# BinLog与RedoLog

1. Server层：查询缓存 连接器 分析器 优化器 执行器。
2. 引擎层：存储引擎
3. BinLog：由Server层产生，默认放在Data文件夹
4. RedoLog：innodb特有，由引擎层生成，保证持久化
5. Mysql引擎设计：先写内存，再写硬盘。
6. RedoLog写入策略：redo log buffer --(write)--> page cache --(fsync)--> 磁盘文件
7. RedoLog写入策略选择：

0: 每次提交事物只写到redo log buffer

1: 每次提交事物都持久化到磁盘（最安全）

2: 每次提交事物都写到文件系统的PageCache

1. RedoLog是循环写，BinLog是追加写。
2. BinLog记录Update、Insert、Create等操作，但不记录数据。

# SQL语句执行过程

1. 客户端发起查询->服务器查询缓存->缓存未命中，则解析器解析SQL、预处理器预处理、由优化器生成执行计划->根据执行计划调用储存引擎的API来执行查询->结果返回
   1. 查询缓存：查询缓存会跟踪查询中涉及的所有表，只要其中一个表发生变化则缓存失效。另外一个会导致缓存失效的因素是执行逻辑相同，但语句有空格之类的差异。如果查询中有不确定的 的数据，如NOW()或者CURRENT\_DATE则查询不会被缓存。
   2. 缓存设置：

query\_cache\_type：OFF、ON、DEMAND(命令指定缓存，才使用缓存)

query\_cache\_size：

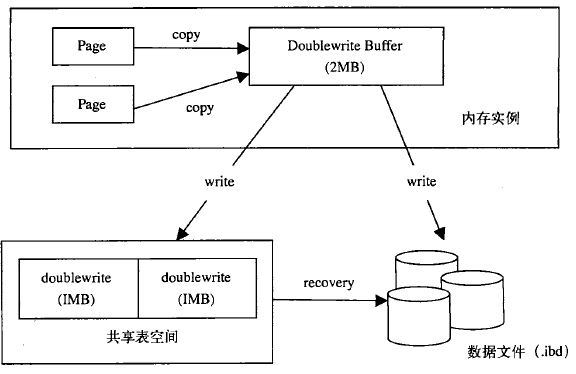
query\_cacha\_limit：不会缓存大于该值的数据

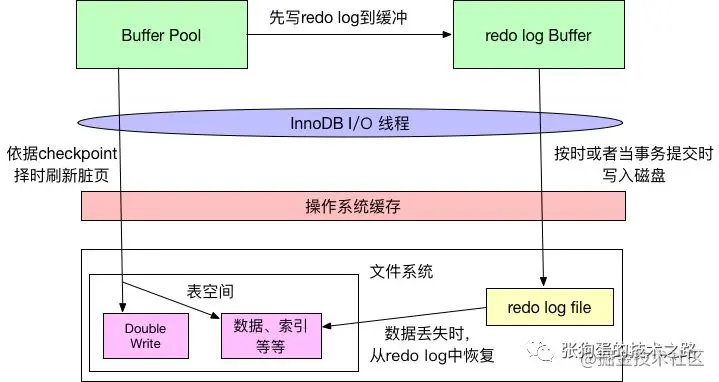
1. 返回客户端：如果查询结果可以被缓存，则会缓存执行结果

4.1 Where执行过程：用到的索引的查询条件为IndexKey，可以使用索引过滤的查询条件为Index Filter，而必须回表过滤的查询条件的称为Table Filter。

4.2 SQL执行过程是依次将IndexKey范围内的索引记录读取，之后mysql服务层按照Table Filter过滤

# INNODB的内存结构和特性

1. 脏页：数据更新发生在内存的页中，此时磁盘的页和内存中的页不一致，内存的页为脏页。而内存脏页刷回磁盘后便变为了干净页。
2. 缓存池：磁盘数据库包括mysql在内，通常会使用缓存池来提升数据库性能。
   1. 缓存池读取：读取操作会把读取的页存入缓存池。读取的页命中缓存池则从缓冲池中读取，否则从磁盘中读取。
   2. 缓存池写入：数据库修改时首先修改缓冲池，随后以一定频率刷回磁盘，该频率由CheckPoint机制决定。
   3. 缓存池数据类型：索引页、数据页、undo页、插入缓存、自适应哈希索引、InnoDB存储的锁信息、数据字典信息。
3. Page：Page是Innodb的最基本结构，也是Innodb磁盘管理的最小的单位。数据库相关的所有内容都存在Page中。
   1. 数据页和索引页是Page最重要的两种类型。
4. 插入缓存(Insert Buffer)：当数据插入时，不会直接更新非聚簇且非唯一。如果缓存池中存在该索引页，则直接插入。不存在，则先放到InsertBuffer，这样可以将这些插入操作合并到一个插入操作中，以提示插入性能。
   1. 两次写(Double Write)：Mysql并不是直接将缓冲池的脏页刷回数据库，而是先拷贝至DoublewriteBuffer(两次写缓存区)，再从DoublewriteBuffer写入共享表空间，最后再把数据从DoublewriteBuffer写入磁盘的数据文件中。
   2. 两次写解决的问题：当DoublewriteBuffer写入数据文件的过程中，mysql宕机，则可以把共享表空间中的副本复制到表空间中再应用重做日志来恢复。
   3. 是否关闭两次写：严重不建议，如果数据表空间的文件系统提供了写失效防范机制，则可以关闭两次写。
5. 
   1. 重做日志：InnoDB先写重做日志，再选择合适的时间将脏页写入数据库。
   2. 重做日志缓存(Redo Log Buffer)：重做日志信息首先写进重做日志缓存中，再按照一定的频率刷进重做日志文件。
   3. 重做日志刷入时机：除了刷新机制外，默认策略是每次事物提交都会将缓存刷入重做日志文件中。

5.4 重做日志刷新策略：1->每次事务提交都刷新(默认策略)，2->事务提交只写入文件系统的缓存，0->事务提交不影响重做日志

6. 自适应哈希索引：InnoDB会根据访问的频率和模式为热点页建立哈希索引。InnoDB会监控对表上各个索引页的查询，如果建立哈希索引可以更快，则建立哈希索引。

7. 锁信息

# INNODB的磁盘文件及落盘机制

1. 磁盘文件组成：系统表空间、用户表空间、redo日志文件及归档文件
2. 系统表空间：数据字典、double write buffer、change buffer、undo logs、表数据、索引数据

# INNODB锁与状态

1. 共享锁不影响共享锁获取，但共享锁全部释放之前不允许获取排它锁。
2. 锁信息获取：SELECT \* FROM information\_schema.INNODB\_LOCKS
3. 事物状态获取：SELECT \* FROM information\_schema.INNODB\_TRX