

# MySQL 复制技术的研究及应用

张伟丽 江春华 魏劲超  
(电子科技大学 成都 610054)

**摘 要** 数据库复制技术是提高数据库系统并发性和容错性的重要技术。通过复制,将数据存储于一个分布式的网络中,由多个数据库系统来提供数据访问服务,能大大地提高数据库的响应速度和并发能力。针对当前数据库应用现状,分析和对比了几大主流数据库 ORACLE、SQL Server、Sybase、MySQL 的复制技术以及原理,着重探讨了 MySQL 的复制原理及其在分布式数据库系统中的应用,并给出了 MySQL 的双向链式复制在分布式系统中的应用,达到了分布式系统中数据同步复制的目的。

**关键词** MySQL,数据库,同步复制,异步复制,分布式数据库

中图分类号 TP392 文献标识码 A

## MySQL Replication Technology and its Applications

ZHANG Wei-li JIANG Chun-hua WEI Jin-chao

(University of Electronic Science and Technology of China, Chengdu 610054, China)

**Abstract** Database replication technology is an important technology which is aimed at improving the concurrency and fault-tolerant of database systems. With replication technology, same data can be stored in distributed system. Because multiple database system provides data access services, database response and concurrency have been greatly improved. This paper is based on the current status of database system applications, analyzes and compares the replication technology of several major databases, such as ORACLE, SQL Server, Sybase and MySQL. This paper focuses on MySQL replication technology and its concrete application in distributed database systems. In addition, this paper also proposed a model of chain replication of MySQL in distributed system. All of these achieve the purpose of data replication in distributed system.

**Keywords** MySQL, Database, Eager replication, Lazy replication, Distributed system

数据复制是将某一个节点的数据及其更新实时同步到其他一个或者多个节点,达到数据的实时更新及其一致性的目的。数据库复制过程中,允许在一个分布式数据库环境中创建和维护数据库对象的副本(通常表,索引,甚至代码)。复制可以提高数据库应用程序的性能,增加使用替代数据访问真实数据的可用性<sup>[1-3]</sup>。文献[5]提出了在 VANET(车辆自组织网络)分布式移动网络系统中,数据复制问题的解决方案。文献[4,7]均对复制机制中的写操作集部分进行了改进。文献[4]在 MySQL 的 InnoDB 存储引擎上实现了一种基于写操作集的多主同步复制模型,该模型能解决异步复制带来的延迟问题,并提高分布式 DBMS 系统的容错性和扩展性。文献[7]提出一种基于订阅写操作集应用的数据库复制策略,即使使用无序可靠的广播而非原子广播。该复制策略继承了基于认证的复制和弱投票复制的特性,在最后的验证提交阶段,写操作集事务并没有广播而是舍弃,这种新方法可减少通信流量和系统响应时间。文献[9]根据分布式数据库系统中的复制机制,选择或改进其中的方法,形成了一种新的解决分布式数据库复制问题的方案,并将其应用到一个实际的出版物发行信息系统之中,较为完善地解决了复制中遇到的问题。文献

[10]是在 MySQL 复制模型的基础上,提出了一种基于通信模型驱动的 Web 托管数据传输策略,该通信模式使用 MySQL 数据库复制和简单的数据库抽象模型,将客户端数据传输到从属托管服务器。该模型保证了异步数据传输的可靠性以及高度的可移植性。

一般而言,数据库复制技术可以从以下 3 个方面改善分布式数据库系统的功能和性能<sup>[4,5]</sup>:

(1)容错性:数据库集群系统具有多个数据库节点,在单个或多个节点出现故障的情况下,其他正常节点可以继续提供服务;

(2)可用性:多个节点一般可以并行处理请求,从而避免单节点的性能瓶颈,一般至少可以提高读操作的并发性能;

(3)可扩展性:单个数据库节点的处理能力毕竟有限,增加节点数量可以显著提高整个集群系统的吞吐量。

## 1 复制机制

数据库复制技术可以分为同步复制(Eager Replication)和异步复制(Lazy Replication)<sup>[4-6]</sup>。同步复制是指在事务提交之前,数据更新操作在所有数据副本中同步完成,主从数据

张伟丽(1988—),女,硕士生,主要研究方向为计算机网络、数据库,E-mail:zhwl-0804@163.com;江春华(1962—),男,副教授,主要研究方向为嵌入式软件技术与应用;魏劲超(1980—),男,硕士,主要研究方向为嵌入式系统。

库中的数据保持完全一致,从数据库对数据表只有读权限,例如数据集群。同步复制可以保障副本之间数据的一致性及较强的容错性,但是也带来了一系列的问题,其中包括死锁、通信量增加、节点规模的限制及事务响应时间的延长<sup>[6]</sup>。异步复制是指在事务提交之后,数据更新操作被传播到其它节点的异步复制中,主数据库中的数据更新操作会同步到所有从数据库中,但从数据库中的数据更新操作不会同步到主数据库中。异步复制的优点是降低了通信量,缩短了响应时间,提高了系统性能。但它最大的缺点是先提交后更改,这样在不同节点的副本之间存在数据不一致的现象<sup>[7]</sup>。

Oracle 数据库主要提供了两种技术来实现数据库的复制过程,一种是通过多主复制,另一种是通过物理视图或快照复制<sup>[1]</sup>。Oracle 自带的数据同步技术有 Data Guard, Streams, Advanced Replication 等。Data Guard 的基本原理是将日志文件从原数据库传输到目标数据库,然后在目标数据库上应用这些日志文件,从而使目标数据库与源数据库保持同步。Streams 是为提高数据库的高可用性和数据的分发和共享功能而设计的,它利用高级队列技术,通过用 Log Miner 挖掘日志文件生成变更的逻辑记录,然后将这些变更应用到目标数据库上,从而实现数据库之间或一个数据库内部的数据同步。

SQL Server 主要采用出版物、订阅的方式来处理数据复制。源数据所在的服务器是出版服务器,负责发表数据<sup>[8]</sup>。出版服务器将要发表的数据的所有改变情况的拷贝复制到分发服务器,分发服务器包含有一个分发数据库,可接收数据的所有改变并对这些改变进行保存,最后再把这些改变分发给订阅服务器。SQL Server 每隔一定时间进行一次复制,如果找不到另一台 SQL Server (比如因为网络故障或是另一台 SQL Server 没有启动),经过  $n$  (默认为 10) 次连接后,它的复制功能将会失效,直至人工将复制功能重新启动。

Sybase 数据复制技术是采用复制服务器来达到数据同步复制的目的<sup>[9]</sup>。在复制服务器环境中,对源数据库中数据的修改由一个称为日志传输管理(Log Transfer Manager 简称 LTM)的进程来实现,根据复制系统的配置,LTM 可以将改变数据的 Log 从一个复制服务器传送到其它复制服务器,最后由这些复制服务器将数据改变传给目标数据库,并且有能力在不同厂商提供的数据库源之间移动传递数据,如 Sybase、Oracle、SQL Server 等系统之间数据的复制和同步。

## 2 MySQL 复制原理及实现

MySQL 发展了一套自有的异步复制方案,其采用主从模式的异步复制机制<sup>[4,10]</sup>。主服务器维护一个记录更新的二进制日志,二进制日志负责记录事件。每个事件都包含一些与从服务器有关的信息,以便能够按照与主服务器相同的方式正确地执行更新。主服务器将更新写入二进制日志文件的同时,会维护文件的一个索引以跟踪日志循环,这些日志可以记录发送到从服务器的更新。当一个从服务器连接主服务器时,它通知主服务器从服务器在日志最后一次成功读取更新的位置。从服务器接收从那时起发生的任何更新,然后封锁并等待主服务器通知新的更新。在二进制日志馈送过程中,主服务器追踪日志序列一旦到达上一个日志的末尾,就自动

切换到下一个日志<sup>[4]</sup>。

MySQL 复制是基于主服务器在二进制日志中跟踪所有对数据库的更改(更新、删除等)<sup>[10]</sup>。因此,要进行复制必须在主服务器上启用二进制日志。而二进制日志只是一个从启用二进制日志的固定时间点开始的记录,任何设置的从服务器都需要主服务器上的二进制日志的数据库拷贝。如果启动从服务器时,其数据库与主服务器上启动二进制日志时的状态不相同,从服务器很可能失败。

MySQL 使用 3 个线程来执行复制功能,其中 1 个在主服务器上,另两个在从服务器上。图 1 为 MySQL 主从复制模型<sup>[4]</sup>。当主服务器发出 START SLAVE 命令时,从服务器创建一个 I/O 线程,以连接主服务器并让它发送记录在其二进制日志中的语句。主服务器创建一个 I/O 线程将二进制日志中的内容发送到从服务器。该线程可以识别为主服务器上 SHOW PROCESSLIST 的输出中的 Binlog Dump 线程<sup>[11]</sup>。从服务器 I/O 线程读取主服务器 Binlog Dump 线程发送的内容并将该数据拷贝到从服务器数据目录的本地文件中,即中继日志。第 3 个线程是 SQL 线程,是从服务器创建用于读取中继日志并执行日志中包含的更新。

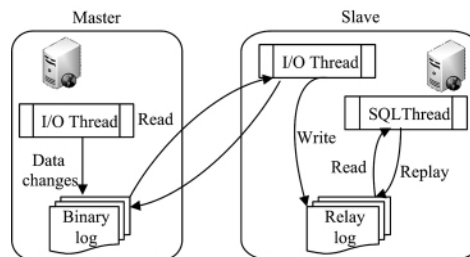


图 1 MySQL 主从复制模型

### 2.1 授予复制权限

假设主服务器 A,从服务器 B,A 修改数据后同步从服务器 B,即  $A \rightarrow B$ ,如图 2 所示。首先在主服务器建立复制账号,然后在主服务器上授予从服务器此账号的复制权限。

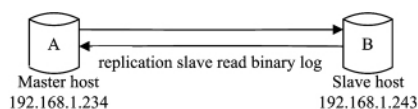


图 2 单项主从复制模型

```
mysql> GRANT REPLICATION SLAVE ON *.* TO
'root'@'192.168.1.234' IDENTIFIED BY '123';
mysql> FLUSH PRIVILEGES;使权限生效。
```

### 2.2 修改主服务上的配置文件

#### 2.2.1 开启二进制日志功能

首先开启主数据库的二进制日志功能:在 my. ini 文件的 [mysqld] 下添加一行 log-bin= binary-log 即可, binary-log 是日志文件的名字。

#### 2.2.2 设置需要同步的数据库或者数据表

在主数据服务器的 my. ini 文件的 [mysqld] 后添加如下代码:

```
server-id=1
log-bin= binary-log
binlog-do-db= database name
```

binlog-ignore-db=database name

server\_id 和 Master\_id 必须为 1 到  $2^{32}-1$  之间的一个正整数,并且从服务器的 ID 必须与主服务器的 ID 不同。binlog-do-db 为需要同步的数据库名字。binlog-ignore-db 为忽略要同步的数据库名字。

mysql> SHOW MASTER STATUS;查看日志状态。

### 2.3 修改从服务器上的配置文件

在从服务器的 my. ini 文件的[mysqld]下添加如下几行代码:

server-id=2

replicate-do-db=database name

replicate-ignore-db=database name

replicate-do-table=database, table name

replicate-ignore-table=database, table name

### 2.4 启动复制

在从服务器上启动复制:

mysql> CHANGE MASTER TO

MASTER\_HOST='192.168.1.134',

MASTER\_PORT=3306,

MASTER\_USER='user',

MASTER\_PASSWORD='123',

MASTER\_LOG\_FILE='binary-log.000001',

MASTER\_LOG\_POS=107;

其中,MASTER\_HOST 是主服务器的 IP 地址,MASTER\_PORT 是数据库的端口号,MASTER\_USER 是从服务器的用户名,MASTER\_PASSWORD 为密码,MASTER\_LOG\_FILE 为主服务器的二进制日志名,MASTER\_LOG\_POS 为日志的偏移量。

正确执行之后再输入命令行:mysql> START SLAVE;用来启动同步进程。

### 2.5 检查系统复制进程

在从服务器下查看复制进程:mysql> SHOW SLAVE STATUS;至此,MySQL 同步复制配置已经完成。修改主服务器数据,查看从服务器数据是否发生同步变化。

## 3 MySQL 双向异步复制模型

双向异步复制也叫链式复制,即从服务器本身也可以充当主服务器。服务器 A 配置了 binlog 且被启用,作为服务器 B 的主服务器,而服务器 B 也启用了自己的 binlog,同时作为服务器 A 的从服务器和服务器 C 的主服务器。服务器 A 修改数据后同步从服务器 B,然后从服务器 B 同步从服务器 C,即 A→B→C,如图 3 所示,图 3 中的 B1、B2 节点既充当主服务器,又充当从服务器。这样的复制架构很好地满足了分布式系统的多层架构,具有良好的扩展性。

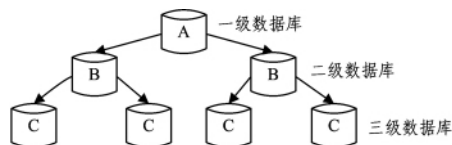


图 3 双向主从复制模型

在设置时,只需在从服务器 B 的配置文件 my. ini 文件的

[mysqld]后添加一行代码:log\_slave\_updates=1。该行代码的作用是在从服务器 B 的数据发生修改的同时将修改信息记录到日志文件中,这样从服务器 C 在同步数据时所读取的二进制文件不仅记录了 B 服务器自身对数据表的修改信息,还记录了 A 服务器对数据表的修改信息。

双向复制虽然提高了系统的可靠性,但是增加了数据冲突的风险。该方法一般用于大规模数据操作时处理数据库均衡负载的问题,负载均衡设备根据数据库的响应时间来判断把数据交给谁处理。

结束语 本文详细研究了 MySQL 数据库系统的复制原理以及实现技术,给出了 MySQL 复制技术在分布式数据库模型中的应用,并提出了双向链式复制模型。所介绍的 MySQL 单向异步复制模型和双向异步复制模型已经成功应用到铁路视频监控项目的分布式数据库系统中,其不仅可以满足源数据库和目标数据库运行在不同类型的操作系统和同一 MySQL 数据库的不同版本的要求,而且能够很好地满足系统的扩充性需求,达到了 MySQL 分布式数据库系统中数据同步复制的目标。

## 参考文献

- [1] Filip I, Vasar C, Robu R. Considerations about an Oracle Database Multi-Master Replication[C]// 5th International Symposium on Applied Computational Intelligence and Informatics, Timișoara, Romania, 2009:147-151
- [2] Lieskovsky A, Janech J, Baca T. Data replication in distributed database systems in VANET environment[C]// 11th Int. Conf. on ITS Telecommunications, 2011:447-451
- [3] de Mendivil J R G, Armendáriz-Inigo J E, Garitagoitia J R, et al. A formal analysis of database replication protocols with SI replicas and crash failures[J]. Springer Science + Business Media, LLC, 2008(50):121-161
- [4] 刘腾. MySQL 复制技术的研究与改进[D]. 杭州:浙江大学, 2011
- [5] 杜凯, 缪嘉嘉, 杨树强, 等. 数据库复制技术研究进展[J]. 计算机工程与科学, 2008, 30(7):117-122
- [6] 仲福建, 何远德. 高校招生数据库的可靠性研究与实现[J]. 绵阳师范学院学报, 2009, 28(11):78-80
- [7] Juarez-Rodriguez J R, Armendariz-Inigo J E, de Mendivil J R G, et al. A Database Replication Protocol Where Multicast Write sets Are Always Committed[C]// The Third International Conference on Availability, Reliability and Security, 2008:120-127
- [8] 盖九宇, 张忠能, 肖鹤. 分布式数据库数据复制技术的分析与应用[J]. 计算机应用与软件, 2005, 22(7):36-38
- [9] 陈宫, 牛秦洲. 分布式数据库复制问题的一种解决方案及应用[J]. 计算机与现代化, 2011, 11:138-141
- [10] Gudi A, Voisan E, Drăgan F. Database Replication Driven Communication Model for Distributed Dedicated Web Hosting Systems[C]// IEEE International Joint Conferences on Computational Cybernetics and Technical Informatics, Timisoara, Romania, 2010:311-314
- [11] 北京万里开源软件有限公司. MySQL 用户手册[K]. 北京, 2005