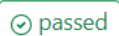





# DBTrain Lab 5 Report

刘泓尊 2018011446 计84

CI job ID #148576

Status	Job	Pipeline	Stage
 passed	#148576  master  a9ed0f4d	#73002 by 	test

## Steal/No-Force策略

为了支持Steal+No-Force策略，需要实现Write-Ahead Log(WAL), 所以我实现了新的日志页面(LogPage)，用于WAL的持久化，页面的一项包括事务ID、事务操作、修改前后的数据及其位置(PageSlotID)。在事务Begin或Commit时，向WAL写入Begin或Commit信息；事务进行修改操作的时候，先向WAL写入修改日志，日志持久化之后再数据落盘。

在Redo的时候，读取WAL内容，拿到已经Commit的事务对应的修改，然后重做这些操作；对于Undo操作，读取WAL的内容，拿到没有Commit的事务对应的修改，按时间逆序进行撤销操作，就可以保证正确的数据恢复。

Redo: 需要注意的是在我们之前的实现中并没有BufferPool, 也就是每次修改操作都同步到了持久存储，所以在本实验中为了简便，只是在Insert接口中数据落盘前进行了WAL的记录，实际上并不会出现事务操作完成（返回）但数据还没有落盘的情况。所以Redo操作看起来没有意义，甚至不经修改也是能通过测试的。但是为了实现的完整性，我还是实现了Redo操作，Redo时如果对应记录存在就不再实际插入了。

Undo: 拿到需要undo的操作序列之后，对于Insert的修改，只需要删除对应的record. 对于delete修改，需要重新insert原来的数据；对于update修改，则需要在WAL中记录修改前后的数据，用旧数据覆盖新数据。所有的操作需要按时间逆序恢复，保证正确性。

### 具体实现：

LogPage: 一系列Page组成链表，以应对Log数据过多的情况。用于实现WAL的持久化。页面的一项包括事务ID、事务操作、修改前后的数据及其位置(PageSlotID)。目前支持的Log操作包括: BEGIN, COMMIT, ABORT, INSERT, UPDATE, DELETE.

TransactionManager: 新增了一个system固有页面用于存放transaction相关的元信息，目前包括LogPage链表首节点的PageID, 还有当前已分配到的TxnID, 保证数据库正确恢复。

RecoveryManager: 用于完成Redo和Undo操作。它会读取TransactionManager的元信息，从WAL中得到日志，根据上面所述的方法进行Redo和Undo操作。

其中比较麻烦的点是LogPage的维护。Undo操作需要注意按时间逆序进行撤销。

## 测试结果

```
Start to test Lab5
[=====] Running 2 tests from 1 test suite.
[-----] Global test environment set-up.
[-----] 2 tests from RecoveryTests
[ RUN      ] RecoveryTests.RedoTest
Database Init.
Build Finish.
[      OK   ] RecoveryTests.RedoTest (8 ms)
[ RUN      ] RecoveryTests.UndoTest
Database Init.
Build Finish.
[      OK   ] RecoveryTests.UndoTest (6 ms)
[-----] 2 tests from RecoveryTests (14 ms total)
[-----] Global test environment tear-down
[=====] 2 tests from 1 test suite ran. (14 ms total)
[ PASSED   ] 2 tests.
Start to send result
Result has been sent
Passed lab5 test
```

## 课程感想

首先很感谢老师和助教团队如此精心地设计实验框架，在5次实验中始终保持了实验框架的简洁性和直观性，我在阅读框架代码的时候能感到通俗易懂，每次实验要求也极具灵活性。整体的课程难度和工作量我认为还是比较符合2学分的要求的，这门课用很清晰的架构和实验让我们熟悉了一个简单DBMS需要哪些部分、哪些功能以及如何实现。

同时，我也希望将来能够开放更多可选的实验，增加考查的多样性，避免不必要的重复。比如LAB1,LAB2,LAB5都涉及了新的页面的增加，也就都需要重复序列化和反序列化这部分代码，简单但是繁琐，所以或许可以通过精心设计避免重复性的内容，给同学们更多精力涉猎新的功能和方向。

我在这门课的巨大收获离不开老师的悉心讲解、助教的耐心答疑和实验框架的简洁完善，衷心祝愿这门课能给学弟学妹们带来更多的惊喜和收获！