Atitit 数据库对事务机制

目录

[1.事务提交的方式 1](#_Toc23999)

[1.显式开启和提交。 1](#_Toc26921)

[2.自动提交。(MySQL默认的提交方式) 1](#_Toc12944)

[3.隐式提交事务 1](#_Toc25817)

# **1.事务提交的方式**

在MariaDB/MySQL中有3种事务提交的方式。

## 1.显式开启和提交。

使用begin或者start transaction来显式开启一个事务，显式开启的事务必须使用commit或者rollback显式提交或回滚。几种特殊的情况除外：行版本隔离级别下的更新冲突和死锁会自动回滚。

在存储过程中开启事务时必须使用start transaction，因为begin会被存储过程解析为begin...end结构块。

另外，MariaDB/MySQL中的DDL语句会自动提交前面所有的事务（包括显示开启的事务），而在SQL Server中DDL语句还是需要显式提交的，也就是说在SQL Server中DDL语句也是可以回滚的。

## 2.自动提交。(MySQL默认的提交方式)

不需要显式begin或者start transaction来显式开启事务，也不需要显式提交或回滚事务，每次执行DML和DDL语句都会在执行语句前自动开启一个事务，执行语句结束后自动提交或回滚事务。

## 3.隐式提交事务

隐式提交事务是指执行某些语句会自动提交事务，包括已经显式开启的事务。

会隐式提交事务的语句主要有：

(1).DDL语句(其中有truncate table)。

(2).隐式修改mysql数据库架构的操作:create user,drop user,grant,rename user,revoke,set password。

(3).管理语句：analyze table、cache index、check table、load index into cache、optimize table、repair table。

通过设置 auto\_commit 变量值为1或0来设置是否自动提交，为1表示自动提交，0表示关闭自动提交，即必须显式提交。但是不管设置为0还是1，显式开启的事务必须显式提交，而且隐式提交的事务不受任何人为控制。

atigmail ati, [04.01.21 01:10]

systems."——维基百科

在计算机领域，WAL（Write-ahead logging，预写式日志）是数据库系统提供原子性和持久化的一系列技术。

在使用WAL的系统中，所有的修改都先被写入到日志中，然后再被应用到系统状态中。通常包含redo和undo两部分信息。

为什么需要使用WAL，然后包含redo和undo信息呢？举个例子，如果一个系统直接将变更应用到系统状态中，那么在机器掉电重启之后系统需要知道操作是成功了，还是只有部分成功或者是失败了（为了恢复状态）。如果使用了WAL，那么在重启之后系统可以通过比较日

atigmail ati, [04.01.21 01:15]

引言

Write Ahead Logging，简称WAL，也被翻译成预写式日志，是数据库技术中实现事务日志(Transaction Journal)的一种标准方法，可以实现单机事务的原子性，同时可以提高数据库的写入效率。

思考如下场景，如何确保原子性：写操作修改数据库中a和b的值，二者是一个事务，需要把a和b的最新值持久化到磁盘，假如保存完a的值，系统宕机了，重新启动后，a的值已经写入，但b待写入的值已经丢失，如何发现事务没有完成呢？

atigmail ati, [04.01.21 01:16]

我们看WAL怎么解决宕机和恢复的问题：

写WAL前宕机了，重启后，数据处于事务未执行的状态。

写WAL时宕机了，重启后，可以检查到WAL数据不正确，回滚当事务前的状态。

写WAL后宕机了，重启后，把WAL中记录的操作，应用到数据库文件中，得到事务执行后的状态。

如此，保证了数据的恢复和事务的原子性

atigmail ati, [04.01.21 01:17]

上面提到的都是写操作，看一下使用WAL时的读操作。WAL中可能包含了未写入到数据库文件中的最新值，如果读最新值就需要从WAL中读取，如果WAL中未读到，从数据库读到的就是最新的数据。

atigmail ati, [04.01.21 01:18]

证事务一致性

并发读写，比如SQLite中，读写、读读都是可以并行的，比如读时需要找到WAL某个值最后写入的位置，就可以从该位置读数据，而写操作是在WAL文件后Append，二者并行。但写写不能并行，因为2次写操作都要向WAL文件Append数据，无法同时进行。

WAL文件中记录了数据的历史版本，因此可以读取历史版本的值，甚至把状态回滚到某个历史版本

atigmail ati, [04.01.21 01:19]

Rollback Journal 的原理是，在修改数据库文件中数据之前，先将修改所在分页中的数据备份在另外一个地方，然后才将修改写入到数据库文件中；如果事务失败，则将备份数据拷贝回来，撤销修改；如果事务成功，则删除备份数据提交修改。2019年10月13日

www.jianshu.com › ...

SQLite 数据库WAL 工作模式原理简介- 简书

atigmail ati, [04.01.21 01:20]

缺点

SQLite提到了WAL的几项缺点：

WAL需要VFS的支持。

所有使用数据库的进程必须在同一个机器上，以为WAL是单机的。

多读少写的场景WAL比rollback-journal类型要慢1%~2%

atigmail ati, [04.01.21 01:21]

的能力，也就是本文标题所讲的 crash-safe。即在 InnoDB 存储引擎中，事务提交过程中任何阶段，MySQL突然奔溃，重启后都能保证事务的完整性，已提交的数据不会丢失，未提交完整的数据会自动进行回滚。这个能力依赖的就是redo log和unod log两个日志。

atigmail ati, [04.01.21 01:24]

如果是主从模式下，binlog是必须的，因为从库的数据同步依赖的就是binlog；

如果是单机模式，并且不考虑数据库基于时间点的还原，binlog就不是必须，因为有redo log就可以保证crash-safe能力了；但如果万一需要回滚到某个时间点的状态，这时候就无能为力，所以建议binlog还是一直开启

atigmail ati, [04.01.21 09:08]

并且是最好的。当然，有一些头条新闻，例如普吉岛的Patong和菲律宾的世界三大最佳岛屿（长滩岛，宿雾和巴拉望岛），它们始终是制胜法宝。但是，如果您想与众不同，这里还有另外三个海滩，可为您带来更加独特的度假体