

MongoDB GridFS

GridFS 用于存储和恢复那些超过16M (BSON文件限制) 的文件(如：图片、音频、视频等)。

GridFS 也是文件存储的一种方式，但是它是存储在MongoDB的集合中。

GridFS 可以更好的存储大于16M的文件。

GridFS 会将大文件对象分割成多个小的chunk(文件片段),一般为256k/个,每个chunk将作为MongoDB的一个文档(document)被存储在chunks集合中。

GridFS 用两个集合来存储一个文件：fs.files与fs.chunks。

每个文件的实际内容被存在chunks(二进制数据)中,和文件有关的meta数据(filename,content_type,还有用户自定义的属性)将会被存在files集合中。

以下是简单的 fs.files 集合文档：

```
{
  "filename": "test.txt",
  "chunkSize": NumberInt(261120),
  "uploadDate": ISODate("2014-04-13T11:32:33.557Z"),
  "md5": "7b762939321e146569b07f72c62cca4f",
  "length": NumberInt(646)
}
```

以下是简单的 fs.chunks 集合文档：

```
{
  "files_id": ObjectId("534a75d19f54bfec8a2fe44b"),
  "n": NumberInt(0),
  "data": "Mongo Binary Data"
}
```

GridFS 添加文件

现在我们使用 GridFS 的 put 命令来存储 mp3 文件。调用 MongoDB 安装目录下bin的 mongofiles.exe工具。

打开命令提示符，进入到MongoDB的安装目录的bin目录中，找到mongofiles.exe，并输入下面的代码：

```
>mongofiles.exe -d gridfs put song.mp3
```

GridFS 是存储文件的数据名称。如果不存在该数据库，MongoDB会自动创建。Song.mp3 是音频文件名。

使用以下命令来查看数据库中文件的文档：

```
>db.fs.files.find()
```

以上命令执行后返回以下文档数据：

```
{
  _id: ObjectId('534a811bf8b4aa4d33fdf94d'),
  filename: "song.mp3",
  chunkSize: 261120,
  uploadDate: new Date(1397391643474), md5: "e4f53379c909f7bed2e9d631e15c1c41",
  length: 10401959
}
```

我们可以看到 fs.chunks 集合中所有的区块，以下我们得到了文件的 _id 值，我们可以根据这个 _id 获取区块(chunk)的数据：

```
>db.fs.chunks.find({files_id:ObjectId('534a811bf8b4aa4d33fdf94d')})
```

以上实例中，查询返回了 40 个文档的数据，意味着mp3文件被存储在40个区块中。

[← MongoDB 管理工具: Rockmongo](#)

[MongoDB 固定集合 \(Capped Collections \) →](#)

[✎ 点我分享笔记](#)