

## MySQL存储引擎

在MySQL中，不需要在整个服务器中使用同一种存储引擎，针对具体的要求，可以对每一个表使用不同的存储引擎。

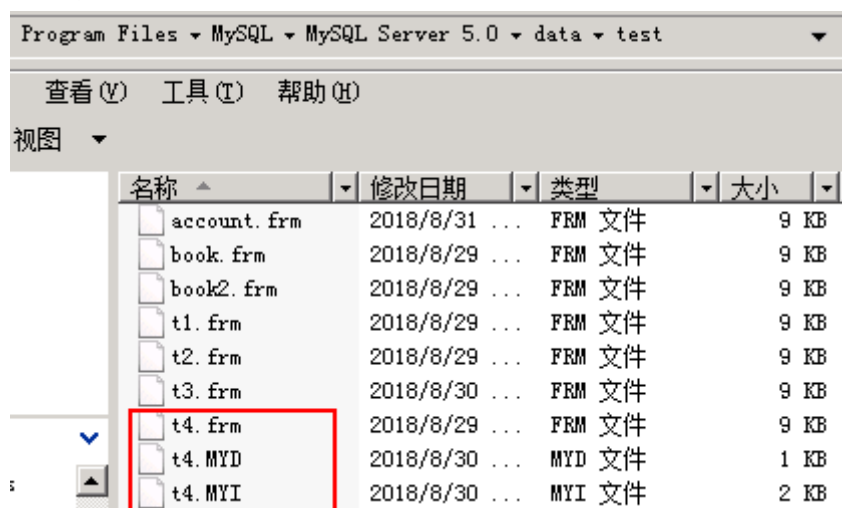
### 一、MyISAM

1. 不支持事务，也不支持外键，但访问速度快，对事务完整性没有要求或者以SELECT、INSERT为主的应用基本都可以使用这个引擎来创建表。
2. 使用MyISAM引擎创建数据库表，将产生3个文件。其中文件名和表名都相同，扩展名分别为：

.frm（存储表定义）、.MYD（MYData，存储数据）、.MYI（MYIndex，存储索引），如下图所示：表t4一共有三个文件

node：数据文件和索引文件放在不同目录，以平均分配IO，获取更快的速度。

3. MyISAM引擎使用B+Tree作为索引结构，叶节点的data域存放的是数据记录的地址。



### 二、InnoDB（MySQL默认的存储引擎）

InnoDB是一个健壮的事务型存储引擎，支持行级锁定和外键约束

在以下场合下，使用InnoDB是最理想的选择：

1. 更新密集的表。InnoDB存储引擎特别适合处理多重并发的更新请求。
2. 事务。InnoDB存储引擎是支持事务的标准MySQL存储引擎。
3. 自动灾难恢复。与其它存储引擎不同，InnoDB表能够自动从灾难中恢复。
4. 外键约束。MySQL支持外键的存储引擎只有InnoDB。
5. 支持自动增加列AUTO\_INCREMENT属性。

一般来说，如果需要事务支持，并且有较高的并发读取频率，InnoDB是不错的选择。

InnoDB 和 MyISAM之间的区别：

- 1>. InnoDB支持事物，而MyISAM不支持事物

- 2>. InnoDB支持外键，而MyISAM不支持
- 3>. InnoDB支持行级锁，而MyISAM支持表级锁
- 4>. InnoDB支持MVCC，而MyISAM不支持
- 5>. InnoDB不支持全文索引，而MyISAM支持。
- 6>. InnoDB和MyISAM索引区别

虽然InnoDB也使用B+Tree作为索引结构，但具体实现方式却与MyISAM截然不同。

- 1). InnoDB的数据文件本身就是索引文件。
- 2). InnoDB的辅助索引data域存储相应记录主键的值而不是地址

### 三、MEMORY

Memory存储引擎的出发点是速度（MEMORY存储引擎将表中的数据存储在系统内存中）

一般在以下几种情况下使用Memory存储引擎：

1. 目标数据较小，而且被非常频繁地访问。在内存中存放数据，所以会造成内存的使用，可以通过参数max\_heap\_table\_size控制Memory表的大小，设置此参数，就可以限制Memory表的最大大小。
2. 如果数据是临时的，而且要求必须立即可用，那么就可以存放在内存表中。
3. 存储在Memory表中的数据如果突然丢失，不会对应用服务产生实质的负面影响。

Memory同时支持散列索引和B树索引。

存储在内存中带来的问题

1. 当mysqld守护进程崩溃时，所有的Memory数据都会丢失。
2. 因为存储在内存中，所以要求存储在Memory数据表里的数据使用的是长度不变的格式，这意味着不能使用BLOB和TEXT这样的长度可变的数据类型，VARCHAR是一种长度可变的类型，但因为它在MySQL内部当做长度固定不变的CHAR类型，所以可以使用。

### 四、ARCHIVE

Archive是归档的意思，在归档之后很多的高级功能就不再支持了，仅仅支持最基本的插入和查询两种功能。

在MySQL 5.5版以前，Archive是不支持索引，但是在MySQL 5.5以后的版本中就开始支持索引了。Archive拥有很好的压缩机制，它使用zlib压缩库，在记录被请求时会实时压缩，所以它经常被用来当做仓库使用。

### 五、MERGE（MySQL5.0中名为：MRG\_MYISAM）

MERGE存储引擎是一组MyISAM表的组合

