

# MVDBS:Knotenspeicherung im MongoDB

Roland Hediger

April 14, 2014

## 1 Storage Model

MongoDB speichert Daten als Dokumente in einer binären Darstellung (Binary JSON) - BSON. BSON erweitert JSON, damit es mehrere Typen einschliesst wie int, long, und float. BSON Dokumente beinhalten ein oder mehrere Felder und jedes Feld beinhaltet den Wert von einem spezifischen Datentyp: Arrays, Binärdaten, und Unterdokumenten. Dokumente haben eine ähnliche Struktur und sind in Sammlungen oder Collections organisiert. Es ist hilfreich, die Collections als Tabellen-ähnliche Strukturen vorzustellen. Dokumenten sind dann nach dieser Vorstellung wie Zeilen und die Felder ähnlich wie Spalten.

### 1.1 BSON Spezifikation

MongoDB Dokumenten haben folgenden Struktur:

```
{
  field1: value1,
  field2: value2,
  field3: value3,
  ...
  fieldN: valueN
}
```

BSON Dokumente sind aber limitiert in der Grösse bis zum 16MB.

## 2 Datenstruktur

Jede MongoDB Knoten hat eine Namespace Datei, Journal Datei, und Dateien für Daten. Dateien mit Daten speichern BSON Dokumente, Indexes, und MongoDB Metadata in speziellen Strukturen, die Extents heissen. Jede Daten-Datei ist aus mehreren Extents zusammengestellt.

### 2.1 Beziehung zwischen Daten und Extents

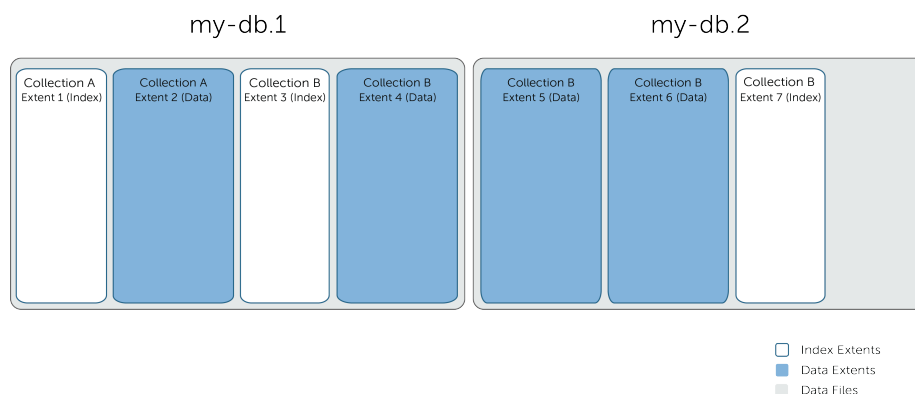


Figure 1: <http://blog.mongolab.com/2014/01/how-big-is-your-mongodb/>

- Daten und indexen (indices) haben beide separaten Mengen von Extents  
Kein Extent hat Informationen über mehr als eine Collection.
- Data und Indexes gehen über mehrere Extents hinaus.
- MongoDB versucht immer zuerst die freien Platz innerhalb existierenden Extents auszunutzen bevor eine neue erzeugt wird.

## 2.2 Speicherung Metriken

**dataSize** Innerhalb Collections sind Dokumenten mit Padding alloziert, das heisst der Behälter ist etwas grösser als die tatsächlich abgespeicherte Daten vom Dokument. Die grössse des Dokuments wird immer inklusiv Padding gemessen.

**fileSize** Gesamtgrösse von alle Extents. Extents sind wie Dokumente mit Padding alloziert. Logischerweise ist dieses Metrik immer grösser als dataSize.

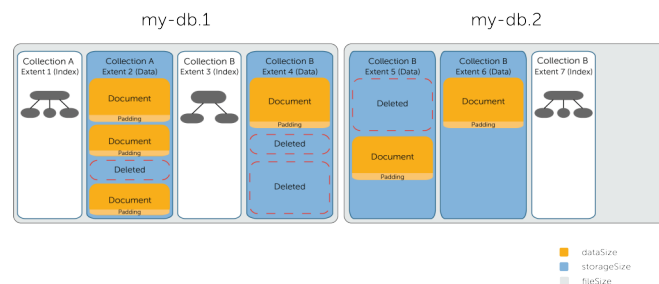


Figure 2: <http://blog.mongolab.com/2014/01/how-big-is-your-mongodb/>

## 3 Grössere Daten

Instead of storing a file in a single document, GridFS divides a file into parts, or chunks, [1] and stores each of those chunks as a separate document. By default GridFS limits chunk size to 255k. GridFS uses two collections to store files. One collection stores the file chunks, and the other stores file metadata.

When you query a GridFS store for a file, the driver or client will reassemble the chunks as needed. You can perform range queries on files stored through GridFS. You also can access information from arbitrary sections of files, which allows you to “skip” into the middle of a video or audio file.

GridFS stores files in two collections:

chunks stores the binary chunks. For details, see The chunks Collection. files stores the file’s metadata. For details, see The files Collection. GridFS places the collections in a common bucket by prefixing each with the bucket name. By default, GridFS uses two collections with names prefixed by fs bucket:

fs.files

fs.chunks

You can choose a different bucket name than fs, and create multiple buckets in a single database.

Each document in the chunks collection represents a distinct chunk of a file as represented in the GridFS store. Each chunk is identified by its unique ObjectId stored in its `_id` field.

GridFS uses a unique, compound index on the chunks collection for the `files_id` and `n` fields. The `files_id` field contains the `_id` of the chunk’s “parent” document. The `n` field contains the sequence number of the chunk. GridFS numbers all chunks, starting with 0. For descriptions of the documents and fields in the chunks collection,

## References

- [1] A MongoDB Whitepaper  
MongoDB 2.6
- [2] How big is your MongoDB  
<http://blog.mongolab.com/2014/01/how-big-is-your-mongodb/>
- [3] Getting Started with MongoDB + Gridfs  
<http://learnmongo.com/posts/getting-started-with-mongodb-gridfs/>
- [4] <http://docs.mongodb.org/manual/core/gridfs/>