Cutters
Guest Lecture

INF115 April 2021
Jakob Manne

Cutters from 2015 - 2021

Cutters, numbers by numbers

Owned by Procuritas AB (2019)

22 Non-cutting stab

3 Countries

ca 100 Salons

Ca 500 hairdressers

3.500 cuts / day.
Customer satisfaction:
9 av 10

	2015:	2016:	2017:	2018:	2019:
Omsetning:	147.000 kr	10 MNOK	67 MNOK	160 MNOK	232 MNOK

Cutters.

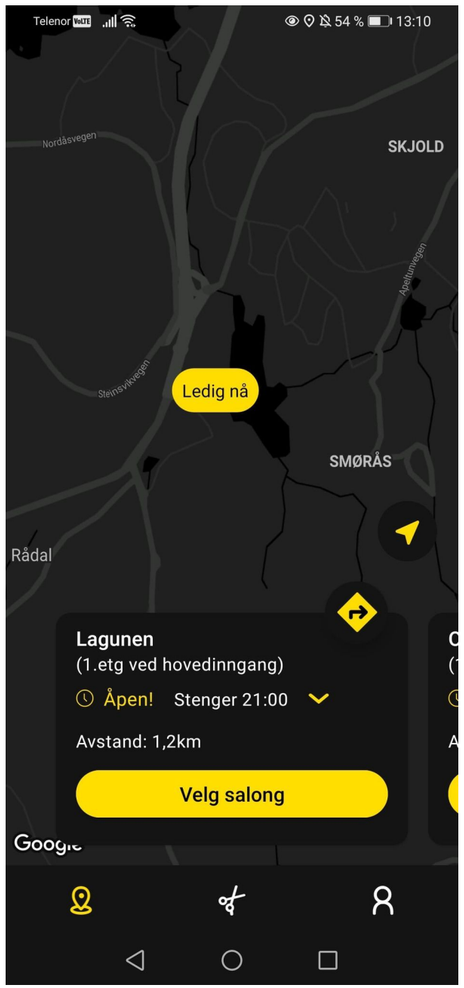
How to scale quickly, and globally?

Technology in Cutters

Small tech team consisting of 4 people

- Technology a key factor in the scaling of Cutters
- Utilizing the latest technologies in the market
- Self-service haircut logistics
- Our goal is to create the smoothest customer experience possible

Customer haircut overview



- Simple purchase through the app or Ipad in store
- Hairdresser tool for administering the queue

Cutters.

Databases - NoSQL vs SQL

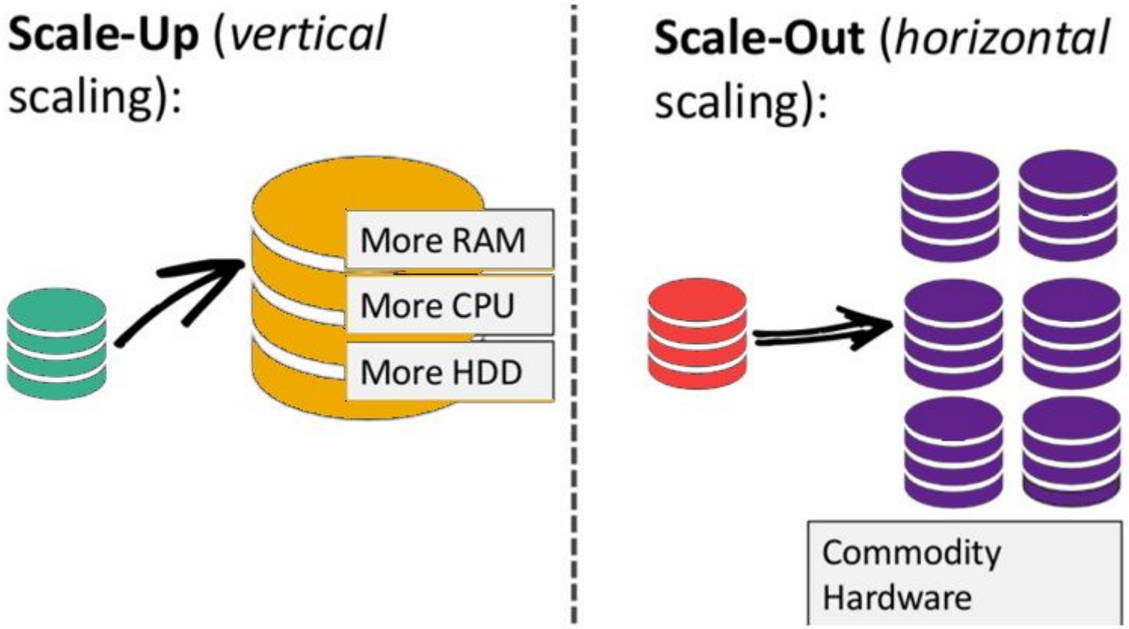
- Quick overview of differences

SQL (RBDMS) (Developed in the 1970's)	**NoSQL** (Developed in the late 2000's)
Standardized implementation	No standardized implementation
Requires ORM (Object Relational Mapping)	Documents map directly to data structures in most popular programming languages.
Stores data in tables with fixed rows and columns	Stores data in documents as JSON structures, Key-Value pairs, among other types.
Vertical scaling (Add more server resources (CPU, Memory etc)	Horizontal scaling (Spread load across more servers)

Cutters.

Database Scaling

- NoSQL is well suited for processing large amounts of data



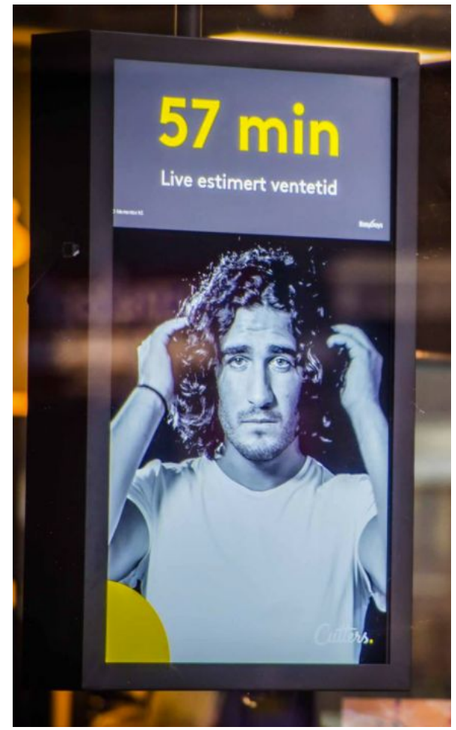
Cutters.

Databases based on NoSQL

- Document Databases (MongoDB, FireStore)
- Key-Value (Redis)
  - Used for caching
- Wide Column (Apache Cassandra)
- Graph-stores (Neo4j)

MongoDB - A Document Database

- Used for high volume data storage
- One of the most popular NoSQL databases with drivers for 10+ languages
- Stores data as JSON-like documents
- Ideal for Cutters
  - We need to deliver live waittime estimation to screens, hairdressers, customers



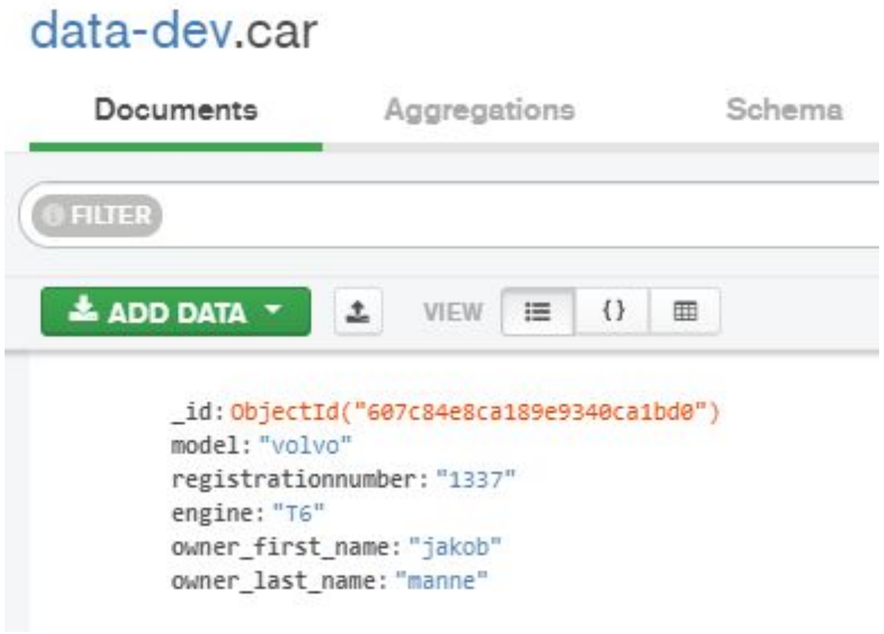
Cutters.

How is data stored in MongoDB?

- Data is stored in BSON-format (JSON)
- Data records are called “documents” which are grouped together in “collections”

SQL	NoSQL
Table	Collection
Row	Document

- Each document stored in a collection require a unique _id field, that acts as a **primary key**



Cutters.

More on Data Storage in MongoDB

- In SQL you split up data in separate tables, where in MongoDB you store the data embedded in the document.

```
0 references
public class Purchase
{
    [BsonId]
    [BsonRepresentation(BsonType.ObjectId)]
    0 references | 0 exceptions
    public string _id { get; set; }
    0 references | 0 exceptions
    public string CustomerName { get; set; }
    0 references | 0 exceptions
    public Order Order { get; set; }
}

1 reference
public class Order
{
    0 references | 0 exceptions
    public int OrderId { get; set; }
    0 references | 0 exceptions
    public string Product { get; set; }
    0 references | 0 exceptions
    public int Quantity { get; set; }
}
```

```
{
        _id : <ObjectId> ,

        CustomerName : Guru99 ,

        Order:
                {
                        OrderID: 111
                        Product: ProductA
                        Quantity: 5
                }
}
```

Cutters.

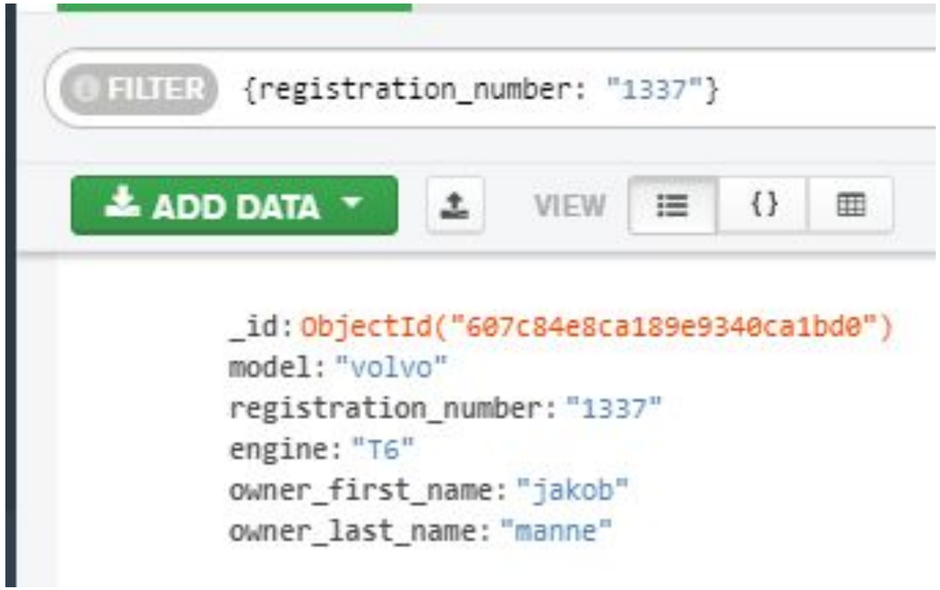
Queries in MongoDB

SQL: Select * from car where registration_number = “1337”

MongoDB: {registration_number: "1337"}

C# implementation

```
0 references
public async Task<CarEntity> GetCarByRegistrationNumber()
{
    var car = await database.GetCollection<CarEntity>("car")
                .Find(x => x.registration_number == "1337")
                .FirstOrDefaultAsync();

    return car;
}
```



Cutters.

More on queries in MongoDB

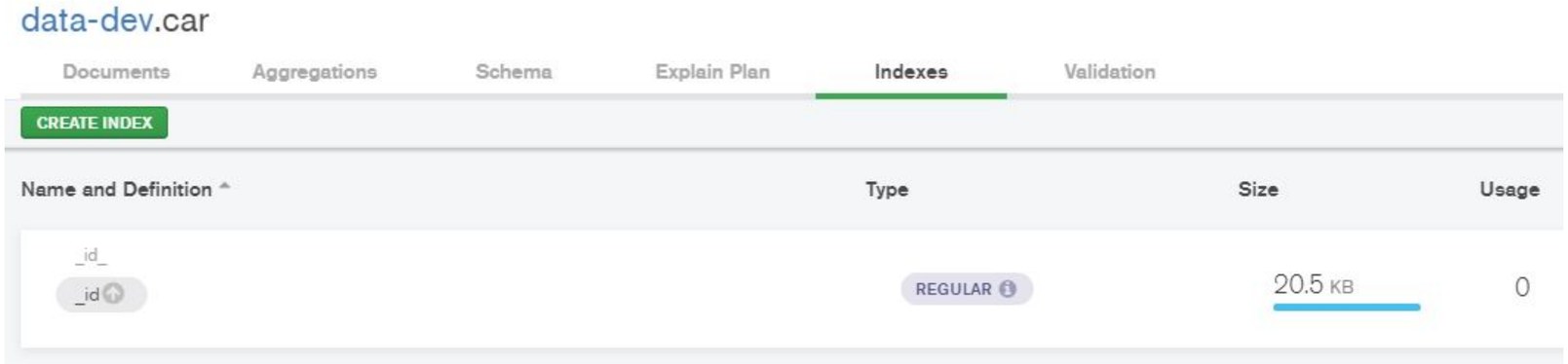
- In SQL you use ‘joins’ to query and select data from different tables
- In MongoDB you typically query a single collection and retrieve the entire document
- Perform multiple queries on different collections and then merge the data together in application code

```
0 references
public async Task<CarEntity> GetCarByRegistrationNumber(string query)
{
    var car = await database.GetCollection<CarEntity>("car")
              .Find(x => x.registration_number == "1337")
              .FirstOrDefaultAsync();

    return car;
}
```

```
0 references
public async Task<ParkingTicket> GetParkingTicketsByRegistrationNumber(string query)
{
    var ticket = await database.GetCollection<ParkingTicket>("parkingtickets")
              .Find(x => x.registration_number == "1337")
              .FirstOrDefaultAsync();

    return ticket;
}
```

Indexing in MongoDB

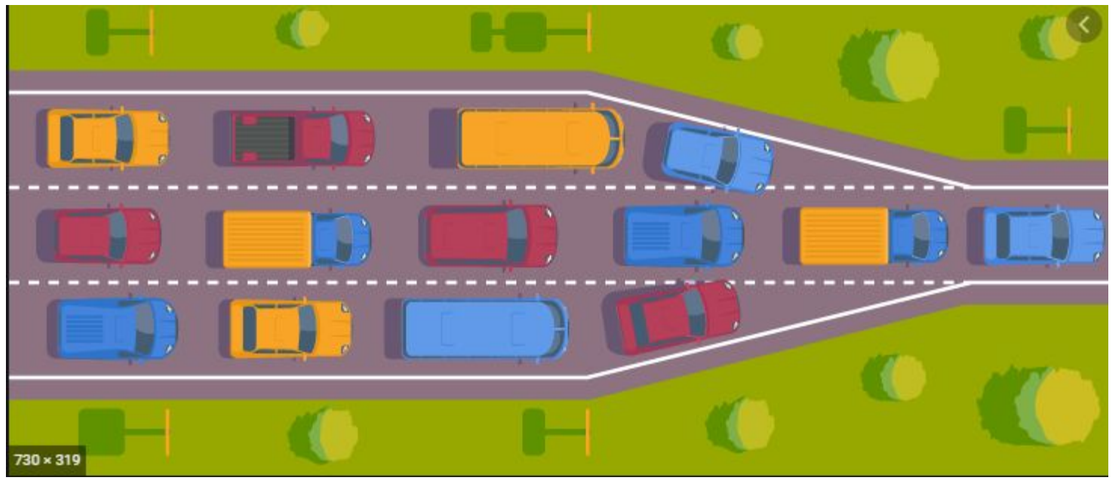
- Indexes makes the query execution times more efficient
- Without indexes MongoDB has to scan every document in a collection, to find the documents that match the query.
- Indexes are used to limit the amount of documents it must inspect



Cutters.

What are the benefits of using NoSql?

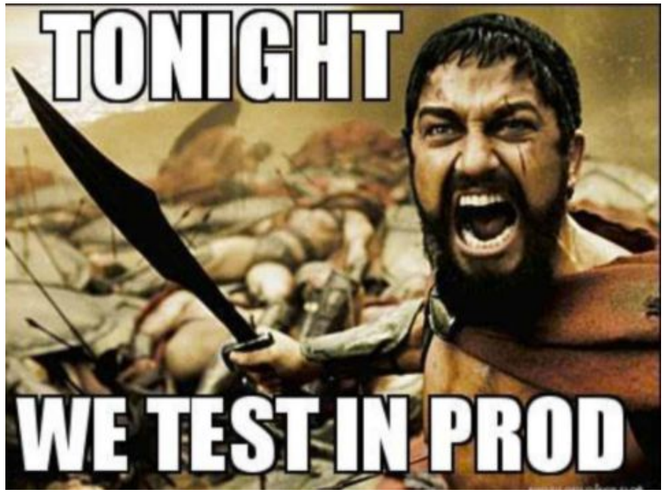
- You can store data as structured, unstructured or semi-structured
- Data is stored closer to the way it is used in the application using it
- Unlike SQL, you don’t need to design a model and then load the data into the database
- Removes bottlenecks as you don’t need to ask a database administrator to create changes
- This is too often the case in big corporations



Cutters.

Databases in a real production environment

- One database for each step in the development process
- Need proper release management
  - Store database changes in version control (GIT)
- Use migration scripts
- Staging environment should replica production environment



Cutters.

Database Monitoring and Performance

- Great, my query works, now what?

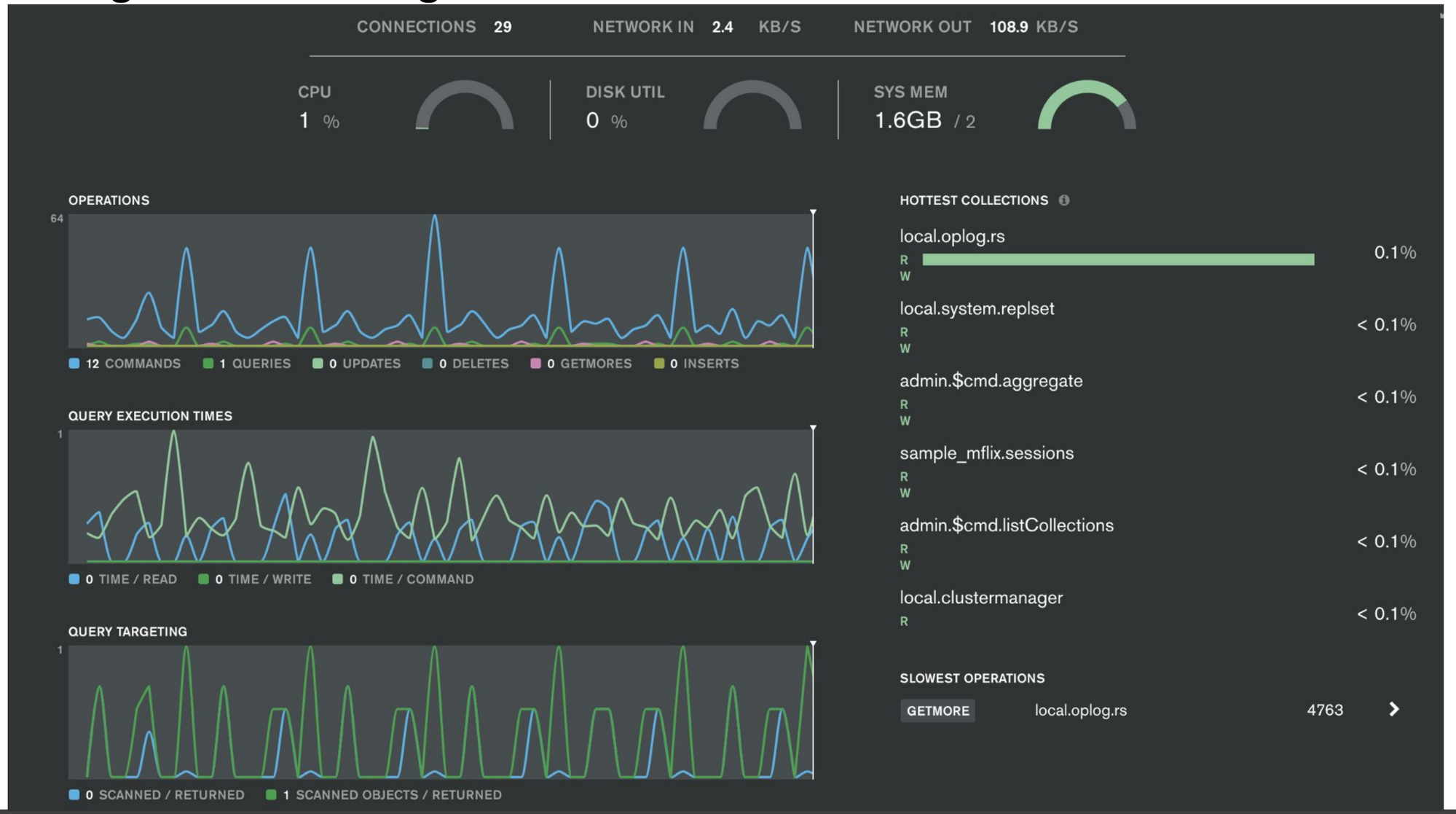
- I took down the entire Cutters platform by performing a Regex query (Pattern search)

- Good practise to test the query execution time

- How long time does it take to retrieve data?

- Several thousands of queries every minute

MongoDB Monitoring



Cutters.

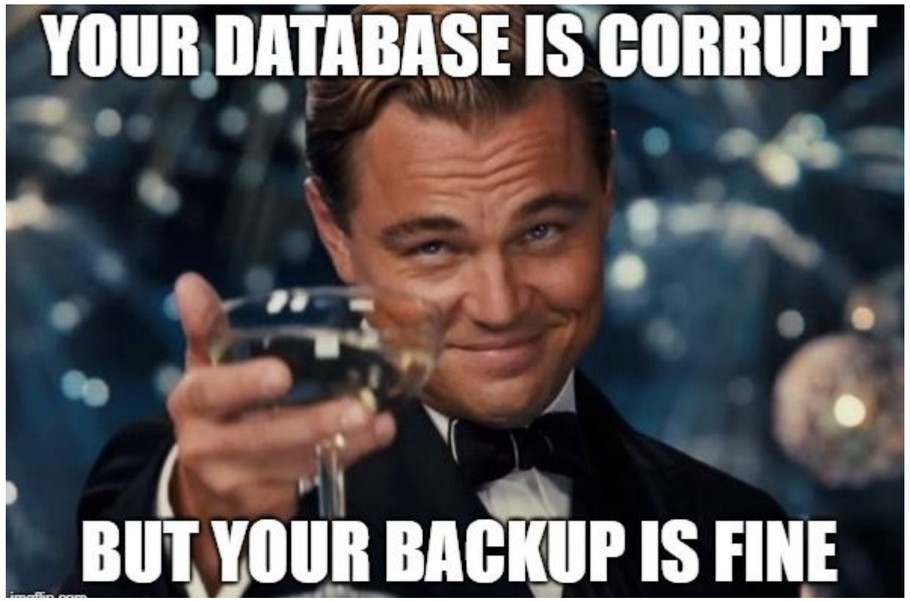
Database Security

- Whitelisting Ip’address
  - Only allow certain addresses to connect

- What if the database has been modified by mistake?

- What if the database is breached?
  - Backup/Snapshots

- Never store database connection strings in source control!



How do I use databases on a daily basis?

- Help hairdressers in investigating technical difficulties
- We rely on several third party' systems, where we store data in our database
- Monitor database performance
- Plan new functionalities that requires storing of data
  - Which types to use?
  - How can I make this query more efficient?
  - Write documentation for our partners

Cutters and databases

- We store live day to day information in our MongoDB
  - purchases, waittime

- We need to provide the rest of the organization with data, how many cuts per salon, customer satisfaction etc.

- We do this by exporting the data from MongoDB to our data warehouse provider

- If we did not export our data, we would have much longer query times

Cutters.

Do I really need to know databases and SQL?

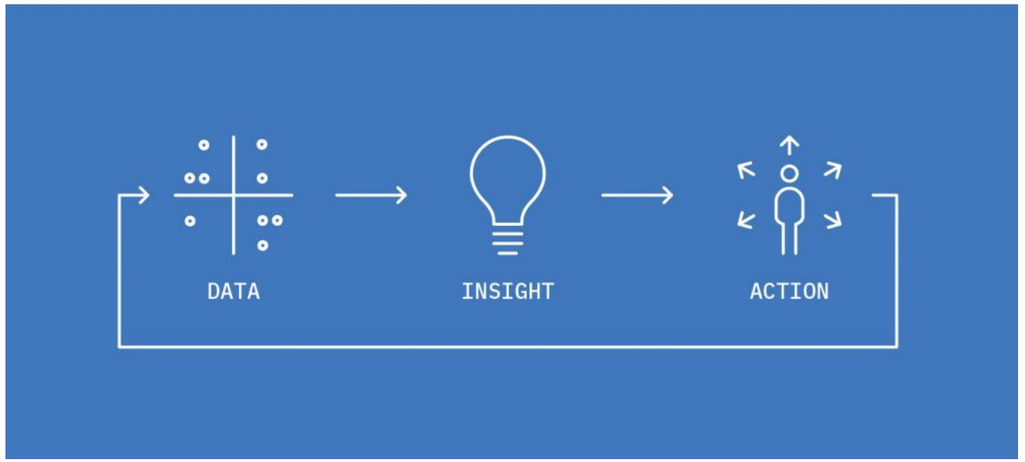
- When developing new functionalities you need test data
- A large part of programming is to obtain, process and store data
- You will be a much more efficient programmer
- A new career path?
  - Several exciting things happening in the field of Business Analytics
- They even teach SQL in courses at NHH

Databases role in the future business?

- Companies are becoming more “data-driven”
  - Key decisions are based on data and not on intuition

- Databases are used to store data and data, warehouses are used to analyze data.

- Data Warehouse (BigQuery/SnowFlake)
  - This is where all the fancy SQL-queries happen



Cutters.

Disadvantages of Nosql

- No standardization rules
- Limited query capabilities
- The learning curve can be stiff
- Can be hard to work with relational data

15 MINUTE **HAIRCUT**

The fastest growing beauty concept in the Nordics