



MySQL 5.6 新特性 *DBA 和开发人员指南*

MySQL® 白皮书

2013 年 2 月



目录

简介	3
提高性能和可伸缩性：改进了 InnoDB 存储引擎.....	3
缩短查询执行时间和增强诊断：改进了优化器.....	7
提高应用程序可用性：联机 DDL/模式更改.....	9
提高开发人员灵活性：对 InnoDB 进行 NoSQL 访问.....	9
提高开发人员灵活性：扩展了 InnoDB 用例.....	10
改进复制和高可用性.....	11
改进 Performance Schema	14
提高安全性.....	14
其他重要增强	15
MySQL 5.6：生产就绪软件和支持	17
总结	18
了解更多信息	18



简介

MySQL 是当今最值得信赖和依靠的开源数据库平台。全球 10 大最热门、流量最高的网站中有 9 家都依赖于 MySQL，这主要归功于其在各种不同平台和软件体系中的普遍应用以及众所周知绝佳的性能、可靠性和易用性。在此基础上，MySQL 5.6 进行了全方位的改进，旨在让富于创新的 DBA 和开发人员能够在最新一代的开发框架和硬件平台上创建和部署下一代 Web、嵌入式和云计算/SaaS/DaaS 之应用程序。

简而言之，MySQL 5.6 只是 MySQL 的一个改进版。它引入了增强数据库内核每个功能领域的多方面改进，包括：

- 提高性能和可扩展性
 - 通过改进 InnoDB 存储引擎来提升事务吞吐量
 - 通过改进优化器来缩短查询执行时间和增强诊断
- 通过在线 DDL/模式更改来提高应用程序可用性
- 支持通过 Memcached API 对 InnoDB 进行 NoSQL 访问，从而使开发人员能灵活的以适当的技术因应开发速和应用系统效能的需求
- 通过改进复制来实现高性能、自我修复的分布式部署
- 通过改进 Performance Schema 来更好地监测 MySQL
- 提高安全性以确保应用程序部署安全无忧
- 以及其他重要增强

该白皮书作为 DBA 和开发人员了解 MySQL 5.6 的指南，将重点介绍以上各个功能领域中的重要新特性，同时辅以相应的实际用例。最后，本文将总结 MySQL 企业版产品的生产就绪支持和服务如何确保 MySQL 5.6 的实施过程遵循最佳实践。

提高性能和可扩展性：改进了 InnoDB 存储引擎

从运营的角度来看，MySQL 5.6 在支持多处理器和高度并发 CPU 线程的系统上，提供更持续的线性性能和扩展性。实现这一点的关键是通过改进 Oracle InnoDB 存储引擎的效率和并发性，来消除 InnoDB 内核中原有的线程争用和互斥锁定的现象。通过这些改进，MySQL 可以充分利用当今基于 x86 的商用硬件先进的多线程处理能力。

以 SysBench 做的读/写和只读负载的内部基准测试显示，MySQL 的持续扩展性在 MySQL 5.5 的最新版本有了显著提升。下图显示了 MySQL 5.6 在最多支持 48 个并发 CPU 线程的系统上的每秒读/写事务处理次数 (TPS) 呈“高开高走”的态势。

SysBench (读写): MySQL 5.6 与 5.5 的对比 (Linux)

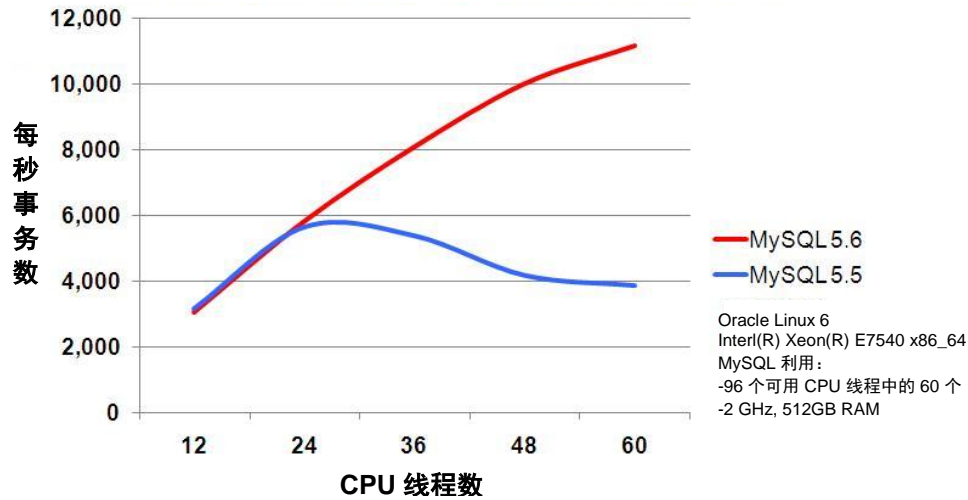


图 1: SysBench 读/写 TPS 持续线性可伸缩性 — 60 个 CPU 线程

只读 TPS 负载的持续可伸缩性也得到了改进，如下所示：

SysBench (只读): MySQL 5.6 与 5.5 的对比 (Linux)

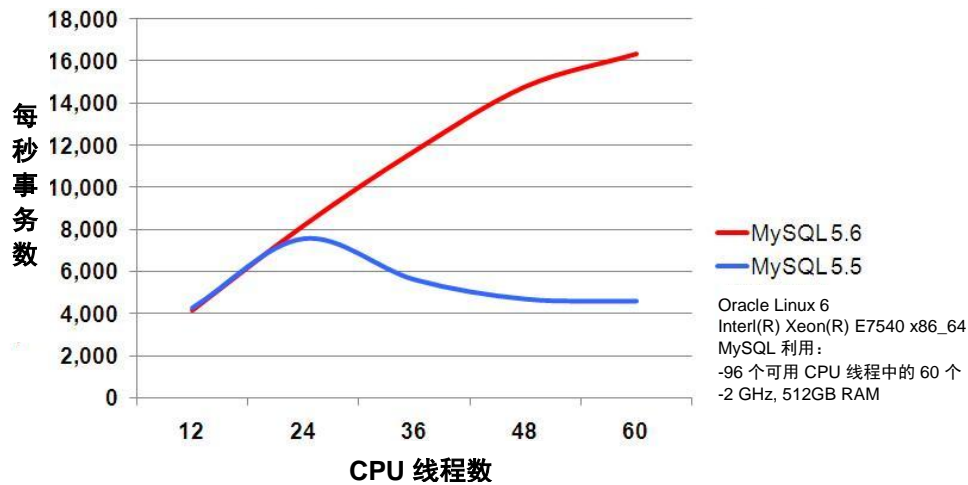


图 2: SysBench 只读 TPS 持续线性可伸缩性 — 60 个 CPU 线程

更高的事务吞吐量

MySQL 5.6 还改进了 InnoDB，从而提升了高度并发、事务性和读取密集型负载上的性能和可扩展性。在这些情况下，衡量性能提升的最佳指标就是应用程序随并发用户数增加所展现的性能和扩展性。为了支持这些用例，InnoDB 采用了一个全新设计的架构，可最大限度减少互斥元争用和瓶颈，同时提供一个更加一致的底层数据访问路径。具体改进包括：

- 通过拆分内核互斥体来消除单点争用
- 新增针对刷新操作的线程
- 新增多线程清除
- 新增调适性哈希运算法



- 减少缓冲池争用
- 透过定时收集的持久性优化统计讯息，从而提高查询执行的效率和一致性

SysBench 读/写基准测试反映了这些改进的成果，如下所示：

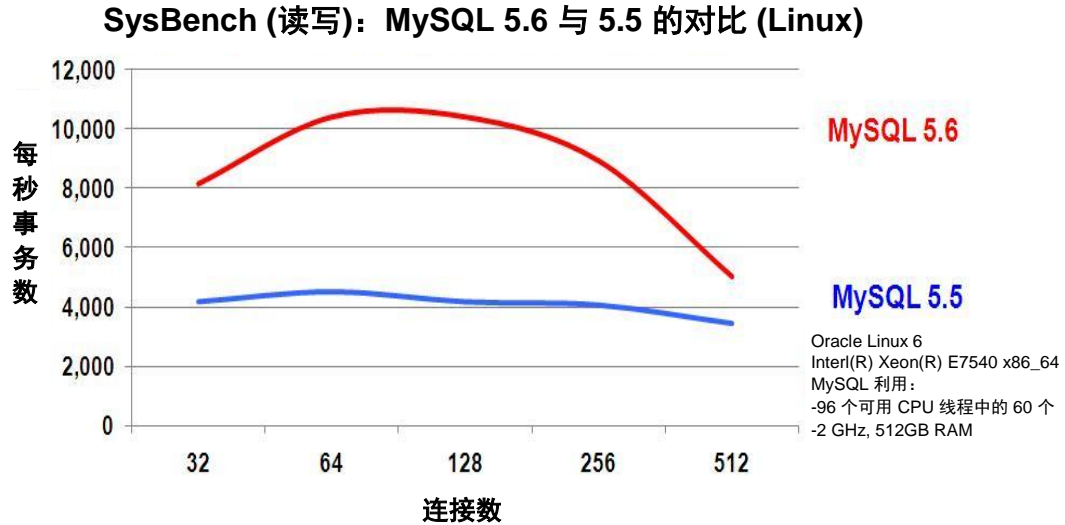


图 3: 与 MySQL 5.5 相比的 SysBench 读写 TPS 性能提升

在 Linux 上，MySQL 5.6 比 MySQL 5.5 的每秒处理事务吞吐量提高了 150%，而在 Windows 2008 上运行类似的测试取得了 47% 的性能提升。

更高的只读负载吞吐量

通过对 InnoDB 的只读事务处理进行新的优化，使基于 Web 的高并发性查找和报告生成应用程序的性能得到了显著提升。这些优化可避免事务开销，默认情况下为启用状态 (**autocommit = 1**)，开发人员也可以使用新增的 **START TRANSACTION READ ONLY** 语法来自动控制：

```
SET autocommit = 0;  
START TRANSACTION READ ONLY;  
SELECT c FROM T1 WHERE id=N;  
COMMIT;
```

这些优化的结果如下所示：

SysBench (只读): MySQL 5.6 与 5.5 的对比 (Linux)

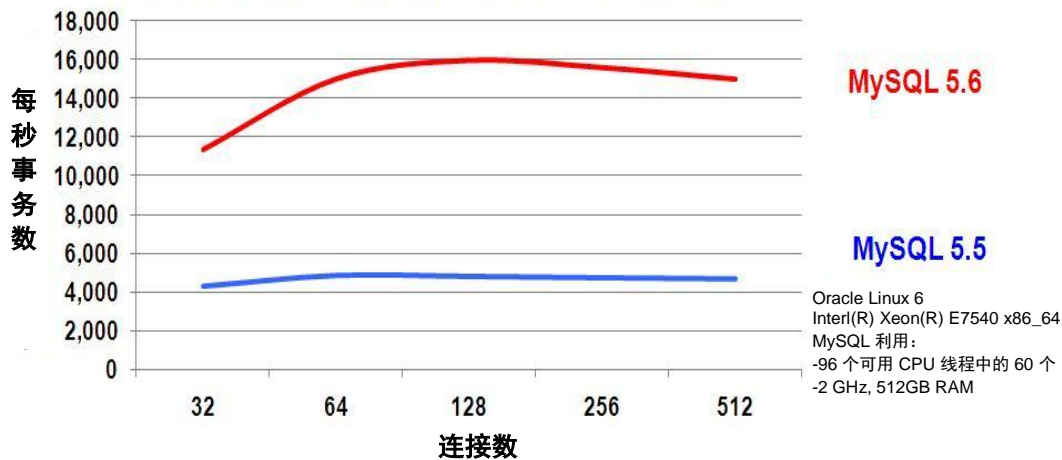


图 4: 与 MySQL 5.5 相比的 SysBench 只读 TPS 性能提升

在 Linux 上, MySQL 5.6 的只读 TPS 吞吐量比 MySQL 5.5 提高了 230%, 而在 Windows 2008 上运行类似的测试取得了 65% 的性能提升。

上述所有基准测试都是在以下平台配置上完成的:

- Oracle Linux 6
- Intel(R) Xeon(R) E7540 x86_64
- MySQL, 利用:
 - 96 个可用 CPU 线程中的 48 个
 - 2 GHz, 512GB RAM

SysBench 是一款针对特定应用程序用例的免费基准测试工具, 可从这里下载:

<http://dev.mysql.com/downloads/benchmarks.html>。

通过固态硬盘 (SSD) 提升性能

旋转式磁盘在任何系统中都是最常见的瓶颈, 仅仅是因为它们的机械部件会从物理上限制并发增加时的扩展性。因此, 许多 MySQL 应用程序都部署在支持 SSD 的系统上, 这些系统可提供基于内存的处理速度和可靠性, 以支持当今基于 Web 的系统实现最高水平的并发性。考虑到这些因素, MySQL 5.6 专门针对 SSD 融入了一些关键增强, 包括:

- 支持更小的 4k 和 8k 页面大小, 以便更好地适应 SSD 的标准存储算法。
- 可移植的 .idb (InnoDB 数据) 文件, 支持轻松将“热”InnoDB 表从默认数据目录移至 SSD 或网络存储设备。
- 为 InnoDB 撤销日志(undo log)提供单独的表空间, 可以选择性地将撤销日志从系统表空间移入一个或多个单独的表空间中。由于撤销日志属于读取密集型的 I/O 模式, 因此非常适合将这些新的表空间移至 SSD 存储, 而系统表空间仍然存放在硬盘驱动器的存储上。

了解所有支持的 SSD 优化, 请访问: <http://dev.mysql.com/doc/refman/5.6/en/innodb-performance.html>。

缩短查询执行时间和增强诊断：改进了优化器

MySQL 5.6 优化器在重新设计后提高了效率和性能，同时提供一系列改进的功能，可缩短查询执行时间和增强诊断。5.6 优化器的关键改进包括：

- **子查询优化** — 通过使用半联接 (semi-join) 和物理化，MySQL 优化器可显著提高子查询性能，从而让开发人员更轻松构建查询。具体来说，优化器现在可以更高效地处理 FROM 子句中的子查询；FROM 子句中针对子查询的物理化操作现在会推迟，直到执行过程中需要这些内容时才进行物理化，大大提高了性能。此外，优化器还可以在执行过程中为衍生表加上一个索引，以加快行检索速度。如下所示，使用 DBT-3 基准测试 Query #13 运行的测试显示，新版本的执行速度相比之前版本有大幅的提升（从数天缩短至数秒）。¹

```
select c_name, c_custkey, o_orderkey, o_orderdate, o_totalprice,
sum(l_quantity)
from customer, orders, lineitem
where o_orderkey in (
    select l_orderkey
    from lineitem
    group by l_orderkey
    having sum(l_quantity) > 313
)
and c_custkey = o_custkey
and o_orderkey = l_orderkey
group by c_name, c_custkey, o_orderkey, o_orderdate, o_totalprice
order by o_totalprice desc, o_orderdate
LIMIT 100;
```

- **小 Limit 值实现文件排序优化** — 对于使用 ORDER BY 和小 LIMIT 值的查询来说，优化器现在使用单表扫描生成一个排序结果集。对于仅显示整个结果集中的少数行的 Web 应用程序来说，这些查询极为常见，例如：

```
SELECT col1, ... FROM t1 ... ORDER BY name LIMIT 10;
```

内部基准测试显示，查询执行速度最多可提升 4 倍，有助于改善用户的体验和缩短响应时间。²

- **索引条件下推 (ICP)** — 默认情况下，优化器现在将 WHERE 条件下推至存储引擎进行评估、执行表扫描并将排序后的结果集返回 MySQL 服务器。

```
CREATE TABLE person (
    personid INTEGER PRIMARY KEY,
    firstname CHAR(20),
    lastname CHAR(20),
    postalcode INTEGER,
    age INTEGER,
    address CHAR(50),
    KEY k1 (postalcode, age)
) ENGINE=InnoDB;

SELECT lastname, firstname FROM person
WHERE postalcode BETWEEN 5000 AND 5500 AND age BETWEEN 21 AND 22;
```

¹ <http://oysteing.blogspot.com/2012/07/from-months-to-seconds-with-subquery.html>

² <http://didrikdidrik.blogspot.com/2011/04/optimizing-mysql-filesort-with-small.html>

针对此类表和查询的内部基准测试显示，ICP 默认行为最多可将执行时间缩短 15 倍。³

- **批量键值访问 (BKA) 和多范围读取 (MRR)** — 优化器现在以批次方式为存储引擎提供和结果集相关的全部主键，让存储引擎可以更加高效地访问、排序和返回数据，从而大幅缩短查询执行时间

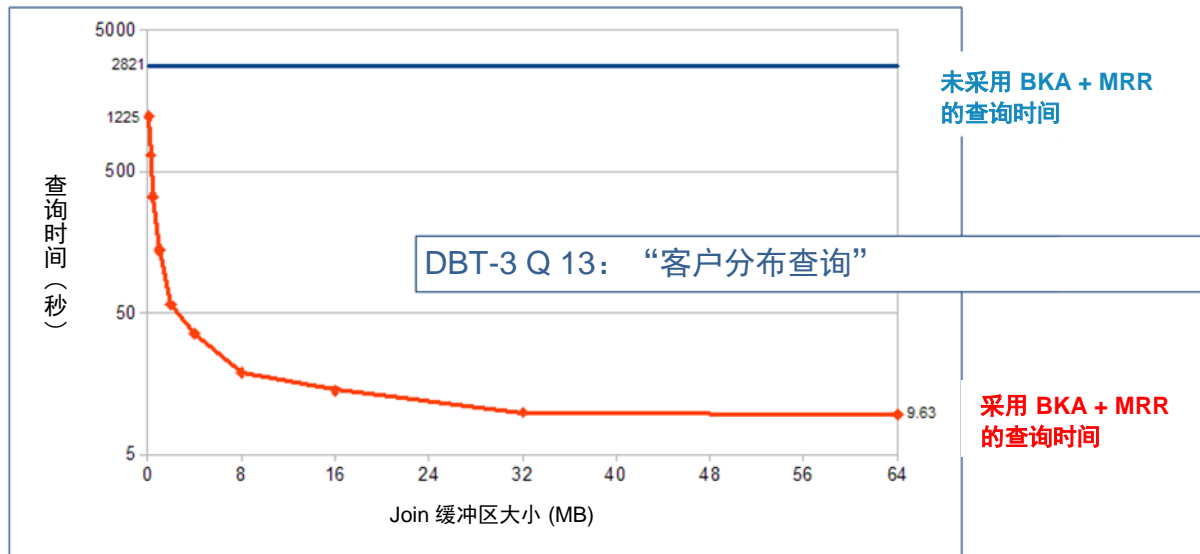


图 5: MySQL 5.6 通过批量密钥访问和多范围读取缩短了查询执行时间

在 DBT-3 Query 13 和其他针对磁盘的查询基准测试中，BKA 和 MRR 最多可将查询执行速度提升 280 倍。⁴

- **增强优化器诊断** — MySQL 5.6 优化器还通过以下改进优化了诊断和侦错：
 - 针对 INSERT、UPDATE 和 DELETE 操作的 EXPLAIN 语句
 - JSON 格式的 EXPLAIN 计划输出可提供更加准确的优化器量度和更好的可读性
 - 通过 Optimizer Traces 跟踪优化器决策过程。

了解 MySQL 5.6 优化器的全部改进和特性以及所有技术文档，请访问：

<http://dev.mysql.com/doc/refman/5.6/en/mysql-nutshell.html>

³ <http://olavsandstaa.blogspot.com/2011/04/mysql-56-index-condition-pushdown.html>

⁴ <http://oysteing.blogspot.com/2011/10/batched-key-access-speeds-up-disk-bound.html>



提高应用程序可用性：在线DDL/模式更改

现在，基于 Web 的应用程序均设计为可快速发展和调整来满足业务和创收的要求。因此，开发 SLA 的标准周期现在已经从数天或数周缩短为数分钟。当应用程序必须快速支持新的产品线或者当前产品线中的新产品时，后端数据库模式必须适应这种变化，同时确保应用程序仍可用于正常业务运营。为了支持这种高水平的在线操作模式所需的灵活性和敏捷性，MySQL 5.6 新增了以在线 ALTER TABLE DDL 语法：

- CREATE INDEX
- DROP INDEX
- Change AUTO_INCREMENT value for a column
- ADD/DROP FOREIGN KEY
- Rename COLUMN
- Change ROW FORMAT, KEY_BLOCK_SIZE for a table
- Change COLUMN NULL, NOT_NULL
- Add, drop, reorder COLUMN

DBA 和开发人员可以添加索引和执行标准的 InnoDB 表更改，而数据库仍可用于应用程序更新。这对于快速变化的应用程序尤为有用，因为开发人员需要操作灵活性来满足不断变化的业务需求。

了解 MySQL 5.6 InnoDB 在线DDL 的全部改进和特性以及所有技术文档，请访问：

<http://dev.mysql.com/doc/refman/5.6/en/innodb-online-ddl.html>。

提高开发人员灵活性：对 InnoDB 进行 NoSQL 访问

许多最新一代的 Web、云计算、社交网和移动应用程序都需要快速操作简单的键值对。与此同时，它们还必须能够对相同的数据运行复杂的查询，并确保数据库受到 ACID 的保护。借助 InnoDB 的新 NoSQL API，开发人员可以发挥事务性 RDBMS 的一切优势，同时获得键值存储的性能水平。

MySQL 5.6 支持通过大家熟悉的 Memcached API 与 InnoDB 数据进行简单的键值互动。mysqld 中新增了一个 Memcached 后台进程插件，可将全新的 Memcached 协议直接转成原生 InnoDB API，让开发人员能够使用现有 Memcached 客户端直接对 InnoDB 数据执行查找和事务兼容的更新，从而避免 SQL 命令解析的开销。通过该 API 可利用标准 Memcached 库和客户端，同时通过集成一个持久、崩溃安全、事务性的数据库后端对 Memcached 的功能进行扩展。该实施如下所示：

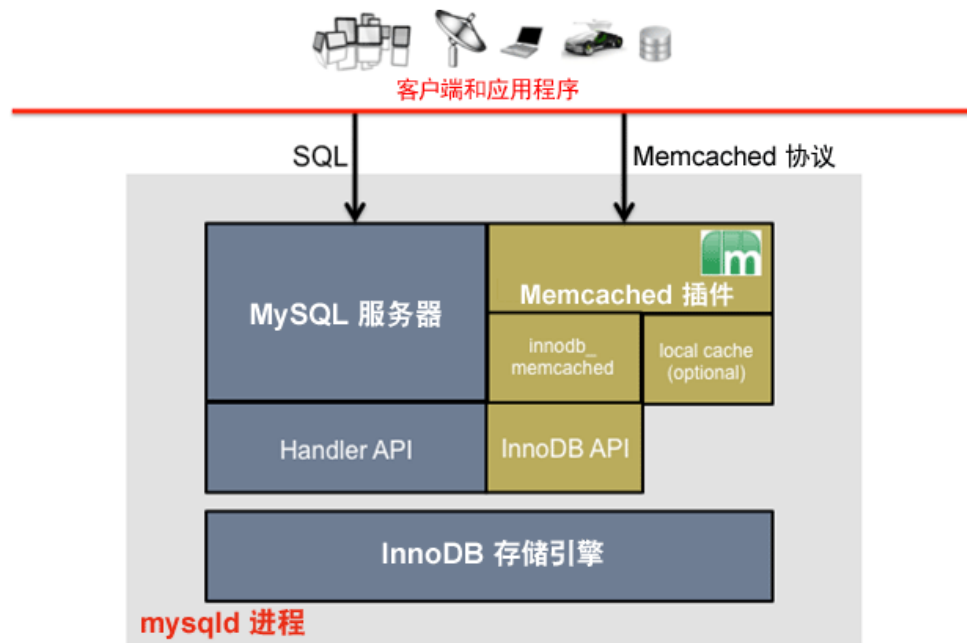


图 6：新增的通过 Memcached API 对 InnoDB 数据进行 NoSQL 键值访问

开发人员和 DBA 不仅获得了更高的性能和灵活性，还降低了工作的复杂性，因为他们可以将之前各自独立的缓存和数据库层压缩成一个数据管理层，同时消除维护缓存一致性的开销。

了解详细信息以及如何通过新的 Memcached API 访问 InnoDB，请访问
<http://dev.mysql.com/doc/refman/5.6/en/innodb-memcached.html>

提高开发人员灵活性：扩展了 InnoDB 用例

MySQL 5.6 新增的优化和特性将 InnoDB 扩展为更多用例，这样开发人员就可以通过在单个存储引擎上实现标准化来简化应用程序。

- **新增全文检索 (FTS)** — 作为比 MyISAM FTS 更好的替代方案，InnoDB 现在可以让开发人员在 InnoDB 表上构建 FULLTEXT 索引，来表示基于文本的内容以及加快应用程序对词和短语的搜索速度。InnoDB 全文本搜索支持自然语言/布林模式、邻近搜索和相关性排序。下面给出了一个简单用例：

```
CREATE TABLE quotes
(id int unsigned auto_increment primary key
, author varchar(64)
, quote varchar(4000)
, source varchar(64)
, fulltext(quote)
) engine=innodb;

SELECT author AS "Apple" FROM quotes
WHERE match(quote) against ('apple' in natural language mode);
```

- **新增可传输表空间** — 现在，采用每个表使用一个表空间 (file-per-table) 模式创建的 InnoDB .ibd 文件可在物理存储设备和数据库服务器之间传输；开发人员现在创建表时，可以在 MySQL 数据目录外部为 .ibd 文件指定一个存储位置。这样一来，就可以轻松地将“热”表或繁忙的表移至不会争用应用程序

序或数据库开销的外部网络存储设备（SSD、HDD）。这个新特性还支持快速、无缝的应用程序伸缩，允许用户轻松地在运行的 MySQL 服务器之间导出/导入 InnoDB 表，如下所示：

导出示例：

```
CREATE TABLE t(c1 INT) engine=InnoDB;
FLUSH TABLE t FOR EXPORT; -- quiesce the table and create the meta data
file
$innodb_data_home_dir/test/t.cfg
UNLOCK TABLES;
```

对应的导入：

```
CREATE TABLE t(c1 INT) engine=InnoDB; -- if it doesn't already exist
ALTER TABLE t DISCARD TABLESPACE;
-- The user must stop all updates on the tables, prior to the IMPORT
ALTER TABLE t IMPORT TABLESPACE;
```

本文无法详尽介绍所有 InnoDB 改进。有关全部 MySQL 5.6 特性的完整列表以及所有技术文档，请访问：
<http://dev.mysql.com/doc/refman/5.6/en/mysql-nutshell.html>。

改进复制和高可用性

复制是使用最广泛的一项 MySQL 特性，用于实现向外扩展和高可用性 (HA)。MySQL 5.6 提供了一些新特性，旨在帮助开发人员使用自我修复式复制拓扑和高性能主从机制打造下一代 Web、云计算、社交和移动应用程序及服务。其关键特性包括：

- **新增全局事务标识 (GTID)** — GTID 通过复制主/从拓扑来跟踪复制事务完整性，为自我修复式恢复奠定了基础，使 DBA 和开发人员可以在主服务器发生故障时轻松识别最新的从服务器。由于 GTID 直接内置在二进制日志流中，因此不必使用复杂的第三方插件来提供相同水平的跟踪能力。



图 7：GTID 在二进制日志中的位置

- **新增 MySQL 复制实用工具** — 一组新的 Python 实用工具，旨在利用新的复制 GTID 提供复制管理和监视，支持在主机发生故障时自动执行故障切换，或者在主机维护时进行切换。这样就无需其他第三方高可用解决方案，可保护 Web 和基于云计算的服务不受计划性停机和意外停机的影响，且无需操作人员介入。
- **新增多线程从服务器** — 根据模式来拆分工作线程之间的处理，允许并行应用更新，而不必按顺序进行。对于那些使用数据库隔离应用程序数据的负载（例如，多承租方系统）来说，这项功能提供了很多好处。

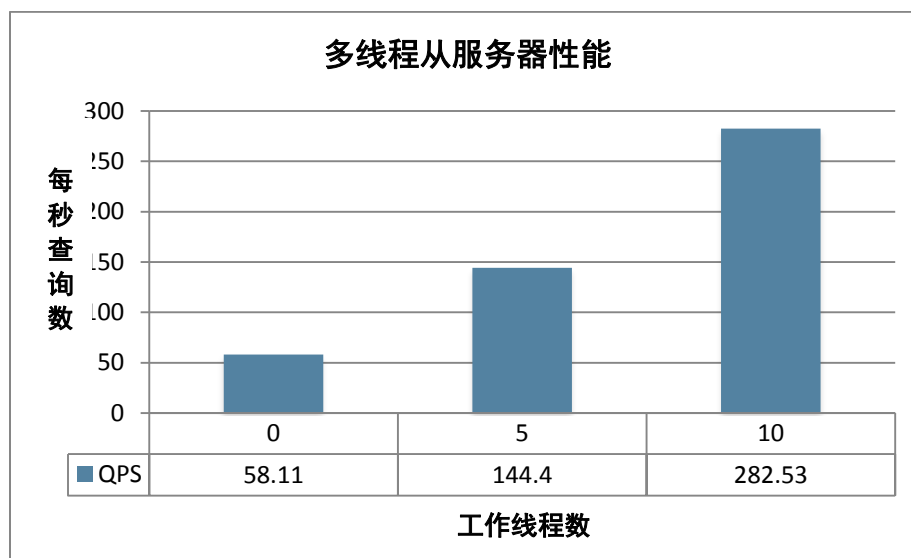


图 8 : MySQL 5.6 多线程从服务器性能提升

跨 10 个模式、累进数量的工作线程进行的 SysBench 基准测试显示，启用多线程时最多可实现 5 倍的性能提升。⁵

- **新增二进制日志分组提交 (BGC)** — 在 MySQL 5.6 复制中，主服务器现在将写入分组提交至二进制日志，而不是一次提交一个，从而显着提升了主从架构的主服务器端的性能。BGC 还支持更加细粒度的锁定，可减少锁等待以及进一步提高性能，如下所示：

⁵ https://blogs.oracle.com/MySQL/entry/benchmarking_mysql_replication_with_multi

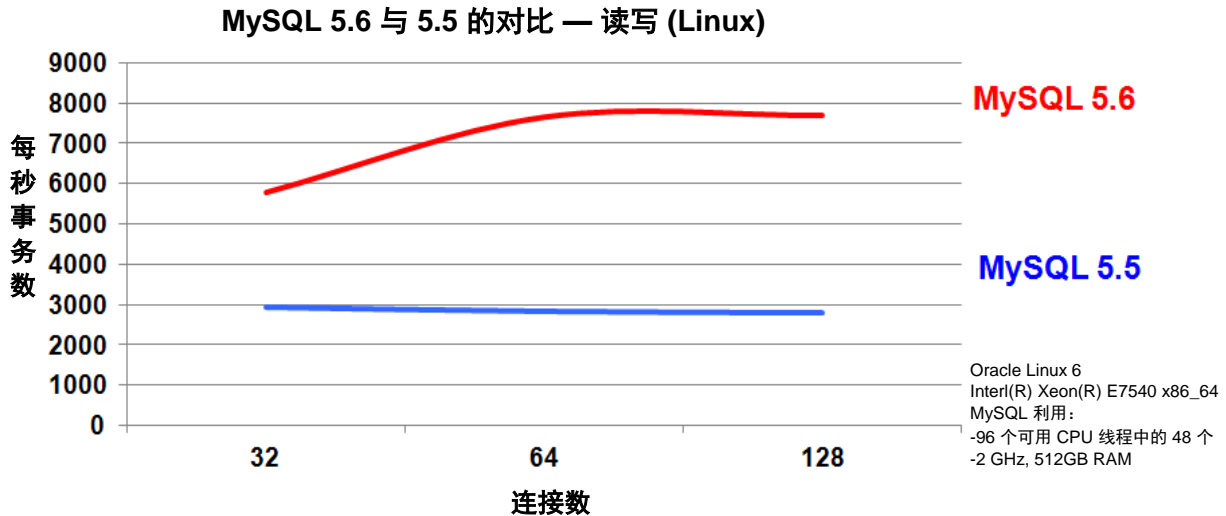


图 9: MySQL 5.6 二进制日志分组提交性能提升

启用了复制 (Binlog=1) 之后, MySQL 5.6 的主服务器吞吐量比 5.5 提升了 180%。借助 BGC, 用户基本上无需在主服务器性能开销与 MySQL 复制带来的向外扩展和高可用性优势之间做取舍。

- **新增经过优化的基于行的复制** — MySQL 5.6 提供了一个新的选项变量 **binlog-row-image=minimal**, 它允许应用程序仅复制行映像中在 DML 操作后发生了更改的数据元素。这有助于提高主服务器和从服务器的吞吐量, 同时最大限度地减少二进制日志磁盘空间、网络资源和服务器内存占用的空间。
- **新增崩溃安全从服务器** — MySQL 5.6 将二进制日志位置数据存储在表中, 这样从服务器便可自动将复制回滚到发生故障前上次提交的状态, 然后恢复复制, 无需管理员介入。这不仅有助于降低运营开销, 而且还可以消除从服务器尝试从受损数据文件进行恢复造成的数据丢失风险。此外, 当主服务器崩溃导致二进制日志受损时, 服务器还会自动将其恢复到可正确读取的位置。
- **新增复制校验和** — MySQL 5.6 确保将完整的数据复制到从服务器中, 具体方法是: 检测数据损坏并在数据损坏应用到从服务器之前返回错误, 从而防止从服务器本身受损。
- **新增延时复制** — MySQL 5.6 通过允许开发人员在复制流中加上指定的延迟时间, 防止主服务器上发生的操作错误传播至相连的从服务器。由于可配置主服务器和从服务器之间的时间延迟, 因此在发生故障或重大事故时, 系统会提示从服务器推送至新的主服务器, 以便将数据库恢复到之前的状态。还可以通过检测数据库状态来避免错误或停机, 无需重新载入备份。

了解 MySQL 5.6 在复制和高可用性方面的全部改进和特性以及所有技术文档, 请访问:
<http://dev.mysql.com/doc/refman/5.6/en/mysql-nutshell.html>。

改进 Performance Schema

MySQL Performance Schema 是在 MySQL 5.5 中引入的，旨在为关键绩效指标提供指定时间点的指标。为了解决 DBA 和开发人员最常见的问题，MySQL 5.6 对 Performance Schema 进行了改进。新增监测包括：

- 语句/阶段
 - 哪些是最占用资源的查询？它们在哪些地方耗费时间？
- 表/索引 I/O、表锁定
 - 哪些应用程序表/索引是产生负载或争用的最大原因？
- 用户/主机/帐户
 - 哪些应用程序用户、主机、帐户消耗的资源最多？
- 网络 I/O
 - 网络负载有多大？连线闲置时间有多长？
- 汇总
 - 按语句、线程、用户、主机、帐户或对象分组的合计的统计信息。

MySQL 5.6 Performance Schema 现在在 my.cnf 文件中默认为启用状态，其配置经过优化和自动调优，可最大限度减少开销（<5%，但具体因情况而异）；因此，在生产环境中使用 Performance Schema 监视最常见的应用程序用例不会造成问题。此外，新的最小单元级监测支持按用户、主机、帐户和应用程序等细粒度捕获资源占用数据，以便在云计算环境中实现计费和收费。

MySQL 文档详细介绍了 5.6 Performance Schema 的功能和作用。有关全面的详细信息，请访问 <http://dev.mysql.com/doc/refman/5.6/en/performance-schema.html>。

提高安全性

MySQL 5.6 对内部口令处理和加密机制进行了重大改进。新的选项和特性包括：

- **master.info 中口令新的替代选择** — MySQL 5.6 扩展了复制 START SLAVE 命令，使 DBA 能够在复制从服务器选项中指定主用户和口令，以及通过外部身份验证插件来验证连接主服务器的帐户（用户自定义或 MySQL 企业版所提供）。借助这些选项，用户和口令无需再以纯文本的形式在 master.info 文件中。
- **针对通用查询日志、慢查询日志和二进制日志中口令的新加密机制** — 写入这些日志的语句中的口令不必再采用纯文本的形式记录。
- **具备适当强度的新口令散列算法** — 针对内部 MySQL 服务器身份验证的默认口令散列不再使用 PASSWORD()，而是采用基于 SHA-256 口令哈希算法随机产生的 salt 值。
- **命令行中的新口令选项** — 命令行中的新口令选项 — MySQL 5.6 引入了一种全新的“加密”选项/配置文件 (.mylogin.cnf)，可用于安全地存放命令行操作所需的用户口令。
- **新增下次登录口令更改机制** — DBA 和开发人员现在可通过 mysql.user 表中新增的 password_expired 标志来控制何时必须更改帐户口令。
- **新增基于策略的口令验证** — 现在可以使用适当强度、长度、大小写组合、特殊字符以及基于 LOW、MEDIUM 和 STRONG 指定设置的其他用户定义的策略来验证口令。了解详细信息以及可用的配置选项，请参见 <http://dev.mysql.com/doc/refman/5.6/en/validate-password-plugin.html>。



了解 MySQL 5.6 在安全性方面的全部改进和特性以及所有技术文档，请访问：
<http://dev.mysql.com/doc/refman/5.6/en/mysql-nutshell.html>。

其他重要增强

- **新增默认配置优化** — MySQL 5.6 对服务器默认设置进行了修改，在当今的系统架构上可提供更好的即用性能。这些新的默认设置旨在最大限度地减少前期花在更改最常更新的变量和配置选项上的时间。

有关具体的修改细节、哪些参数会根据环境自动调整以及哪些设置可在服务器启动时设置和控制，请访问 <http://dev.mysql.com/doc/refman/5.6/en/server-default-changes.html>。

- **改进了 TIME/TIMESTAMP/DATETIME 数据类型：**
 - TIME/TIMESTAMP/DATETIME — 现在支持微秒级精度，可实现更加精准的时间/日期对比和数据选择。
 - TIMESTAMP/DATETIME — 在 5.5 的基础上进行了改进，开发人员可将当前时间戳和/或自动更新值指定为 TIMESTAMP 和 DATETIME 列的默认值。
 - TIMESTAMP — 现在列的默认值可以为空。如果没有明示指定，TIMESTAMP 不会再自动获得 DEFAULT NOW() 或 ON UPDATE NOW() 属性，未明示指定默认值的非 NULL 值 TIMESTAMP 列会被视为没有默认值。

- **改进了状况处理 — GET DIAGNOSTICS**

MySQL 5.6 引入了新的 MySQL 诊断区和相应的 GET DIAGNOSTICS 接口命令，允许开发人员轻松地检查错误条件和异常代码。可通过多个选项来填充诊断区，它提供了两类信息：

- 语句 — 提供受影响的行数以及所发生的条件数
- 条件 — 针对之前操作返回的所有条件提供错误代码和消息

它们各自可用的项目如下：

<i>condition_information_item_name:</i>	<i>statement_information_item_name:</i>
CLASS_ORIGIN	NUMBER
SUBCLASS_ORIGIN	ROW_COUNT
RETURNED_SQLSTATE	
MESSAGE_TEXT	
MYSQL_ERRNO	
CONSTRAINT_CATALOG	
CONSTRAINT_SCHEMA	
CONSTRAINT_NAME	
CATALOG_NAME	
SCHEMA_NAME	
TABLE_NAME	
COLUMN_NAME	
CURSOR_NAME	

新增的 GET DIAGNOSTICS 命令提供了一个标准的诊断区接口，可以通过命令行介面使用或者在应用程序代码中使用，方便检索和处理最新语句执行的结果：

```
mysql> DROP TABLE test.no_such_table;
ERROR 1051 (42S02): Unknown table 'test.no_such_table'
mysql> GET DIAGNOSTICS CONDITION 1
-> @p1 = RETURNED_SQLSTATE, @p2 = MESSAGE_TEXT;
mysql> SELECT @p1, @p2;
+-----+-----+
| @p1   | @p2                                     |
+-----+-----+
| 42S02 | Unknown table 'test.no_such_table' |
+-----+-----+
```

了解有关如何利用 MySQL 诊断区的选项的详细信息，请访问：

<http://dev.mysql.com/doc/refman/5.6/en/diagnostics-area.html>。GET DIAGNOSTICS 的详尽文档如下：<http://dev.mysql.com/doc/refman/5.6/en/get-diagnostics.html>。

• 改进了 IPv6 支持

- MySQL 5.6 改进了 INET_ATON()，将基于字符串的 IPv6 地址转换为二进制数据进行存储，最大限度地减少占用的空间。
- MySQL 5.6 将 bind-address 选项的默认值从 “0.0.0.0” 更改为 “0::0”，这样 MySQL 服务器便可接受所有 IPv4 和 IPv6 地址的连接。有关详细信息，请访问 http://dev.mysql.com/doc/refman/5.6/en/server-options.html#option_mysqld_bind-address。

• 改进了分区

- 提高了包含大量分区的表的性能 — MySQL 5.6 现在可在高度分区的系统上运行和扩展，特别是跨数百个分区的 INSERT 操作。
- 从分区的表导入表/将表导入到分区的表 — MySQL 5.6 允许用户使用 ALTER TABLE ...EXCHANGE PARTITION 语句将表分区或子分区与表进行交换；分区或子分区中的已有行可移至未分区的表中，相反，未分区的表中的任何已有行均可移至已有的表分区或子分区中。有关详细信息，请访问 <http://dev.mysql.com/doc/refman/5.6/en/partitioning-management-exchange.html>。
- 显式分区选择 — MySQL 5.6 支持显式选择待检查的分区和子分区（即检查是否存在匹配特定 WHERE 条件的行）。与自动分区修剪类似，待检查分区由语句发起方指定/控制，并且支持查询和众多 DML 语句（SELECT、DELETE、INSERT、REPLACE、UPDATE、LOAD DATA、LOAD



XML)。有关详细信息，请访问

<http://dev.mysql.com/doc/refman/5.6/en/partitioning-selection.html>。

- **改进了 GIS：精准的空间操作** — MySQL 5.6 通过符合 OpenGIS 标准的精确对象形状提供几何操作，用于测定两个几何值之间的关系。有关详细信息，请参考 <http://dev.mysql.com/doc/refman/5.6/en/functions-for-testing-spatial-relations-between-geometric-objects.html#functions-that-test-spatial-relationships-between-geometries>

了解 MySQL 5.6 的全部改进和特性以及所有技术文档，请访问：

<http://dev.mysql.com/doc/refman/5.6/en/mysql-nutshell.html>。

MySQL 5.6：生产就绪软件和支持

MySQL 是当今最流行的开源数据库，依据 GNU 通用公共许可 (GPL) 提供。对于构建下一代业务关键应用程序和服务的 DBA 和开发人员来说，MySQL 5.6 完全可用于开发、QA 和生产使用，Oracle 终身标准支持服务将为其提供全方位的保障，用户可以放心部署。MySQL 5.6 支持多种商业许可方案，允许用户开发、部署、监视和备份最安全、最新版本的 MySQL，Oracle 将为整个过程提供 24x7x365 的技术支持。

MySQL 市售软件和服务作为 MySQL 企业版的一部分提供，包括：

- **MySQL 数据库** — 最安全、最新版本的 MySQL 用于支持要求最严苛的联机、Web、云和 OLTP 应用程序及服务。MySQL 商用服务器是一个全面集成、事务安全、符合 ACID 的数据库，具备全面的提交、回滚、崩溃恢复和行级锁定功能。MySQL 因其在所有 Linux、UNIX、Mac OSX 和 Windows 平台上的性能、可靠性和易用性而著称。
- **MySQL 企业级可伸缩性** — 为了满足不断增长的用户、查询和数据负载对性能和可伸缩性的要求，MySQL 企业版提供了 MySQL 线程池。线程池提供了一个高度可伸缩、基于队列的线程处理模型，旨在降低客户端连接和语句执行线程的管理开销。
- **MySQL 企业级安全性** — MySQL 企业版提供了一些随时可用的外部身份验证模块，这样 DBA 和开发人员就可以轻松地将 MySQL 与现有的安全基础架构集成，包括 Linux 可插入身份验证模块 (PAM) 和 Windows Active Directory。
- **MySQL 企业级审计** — MySQL 企业版提供了一个易于使用、基于策略的审计解决方案，无需对现有应用程序进行任何修改，即可帮助开发人员实施更加严格的安全控制和满足法规要求。
- **MySQL 企业级高可用性** — MySQL 企业版提供了丰富的数据库高可用性解决方案，可自动检测故障以及从故障中恢复 — 无论故障发生在网络、主机、操作系统还是数据库层，同时最大限度地减少计划内维护活动导致的停机。
- **MySQL 企业级备份** — 执行 MySQL 数据库的无阻塞联机备份。MySQL 企业级备份还支持完整、增量和部分备份，同时支持压缩和时间点恢复。
- **MySQL 企业级监控器和查询分析器** — 持续监视 MySQL 数据库并主动向 DBA 提醒潜在问题、查询和调优机会，避免对关键系统或应用程序产生影响。该监控器提供了一系列 MySQL 专业顾问程序，可提供有关如何修复和调优 MySQL 配置和变量的洞察和详细指导，以实现最佳安全性、性能和可用性。内置的高级查询分析器可帮助开发人员以直观的方式查找和调优资源占用高的查询代码，而不需要使用慢查询日志、显示进程列表或其他劳动密集型方法。
- **MySQL Workbench** — 为数据库架构师、开发人员和 DBA 提供基于 GUI 的数据建模、SQL 开发、部署、数据库迁移和全面的管理工具（服务器配置、用户管理、对象管理）。



- **Oracle MySQL 终身标准支持服务** — 用户可直接联系 MySQL 专业支持工程师，获取关于 MySQL 应用程序开发、部署和管理的全方位支持。MySQL 支持团队由经验丰富的 MySQL 开发人员和数据库专家组成，他们了解用户面临的问题和挑战，因为他们之前都曾独立成功应对过同样的挑战。

总结

MySQL 凭借其性能、可靠性和易用性，成为全球最受欢迎、使用最广泛的开源数据库。在此基础上，MySQL 5.6 为 DBA 和开发人员提供了更高的性能、可伸缩性和全面的改进，可帮助他们构建下一代基于 Web、基于云和嵌入式应用程序及服务。简要总结如下：

- MySQL 5.6 的内部“体系”经过全面改造和重新设计，可以在各种平台上充分利用当今开发框架和多处理架构的计算能力。
- 为了实现更出色、更优化性能和可伸缩性，MySQL 5.6 通过改进 InnoDB 消除了原有的瓶颈和争用。
- 凭借联机模式更改和对数据库的 NoSQL 访问等创新，MySQL 可满足最严苛的应用程序需求，不会牺牲任何 ACID 合规性和 SQL 查询能力。
- 优化器进行了全新设计，支持更高效的查询执行和诊断报告，可以更加有效地调试和分析底层决策路径。
- 复制进行了改进，可确保主从服务器保持同步，同时改进了自动恢复选项、提高了数据完整性和应用程序整体可用性。新增的复制实用工具为 DBA 提供了可配置的自动选项，可确保最高水平的应用程序故障切换和恢复。
- 最后，新增的 Performance Schema 使 DBA 和开发人员能够更好地监测和洞察 MySQL 数据库性能量度，这样他们可以了解应用程序将宝贵的资源用于何处以及如何调优来实现最佳性能。

这些改进进一步展现了 Oracle 推进 MySQL 创新以及将 MySQL 5.6 打造成有史以来最佳数据库的坚定决心。

了解更多信息

MySQL 5.6 下载

<http://www.mysql.com/downloads/>

MySQL 5.6 文档

<http://dev.mysql.com/doc/refman/5.6/en/>

MySQL 企业版

<http://mysql.com/products/enterprise/>

MySQL 客户和案例研究

<http://www.mysql.com/customers>

MySQL 专业服务和咨询

<http://mysql.com/consulting/>

有关 Oracle MySQL 产品和服务的更多信息，请访问：

<http://www.mysql.com/products/>。



请直接联络甲骨文公司（免费电话）
可以直接联系 MySQL 部门获得更多的产品和服务信息
[MySQL Business Development]
中国北方地区：10800-811-0823
中国南方地区：10800-281-2682
香港：800-96-2447
台湾：00801-81-2732
【受理时间】
平时 9:00-12:00/13:00-18:00 （节假日与年末年初停业日除外）
mysql-sales_cn_grp@oracle.com