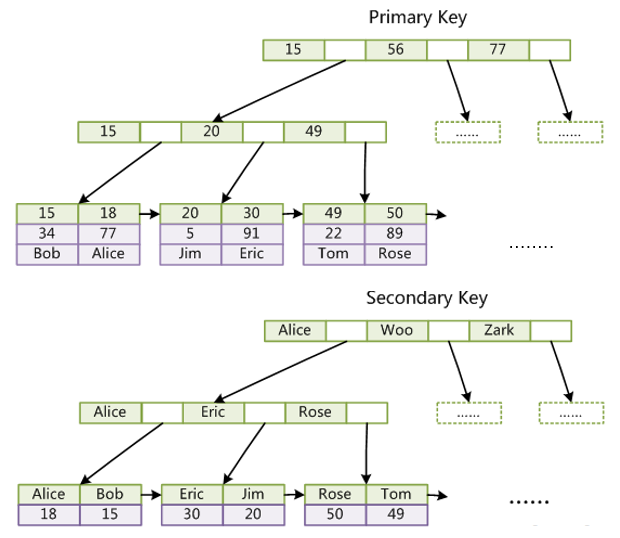
**1.InnoDB存储引擎**

**支持事务**，是面向在线事务处理的应用、特点是**行锁设计**、支持外键。默认的**select**方式**不加锁**，而**增删改会加排它锁**。InnoDB存储引擎是MySQL**默认的存储引擎**。

InnoDB通过使用**多版本并发控制MVCC**来获得高并发性，并且实现了SQL标准的四种隔离级别，默认为**REPEATABLE\_READ**级别，同时使用**next key locking**的策略来避免幻读。**对于表中数据的存储**，InnoDB采用**聚集索引**的方式，每张表的存储都是按**主键顺序**进行存放的。如果没有显式地在表定义时指定主键，找unique字段作为主键，否则就会生成一个**6字节的rowId**作为主键，即默认聚集索引，索引页与数据页是连在一起的，叶子节点直接包含表的行数据，即数据页，非聚集索引中，叶子节点按照非聚集索引列排序，包含的是**非聚集索引列**，以及指向聚集索引的**主键值**。

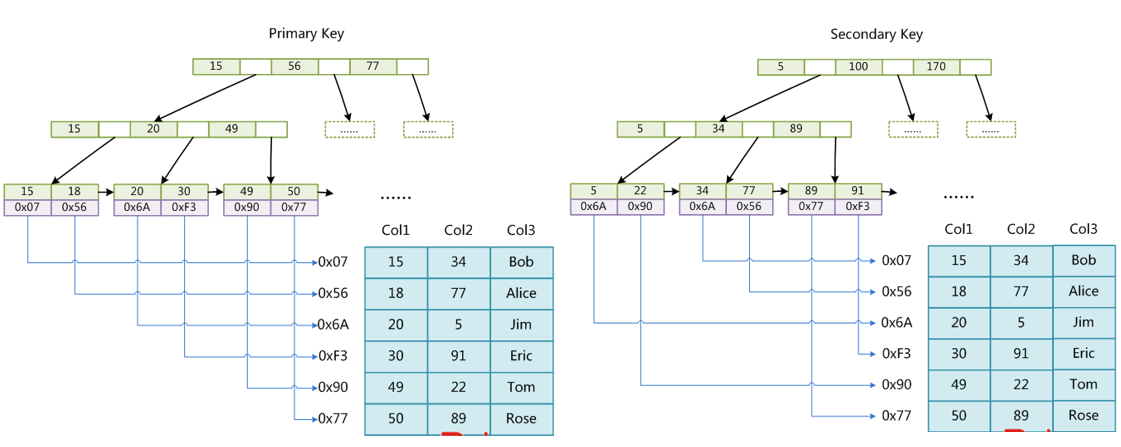


索引采用b+树索引，聚集索引中叶子节点

**2.MyISAM存储引擎**

**不支持事务**，特点是**表锁设计**，支持**全文索引**，，MyISAM存储引擎它的缓冲池只**缓存索引文件**，**不缓存数据文件**。

表数据由**MYD和MYI**组成，MYD用来存储**数据文件**，MYI用来存储**索引文件**，**MySQL数据库**只负责**缓存索引文件**，**数据文件的缓存**交给**操作系统**完成。



主键索引和辅助索引基本没什么大的区别，只不过主键索引按照主键进行排序，辅助索引按照指定索引列排序。但是一个重大不同是，主键索引和辅助索引，虽然用的也是b+树，但是叶子节点不直接保存表数据或者主键值，保存的是**数据记录的地址**

**3.MyISAM与InnoDB引擎的区别**

**3.1锁差异**

**MyISAM只支持表锁**，**不会产生死锁**，**粒度大，冲突概率高，并发度低**。用户在select的时候自动加**表共享读锁**，在增删改的时候加**表独占写锁**，自动加锁时，会对sql语句**所有相关表进行加锁**；也可以使用lock table命令加锁，但是使用lock table，需要对**sql语句相关表进行加锁**，即使同一张表，只要被定义使用多次别名，就得对同一张表的**不同别名进行加锁**。**并发插入**，Concurrent\_insert控制并发插入，0表示不允许，1表示表**中间的数据没有删除**的情况下，可以在**表尾插入**；2表示允许在表尾插入。

Session1使用lock table 对a表**进行加锁**，并**查询**操作，session2不能对a表进行**更新删除**，但是可以插入数据t，插入结束，如果session2要更新数据，那么就会**阻塞**，直到session1释放锁才可更新。

**InnoDB支持事务和行锁，会出现死锁，粒度小，冲突概率低，并发度高。**使用select的时候，默认不加锁，使用增删改的时候默认排它锁。

**插入讲解事务**：

**原子性（atomicity）**：事务包含的操作要么全部成功，要么全部失败

例如转账:a-100,b+100,这两个操作要么都发生，要么都不发生

**一致性（consistency）**：数据库从一个一致性状态变换到另一个一致性状态，就是事务执行前和执行后都必须处于一致性状态。

**隔离性（isolation）**：当多个用户并发访问数据库时，事务之间相互隔离不能干扰

隔离级别稍后介绍。

**持久性（durablity）**：一旦事务被提交，数据库的数据改变就是永久性的。

讲隔离级别之前先得理解四个并发事务的问题：

**丢失更新**：①t1查询一行a，存入内存；②t2查询一行a，存入内存；③t1先修改a数据，④t2之后修改a数据；⑤t1提交⑥t2提交；此时**t2的修改覆盖了t1的修改**。

解决方法：将这几种事务操作**串行化**。即在①加入**排它锁**，②也加入**排它锁**，那么②就必须等待①③结束之后才能**获得锁并取得修改以后的数据**，这样不会产生丢失更新。

**脏读**：事务读取到了**未提交的数据**，t1开始事务，t2开始事务 ，t2查询t表中的数据此时只有一条a，t1向t表插入一条数据b此时事务未提交，然后t2再次查询t表中的数据，查询到了两条数据a和b，此时发生了脏读。

**不可重复读**：a事务多次**读取同一数据**，b事务也**访问同一数据**，并进行**修改提交**，a事务读取的**数据发生变化**。

**幻读：**a事务读取到了b事务提交的**新增数据**。例如统计总金额，a事务开始，统计总金额为1000，b事务开始，增加存款账户，存款为100，提交事务，a事务再次统计存款此时统计金额为1100；

注：不可重复读和幻读的区别：前者是读到了**其他事务**已经**提交**的**更改数据**，后者是指已经读到**其他事务**已经**提交**的**新增数据**

**隔离级别**：

**读未提交（READ\_UNCOMMITED）**:允许**脏读、不可重复读、幻读**，不允许**丢失更新**，因为如果一个事务已经开始**写数据**，另一个事务**不允许同时进行写操作**，可以通过**排它锁**实现。

**读已提交（READ\_COMMITTED）**：允许**不可重复读、幻读**，不允许**脏读**。读取数据的事务允许其他事务继续访问该行数据，但是**未提交的写事务**会**禁止其他事务读取该行数据**

**可重复读（REPEATABLE READ）：**禁止**不可重复读取和脏读取**，可能出现**幻读**。只对该**范围内存在的数据进行隔离限制**，不能修改数据内容，**不对未存在但还在查询条件范围**中的数据进行隔离限制。例如，查找a>100的记录，此时数据库中至于101,102，该级别能保证101,102不能被其他事务修改，但是其他事务可以插入103数据，此时隔离级别就会产生幻读，因为不存在的数据没有隔离级别限制。

**可串行（SERIALIZABLE）：**不仅锁定**第一次查询的数据行**，而且**锁定未来满足条件的数据行**，不会引起幻读。

InnoDB默认的隔离级别为REPEATABLE\_READ，默认的锁机制为next\_key\_lock，通过两者结合已经可以避免幻读，达到事务最高隔离级别。

**3.2数据库文件差异**

MyISAM：**frm用于存储表定义**，MYD用于存放表**数据**，MYI用于存放表**索引**

InnoDB：**frm存储表定义**，存储方式为**共享表空间**和**多表空间**，共享表空间，所有表的**数据文件和索引文件都保存在表空间里**；如果使用多表空间，那么**每个表都有一个表空间文**件，用于存储每个表的**数据页和索引页**。以**ibd**为扩展名

**3.3索引差异**

①MyISAM引擎的**自动增长列必须是索引列**，如果是组合索引，自动增长**可以不是第一列**，InnoDB的引擎的**自动增长列必须是索引列**，如果是组合索引，也**必须是组合索引第一列**。

②关于主键

MyISAM允许**没有任何索引和主键的表存在**，其索引都是保存**行记录的地址**；InnoDB如果**没有主键或者非空唯一列**，就会自动生成**6字节rowID**主键，数据是聚集索引的一部分，即叶**子节点就是数据页**，辅助索引保存的是**key值**和指向聚集索引的**主键**

③关于count()函数

MyISAM保存了表的**总行数**，如果count（\*）会**直接取出该值**，**效率高**；InnoDB没有保存表的总行数，使用count（\*）会**遍历整表**，当时加了where之后MyISAM与InnoDB处理方式一样。

④索引保存位置

MyISAM的索引以表名+.MYI文件保存;InnoDB的索引和数据一起保存在表空间。