



MySQL 5.7的新增功能 白皮书

MySQL白皮书

2015年10月



综述	4
性能提升	4
3 倍于 MySQL 5.6 : SQL Point Selects	4
原生 JSON 支持	5
Performance Schema	7
SYS Schema	7
InnoDB 相关改进	9
InnoDB 在线操作	9
InnoDB 一般表空间	9
InnoDB 透明的页级别压缩	9
InnoDB 原生分区	11
InnoDB 原生全文检索	11
InnoDB 缓存保留	13
优化器相关	13
优化器代价模型 (Optimizer Cost Model)	14
JSON Explain	16
改进后的新优化器 Hints	17
Query Rewrite 插件 (Query Rewrite Plugin)	17
Generated Columns	17
服务器端语句执行超时	19
复制方面改进	19
多源复制	19
基于事务的并行复制	20
在线复制变更	21
半同步复制相关改进	22
加强版监控	22
Group Replication	22
高可用相关改进	23
GIS 相关改进	25
GeoJSON	25
GeoHash	25
Spatial Aware Client Tools (空间感知客户端工具)	25



安全性相关改进	27
AES 256 加密.....	27
密码轮换策略	29
MySQL 安全模式安装	29
MySQL 企业版.....	29
总结	33
附加资源	34



综述

时至今日，MySQL 是最受信赖和可依赖的开源数据库平台。在全球最流行、访问量最大的前十名网站中，有九个是 MySQL 支撑的，这主要是因为其广泛的跨平台和适应多种技术栈，以及被普遍认可的高性能、可靠性和易用性。MySQL 5.7 在这些成就之上做了全面的改进，使那些有创新精神的 DBA 和开发者能够很容易地在最新的开发框架和硬件平台上开发和部署下一代的 web 应用、嵌入式系统以及 Cloud/SaaS/PaaS/DBaaS 应用。

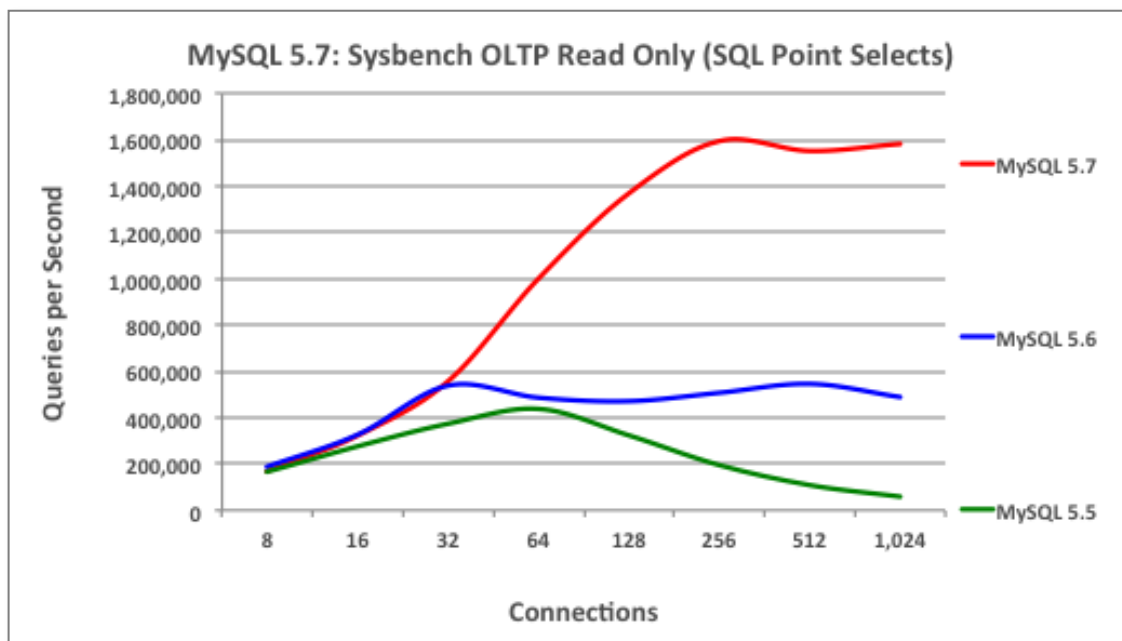
本白皮书会重点介绍 MySQL 5.7 的关键新特性，作为 MySQL 5.7 的 DBA 和开发人员指南进行发布。在本文尾部，会提供一个关于 MySQL 5.7 的部署的概要，介绍基于使用已经可以发布的 MySQL 企业版服务提供的支持和服务的最佳实践。

性能提升

MySQL 5.7 在所有负载模型上都有显著的性能改进，并创造了新的基准测试纪录。

3 倍于 MySQL 5.6 : SQL Point Selects

MySQL 5.7 在 Point Select 查询测试中，测试成绩可以达到 1600000 QPS，是 MySQL 5.6 的 3 倍多。





测试环境如下：

- Intel(R) Xeon(R) CPU E7-8890 v3
- 4 sockets x 18 cores-HT (144 CPU threads)
- 2.5 Ghz, 512GB RAM
- Linux kernel 3.16

关于 MySQL 5.7 基准测试更多信息，请访问：

<http://www.mysql.com/why-mysql/benchmarks/>

适用于特定测试应用案例的 Sysbench 基准测试工具是免费的，可以在这里下载：

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

原生 JSON 支持

在 MySQL 5.7 中，新增了一种新的数据类型，用来在 MySQL 的表中存储 JSON 格式的数据。原生支持 JSON 数据类型主要有如下好处：

- **文档校验** – 只，有符合JSON规范数据段的才能被写入类型为 JSON 的列中，所以其实相当于有了自动JSON语法校验。
- **高效访问** – 更重要的是，当你在一个 JSON 类型的列中存储 JSON 文档的时候，数据不会被视作纯文本进行存储。实际上，数据用一种优化后的二进制格式进行存储，以便可以更快速地访问其对象成员和数组元素。
- **性能** – 可以在 JSON 类型列的数据上创建索引以提升 query 性能。这种索引可以由在虚拟列上所建的“函数索引”来实现。
- **便捷** – 针对 JSON 类型列附加的内联语法可以非常自然地在 SQL 语句中集成文档查询。例如 (features.feature 是一个 JSON 字段)：

```
SELECT feature->"$.properties.STREET" AS property_street FROM features  
WHERE id = 121254;
```

使用 MySQL 5.7，你现在可以在一个工具中无缝地混合最好的关系和文档范例，在不同的应用和使用案例中应用关系型范例或文档性范例当中最适合的范例。这为 MySQL 用户大大扩大了应用范围。

更多信息：

- [Native JSON Data Type and Binary Format](#)
- [JSON Functions, Part 1 - Manipulation JSON Data](#)
- [JSON Functions, Part 2 – Querying JSON Data](#)



- [Virtual Columns and Effective Functional Indexes in InnoDB](#)
- [JSON Manual](#)

Performance Schema

对于任何数据管理系统而言，监控是必要的，从用户和客户的角度来说，监控同样很重要。我们的核心的监控策略是提供 Performance Schema。Performance Schema 在 MySQL 5.7 中的改进包括大量新加入的监控项、降低占用空间和负载，以及通过新的 SYS Schema 机制显著提升易用性。在监控方面，我们已经提供了：

- **元数据锁 (Metadata Locking)** – 对于了解会话之间 metadata 锁依赖至关重要。
- **进度跟踪 (Stage Tracking)** – 跟踪长时间操作的进度（比如 ALTER TABLE）。
- **事务** – 监控服务层和存储引擎层的事务的全部方面。
- **内存使用** – 提供聚合内存使用信息统计以了解和调解服务的内存消耗。
- **存储程序 (Stored Programs)** – 存储过程、存储方法、事件调度器和表触发器的检测器。
- **预编译语句** – 通过预编译语句提供聚合统计信息，并且展示服务使用的预编译语句。

我们已经在 Performance Schema 中展示了 SHOW SLAVE STATUS 的信息和用户变量。我们在保持向后兼容的同时，还展示了全局 (GLOBAL) 和会话 (SESSION) 级的状态 (STATUS) 与变量 (VARIABLES)。

更多信息：

- [Performance Schema Reference Manual](#)
- [Performance Schema and SYS Schema in MySQL 5.7](#)
- [Performance Schema Overview & New Features](#)

SYS Schema

MySQL SYS Schema 是一个由一系列对象（视图、存储过程、存储方法、表和触发器）组成的 database schema，以使主要存储在 Performance Schema 和 INFORMATION_SCHEMA 中诸多的表中监测数据资源可以以方便、可读、对 DBA 和开发者的友好的方式进行访问。

MySQL SYS Schema 默认包含在 MySQL 5.7 中，并提供摘要视图以回答诸如下面所列的常见问题：

- “谁占了我数据库服务的所有资源？”



- “哪些主机对数据库服务器的访问量最大？”
- “我实例上的内存都上哪去了？”



更多信息：

- [The MySQL SYS Schema in MySQL 5.7](#)
- [MySQL 5.7 中的 Performance Schema和SYS Schema](#)
- [MySQL SYS Schema Version 1.5.0 Released](#)

InnoDB 相关改进

InnoDB 在线操作

MySQL 5.7 支持在线操作以满足需要在生产环境中管理 MySQL 的开发者和 DBA的需求。我们在这方面新加入的最重要特性就是在线更改 InnoDB Buffer Pool。现在你可以在不重启 MySQL 的情况下，动态地调整你的 Buffer Pool size 以适应需求的变化。现在 InnoDB 也可以在线自动清空 InnoDB 的 UNDO 日志和表空间，这样就消除了产生大共享表空间文件 (ibdata1) 问题的一个常见原因。最后，我们添加了对在线重命名索引和增大 VARCHAR 列长的在线变更的支持。

更多信息：

- [Truncate Undo Tablespace](#)
- [InnoDB Buffer Pool Online Resize](#)
- [InnoDB Overview of Online DDL](#)

InnoDB 一般表空间

现在，您可以把数据库和表归组到逻辑和物理表空间中，这样做可以提高资源的利用率，并且可以支持在诸多情况下（如每个客户/用户使用专门的表空间）都以简易而高效的方式来做数据迁移。

更多信息：

- [InnoDB General Tablespaces](#)

InnoDB 透明的页级别压缩

自 MySQL 5.1 开始，InnoDB 支持了表压缩特性。InnoDB 页级别的压缩 — 这个想



法最初 FusionIO (现在是SanDisk的一部分) 提出——是 MySQL 5.7 的一个新特性，它补充了 InnoDB 的表级压缩，它们可以在同一个服务实例上并存。用户现在可以选择最适合他们使用场景的压缩方式，甚至基于逐表选择。

关于压缩算法，目前支持 Zlib 和 LZ4。当一个页被写入时，它就被指定的压缩算法压缩。压缩后的数据写到磁盘上，随后 "hole punching" 机制会在页的末尾出释放空块。如果压缩失败，数据则被原样写入。

InnoDB 现在也支持 32K 和 64K 的页大小设置，这对于页级的压缩来说是一个很好的补充。一般来说，更大的页通常会增加冗余的数据量，从而提高潜在的压缩比。

除了上述工作，我们还为 MySQL 5.7 增加了用户可配置填充因子和页合并抑制的新特性，这样就可以让 InnoDB 更好地使用你的存储空间。

更多信息：

- [InnoDB Transparent Page Compression](#)
- [InnoDB Supporting Page Sizes of 32k and 64k](#)

InnoDB 原生分区

在 MySQL 5.7 InnoDB 中包含了对分区的原生支持。InnoDB 原生分区会降低负载，减少多达 90% 的内存需求。这项工作也为我们消除分区表上的限制、新加全局索引和并行查询执行铺平了道路。

更多信息：

- [InnoDB Native Partitioning](#)

InnoDB 原生全文检索

我们在 5.6 中引入了全文检索。在 MySQL 5.7 中，我们加入了更灵活和更进一步的优化。例如，InnoDB 中的全文索引现在支持外部解析器。插件既可以替换掉内建的解析器，也可以作为是内部解析器的一个前端。我们也还实现了查询传给优化器的提示功能，这样 InnoDB 就可以略过部分全文检索的步骤，比如，如果不需要计算排名，则不计算。

从 MySQL 5.7.6 开始，MySQL 还提供了一个内建的全文 -gram 解析器插件，它可以支持中文、日文和韩文 (CJK)，以及可额外安装的 MeCab 日文全文解析器。



更多信息：

- [InnoDB Supports Full-Text Plugin Parser](#)
- [MySQL Full-Text Plugins](#)
- [InnoDB Full-Text N-gram Parser](#)
- [InnoDB MeCab Full-Text Parser Plugin](#)



InnoDB 缓存保留

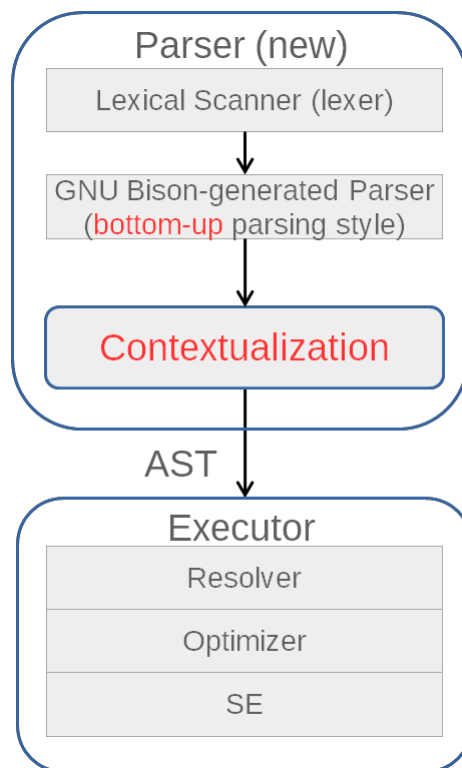
现在，当 MySQL 重启时，InnoDB 自动保留你缓存池中最热的 25% 的数据。你再也不需要任何预加载或预热你数据缓存的工作，也不需要承担 MySQL 重启带来的性能损失。

更多信息：

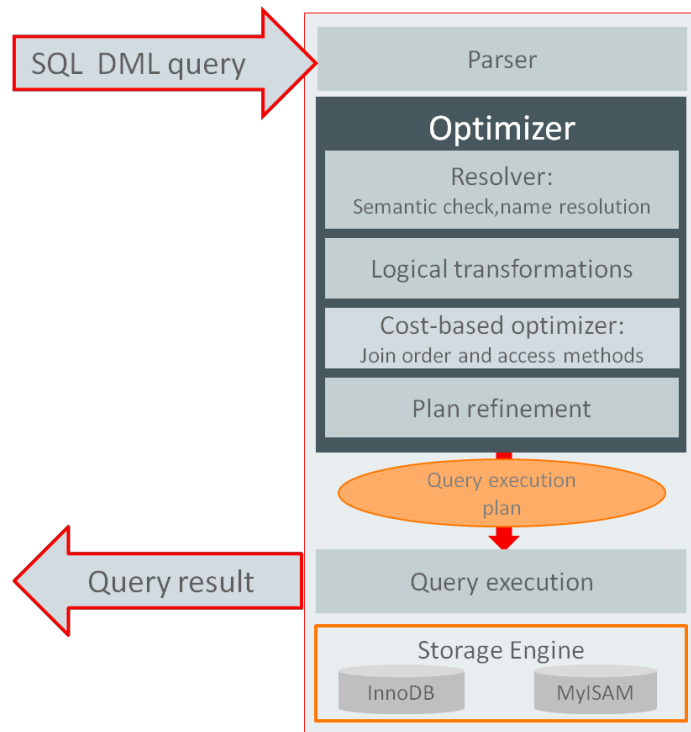
- [InnoDB Preload Buffer Pool](#)
- [MySQL Dumping and Reloading the InnoDB Buffer Pool](#)

优化器相关

我们对 MySQL 的优化器进行了重构，其中 parser、optimizer、cost model 的大部分模块都进行重写，以提升可维护性、扩展性和性能。



New MySQL Parser



New MySQL Optimizer

优化器代价模型 (Optimizer Cost Model)

为了生成一个合理的执行计划，优化器使用代价模型，这种模型基于估算 query 执行过程中各种操作的资源消耗成本。一直以来，优化器都是依赖一组编译的默认“成本常量”来做出执行计划决策。

MySQL 5.7 中，代价模型优化器被加以改进，以做出动态的，智能的，最终更优的资源消耗决策。比如，代价模型考虑了新的硬件架构（大容量的 buffers&cache，SSD 等）。好的代价模型可以让优化器做出更好的优化决策，进而可以优化 query 性能。

在 MySQL 5.7 中，优化器提供了一个可以配置的数据库。这个数据库中会存储在执行计划被构建的时候生成的将要被使用的资源消耗预估数据。这些数据存储在 mysql 系统库中的 servercost 和 enginecost 表中。

更多信息：



- [The MySQL Optimizer Cost Model Project](#)
- [Optimizer Cost Model Manual](#)
- [Optimizer Cost Model Presentation](#)

JSON Explain

我们改进了 JSON EXPLAIN 的输出，现在的输出包括了相关的资源消耗预估值、需要被处理的数据总量的预估值。这些信息可以帮助 DBA 去查看不同执行计划之间的重要差别。

Cost in Explain JSON

- Total cost for query
- Cost per table:
 - Cost for reading data
 - Cost for evaluating conditions
- Cost for “join prefix”

```
EXPLAIN FORMAT=JSON
SELECT * FROM t1 WHERE a BETWEEN 20 AND 23;
{
  "query_block": {
    "select_id": 1,
    "cost_info": {
      "query_cost": "112709.41"
    },
    "table": {
      "table_name": "t1",
      "access_type": "range",
      "possible_keys": [
        "idx1"
      ],
      "rows_examined_per_scan": 80506,
      "rows_produced_per_join": 80506,
      "filtered": 100,
      "index_condition": "{'test': 't1', 'a' between 20 and 23}",
      "cost_info": {
        "read_cost": "96608.21",
        "eval_cost": "16101.20",
        "prefix_cost": "112709.41",
        "data_read_per_join": "19M"
      }
    },
    "rows_examined_per_scan": 80506,
    "rows_produced_per_join": 80506,
    "filtered": 100,
    "index_condition": "{'test': 't1', 'a' between 20 and 23}",
    "cost_info": {
      "read_cost": "96608.21",
      "eval_cost": "16101.20",
      "prefix_cost": "112709.41",
      "data_read_per_join": "19M"
    }
  },
  "rows_examined_per_scan": 80506,
  "rows_produced_per_join": 80506,
  "filtered": 100,
  "index_condition": "{'test': 't1', 'a' between 20 and 23}",
  "cost_info": {
    "read_cost": "96608.21",
    "eval_cost": "16101.20",
    "prefix_cost": "112709.41",
    "data_read_per_join": "19M"
  }
}
```

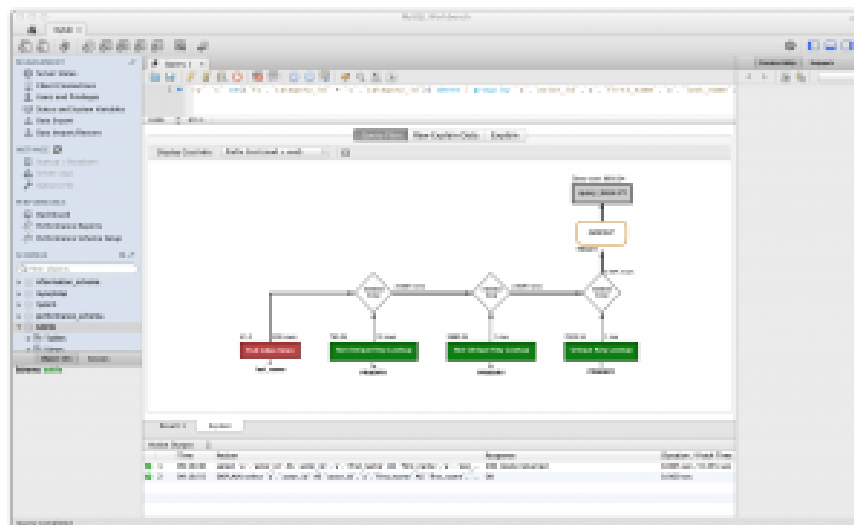
Explain Costs

更多信息：

- [Optimizer Cost Model Improvements in MySQL 5.7.5 DMR](#)
- [Explain Output Manual](#)

Workbench 可视化 Explain

这些新的 insights 还允许客户端提供此数据的其他可视化表示。MySQL Workbench 的可视化 Explain 就是一个很好的例子。



更多信息：

- [MySQL EXPLAIN Explained](#)

改进后的新优化器 Hints

我们添加了新的解析器规则以支持新的优化器 Hints 的 hint 语法。现在 hint 可以直接写在语句的 SELECT , INSERT , REPLACE , UPDATE , DELETE 关键字后面，使用类似注释 /*+*/ 的语法风格。这种新的语法不仅会给用户提供更加一致、强大、易用的方式去管理 hint，而且我们还对这部分进行了内部重构，使得后期可以很方便的增加新的 hint。例如，在 MySQL 5.7 中我们增加了很多索引和 join 连接的 hint。

更多信息：

- [New Optimizer Hints in MySQL](#)
- [Optimizer Hints Manual](#)

Query Rewrite 插件 (Query Rewrite Plugin)

MySQL 5.7 内部集成了 Query Rewrite 插件。Query Rewrite 插件可以让用户在 query 被服务器端执行之前按照用户设置的规则进行改写。比如 Rewrite 插件可以通过对一个 query 添加 hints 的方式来阻止或者加强这个语句的执行计划。Query Rewrite 插件可以近乎完美地解决下面的问题，你的应用有很多很不好的 query，但是你又没有什么好的手段去优化他们，因为他们从属于其他的外部工具（比如像 Hibernate 这样的对象关系映射工具）或者其他的第三程序。作为 query rewrite 插件的一部分，我们还创建了两个新的 API，用户可以使用这两个 API 来定制自己的 Query Rewrite 插件。

更多信息：

- [Query Rewrite Plugins](#)
- [Write Yourself a Query Rewrite Plugin Part 1](#)
- [Write Yourself a Query Rewrite Plugin Part 2](#)

Generated Columns



Generated Columns 是一种虚拟的列，数据并不会存储在表中。除非你在定义的时候声明该列是 STORED。该功能提供了一种便捷的途径去“缓存”一些经常需要通过计算才能获得的值。并且通过 STORED Generated Columns 你可以实现类似“函



数索引”的功能 (STORED Generated Columns 可以作为索引的一部分，会在数据发生更新的时候被更新并存储)。当你将该功能和 schema-less 列混合使用的时候会方便很多，比如使用 JSON 文档这个数据类型。

更多信息：

- [Generated Columns in MySQL 5.7.5](#)
- [Create Table Manual](#)

服务器端语句执行超时

该功能可以让用户在不同的维度设置 query 执行超时时间，可选的维度有全局设置、连接设置，也可以针对单个的语句进行设置。例如：

```
SELECT /*+ MAX_EXECUTION_TIME(1000) */ * FROM my_table;
```

更多信息：

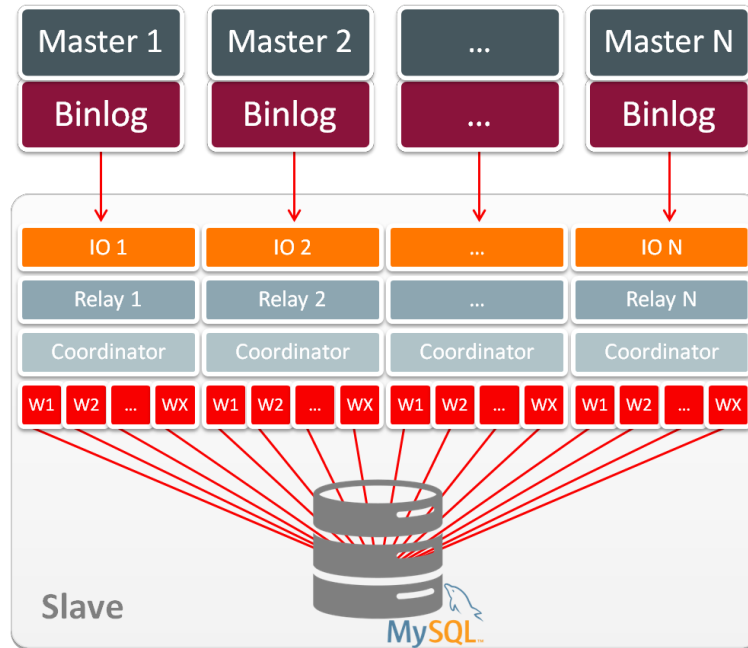
- [Server Side SELECT Statement Timeouts](#)

复制方面改进

多源复制

MySQL 多源复制可以使从节点同时接收多个源端的事务，多源复制可以用于：

- 多个 Server 数据合并到一个节点上
- 多个 Server 数据备份到一个节点上
- 合并表分片



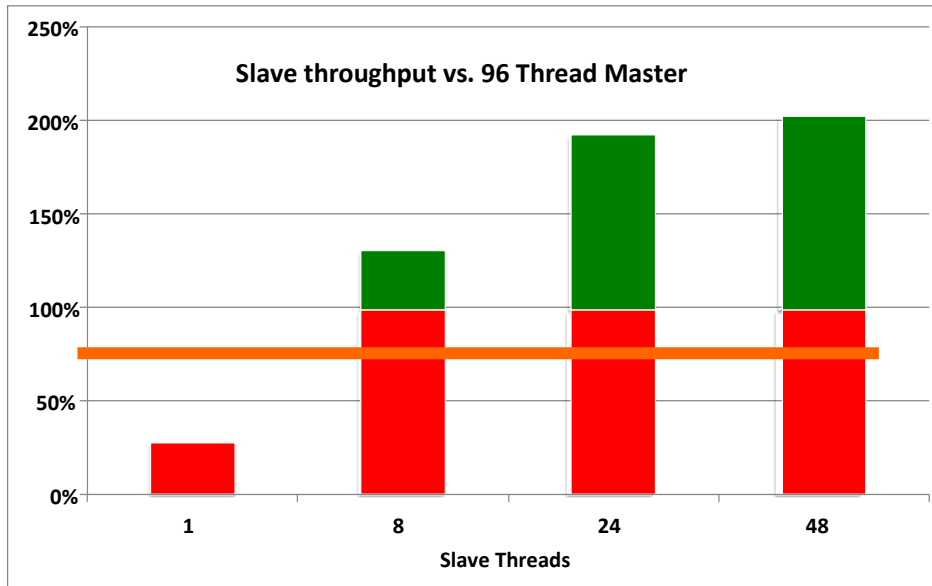
监控接口提供了每个复制通道的详细数据，并且你可以很容易地通过新的 `performance_schema` 中的表来监控你的整个多源复制的拓扑结构。

更多信息：

- [Introducing Multi-Source Replication](#)
- [Multi-Source Replication Manual](#)

基于事务的并行复制

MySQL 5.7 添加了 intra-schema 多线程复制功能，通过设置参数 [slave-parallel-type=LOGICAL-CLOCK](#)，即便是在同一个 DB 或者 Schema 中，只要存在非关联的读集和写集，从库节点就可以并行地接收事务。这样复制能够更快地跟上主库节点，消除最常见的导致“从库延迟”的问题。

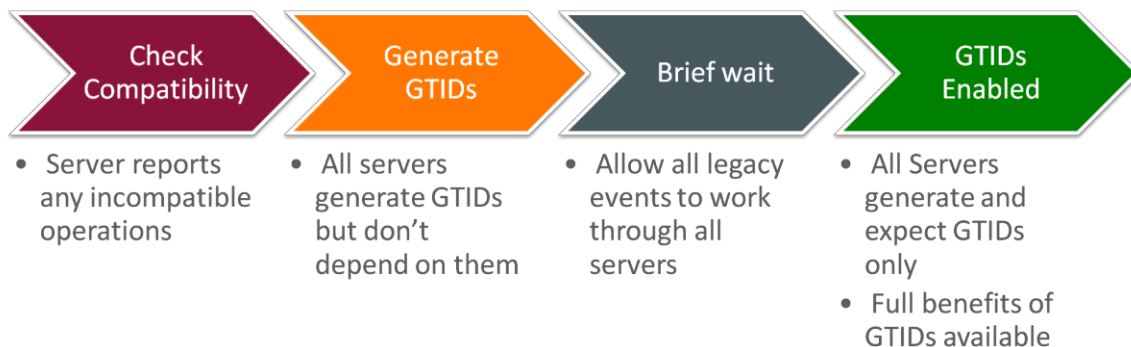


更多信息：

- [MySQL 5.7 Enhanced MTS: configuring slave for Intra-database parallelization](#)

在线复制变更

利用下一代复制特性，你可以在无需暂停 MySQL 生产环境的服务的情况下在线开启基于 GTID 的复制。



现在，你还可以对复制过滤器进行在线变更，可以使用多种不同的方法来配置你的 MySQL 集群的数据复制，最后，你现在可以在无需停止从库节点的复制执行的情况下执行主库节点故障转移操作 (CHANGE MASTER)。

更多信息：

- [Replication Enable GTIDs Online](#)
- [Enabling GTIDs Without Downtime](#)
- [Change Replication Filter](#)

半同步复制相关改进

半同步复制插件做了相关的语义改进，这些改进可以提供更好的性能和可靠性。

更多信息：

- [Loss-Less Semi-Sync Replication](#)

加强版监控

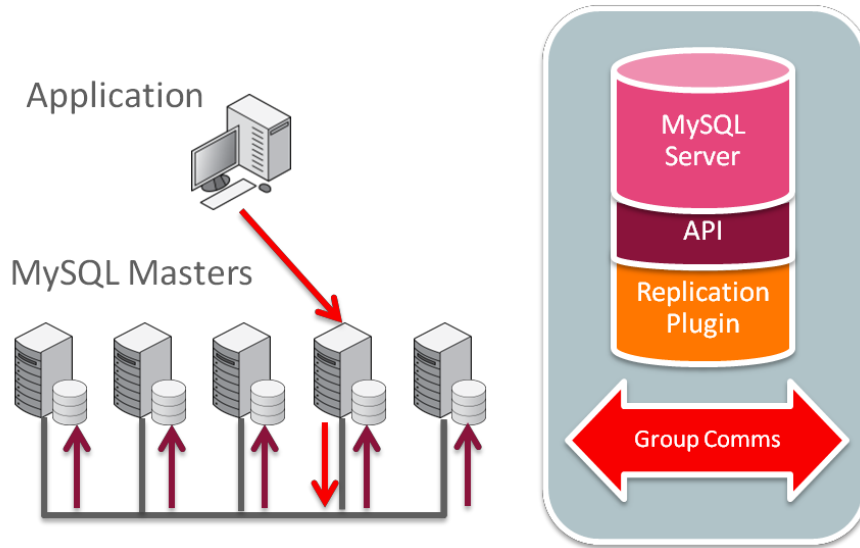
作为传统的 SHOW 命令的补充，我们添加了诸多新的 Performance Schema 表。这些表可以展示出前所未有的关于“正在发生什么”，“正在干什么”的细节。这可以让你轻易地确认你的复制拓扑是否健康，相关行为是否正常。同时这些信息也包括你在调试可能发生的问题时所需要的信息。

更多信息：

- [Performance Schema Replication Tables](#)

Group Replication

随着该特性的引入，MySQL 就可以组成一个 replication cluster，在这个 cluster 里面，原生的支持多节点写入。现在你可以根据特定的需求选择合适的复制模型组合：异步复制、半同步复制、（虚拟）同步复制 (Group Replication)。



Group Replication 负责节点之间关系、一致性和其他的管理相关的功能，不需要人为介入或其他工具协助。MySQL 简单高可用时代已经到来。

更多信息：

- [MySQL Group Replication: Hello World](#)
- [MySQL Group Replication Blog Posts](#)
- [State of Group Replication Presentation](#)

高可用相关改进

对跟踪事务会话状态进行了支持，以便于更好地在集群的不同节点中进行负载均衡。

Server version tokens可以更好地支持分布式系统中的缓存了。

一个新的数据迁移工具 – mysqlpump – 改进了节点之间的数据迁移和分片操作（如：分片再平衡）

高可用组中改进后的复制选项：

- 基于事务的并行复制的性能改善
- 无损的半同步复制插件支持多节点的 acks
- 同步复制 (Group Replication plugin)



更多信息：

- [MySQL Session Tracking](#)
- [MySQL Version Tokens](#)
- [Introducing mysqlpump](#)

GIS 相关改进

我们认识到了移动端应用规模迅速增长及基于地理位置的服务需求的大规模增加。为了满足这些需求，MySQL 5.7 在 GIS 方面做了重大的改进。

改进的新版 Spatial 扩展

MySQL 5.7 使用了一个外部库替换了原有的用户几何计算的相关旧代码。我们将这个库命名为 Boost.Geometry.

MySQL 5.7 用 Boost.Geometry 中的函数库取代了之前用于几何计算的代码。Boost.Geometry 是一个强大可靠的开源几何引擎，它拥有充满活力的开发社区，我们也活跃于其中。在我们内部有两个重要的 Boost.Geometry 开发者，我们很乐于为这个上游项目 (upstream project) 回馈代码。所有这些的最终结果是使我们有了一个更好性能、更全面特性、可靠并符合标准的 MySQL GIS 支持。

Innodb Spatial 索引

Innodb 现在支持 Spatial 索引（以 R 树索引实现），这使与 Spatial 相关的检索更高效，同时还增添了现代企业所需要的事务支持、MVCC 和 ACID 保证。

GeoJSON

近年来，GeoJSON 因其简单轻量容易阅读的特点成为了交换GIS数据的流行数据格式。在MySQL5.7中，我们添加了两个新的函数 [ST_GeomFromGeoJson\(\)](#) 及 [ST_AsGeoJson\(\)](#) 以实现 对 GeoJSON 类型的文档的解析与生成的支持。这些函数使得支持 GeoJSON 的软件和服务可以很容易的连接 MySQL，例如谷歌地图 JS API。

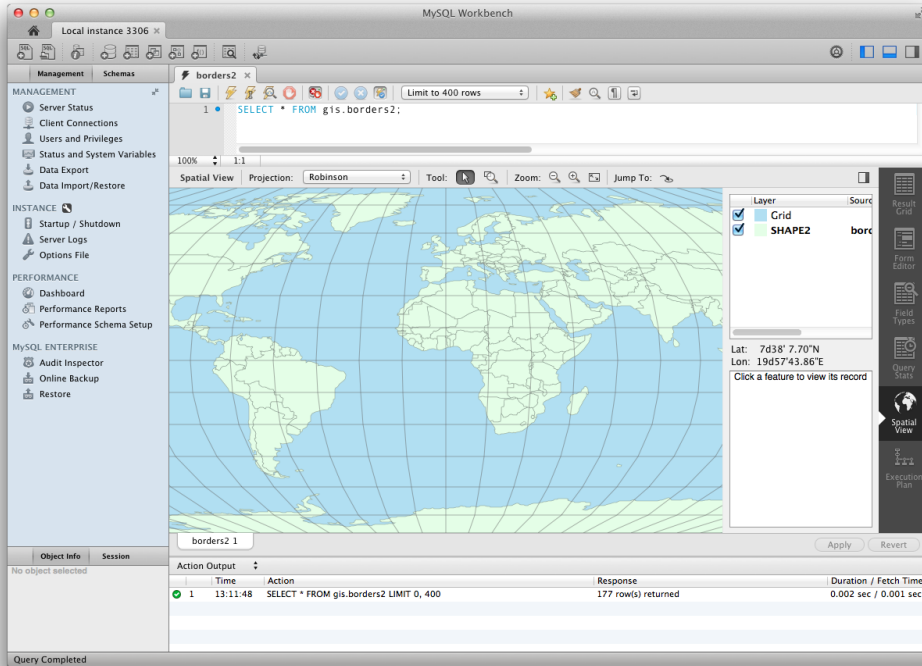
GeoHash

在 MySQL5.7 中我们引入了编码和解码 Geohash 数据的新函数，Geohash 是一个众所周知用于编码和解码经度纬度坐标为文本字符串的系统（在 WGS 84 坐标系中）。

Spatial Aware Client Tools (空间感知客户端工具)



我们的图形客户端工具 MySQL Workbench 新添加了一个Spatial查看器，在结果中检测到 GEOMETRY 类型列时他会自动被显示为一个查看选项。我们现在也在 Sakila 样例数据库中合并了有效的空间数据 [\(address.location column\)](#) 使测试新的 GIS 功能更简单便利。



MySQL Workbench 为开发 GIS 应用提供了可视化工具

更多信息：

- [MySQL 5.7 and GIS, an Example](#)
- [GeoJSON Functions](#)
- [Geohash Functions](#)
- [Spatial Extensions Manual](#)

安全性相关改进

MySQL 5.7 新增了“默认为安全模式”的安装形式，以及很多其他特性来避免用户数据库实例遭到非计划使用，包括更好的数据加密、更好的密码处理、更好的传输层安全，等等

AES 256 加密

MySQL 支持多种 AES 加密模式。我们通过对更大的密钥长度和不同的块模式支持来增强了高级加密标准 (AES) 的加密/解密函数 - [AES_ENCRYPT\(\)](#) 和



[AES_DECRYPT\(\)](#) 的安全力度。



密码轮换策略

MySQL 允许数据库管理员手动设置账户密码过期，也可以建立一个自动密码超期的策略。任何密码超期的账户想要连接服务端时都必须更改密码。

更多信息：

- [Password Expiration Policy in MySQL Server 5.7](#)

MySQL 安全模式安装

mysql_secure_installation 允许用户通过以下方式改善 MySQL 安装的安全性：

- 你可以为root账户设置密码
- 你可以移除能从本地主机以外的地址访问数据库的root账户
- 你可以移除匿名账户
- 你可以移除test数据库(该数据库默认可被任意用户甚至匿名账户访问)

使用 `mysqld --initialize` 来安装的 MySQL 实例默认是安全的

- 安装过程只创建一个 root 账户'`root`'@'`localhost`'，自动为这个账户生成一个随机密码并标记了密码过期。数据库管理员必须用 root 账户及该随机密码登录并设置一个新密码后才能对数据库进行正常操作。
- 安装过程不创建任何匿名账户
- 安装过程不创建 test 数据库

MySQL 企业版

- **MySQL Database** – 最安全和最新版本的 MySQL 数据库被用来驱动要求最为严苛的在线，web，云，OLTP 等类型的应用及服务。MySQL 企业版是一个完全集成的事务安全，ACID 兼容的数据库，同时具有完全提交，回滚，崩溃恢复和行级锁定功能。MySQL 以其在所有 Linux，UNIX，Mac OS X 和 Windows 平台上的性能，可靠性和易用性而闻名。



- **MySQL Enterprise Backup** – MySQL Enterprise Backup 可在线不阻塞的备份 MySQL 数据库。在 MySQL 对事务操作都完全可用的情况下可执行所有 Innodb 数据的全量、增量和部分备份。所有的备份操作可平行执行，而且支持压缩可将备份文件大小减小高达 90%。恢复包括备份完全恢复、精细的定点恢复以及对于存储了特定对象的部分恢复。

- **MySQL Enterprise Firewall** – 阻断会导致有价值的个人和财务数据损失的 SQL 注入攻击。创建白名单，实时威胁监控，SQL 语句阻塞，报警提醒数据库管理员保护重要的数据资产。作为一种入侵检测系统，MySQL Enterprise Firewall 可以通知管理员不符合白名单的 SQL 语句情况。
- **MySQL Enterprise Audit** – MySQL Enterprise Audit 提供了一种简单易用，基于策略的审核解决方案来帮助开发人员在更不更改现有应用的情况下执行更强的安全控制并满足监管要求。
- **MySQL Enterprise Scalability** – 为了满足可持续的性能，不断增长的用户、查询和数据负载的可扩展性 — MySQL 企业版提供了 MySQL 线程池。企业版线程池的高扩展性、基于队列的线程处理模式旨在减少客户端连接管理以及语句处理线程的开销。
- **MySQL Enterprise Authentication** – MySQL 企业版提供了外部验证模块接口，所以 DBA 和开发者可以将现有的安全设施，包括 Linux 插件式认证模块 (PAM) 以及 Windows 活动目录等与 MySQL 简单的整合在一起。
- **MySQL Enterprise Encryption** – 为了保护敏感数据的安全性，MySQL 企业版加密策略为非对称加密提供了行业标准的功能(公钥加密系统)。同时也提供了加密、密钥生成、数字签名和其他的加密功能以保护机密数据并符合如 HIPAA (健康保险流通与责任法案)、Sarbanes-Oxley (萨班斯-奥克斯利法案) 以及 PCI 数据安全标准的监管需求。
- **MySQL Enterprise High Availability** – MySQL 企业版提供了很多数据库高可用的解决方案来自动检测故障的发生，并不论这些故障发生在网络、主机、操作系统还是数据库层面，都可以自动恢复，而且减少因维护导致的停机时间。
- **MySQL Enterprise Monitor and Query Analyzer** – 持续性的监控 MySQL 数据库，在影响重要系统和应用之前发现潜在危险、有问题的查询语句以及可调优的机会并主动报警给 DBA。这个监控系统提供了一系列的 MySQL 专业指导，在为了优化数据库的安全、性能、可用性而修复和调整 MySQL 配置和参数时提供了详细清晰地指导。其中内置的查询分析器可以让开发人员不再需



要查看慢查询日志，SHOW PROCESSLIST 或者其他耗时耗力的方法就方便的找到并优化开销较大的查询代码。



- **Oracle Enterprise Manager for MySQL** – 用于 MySQL 的 Oracle Enterprise Manager 为 MySQL 数据库提供了实现的监控，全面的性能，以及可用性和配置信息。除了默认的MySQL规则，用户还可以根据自身的需要去定制临界点和报警阈值。
- **MySQL Workbench** – 是为数据库架构师，开发人员和 DBA 提供的以 GUI 为基础，用于数据建模、SQL 开发，部署，数据库迁移和全面管理（如服务器配置，用户管理，对象管理等）的工具。
- **Oracle Premier Lifetime Support for MySQL** – 有专业的 MySQL 支持工程师随时可以为用户在 MySQL 应用的开发、部署和管理等方面的需求提供直接的帮助。MySQL 支持组由众多经验丰富曾解决过很多用户现在面对的问题和挑战的 MySQL 数据库专家和开发人员组成。

总结

MySQL 因其性能、可靠性和易于使用等特性成为了目前世界上最流行并被广泛使用的开源数据库。在此基础上，MySQL 5.7 为 DBA 和开发人员提供了有着更高性能、可扩展性和全面改善的新一代可基于网络，基于云的可嵌入式应用和服务。这些改进进一步证明 Oracle 促进了 MySQL 的创新，使得 MySQL5.7 成为了历史上拥有最全面最丰富特性的应用版本。



附加资源

MySQL 5.7 下载

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

MySQL 5.7 文档

<http://dev.mysql.com/doc/refman/5.7/en/>

MySQL Enterprise Edition

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

MySQL Customers and Case Studies

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

更多 Oracle 的 MySQL 产品和服务可访问：

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