

Architecture

Mag. Thomas Griesmayer

JSON

- JavaScript Object Notation, is a human-readable data interchange format, specified in the early 2000s.
- Even though JSON is based on a subset of the JavaScript programming language standard, it's completely language-independent.
- JSON objects are associative containers, wherein a string key is mapped to a value
 - number
 - string
 - boolean
 - array
 - null
 - object
- hash tables in Java
- JSON objects are easy for humans to understand and for machines to parse and generate.

```
{
  "_id": "456f1456bcf86456ff343401",
  "name": { "first" : "Thomas", "last" : "Griesmayer" },
  "programLanguages": [ "Java", "php", "C" ],
  "address": [
    {
      "street": "Spengergasse",
      "houseNumber",
      "zip": 1050,
      "city": "Vienna"
    },
    {
      "zip": 2700,
      "city": "Wr. Neustadt"
    }
  ]
}
```

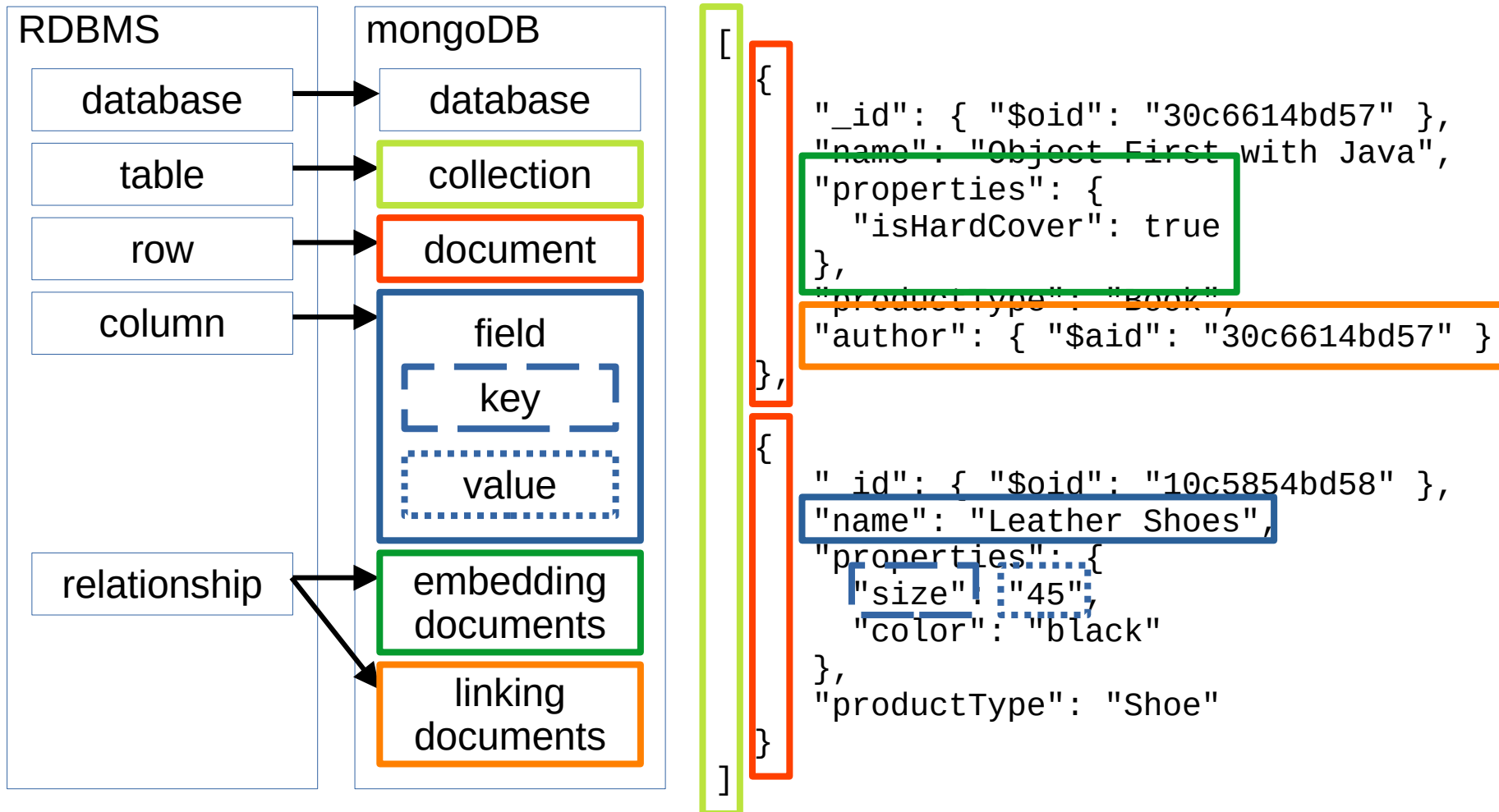
BSON

- BSON stands for “Binary JSON” and that’s exactly what it was invented to be.
- BSON’s binary structure encodes type and length information, which allows it to be traversed much more quickly compared to JSON.
- BSON adds some non-JSON-native data types, like dates and binary data, without which MongoDB would have been missing some valuable support.
- MongoDB stores BSON documents.

```
{"hello": "world"}  
\x16\x00\x00\x00          // total document size  
\x02                      // 0x02 = type String  
hello\x00                 // field name  
\x06\x00\x00\x00world\x00  // field value  
\x00                      // 0x00 = type E00 ('end of object')
```

<https://www.mongodb.com/json-and-bson>
Mag. Thomas Griesmayer

RDBMS / mongoDB



RDBMS / mongoDB

RDBMS	mongoDB
Database	Database
Table	Collection
Tuple or Row	Document
Column	Field
Table Join	Embedded Documents
Primary Key	Primary key / Default key
Mysqld / Oracle	mongod

<https://www.w3schools.in/mongodb/overview>
Mag. Thomas Griesmayer

```
$ mongosh
```

```
Connecting to:      mongodb://127.0.0.1:27017/
```

```
test> show dbs
```

```
admin      40.00 KiB
```

```
config    108.00 KiB
```

```
local      72.00 KiB
```

```
test> use shop
```

```
switched to db shop
```

```
shop> show collections
```

```
shop> db.customers.insertOne(
```

```
{  "_id"      : "1",  
    "name"    : {"firstName": "Thomas", "lastName": "Griesmayer"},  
    "gender"  : "male"  
} )
```

```
{ acknowledged: true, insertedId: '1' }
```

```
shop> show collections
customers
```

```
shop> show dbs
admin      40.00 KiB
config    108.00 KiB
local      72.00 KiB
shop       40.00 KiB
```

```
shop> db.customers.insertOne(
{ "_id"      : "2",
  "name"     : {"firstName": "Andrea", "lastName": "Griesmayer"},
  "gender"   : "female"
})
{ acknowledged: true, insertedId: '2' }
```

```
shop> db.customers.insertOne(  
  { "_id"      : "3",  
    "name"     : {"firstName": "Verena", "lastName": "Griesmayer"},  
    "gender": "female"  
  } )  
{ acknowledged: true, insertedId: '3' }
```

```
shop> db.customers.insertOne(  
  { "_id"      : "4",  
    "name"     : {"firstName": "Marion", "lastName": "Griesmayer"},  
    "gender": "female"  
  } )  
{ acknowledged: true, insertedId: '4' }
```

```
shop> db.customers.insertOne(  
  { "_id"      : "5",  
    "name"     : {"schoolName": "HTL Spengergasse"}  
  } )  
{ acknowledged: true, insertedId: '5' }
```

```
shop> db.customers.insertOne(  
  { "_id"      : "6",  
    "name"     : {"companyName": "IBM"},  
    "products" : ["SPSS", "Watson Studio", "Cognos Analytics"],  
    "employees": 345900  
  } )  
{ acknowledged: true, insertedId: '6' }
```

```
shop> db.customers.find()  
[  
  {  
    _id: '1',  
    name: { firstName: 'Thomas', lastName: 'Griesmayer' },  
    gender: 'male'  
  },  
  ...  
  { _id: '5', name: { schoolName: 'HTL Spengergasse' } },  
  {  
    _id: '6',  
    name: { companyName: 'IBM' },  
    products: [ 'SPSS', 'Watson Studio', 'Cognos Analytics' ],  
    employees: 345900  
  }  
]
```

```
shop> db.customers.findOne()  
{  
  _id: '1',  
  name: { firstName: 'Thomas', lastName: 'Griesmayer' },  
  gender: 'male'  
}
```

```
shop> db.customers.drop()  
true
```