才开始真正行动起来，他能够以不可思议的速度编写出高质量的内容。

Vadim则在深入检查MySQL源代码和提供基准测试及其他研究来巩固我们的论点时提供了巨大的帮助。

当我们编写本书时，我们发现越来越多的领域需要刨根问底。

本书的大部分主题，如复制、查询优化、InnoDB、架构和设计都足以单独成书。

因此，有时候我们不得不在某个点停止深入，把余下的材料用在将来可能出版的新版本中，或者我们的博客、演讲和技术文章中。

本书的评审者给了我们非常大的帮助，无论是来自MySQL AB公司内部的人员，还是外部的人员，他们都是MySQL领域最优秀的世界级专家。

其中包括MySQL的创建者Michael Widenius、InnoDB的创建者Heikki Tuuri、MySQL优化器团队的负责人Igor Babaev，以及其他人员。

我还要感谢我的妻子Katya Zaytseva，我的孩子Ivan和Nadezhda，他们允许我把家庭时间花在了本书的写作上。

我也要感谢Percona的员工，当我在公司里“人间蒸发”去写书时，他们承担了日常事务的处理工作。当然，我也要感谢O’Reilly和Andy Oram让这一切成为可能。

来自Vadim我要感谢Peter，能够在本书中和他合作，我感到十分开心，期望在其他项目中能继续共事。

我也要感谢Baron，他在本书的写作过程中起了很大的作用。

还有Arjen，跟他一起工作非常好玩。

我还要感谢我们的编辑Andy Oram，他抱着十二万分的耐心和我们一起工作。

<<高性能MySQL>>

此外，还要感谢MySQL 团队，是他们创造了这个伟大的软件。

我还要感谢我们的客户给予我调优MySQL 的机会。

最后，我要特别感谢我的妻子Valerie，以及我们的孩子Myroslav 和Timur，他们一直支持我，帮助我一步步前进。

来自Arjen我要感谢Andy 的睿智、指导和耐心，感谢Baron 中途加入到我们当中来、感谢Peter 和Vadim 坚实的背景信息和基准测试。

也要感谢Jeremy 和Derek 在第一版中打下的基础。

在我的书上，Derak 题写着：“要诚实——这就是我的所有要求”。

我也要感谢我在MySQL AB 公司时的所有同事，在那里我获得了关于本书主题的大多数知识。

在此，我还要特别提到Monty，我一直认为他是令人自豪的MySQL 之父，尽管他的公司如今已经成为Sun 公司的一部分。

我要感谢全球MySQL 社区里的每一个人。

最后同样重要的是，我要感谢我的女儿Phoebe，在她尚年少的生活舞台上，不用关心什么是MySQL，也不用考虑Wiggles 指的是什么东西。

从某些方面来讲，无知就是福。

它能给予我们一个全新的视角来看清生命中真正重要的是什么。

对于读者，祝愿你们的书架上又增添了一本有用的书，还有，不要忘记你的生活。

本书第一版的致谢要完成这样一本书的写作，离不开许许多多人的帮助。

没有他们的无私援助，你手上的这本书就可能仍然是我们显示器屏幕四周的那一堆贴纸。

这是本书的一部分，在这里，我们可以感谢每一个曾经帮助我们脱离困境的人，而无须担心突然奏响的背景音乐催促我们闭上嘴巴赶紧走掉——如同你在电视里看到的颁奖晚会那样。

如果没有编辑Andy Oram 坚决的督促、请求、央求和支持，我们就无法完成本书。

如果要找对于本书最负责的一个人，那就是Andy。

我们真的非常感谢每周一次的唠唠叨叨的会议。

然而，Andy 不是一个人在战斗。

在O'Reilly，还有一批人都参与了将那些小贴纸变成你正在看的这本书的工作。

所以我也要感谢那些在生产、插画和销售环节的人们，感谢你们把这本书变成实体。

当然，也要感谢Tim O'Reilly，是他持久不变地承诺为广大开源软件出版一批业内最好的图书。

最后，我们要把感谢给予那些同意审阅本书不同版本草稿，并告诉我们哪里有错误的人们：我们的评审者。

他们把2003 年假期的一部分时间用在了审阅这些格式粗糙，充满了打字符号、误导性的语句和彻底的数学错误的文本上。

我们要感谢（排名不分先后）：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, Kasia “Unix Girl” Trapszo。

来自Jeremy我要在此感谢Andy，是他同意接纳这个项目，并持续不断地鞭策我们加入新的章节内容。Derek 的帮助也非常关键，本书最后的20% ~ 30% 内容由他一手完成，这使得我们没有错过下一个目标日期。

感谢他同意中途加入进来，代替我只能偶尔爆发一下的零星生产力，完成了关于XML 的烦琐工作、第10 章、附录F，以及我丢给他的那些活儿。

我也要感谢我的父母，在多年以前他们就给我买了Commodore 64 电脑，他们不仅在前10 年里容忍了我就像要以身相许般的对电子和计算机技术的痴迷，并在之后还成为我不懈学习和探索的支持者。

接下来，我要感谢在过去几年里在Yahoo！

布道推广MySQL 时遇到的那一群人。

跟他们共事，我感到非常愉快。

在本书的筹备阶段，Jeffrey Friedl 和Ray Goldberger 给了我鼓励和反馈意见。

<<高性能MySQL>>

在他们之后还有Steve Morris、James Harvey 和Sergey Kolychev 容忍了我在Yahoo! Finance MySQL 服务器上做着看似固定不变的实验，即使这打扰了他们的重要工作。

我也要感谢Yahoo! 的其他成员，是他们帮我发现了MySQL 上的那些有趣的问题和解决方法。

还有，最重要的是要感谢他们对我有足够的信任和信念，让我把MySQL 用在Yahoo! 重要和可见的部分业务上。

Adam Goodman，一位出版家和Linux Magazine 的拥有者，他帮助我轻装上阵为技术受众撰写文章，并在2001 年下半年第一次出版了我的MySQL 相关的长篇文章。

自那以后，他教授给我更多他所能认识到的关于编辑和出版的技能，还鼓励我通过在杂志上开设月度专栏在这条路上继续走下去。

谢谢你，Adam。

我要感谢Monty 和David，感谢你们与这个世界分享了MySQL。

说到MySQL AB，也要感谢在那里的其他“牛”人，是他们鼓励我写成本书：Kerry, Larry, Joe, Marten, Brian, Paul, Jeremy, Mark, Harrison, Matt，以及团队中的其他人。

他们真的非常棒。

最后，我要感谢我博客的读者，是他们鼓励我撰写基于日常工作的非正式的MySQL 及其他技术文章。

最后同样重要的是，感谢Goon Squad。

来自Derek就像Jeremy 一样，有太多同样的原因，我也要感谢我的家庭。

我要感谢我的父母，是他们不停地鼓励我去写一本书，哪怕他们头脑中都没有任何和它相关的东西。

我的祖父母给我上了两堂很有价值的课：美元的含义，以及我跟电脑相爱有多深，他们还借钱给我去购买了我平生第一台电脑：Commodore VIC-20。

我万分感谢Jeremy 邀请我加入他那旋风般的写作“过山车”中来。

这是一个很棒的体验，我希望将来还能跟他一起工作。

我要特别感谢Raymond De Roo, Brian Wohlgemuth, David Calafrancesco, Tera Doty, Jay Rubin, Bill Catlan, Anthony Howe, Mark O'Neal, George Montgomery, George Barber，以及其他无数耐心听我抱怨的人，我从他们那里了解到我所讲述的内容是否能让门外汉也能理解，或者仅仅得到一个我所希望的笑脸。

没有他们，这本书可能也能写出来，但我几乎可以肯定我会在这个过程中疯掉。

<<高性能MySQL>>

内容概要

《高性能mysql(第3版)》是mysql领域的经典之作，拥有广泛的影响力。

第3版更新了大量的内容，不但涵盖了最新mysql 5.5版本的新特性，也讲述了关于固态硬盘、高可扩展性设计和云计算环境下的数据库相关的新内容，原有的基准测试和性能优化部分也做了大量的扩展和补充。

全书共分为16章和6个附录，内容涵盖mysql架构和历史，基准测试和性能剖析，数据库软硬件性能优化，复制、备份和恢复，高可用与高可扩展性，以及云端的mysql和mysql相关工具等方面的内容。每一章都是相对独立的主题，读者可以有选择性地单独阅读。

《高性能mysql(第3版)》不但适合数据库管理员（dba）阅读，也适合开发人员参考学习。不管是数据库新手还是专家，相信都能从本书有所收获。

<<高性能MySQL>>

作者简介

作者：（美国）施瓦茨（美国）扎伊采夫（美国）特卡琴科 译者：宁海元 周振兴 彭立勋 Baron Schwartz 是一位软件工程师，居住在弗吉尼亚州的Charlottesville，网络常用名是Xaprb，这是按照QWERTY 键盘的顺序在Dvorak 键盘上打出来的名字。

在不忙于解决有趣的编程挑战时，Baron 会和他的妻子Lynn 以及小狗Carbon 一起享受闲暇的时光。他有一个软件工程方面的博客，地址是[http://www.xaprb.com/blog/Peter Zaitsev](http://www.xaprb.com/blog/Peter_Zaitsev) 曾经是MySQL AB 公司高性能组的经理，目前在运作mysqlperformance blog.com 网站。

他擅长于帮助那些每天有数以百万计访问量的网站的管理员解决问题，这些网站通常需要几百台机器来处理TB 级的数据。

他常常为了解决一个问题而不停地升级硬件和软件（比如查询优化）。

Peter 还经常在各种会议上演讲。

Vadim Tkachenko 曾经是MySQL AB 公司的性能工程师。

作为一名在多线程编程和同步方面的专家，他的主要工作是基准测试、性能剖析，以及找出系统的性能瓶颈。

他还在性能监控和调优方面做了一些工作，使得MySQL 在多核机器上有更好的可扩展性。

宁海元：有超过十年的数据库管理经验，从最初的SQL Server 2000 到Oracle 再到MySQL，擅长数据库高可用架构、性能优化和故障诊断。

2007 年加入淘宝，带领淘宝DBA 团队完成了数据库的垂直拆分、水平拆分，迁移到MySQL 体系等主要工作，为淘宝业务的快速增长提供支撑。

目前专注于无线数据领域。

网络常用名NinGoo，个人博客：<http://www.ningoo.net>周振兴：毕业于北京师范大学数学系，2009 年加入淘宝数据库团队，负责MySQL 运维管理工作，有丰富的MySQL 性能优化、Troubleshooting 经验，对MySQL 主要模块的实现和原理有深入的研究，经历了淘宝MySQL 实例从30 到3000 的发展，对系统架构、高可用环境规划都有深入理解。

个人博客：<http://orczhou.com>彭立勋：2010 年大学毕业后加入阿里巴巴运维部。

作为一名MySQL DBA，在运维MySQL 的过程中对MySQL 和InnoDB 的一些功能和缺陷进行了补充，编写了多主复制和数据闪回等补丁。

目前在阿里集团核心系统研发部数据库组，专注于MySQL 数据库相关的开发工作。

后来一些补丁被MySQL 之父Mony 看中，多主复制、线程内存监控等补丁被合并到了MariaDB 10.0 版本，本人也因此成为MariaDB 提交组（Maria-captains）成员。

翟卫祥：毕业于武汉大学，研究生阶段从事数据库相关研究。

毕业后就职于阿里巴巴集团数据库技术团队至今，主要负责阿里内部MySQL 代码分支维护，包括MySQL Bug Fix及新特性开发。

对MySQL 内核有一定的研究。

刘辉：2008 年毕业于西安电子科技大学计算机系，硕士学位。

2011 年加入阿里巴巴集团数据库技术团队，花名希羽，MySQL 内核开发工程师。

书籍目录

推荐序 前言 第1章MySQL架构与历史1 1.1MySQL逻辑架构1 1.1.1连接管理与安全性2 1.1.2优化与执行3 1.2并发控制3 1.2.1读写锁4 1.2.2锁粒度4 1.3事务6 1.3.1隔离级别8 1.3.2死锁9 1.3.3事务日志10 1.3.4MySQL中的事务10 1.4多版本并发控制12 1.5MySQL的存储引擎13 1.5.1InnoDB存储引擎16 1.5.2MyISAM存储引擎17 1.5.3MySQL内建的其他存储引擎19 1.5.4第三方存储引擎22 1.5.5选择合适的引擎24 1.5.6转换表的引擎27 1.6MySQL时间线 (Timeline) 29 1.7MySQL的开发模式32 1.8总结33 第2章MySQL基准测试35 2.1为什么需要基准测试35 2.2基准测试的策略37 2.2.1测试何种指标38 2.3基准测试方法40 2.3.1设计和规划基准测试41 2.3.2基准测试应该运行多长时间42 2.3.3获取系统性能和状态43 2.3.4获得准确的测试结果44 2.3.5运行基准测试并分析结果46 2.3.6绘图的重要性47 2.4基准测试工具49 2.4.1集成式测试工具49 2.4.2单组件式测试工具50 2.5基准测试案例52 2.5.1http_load53 2.5.2MySQL基准测试套件54 2.5.3sysbench55 2.5.4数据库测试套件中的dbt2TPC—C测试60 2.5.5Percona的TPCC—MySQL测试工具63 2.6总结65 第3章服务器性能剖析67 3.1性能优化简介67 3.1.1通过性能剖析进行优化69 3.1.2理解性能剖析71 3.2对应用程序进行性能剖析72 3.2.1测量PHP应用程序74 3.3剖析MySQL查询77 3.3.1剖析服务器负载77 3.3.2剖析单条查询81 3.3.3使用性能剖析87 3.4诊断间歇性问题88 3.4.1单条查询问题还是服务器问题89 3.4.2捕获诊断数据93 3.4.3一个诊断案例98 3.5其他剖析工具106 3.5.1使用USER_STATISTICS表106 3.5.2使用strace107 3.6总结108 第4章Schema与数据类型优化111 4.1选择优化的数据类型111 4.1.1整数类型113 4.1.2实数类型113 4.1.3字符串类型114 4.1.4日期和时间类型121 4.1.5位数据类型123 4.1.6选择标识符 (identifier) 125 4.1.7特殊类型数据127 4.2MySQLschema设计中的陷阱127 4.3范式和反范式129 4.3.1范式的优点和缺点130 4.3.2反范式的优点和缺点130 4.3.3混用范式和反范式131 4.4缓存表和汇总表132 4.4.1物化视图134 4.4.2计数器表135 4.5加快ALTER TABLE操作的速度136 4.5.1只修改.frm文件137 4.5.2快速创建MyISAM索引139 4.6总结140 第5章创建高性能的索引141 5.1索引基础141 5.1.1索引的类型142 5.2索引的优点152 5.3高性能的索引策略153 5.3.1独立的列153 5.3.2前缀索引和索引选择性153 5.3.3多列索引157 5.3.4选择合适的索引列顺序159 5.3.5聚簇索引162 5.3.6覆盖索引171 5.3.7使用索引扫描来做排序175 5.3.8压缩 (前缀压缩) 索引177 5.3.9冗余和重复索引178 5.3.10未使用的索引181 5.3.11索引和锁181 5.4索引案例学习183 5.4.1支持多种过滤条件183 5.4.2避免多个范围条件185 5.4.3优化排序186 5.5维护索引和表187 5.5.1找到并修复损坏的表187 5.5.2更新索引统计信息188 5.5.3减少索引和数据的碎片190 5.6总结192 第6章查询性能优化195 6.1为什么查询速度会慢195 6.2慢查询基础：优化数据访问196 6.2.1是否向服务器请求了不需要的数据196 6.2.2MySQL是否在扫描额外的记录198 6.3重构查询的方式201 6.3.1一个复杂查询还是多个简单查询201 6.3.2切分查询202 6.3.3分解关联查询203 6.4查询执行的基础204 6.4.1MySQL客户端 / 服务器通信协议205 6.4.2查询缓存208 6.4.3查询优化处理208 6.4.4查询执行引擎222 6.4.5返回结果给客户端223 6.5MySQL查询优化器的局限性223 6.5.1关联子查询223 6.5.2UNION的限制228 6.5.3索引合并优化228 6.5.4等值传递229 6.5.5并行执行229 6.5.6哈希关联229 6.5.7松散索引扫描229 6.5.8最大值和最小值优化231 6.5.9在同一个表上查询和更新232 6.6查询优化器的提示 (hint) 232 6.7优化特定类型的查询236 6.7.1优化COUNT () 查询236 6.7.2优化关联查询239 6.7.3优化子查询239 6.7.4优化GROUPBY和DISTINCT239 6.7.5优化LIMIT分页241 6.7.6优化SQL_CALC_FOUND_ROWS243 6.7.7优化UNION查询243 6.7.8静态查询分析244 6.7.9使用用户自定义变量244 6.8案例学习251 6.8.1使用MySQL构建一个队列表251 6.8.2计算两点之间的距离254 6.8.3使用用户自定义函数257 6.9总结258 第7章MySQL高级特性259 7.1分区表259 7.1.1分区表的原理260 7.1.2分区表的类型261 7.1.3如何使用分区表262 7.1.4什么情况下会出问题263 7.1.5查询优化266 7.1.6合并表267 7.2视图270 7.2.1可更新视图272 7.2.2视图对性能的影响273 7.2.3视图的限制274 7.3外键约束275 7.4在MySQL内部存储代码276 7.4.1存储过程和函数278 7.4.2触发器279 7.4.3事件281 7.4.4在存储程序中保留注释283 7.5游标283 7.6绑定变量284 7.6.1绑定变量的优化286 7.6.2SQL接口的绑定变量286 7.6.3绑定变量的限制288 7.7用户自定义函数289 7.8插件290 7.9字符集和校对291 7.9.1MySQL如何使用字符集292 7.9.2选择字符集和校对规则295 7.9.3字符集和校对规则如何影响查询296 7.10全文索引299 7.10.1自然语言的全文索引300 7.10.2布尔全文索引302 7.10.3MySQL5.1中全文索引的变化303 7.10.4全文索引的限制和替代方案304 7.10.5全文索引的配置和优化306 7.11分布式 (XA) 事务307 7.11.1内部XA事务307 7.11.2外部XA事务308 7.12查询缓存309 7.12.1MySQL如何判断缓存命中309 7.12.2查询缓存如何使

用内存311 7.12.3什么情况下查询缓存能发挥作用313 7.12.4如何配置和维护查询缓存316 7.12.5InnoDB和查询缓存319 7.12.6通用查询缓存优化320 7.12.7查询缓存的替代方案321 7.13总结321 第8章优化服务器设置325 第9章操作系统和硬件优化377 第10章复制433 第11章可扩展的MySQL501 第12章高可用性543 第13章云端的MySQL563 第14章应用层优化577 第15章备份与恢复593 第16章MySQL用户工具635 附录A MySQL分支与变种649 附录B MySQL服务器状态655 附录C大文件传输683 附录DEXPLAIN687 附录E 锁的调试703 附录F在MySQL上使用Sphinx713 索引739

章节摘录

版权页：插图：第一个趋势，采用了InnoDB plugin的版本，在高并发的时候性能明显更好，可以说InnoDB plugin的扩展性更好。

这是可以预期的结果，旧的版本在高并发时确实存在问题。

第二个趋势，新的版本在单线程的时候性能比旧版本更差。

一开始可能无法理解为什么会这样，仔细想想就能明白，这是一个非常简单的只读测试。

新版本的SQL语法更复杂，针对复杂查询增加了很多特性和改进，这对于简单查询可能带来了更多的开销。

旧版本的代码简单，对于简单的查询反而会更有利。

原计划做一个更复杂的不同并发条件下的读写混合场景的测试（类似TPC—C），但要在不同版本间做到可比较基本是不可能的。

一般来说，新版本在复杂场景时性能有更多的优化，尤其是高并发和大数据集的情况下。

那么该如何选择版本呢？

这更多地取决于业务需求而不是技术需求。

理想情况下当然是版本越新越好，当然也可以选择等到第一个bug修复版本以后再采用新的大版本。

如果应用还没有上线，也可以采用即将发布的新版本，以尽可能地延迟应用上线后的升级操作。

1.7 MySQL的开发模式 MySQL的开发过程和发布模型在不同的阶段有很大的变化，但目前已经基本稳定下来。

在Oracle定期发布的新里程碑开发版本中，会包含即将在下一个GA版本发布的新特性。

这样做是为了测试和获得反馈，请不要在生产环境使用此版本，虽然Oracle宣称每个里程碑版本的质量都是可靠的，并随时可以正式发布（到目前为止也没有任何理由去推翻这个说法）。

Oracle也会定期发布实验室预览版，主要包含一些特定的需要评估的特性，这些特性并不保证会在下一个正式版本中包括进去。

最终，Oracle会将稳定的特性打包发布一个新的GA版本。

MySQL依然遵循GPL开源协议，全部的源代码（除了一些商业版本的插件）都会开放给社区。

Oracle似乎也理解，为社区和付费用户提供不同的版本并非明智之举。

MySQLAB曾经尝试过不同版本的策略，结果导致付费用户变成了“睁眼瞎”，无法从社区的测试和反馈中获得好处。

不同版本的策略并不受企业用户的欢迎，所以后来被Sun废除了。

现在Oracle为付费用户单独提供了一些服务器插件，而MySQL本身还是遵循开源模式。

尽管对于私有的服务器插件的发布有一些抱怨，但这只是少数的声音，并且慢慢地在平息。

媒体关注与评论

每一章均别具匠心，力求理论与实践的精确平衡，且布满无价之宝，有时甚至超越MySQL舞台，完全适用于任一数据库。

其中第二章“MySQL基准测试”及第3章“服务器性能剖析”是非常必要的基础，推荐提前阅读。

纵观全书，作者推荐的工具、实战案例及经验过的诊断技术，可大大提高你的性能急救技能，以及加深对MySQL本质的理解。

然而，本书最值得推崇的，还是其在探讨性能的同时，将数据库结构的客观方面纳入思考，这是其他书里难以看到的。

此外，增补的MySQL高可用性及云特性，也让人更加欣喜。

相信不少人会因为找不到某些书中引用的资料或工具而苦恼，但从本书中按图索骥，会发现这些东西正是作者对MySQL社区的杰出贡献，也就是说，你可以直接用这些工具！

很多年前我就是这本书的“粉丝”了，这是一本伟大的书，第三版尤其如此。

这些世界级的专家不仅仅分享他们的专业知识，也花了很多时间来更新和添加新的章节，且都是高品质的内容。

本书有大量关于如何获得MySQL高性能的细节信息，并且关注的是提升性能的过程，而不仅仅是描述事实结果和琐碎的细枝末节。

这本书将告诉读者如何将事情做得更好，不管MySQL在不同版本中的行为有多么大的改变。

毫无疑问，本书的作者是唯一有资格来写这么一本书的人，他们经验丰富，有合理的方法，关注效率，并且精益求精。

说到经验丰富，本书的作者已经在MySQL性能领域工作多年，从MySQL还没有什么可扩展性和可测量性的时代，直到现在这些方面已经有了长足的进步。

而说到合理的方法，他们简直把这件事情当成了科学，首先定义需要解决的问题，然后通过合理的猜测和精确的测量来解决问题。

我对作者在效率方面的关注尤其印象深刻。

作为顾问，他们时间宝贵。

客户是按照他们的时间付费的，所以都希望能更快地解决问题。

所以本书作者定义了一整套的流程，开发了很多的工具，让事情变得正确和高效。

在本书中，作者详细描述了这些流程，并且发布了工具的源代码。

最后，本书作者在工作上一一直精益求精。

比如从吞吐量到响应时间的关注，致力于了解MySQL在新硬件上的性能表现，追求新的技能如排队理论对性能的影响，等等。

我相信本书预示了MySQL的光明前景。

MySQL已经支持高要求的工作负载，本书作者也在努力提升MySQL社区内对性能的认识。

同时，他们还直接为性能提升做出了贡献，包括XtraDB和XtraBackup。

一直以来我从他们身上学到了不少东西，也希望读者多花点时间读读本书，一定会同样有所收益。

——Mark Callaghan，Facebook 软件工程师

编辑推荐

《高性能MySQL(第3版)》编辑推荐：“ 只要你不敢以MySQL专家自诩，又岂敢错过这本神书？”
“ 一言以蔽之，写得好，编排得好，需要参考时容易到爆！”
“ 我可是从头到尾看了一遍上一版，可还是毫不犹豫拿起了这本书，而且看完后一点都不后悔……”

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>