



INSTITUTE OF TECHNOLOGY TRALEE
WINTER EXAMINATIONS AY 2015-2016

Advanced Database Programming

Module Code DMDS 81001

CRN 48064

External Examiner: Mr Sean McHugh

Internal Examiner: Mr. P Given

Duration: 2 Hours

Instructions to Candidates:

- i) Answer any **three** questions.
 - ii) All questions carry equal marks. Submit all your rough-work, marks may be lost otherwise.
-

Question 1:

- i) CouchDB does not support ad-hoc queries. Explain how queries are developed in CouchDB, giving an example to support your answer. **(13 marks)**

Step 1.Views

- Views are a principal way that documents can be accessed.
- Filtering documents in the database to find those relevant to a particular process.
- Extracting data from document and displaying in specific order.
- Building efficient indexes to find documents of any value
- Using indexes to form relations in document.
- With view you can also perform all sorts of calculations with data in document.

Step 2. Map Reduce Queries

- CouchDB uses map reduce queries to query data from view.
 - Each view will have a map function

- Can have optional reduce function to SUM values.

Step 3. Example

```
function(doc)
(doc.albums || []).forEach(function(albums){
emit(doc.name , 1)
```

Step 4: Map Function

- The map function will be applied to all input documents that match the condition.
- In the case that there is multiple values per key the reduce phase will be used to condense the aggregated data.
- The map function will run in isolation and cannot modify the document.
- This is needed so couch doesn't have to recalculate a result given a guaranteed result

Step 5 : Reduce Phase

- The reduce phase is used when there is multiple values per key.
 - function(key,values,rereduce)
 - return sum(values)
- **Key** - Array of tuples , two element arrays containing key emitted by mapper and _id.
- **Values** - Array of values corresponding to the keys.
- **ReReduce** - A boolean value which will return true if Rereduction is invoked.

Step 4. Output

KEY	Value
The beatles	3

- ii) You have to choose a database for a project. What advantages do you think CouchDB has when compared to Relational Databases? **(10 marks)**

4 Advantages

Advantage 1,

- CouchDB does not enforce a schema uses JSON documents. Relational models structure data into rows and tables.

Advantage 2,

- CouchDB data can be structured, semi-structured and unstructured whereas data in relational must be normalized.

Advantage 3,

CouchDB JSON documents allow for nested/embedded objects. In relational models this may span several tables.

Example,

```
{ "id": 1234,
  "customer": "martin",
  "items": [
    {"product": "talisker", "quantity": 500},
    {"product": "macallan", "quantity": 800},
    {"product": "ledaig", "quantity": 1100}
  ]
}
```

Advantage 4,

- CouchDBs Asynchronous operation and optimistic concurrency enables applications for high throughput.
- In relational models there is strict enforcement of data integrity and normalisation, with tradeoff of lower performance and slower response times.

Advantage 5,

- Data is stored in key-document pairs, suited for applications which handle growing lists.
- With relational models data is stored in tables with fixed relations between tables.

iii) Write a note on the use of REST and JSON in couch. (10 marks)

JSON

- All documents in CouchDB are JSON documents.
- A JSON document consists of key-value pairs separated by commas,
Example,

“title” : “The social network”

- More advanced JSON objects consists 2 constructors objects and arrays.
- An object is a set of unordered name-value pairs.

Example,

```
    "members": [  
      {  
        "name": "Molecule Man",  
        "age": 29,  
        "secretIdentity": "Dan Jukes",  
        "powers": [  
          "Radiation resistance",  
        ]  
      }  
    ]  
  }  
}
```

- When CouchDB connects with the client the server responds with HTTP messages encoded in JSON.
- JSON uses a flexible data structure and stores data into documents instead of tables which consist of rows.

Advantages

- Supported by many javascript libraries
- Simple API.

REST

- The CouchDB API is the primary way of interfacing to the CouchDB instance.
- Requests are made through HTTP requests and can be used to request information from database.
- Storing new data in documents and performing formatting within documents.
- Different types of requests,
 - **GET** - Can be used to retrieve information from database.
 - **PUT** - Can be used to update data/entity.
 - **POST** - Can be used to insert data/entity.

Question 2:

- i) Describe an installation architecture in MongoDB which includes both replication and sharding and discuss the benefits of replication and sharding . (13 marks)

<https://intellipaat.com/blog/what-is-mongodb/>
<https://docs.mongodb.com/manual/replication/>
<https://docs.mongodb.com/manual/sharding/>

- ii) Compared to Relational Databases, MongoDB has a flexible data model. Discuss giving examples where appropriate. **(10 marks)**

MongoDB	RDBMS
Document oriented and non-relational database	Relational database
Document based	Row based
Field based	Column based
Collection based and key value pair	Table based
Gives Javascript client for querying	Doesn't give Javascript for querying
Relatively easy to setup	Comparatively not that easy to setup
It is unaffected by SQL injection	It is quite vulnerable to SQL injection
Ideal for hierarchical data storage	Not good for hierarchical data storage
Has dynamic schema	Contains predefined schema
100 times faster	Through increasing RAM vertical scaling can happen
It is horizontally scalable through sharding	Through increasing RAM vertical scaling can happen

- Mongo Structures data into a collection of JSON documents.
- JSON is a human readable format.
- Mongo has a flexible schema and you dont have to determine a tables shcemas before inserting.
- Mongo does not require documents to have same schema.
- Mongo allows for related data to be embedded whitin a single document.



- You can change structure of document as you like just add new fields, update existing etc.
- You can use references to store relationships between data by including links or references from one document to another.
- A write operation is atomic on the level of a single document, even if operation modifies multiple documents within single document.

Advanced Database Programming\NOSQL summary\Wrapping Up.docx

//*****

- iii) Compare and contrast CouchDB's and MongoDB's differing approaches to the CAP theorem. **(10 marks)**

Overview of CAP

Consistency : Every read will get most recent write

Availability : Every node executes queries.

Partition Tolerance : Even if connections are down between nodes A+C promises are kept.

MONGO is (Consistent + Partition Tolerant).

In terms of consistency for example if you were to do a write and then a read considering the write was successful you will always be able to read the result of the write you just read.

In terms of Partition Tolerance it is achieved through replica sets. As long as more than half of servers are connected a new primary can be chosen

CouchDB is (Availability and Partition Tolerant).

- Couch DB is Available and Partition Tolerant.
- Even thou

Question 3:

- i) Discuss, using examples, the Redis Sorted Set data structure and explain, using an example, how a union of sorted sets can be weighted in favour of one of the keys. What performance disadvantage do sorted sets have? **(10 marks)**

Sorted Sets

- Sorted sets are ordered like lists and unique like sets.
- They have field value pairs like hashes but instead of string fields they use numeric scores which denote the order of values.
- Every member of the sorted set gets a score that is used to take sorted set order from smallest to greatest.

- You can update and remove elements very fast.

Union

- You can create a destination key that contains a union of one or more keys.

Example of a union,

```
ZUNIONSTORE destination numkeys[key...]
[WEIGHTS weight][AGGREGATION SUM|MIN|MAX]
```

- **destination** key is the key to store into and key is one or more keys to union.
- **numkeys** is simply the value you are joining.
- **aggregate** is optional rule for resolving each weighted score and sum by default.

Performance Disadvantages

ii) Describe, using an example, how the Redis Publish Subscribe model works
(11 marks)

- SUBSCRIBE, UNSUBSCRIBE and PUBLISH are all implementing the message paradigm where senders(PUBLISHERS) are not programmed to send messages to specific receivers.
- PUBLISHED messages are characterized into channels without knowledge of what SUBSCRIBERS there will be
- SUBSCRIBERS Express interest in one or more channels and only receive message of interest not knowing if publisher is there.
- Example,

```
Subscribe foo bar.
```

This will subscribe you to both foo and bar channels.

- Messages sent by other clients to these channels will be pushed by redis to all subscribers.
- A client can also UNSUBSCRIBE,
UNSUBSCRIBE foo bar;

- Publishing a comment,
PUBLISH Comments "How are you"

```
//*****
```

iii) Describe how the Redis durability model works and list the advantages and disadvantages of this model. (12 marks)

For Reference these models can be located here :

<https://redis.io/topics/persistence>

Durability options = RDB + AOF + In Memory

1. In Memory

- The first persistence option is having none at all where all values are kept in memory

2. RDB

- The RDB persistence provides point in time snapshots
- snapshots of your dataset in specific intervals.

Advantages

- RDB is a compact single file point in time representation of redis data which is perfect for backups.

Disadvantages

- RDB is not good if you want to minimize data loss in case server goes down.

3. AOF

AOF persistence logs every write operation received by server that plays again at startup.

Advantages

The AOF log is an appended log that holds every write so there is no corruption if outage.

Disadvantages

The AOF file is considerably bigger than the equivalent RDB files.

Redis's Strengths

The obvious strength of Redis is speed, like so many key-value stores of its ilk.

But more than most key-value stores, Redis provides the ability to store complex values like lists, hashes, and sets, and retrieve them based through operations specific to those datatypes.

Beyond even a data structure store, however, Redis's durability options allow you to trade speed for data safety up to a fairly fine point.

Built-in master-slave replication is another nice way of ensuring better durability without requiring the slowness of syncing an append-only file to disk on every operation.

Additionally, replication is great for very high-read systems.

Redis's Weaknesses

Redis is fast largely because it resides in memory.

Some may consider this cheating, since of course a database that never hits the disk will be fast.

A main memory database has an inherent durability problem; namely, if you shut down the database before a snapshot occurs, you can lose data.

Even if you set the append-only file to disk sync on every operation, you run a risk with playing back expiry values, since time-based events can never be counted on to replay in exactly the same manner—though in fairness this case is more hypothetical than practical.

Redis also does not support datasets larger than your available RAM (Redis is removing virtual memory support), so its size has a practical limitation.

Although there is a Redis Cluster currently in development to grow beyond a single-machine's RAM requirements, anyone wanting to cluster Redis must currently roll their own with a client that supports it.

```
//*****
```

Question 4:

i) Appendix 1 shows a graph database. Explain how the following Gremlin queries arrive at a result (13 marks)

- a. `g.V.filter{it.name=='Ardagh Castle Cheese'}.outE.inV.name`
- b. `alice.bothE('friends').bothV.name` (Note *alice* is a reference to the Vertex named "Alice")
- c. `alice.bothE('friends').bothV.except([alice]).loop(3){it.loops <= 2}.name` (Note *alice* is a reference to the Vertex named "Alice")
- d. `cheese_count = [:]
g.V.outE('likes').outV.name.groupCount(cheese_count)`

```
cheese_count
```

```
//*****
```

ii) Discuss the architecture of the Riak ring and discuss the advantage this ring gives to Riak. (12 marks)

Week 7

Riak

```
//*****
```

iii) Riak allows us to control reads and writes into the cluster by altering three values: N, W, and R. Describe these three parameters and discuss a scenario where the W parameter can be used. (8 marks)

Appendix 1:

