Big Data Technologies - CSP 554
Assignment 11 - MongoDB
Prabhu Avula | A20522815
Illinois Institute of Technology, Chicago

Setup:

```
test> use assignment
switched to db assignment
assignment> load('./load.js')
true
assignment> db.unicorns.find()
[
  {
    _id: ObjectId('66ac378abf899bdfc0838726'),
    name: 'Horny',
    dob: ISODate('1992-03-13T07:47:00.000Z'),
    loves: [ 'carrot', 'papaya' ],
    weight: 600,
    gender: 'm',
    vampires: 63
  }
```

Exercise 1:

```
assignment> db.unicorns.find({weight: {$lt: 500}})
[
  {
    _id: ObjectId('66ac378abf899bdfc0838727'),
    name: 'Aurora',
    dob: ISODate('1991-01-24T13:00:00.000Z'),
    loves: [ 'carrot', 'grape' ],
    weight: 450,
    gender: 'f',
    vampires: 43
  },
  {
    _id: ObjectId('66ac378abf899bdfc083872d'),
    name: 'Raleigh',
    dob: ISODate('2005-05-03T00:57:00.000Z'),
    loves: [ 'apple', 'sugar' ],
    weight: 421,
    gender: 'm',
    vampires: 2
  }
]
```

Exercise 2:

```
assignment> db.unicorns.find({loves: "apple"})
[
  {
    _id: ObjectId('66ac378abf899bdfc0838729'),
    name: 'Roooooodles',
    dob: ISODate('1979-08-18T18:44:00.000Z'),
    loves: [ 'apple' ],
    weight: 575,
    gender: 'm',
    vampires: 99
  },
  {
    _id: ObjectId('66ac378abf899bdfc083872a'),
    name: 'Solnara',
    dob: ISODate('1985-07-04T02:01:00.000Z'),
    loves: [ 'apple', 'carrot', 'chocolate' ],
    weight: 550,
    gender: 'f',
    vampires: 80
  },
  {
    _id: ObjectId('66ac378abf899bdfc083872d'),
    name: 'Raleigh',
    dob: ISODate('2005-05-03T00:57:00.000Z'),
    loves: [ 'apple', 'sugar' ],
    weight: 421,
    gender: 'm',
    vampires: 2
  },
```

Exercise 3:

```
assignment> db.unicorns.insertOne({
...     name: "Malini",
...     dob: new Date("2008-11-03"),
...     loves: ["pears", "grapes"],
...     weight: 450,
...     gender: "F",
...     vampires: 23,
...     horns: 1
... })
{
  acknowledged: true,
  insertedId: ObjectId('66ac381ebf899bdfc0838733')
}
```

Exercise 4:

```
assignment> db.unicorns.updateOne(
...     { name: "Malini" },
...     { $push: { loves: "apricots" } }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Exercise 5:

```
assignment> db.unicorns.deleteMany({ weight: { $gt: 600 } })
{ acknowledged: true, deletedCount: 6 }
```

Exercise 6: Article Summary

The emergence of Web 2.0, IoT, and the vast amount of data generated has created a need for new data storage solutions. Traditional RDBMS struggle to handle the scalability and flexibility required for modern applications. This paper discusses the advantages of NoSQL databases, specifically MongoDB, for managing the temporal aspects of sensor data.

The paper addresses the challenges of handling structural, semi-structural, and unstructured data with traditional RDBMS. It highlights the need for horizontal scaling, schema flexibility, and the ability to process various data formats. The authors propose a middleware-based schema model for MongoDB to manage real-time ANT+ sensor data as hierarchical documents. This model ensures logical, compact storage and seamless schema evolution.

NoSQL databases offer schema flexibility and are ideal for storing various data formats without prior structural declarations. MongoDB, a document-oriented NoSQL database, supports dynamic schemas, making it suitable for IoT data. The paper focuses on efficiently storing and processing timestamped data from sensors.

Traditional RDBMS face several challenges, including scalability, flexibility, and performance issues with large datasets. RDBMS systems struggle with horizontal scaling, fixed schemas in RDBMS are unsuitable for dynamic applications, and multi-join queries in RDBMS can lead to performance issues with large datasets.

Big data requires frameworks that offer schema flexibility, scalability, and efficient processing. NoSQL databases, such as MongoDB, are designed to handle these requirements. MongoDB supports various data models, including key-value, document, column-oriented, and graph databases.

Real-time applications, like healthcare monitoring systems using Body Area Networks (BAN), require efficient time-series data management. The authors propose a data model for storing real-time temporal data in MongoDB, which involves using hierarchical documents to represent complex data structures and adapting the schema to accommodate new data formats without losing the advantages of big data solutions.

The proposed model addresses the need for schema flexibility and efficient temporal data management in NoSQL environments. MongoDB's document-oriented storage and dynamic schemas make it an ideal solution for real-time, data-driven applications. This summary highlights the paper's key points, emphasizing the benefits of using MongoDB for managing temporal aspects of sensor data in modern applications.