**MongoDB的存储结构及对空间使用率的影响**

使用MongoDB一段时间的同学肯定会发现，MongoDB往往会占用比实际数据大小多不少空间的问题。如果利用db.stats()命令去 查看，会发现MongoDB会报告几种不同的空间大小信息，如dataSize, storageSize以及fileSize。这些大小到底指的是什么意思呢？让我们来通过了解MongoDB的存储机制来解析这几个数值的含义。

**数据库文件类型**

MongoDB的数据库文件主要有3种：

* journal 日志文件
* namespace 表名文件
* data 数据及索引文件

**日志文件**

跟一些传统数据库不同，MongoDB的日志文件只是用来在系统出现宕机时候恢复尚未来得及同步到硬盘的内存数据。日志文件会存放在一个分开的目录 下面。启动时候MongoDB会自动预先创建3个每个为1G的日志文件（初始为空）。除非你真的有持续海量数据并发写入，一般来说3个G已经足够。

**命名文件 dbname.ns**

这个文件用来存储整个数据库的集合以及索引的名字。这个文件不大，默认16M，可以存储24000个集合或者索引名以及那些集合和索引在数据文件中 得具体位置。通过这个文件MongoDB可以知道从哪里去开始寻找或插入集合的数据或者索引数据。这个值可以通过参数调整至2G。

**数据文件 dbname.0, dbname.1,… dbname.n**

MongoDB的数据以及索引都存放在一个或者多个MongoDB数据文件里。第一个数据文件会以“数据库名.0”命名，如 my-db.0。这个文件默认大小是64M，在接近用完这个64M之前，MongoDB 会提前生成下一个数据文件如my-db.1。数据文件的大小会2倍递增。第二个数据文件的大小为128M，第三个为256M。一直到了2G以后就会停止， 一直按这个2G这个大小增加新的文件。

当然MongoDB还会生成一些临时文件如 \_tmp 和 mongod.lock等， 不过他们跟我们的讨论都没有太大相关性。

**数据文件结构**

**Extent**

在每一个数据文件内，MongoDB把所存储的BSON文档的数据和B树索引组织到逻辑容器“Extent”里面。如下图所示（my-db.1和my-db.2 是数据库的两个数据文件）：

一个文件可以有多个Extent

* 每一个Extent只会包含一个集合的数据或者索引
* 同一个集合的数据或索引可以分布在多个Extent内。这几个Extent也可以分步于多个文件内
* 同一个Extent不会又有数据又有索引

Extend之间使用Double-Link-List实现关联。

**Record 记录**

在每个Extent里面存放有多个”Record“, 每一个记录里包含一个记录头以及MongoDB的BSON文档，以及一些额外的padding空间。Padding是MongoDB在插入记录时额外分配 一些未用空间，这样将来文档变大的时候不至于需要把文档迁移到别处。 记录头以整个记录的大小开始，包括该记录自己的位置以及前一个记录和后一个记录的位置。可以想象成一个Double Linked List。

**数据库大小参数**

在之前的基础上，我们可以来理解一下db.stats()里面关于空间大小参数的含义。

**dataSize**

dataSize是最接近真实数据大小的一个参数。你可以用来检查你的数据有多少。这个大小包括了数据库（或者集合）的每条记录的总和。注意每条记录除了BSON文档外还有header及padding这些额外开销。所以实际大小会比真正数据所占空间会稍大。

当删除文档的时候，这个参数会相应变小因为它是所有文档数的大小总和。如果你的文档没有删除，只是文档内部的字段被删除或缩小，则不会对dataSize 有影响。原因就是因为文档所在记录还在，并且整条记录所占空间并无改动，只不过记录内的未用空间变多了而已。

**storageSize**

这个参数等于数据库或者某个集合所有用到的Data Extents的总和。注意这个数字会大于dataSize因为Extent里面会有一些删除文档之后留下来的碎片（deleted）。及时你的 storageSize大出dataSize很多，这个也不一定就是很糟糕的情况。 如果有新插入的文档小于或等于碎片的大小，MongoDB会重新利用这个碎片来存储新的文档。不过在这之前这些碎片将一直会被保留在那里占用空间。由于这 个原因，你删除文档的时候这个参数不会变小。

碎片问题会因为运行的时间变长而变得严重。你可以通过 compact 命令来进行碎片清理或者通过新架一台从机复制所有数据，然后变成主节点的方式来解决这些碎片。

**fileSize**

这个参数只在数据库上有效，指的是实际文件系统中用到的文件的大小。它包括所有的数据Extents的总和，索引Extent的总和，以及一些未被 分配的空间。之前提到MongoDB会对数据库文件创建时候进行预分配，例如最小就是64M，哪怕你只有几百个KB的数据。所以这个参数可能会比实际的数 据大小会大不少。 这些额外未用空间是用来保证MongoDB可以在新的数据写入时候快速的分配新的Extent，避免引起磁盘空间分配引起的延迟。

值得注意的是，当你删除文档，或甚至集合和索引，这个参数不会变小。换句话说，数据库所使用的硬盘空间只会上升（或者不变），而不会因为删除数据而变小。当然需要知道的是这并不就意味着浪费，只是说有很多预留空间而已。