Distributed Database Management

# Module 06 Assignment – Aggregation Framework

Jason Kendall – Professor Grinberg



MongoDB is a database that we have been using and studying quite extensively in this course. It is a great example of distributed databases and cloud computing technologies. One of the great features of this type of database which stores key and value pairs, is that functions can be created to operate on entire data models. This can be particularly useful as functions can be custom or designed to accomplish a set of tasks that a developer is trying to accomplish.

An example of what this type of code would look like can be seen here:

**Various Aggregate Functions**

1. Count()

2. Average()

3. Max()

For MongoDB using key value pairs to define the data, our functions would operate on a like this:

Key | Value

employeeID | gender

01 | Male

02 | Female

03 | Male

Count(name) and the return would return the sum of the employes

MongoDB runs their code a little different from the general function we described above. To apply a function on a database in Mongo, we provide a filter that operates like a query and document transformations that modify the form of the output document.

A better example would look like this:



By applying the aggregate filter on the orders collection in our MongoDB, we can then create a pipeline that matches “Status” and “CustomerID” somewhat like in our previous example where the results are considered aggregated from the collection Orders. The features and benefits of working with this type of database is that you can work within much fewer confines of standard SQL databases and use more “written language”. This makes learning the technology easier for new developers and makes storing and managing data easier as well.

**Resources**:

<https://docs.mongodb.com/manual/aggregation/>

<https://www.geeksforgeeks.org/database-management-system-aggregate-functions/>

<https://swcarpentry.github.io/sql-novice-survey/06-agg/>