

Cedar 0.2 版本说明

修订历史

版本	修订日期	修订描述	作者	备注
Cedar 0.2	2016-07-01	Cedar 0.2 版本说明	朱涛	无

特征列表

1 功能特征

1.1 SNAPSHOT ISOLATION 隔离级别

Cedar 0.2 增加了事务级的Snapshot Isolation隔离级别，并允许事务读取自己修改但未提交的数据。Snapshot Isolation 在事务启动时确定数据快照，事务将始终读取该快照的数据。

注意：

- 隔离级别的使用需要配合显式事务；设置隔离级别时，需要保证隔离级别名称的正确拼写。由于设计上的原因，目前客户端通过指定REPEATABLE READ来使用SNAPSHOT ISOLATION。

1.2 表锁

Cedar 0.2 增加了对单张表进行排它访问的表锁。该机制弥补了行锁无法解决的幻读异常，并可以满足某些特殊应用的需求：对于读事务，使用表锁可以实现在表上事务执行的串行化，避免幻读问题；对于写事务，使用表锁可以在一定程度上避免长事务的饥饿。

注意：

- 为表添加表锁需要在事务执行过程中显式调用 `lock table table_name` 。如果此时没有事务对该表进行修改，则加锁成功，该事务持有该表的排它锁，其他事务无法再修改该表（但是依然可以读）；如果此时有事务正在对该表进行修改，则重试加锁操作直到该语句超时（Timeout）。
- 不使用表锁时，性能几乎与上一个版本相同，使用表锁会较大限制事务的并发量。

2 性能特征

2.1 基于布隆过滤器的连接

Cedar 0.2 针对大表连接，增加了一种基于布隆过滤器的连接算法(Bloom-Filter Join)。该算法使用布隆过滤器压缩左表的数据信息，对右表数据进行过滤，过滤后能够减少处理右表所需的网络，CPU和内存开销。

当数据和负载满足以下两点特征时，该算法有较高的性能。1、左表数据对右表有较好的选择率。这保证了能够右表数据被大量的过滤，从而较大的减少处理右表的资源开销。经验数据为当能够过滤掉右表500万行时，优化效果明显。2、左表基数较小。它保证了生成的布隆过滤器不会太大，确保布隆过滤器的维护代价不会太高。经验数据为当左表基数不超过100万时，布隆过滤器的维护不会影响性能。

注意：

- 不支持在不同类型的连接列上做等值连接。
- 不支持right、full out join算法。

2.2 日志同步优化

Cedar 0.2 基于已有的集群间选主、集群角色自动切换机制，优化了日志同步以及容灾恢复流程。集群日志同步性能提升了1.3倍；主备切换时，旧主UPS不需要重启；并支持在任意数量的集群间同步日志。

- 增加主备UPS同步日志提交点信息的机制
- 修改备UPS将日志写盘、回放日志、回复主UPS的执行顺序
- 修改主备UPS宕机重启后的恢复机制

注意：

- 现有架构下，为了保证读写数据的一致性，增量数据的读写操作均由主集群中的UPS负责；每进行一次集群切换，管理员应确保集群的整体状态正常；主备UPS宕机重启后首先都恢复到日志提交点，日志提交点信息由主UPS根据备UPS的日志回复情况更新，并借由日志同步给其他备UPS。