

Методы сбора, хранения, обработки и анализа данных

Лекция 18

СУБД MongoDB – управление

Чек-лист устройства СУБД

- архитектура
- носители данных
- модель данных SQL/noSQL
- стандарты
- типы данных
- ограничения целостности
- объекты базы данных
- программные объекты

Архитектура MongoDB

- Одиночный сервер
- Набор реплик
- Шардинг

Репликация MongoDB

- Репликация – способ хранения идентичных копий данных на нескольких серверах
- Набор реплик – группа серверов с одним первичным узлом и несколькими вторичными узлами
 - Первичный узел – основной узел записи
 - Вторичный узел – хранит копии данных

Шардинг MongoDB

- Шардинг – автоматическая сегментация по наборам реплик
- Диапазон – определен для всех документов по сегментному ключу (*shard key*)
- Распределение нагрузки происходит равномерно

Носители данных MongoDB

- Каждая коллекция – отдельный файл
- Каждый индекс – отдельный файл
- Используются подсистемы записи
- WiredTiger (ориентировано на запись)
- MMAP (ориентировано на чтение)

WiredTiger

- NoSQL
- Open Source
- Разработана и поддерживается MongoDB
- Поддерживает построчное хранение
- Поддерживает столбцовое хранение
- Поддерживает журнально-структурированные деревья слияния (LSM – Log-structured merge-tree)
- Поддерживает транзакции

LSM – Log-structured merge-tree

LSM – Log-structured merge-tree

- Запись добавляется в Memtable
- Когда размер Memtable превышает порог – его записывают на диск как SSTable
- SSTable отсортировано по ключам
- Периодически SSTable сливаются (merge)
- Для поиска данных сначала ищется ключ в memtable, затем в последнем SSTable, затем в предыдущем SSTable и т. д.

MMAP

- Memory Mapped Storage Engine
- Загружает файлы в оперативную память
- Выделяет память блоками размера степени 2 (32, 64, 128, 256, ... 2 МБ)
- Блокировка устанавливается на уровне коллекции
- Может вызвать фрагментацию
- Можно использовать стратегию предварительного распределения

Стандарты MongoDB

- Хранение документов – в формате BSON
- BSON – двоичный JSON
- Есть спецификация 1.1

Типы данных MongoDB

- String – строковый тип данных, UTF-8
- Array – массив
- Binary data – данные в двоичном формате
- Boolean – логические значения, TRUE/FALSE
- Date – дата
- JavaScript – код JavaScript
- Object – объект, который содержит набор свойств
- ObjectId – идентификатор документа

Типы данных MongoDB

- Integer – целочисленные значения, 32 бита
- Long – целочисленные значения, 64 бита
- Double – числа с плавающей точкой
- Decimal128 – десятичные числа, 128 бит
- Regular expression - регулярные выражения

Ограничения целостности MongoDB

- Ограничения целостности задаются через Validation

all_subjects.locations

The screenshot shows the MongoDB Compass interface for the 'all_subjects.locations' collection. The 'Documents' tab is selected, and a filter is applied: { field: 'value' }. The interface includes buttons for 'ADD DATA', 'VIEW', and icons for list, JSON, and grid views. Three documents are displayed:

```
_id: ObjectId('67d94036c17e3f603c59c4cc')
name: "Minsk"
> loc: Object
```

```
_id: ObjectId('67d94040c17e3f603c59c4cd')
name: "Svisloch"
> loc: Object
```

```
_id: ObjectId('67d94090c17e3f603c59c4ce')
name: "Belarus"
> loc: Object
```

Ограничения целостности MongoDb

all_subjects.locations

Documents

Aggregations

Schema

Explain Plan

Indexes

Validation

Validation Action ⓘ

ERROR ▾

Validation Level ⓘ

STRICT ▾

```
1 {  
2   'loc.type': {}  
3   $in: [  
4     'Point',  
5     'LineString',  
6     'Polygon'  
7   ]  
8 }  
9 }
```

Ограничения целостности MongoDb

Файл Правка Формат Вид Справка

```
[{
  "name": "Minsk_2",
  "loc": {
    "type": "Pointy",
    "coordinates": [
      150,
      150
    ]
  }
}, {
  "name": "Svisloch_2",
  "loc": {
    "type": "LineString",
    "coordinates": [
      [
        150,
        10
      ],
      [
        120,
        20
      ],
      [
        160,
        120
      ]
    ]
  }
}]
```

Import To Collection all_subjects.locations

Select File

1.json

Select Input File Type

JSON

CSV

Options

☐ Stop on errors

Import completed with following errors:

2 / 3

Document failed validation

DONE

Ограничения целостности MongoDb

all_subjects.locations

Documents

Aggregations

Schema

Explain Plan

Indexes

Valida

FILTER { field: 'value' }

ADD DATA



VIEW



```
_id: ObjectId('67d94036c17e3f603c59c4cc')  
name: "Minsk"  
> loc: Object
```

```
_id: ObjectId('67d94040c17e3f603c59c4cd')  
name: "Svisloch"  
> loc: Object
```

```
_id: ObjectId('67d94090c17e3f603c59c4ce')  
name: "Belarus"  
> loc: Object
```

```
_id: ObjectId('67fd09f06148cb09f3be55a4')  
name: "Svisloch_2"  
> loc: Object
```

```
_id: ObjectId('67fd09f06148cb09f3be55a5')  
name: "Belarus_2"  
> loc: Object
```

Объекты базы данных MongoDB

- Коллекции
- Документы
- Поля

Программные объекты MongoDB

- Нет, но можно писать код на JavaScript

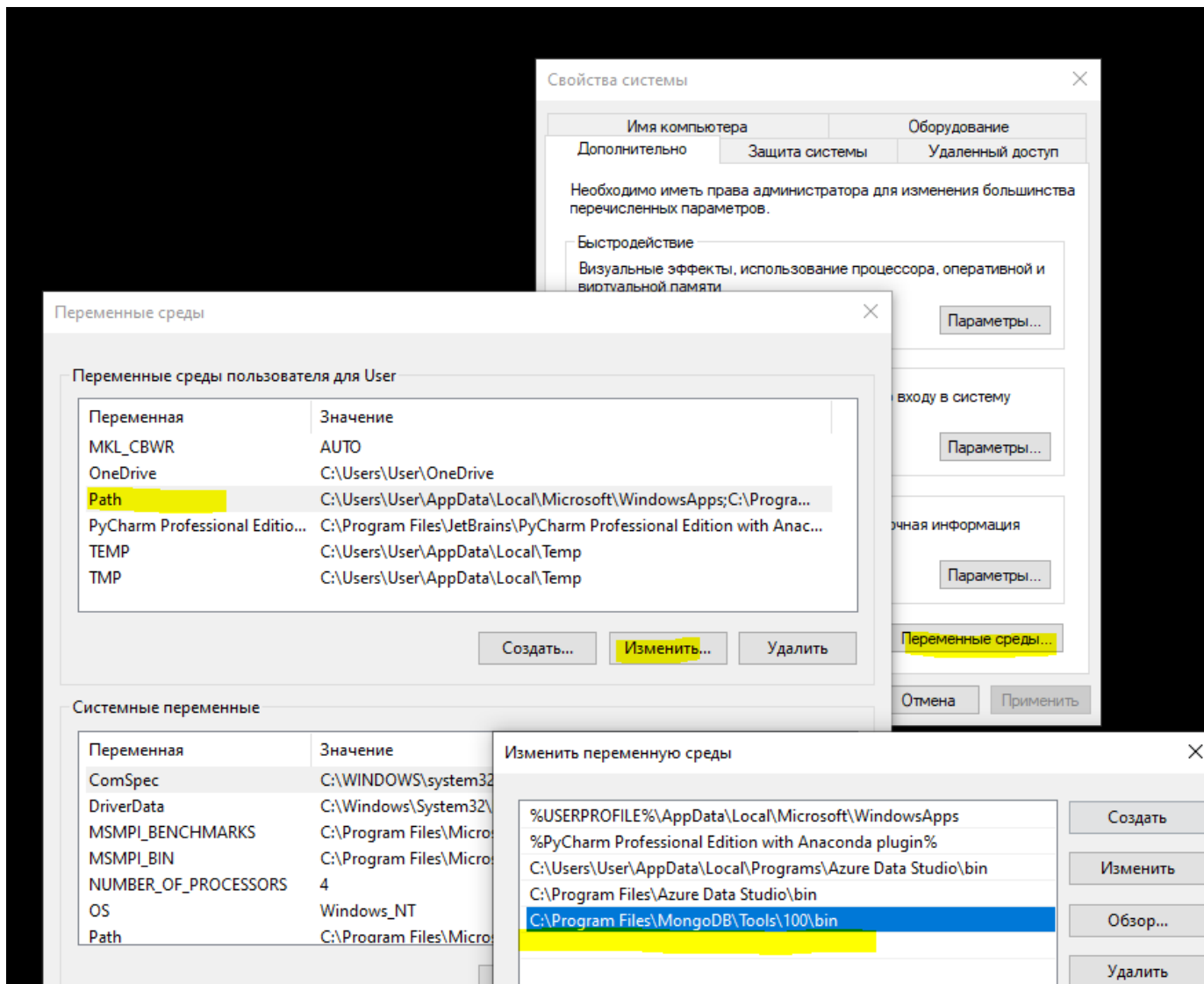
Место использования MongoDB

- Социальные сети, чаты, генераторы новостей
- Большие данные
- Каталоги для магазинов (с разнородными товарами)
- Работа с геопространственными данными
- Сбор и обработка информации с датчиков и считывающих устройств
- Сервисы блогосферы

Резервное копирование

- Используется набор утилит MongoDB Database Tools:
 - `mongodump` – полный дамп всех данных
 - `mongorestore` – восстановление из дампа
 - `bsondump` – конвертация данных в читаемый формат
 - `mongoexport` / `mongoimport` – экспорт / импорт данных из/в MongoDB в/из JSON или CSV
 - `mongostat` – статистика на уровне сервера
 - `mongotop` – статистика на уровне коллекции
 - `mongofiles` – управление большими файлами

MongoDB Database Tools – установить переменные среды



MongoDB Database Tools – mongodump

- Выполнение полной резервной копии

```
C:\Users\User>mongodump
2025-04-16T17:20:32.820+0300   writing admin.system.users to dump\admin\system.users.bson
2025-04-16T17:20:32.829+0300   done dumping admin.system.users (5 documents)
2025-04-16T17:20:32.830+0300   writing admin.system.roles to dump\admin\system.roles.bson
2025-04-16T17:20:32.839+0300   done dumping admin.system.roles (2 documents)
2025-04-16T17:20:32.840+0300   writing admin.system.version to dump\admin\system.version.bson
2025-04-16T17:20:32.841+0300   done dumping admin.system.version (2 documents)
2025-04-16T17:20:32.842+0300   writing all_subjects.locations to dump\all_subjects\locations.bson
2025-04-16T17:20:32.842+0300   writing all_subjects.places to dump\all_subjects\places.bson
2025-04-16T17:20:32.842+0300   writing all_subjects.reviews to dump\all_subjects\reviews.bson
2025-04-16T17:20:32.842+0300   writing all_subjects.areviews to dump\all_subjects\areviews.bson
2025-04-16T17:20:32.846+0300   done dumping all_subjects.locations (3 documents)
2025-04-16T17:20:32.849+0300   done dumping all_subjects.places (70 documents)
2025-04-16T17:20:32.869+0300   done dumping all_subjects.areviews (7 documents)
2025-04-16T17:20:33.507+0300   done dumping all_subjects.reviews (5555 documents)
```

C:\Users\User>

диск (C:) > dump > all_subjects

диск (C:) > dump > admin

Имя

- system.roles.bson
- system.roles.metadata.json
- system.users.bson
- system.users.metadata.json
- system.version.bson
- system.version.metadata.json

Имя

- areviews.bson
- areviews.metadata.json
- locations.bson
- locations.metadata.json
- places.bson
- places.metadata.json
- reviews.bson
- reviews.metadata.json

MongoDB Database Tools – mongoexport

- Восстановление из дампа

```
_id: ObjectId('67d94036c17e3f603c59c4cc')  
name: "Minsk"  
> loc: Object
```

```
_id: ObjectId('67d94040c17e3f603c59c4cd')  
name: "Svisloch"  
> loc: Object
```

```
_id: ObjectId('67d94090c17e3f603c59c4ce')  
name: "Belarus"  
> loc: Object
```

Drop Collection

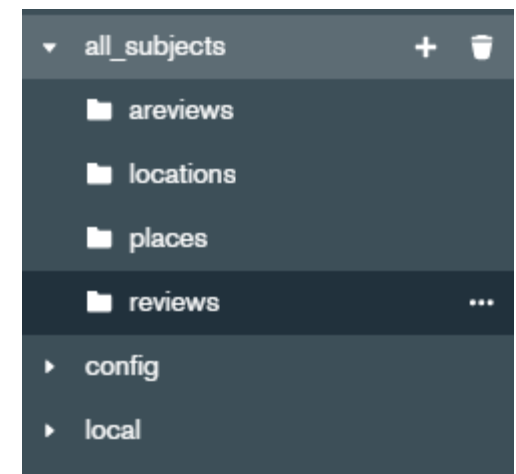
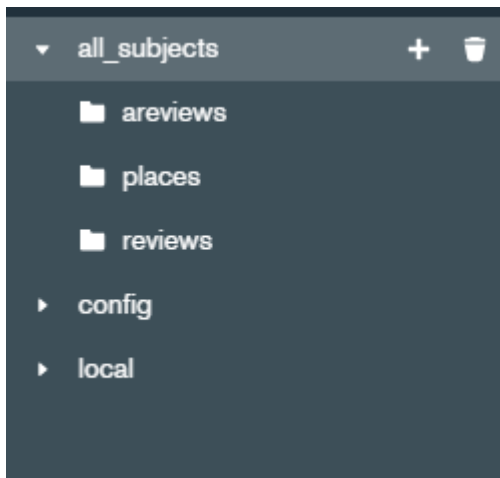
⚠ To drop **locations** type the collection name **locations**.

Cancel

Drop Collection

MongoDB Database Tools – mongorestore

```
2025-04-16T17:37:36.036+0300 finished restoring all_subjects.reviews (0 documents, 5555 failures)
2025-04-16T17:37:36.036+0300 restoring users from dump\admin\system.users.bson
2025-04-16T17:37:36.073+0300 restoring roles from dump\admin\system.roles.bson
2025-04-16T17:37:36.105+0300 restoring indexes for collection all_subjects.areviews from metadata
2025-04-16T17:37:36.105+0300 index: &idx.IndexDocument{Options:primitive.M{"default_language":"english", "language_override":"language",
"name":"name_text_summary_text", "textIndexVersion":3, "v":2, "weights":primitive.M{"name":2, "summary":4}}, Key:primitive.D{primitive.E{Key:
:"_fts", Value:"text"}, primitive.E{Key:"_ftsx", Value:1}}, PartialFilterExpression:primitive.D(nil)}
2025-04-16T17:37:36.106+0300 restoring indexes for collection all_subjects.locations from metadata
2025-04-16T17:37:36.106+0300 index: &idx.IndexDocument{Options:primitive.M{"max":1000, "min":-1000, "name":"coordinates_2d", "v":2}, Key:
primitive.D{primitive.E{Key:"coordinates", Value:"2d"}}, PartialFilterExpression:primitive.D(nil)}
2025-04-16T17:37:36.107+0300 no indexes to restore for collection all_subjects.places
2025-04-16T17:37:36.107+0300 restoring indexes for collection all_subjects.reviews from metadata
2025-04-16T17:37:36.107+0300 index: &idx.IndexDocument{Options:primitive.M{"max":1000, "min":-1000, "name":"loc.coordinates_2d", "v":2},
Key:primitive.D{primitive.E{Key:"loc.coordinates", Value:"2d"}}, PartialFilterExpression:primitive.D(nil)}
2025-04-16T17:37:36.107+0300 index: &idx.IndexDocument{Options:primitive.M{"default_language":"english", "language_override":"language",
"name":"name_text_summary_text", "textIndexVersion":3, "v":2, "weights":primitive.M{"name":1, "summary":7}}, Key:primitive.D{primitive.E{Key:
:"_fts", Value:"text"}, primitive.E{Key:"_ftsx", Value:1}}, PartialFilterExpression:primitive.D(nil)}
2025-04-16T17:37:36.141+0300 3 document(s) restored successfully. 5632 document(s) failed to restore.
```



MongoDB Database Tools – bsondump

```
2025-04-16T17:51:33.626+0300 70 objects found  
C:\Users\User>bsondump --outFile=place_from_bson.json C:\Users\User\dump\all_subjects\places.bson  
2025-04-16T17:51:33.626+0300 70 objects found
```

place_from_bson.json – Блокнот

Файл Правка Формат Вид Справка

```
{ "_id": { "$oid": "67d99ae5ec3dde719eb89915" }, "type": "Feature", "prop  
Westbound", "Town": "Brighouse", "Description": "3 Lockers", "latitud  
{"type": "Point", "coordinates": [{" $numberDouble": "-1.77908" }, {" $nu  
{"_id": { "$oid": "67d99ae5ec3dde719eb89916" }, "type": "Feature", "prop  
Sainsbury's", "Town": "Brighouse", "Description": "4 Stands", "latitu  
{"type": "Point", "coordinates": [{" $numberDouble": "-1.778372" }, {" $n  
{"_id": { "$oid": "67d99ae5ec3dde719eb89917" }, "type": "Feature", "prop  
Street", "Town": "Brighouse", "Description": "1 Stand", "latitude": "5  
{"type": "Point", "coordinates": [{" $numberDouble": "-1.779383" }, {" $n  
{"_id": { "$oid": "67d99ae5ec3dde719eb89918" }, "type": "Feature", "prop  
Pool", "Town": "Brighouse", "Description": "5 Stands & Shelter", "lat  
1.7793842" }, "geometry": {"type": "Point", "coordinates": [{" $numberDo  
{"_id": { "$oid": "67d99ae5ec3dde719eb89919" }, "type": "Feature", "prop  
Park", "Town": "Brighouse", "Description": "5 Stands & Shelter", "lat  
{"type": "Point", "coordinates": [{" $numberDouble": "-1.783504" }, {" $n
```

MongoDB Database Tools – mongostat и mongotop

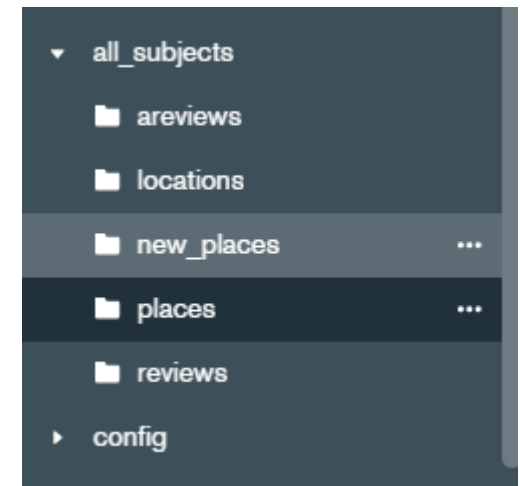
```
C:\Users\User>mongostat
insert query update delete getmore command dirty used flushes vsize res qrw arw net_in net_out conn
*0 *0 *0 *0 0 0|0 0.0% 5.2% 0 6.73G 455M 0|0 0|0 111b 60.3k 20 Apr 16 18:01:11.622
*0 *0 *0 *0 0 1|0 0.0% 5.2% 0 6.73G 455M 0|0 0|0 112b 60.4k 20 Apr 16 18:01:12.621
*0 *0 *0 *0 0 0|0 0.0% 5.2% 0 6.73G 455M 0|0 0|0 111b 60.3k 20 Apr 16 18:01:13.622
*0 *0 *0 *0 0 1|0 0.0% 5.2% 0 6.73G 455M 0|0 0|0 112b 60.4k 20 Apr 16 18:01:14.622
*0 *0 *0 *0 0 1|0 0.0% 5.2% 0 6.73G 455M 0|0 0|0 112b 60.4k 20 Apr 16 18:01:15.621
*0 *0 *0 *0 0 0|0 0.0% 5.2% 0 6.73G 455M 0|0 0|0 111b 60.3k 20 Apr 16 18:01:16.622
*0 *0 *0 *0 0 4|0 0.0% 5.2% 0 6.73G 455M 0|0 0|0 429b 61.3k 20 Apr 16 18:01:17.621
*0 *0 *0 *0 0 1|0 0.0% 5.2% 0 6.73G 455M 0|0 0|0 112b 60.4k 20 Apr 16 18:01:18.621
*0 *0 *0 *0 0 1|0 0.0% 5.2% 0 6.73G 455M 0|0 0|0 166b 60.6k 20 Apr 16 18:01:19.621
*0 *0 *0 *0 0 2|0 0.0% 5.2% 0 6.73G 455M 0|0 0|0 321b 61.0k 20 Apr 16 18:01:20.622
```

```
ns total read write 2025-04-16T18:05:11+03:00
admin.system.roles 0ms 0ms 0ms
admin.system.users 0ms 0ms 0ms
admin.system.version 0ms 0ms 0ms
all_subjects.all_subjects 0ms 0ms 0ms
all_subjects.areviews 0ms 0ms 0ms
all_subjects.locations 0ms 0ms 0ms
all_subjects.places 0ms 0ms 0ms
all_subjects.reviews 0ms 0ms 0ms
config.collections 0ms 0ms 0ms
config.system.sessions 0ms 0ms 0ms
```

MongoDB Database Tools – mongoexport и mongoimport

```
C:\Users\User>mongoimport --db all_subjects --collection "new_places" --drop
--type json --host "localhost:27017" --file "C:\Users\User\place_from_bson.
json"
2025-04-16T18:10:28.436+0300    connected to: mongodb://localhost:27017/
2025-04-16T18:10:28.445+0300    dropping: all_subjects.new_places
2025-04-16T18:10:28.524+0300    70 document(s) imported successfully. 0 docu
ment(s) failed to import.
```

- --db – имя базы данных
- --collection – название коллекции
- --drop – если коллекция была, то удалить
- --type – тип документа, из которого идет импорт, по умолчанию JSON
- --host – порт и хост инстанса
- --file – путь к файлу



Вопросы?