Group 149

**SQL-Mongo Project – Spatial Data of US Wildfires**

**Khushboo Rathore**

|  |
| --- |
| **Activity** |
| Prepared Data Model and Created Physical DB |
| Loaded Data into Database |
| Wrote SQL Queries |
| Prepared Mongo Database |
| Loaded data into Mongo DB |
| Wrote Mongo Queries |
| Prepared Report |
| Reviewed Report |

Contents

Data Model 5

Assumptions/Notes About Data Entities and Relationships 5

Entity-Relationship Diagram 5

Physical Database 6

Assumptions/Notes About Data Set 6

Screen shot of Physical Database objects 6

Data in the Database 6

SQL Queries 7

Query 1 7

Question 7

Notes/Comments About SQL Query and Results (Include # of Rows in Result) 7

Translation 7

Screen Shot of SQL Query and Results 7

Query 2 8

Question 8

Notes/Comments About SQL Query and Results (Include # of Rows in Result) 8

Translation 8

Screen Shot of SQL Query and Results 8

Query 3 9

Question 9

Notes/Comments About SQL Query and Results (Include # of Rows in Result) 9

Translation 9

Screen Shot of SQL Query and Results 9

Query 4 10

Question 10

Notes/Comments About SQL Query and Results (Include # of Rows in Result) 10

Translation 10

Screen Shot of SQL Query and Results 10

Query 5 11

Question 11

Notes/Comments About SQL Query and Results (Include # of Rows in Result) 11

Translation 11

Screen Shot of SQL Query and Results 11

Query 6 12

Question 12

Notes/Comments About SQL Query and Results (Include # of Rows in Result) 12

Translation 12

Screen Shot of SQL Query and Results 12

Data Review for MongoDB 13

Assumptions/Notes About Data Collections, Attributes and Relationships between Collections 13

Physical Mongo Database 14

Assumptions/Notes About Data Set 14

Screen shot of Physical Database objects (Database, Collections and Attributes) 14

Data in the Database 14

MongoDB Queries/Code 15

Query 1 15

Question 15

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result) 15

Translation 15

Screen Shot of MongoDB Query/Code and Results 15

Query 2 16

Question 16

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result) 16

Translation 16

Screen Shot of MongoDB Query/Code and Results 16

Query 3 17

Question 17

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result) 17

Translation 17

Screen Shot of MongoDB Query/Code and Results 17

Query 4 18

Question 18

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result) 18

Translation 18

Screen Shot of MongoDB Query/Code and Results 18

Query 5 19

Question 19

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result) 19

Translation 19

Screen Shot of MongoDB Query/Code and Results 19

Query 6 20

Question 20

Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result) 20

Translation 20

Screen Shot of MongoDB Query/Code and Results 20

Data Model

## Assumptions/Notes About Data Entities and Relationships

Include assumptions about data entities and their relationships with each other.

Include reasons why the data model is in 3NF.

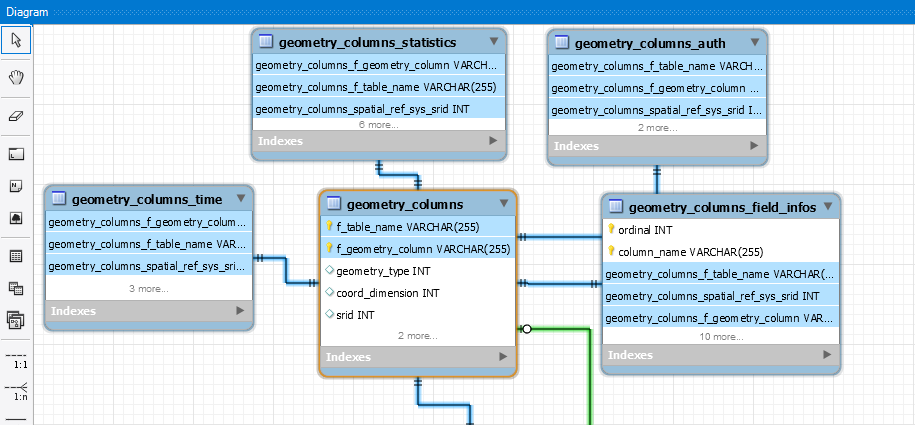
**Project Assumptions:**

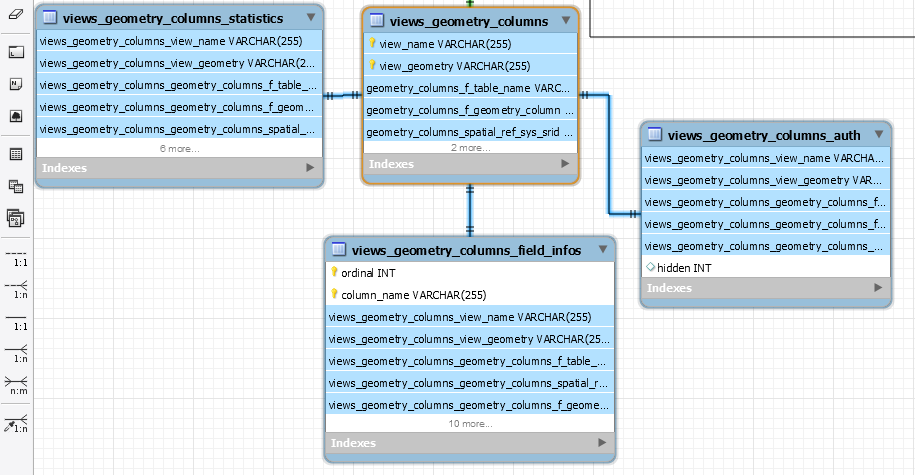
1. spatial\_ref\_sys\_aux table will only have srid’s which are present in spatial\_ref\_sys table. There can be one or more spatial\_ref\_sys\_aux\_ID in spatial\_ref\_sys table.
2. For spatial\_ref\_sys to be in 1NF, we are breaking spatial\_ref\_sys\_name into 2 parts: ref\_sys\_name\_1 and ref\_sys\_name\_2. Since as per 1NF, “There can be no multipart or multivalued fields.”
3. idx\_Fires\_Shape\_Parent contains node number of each idx\_Fires\_Shape\_node node number with the exception of a node which does not have any parent node number.
4. For NWCG\_UnitIDActive\_20170109 table to be in 3NF, we are creating Location table which contains Country, State and Code. Since as per 3NF, “No transitive dependency: every non-key column is functionally depending only on the entire primary key, and nothing else.”
5. Fires table contains many nodes from idx\_Fires\_Shape\_node table. So there is one to many relationship between these two tables.
6. There will be zero to many relationships between Fires and FIPS table. Every FIPS code will have corresponding Object ID in Fires table but Fires table may or may not have FIPS code.
7. For every fire, there is a fire size class assigned to it. Fire class is a derived column in Fires table which can be defined from Fires\_Size\_Class table.
8. Every fire may or may not have a MTBS ID and one MTBS ID can be used for multiple fires, so there exists zero to many relationship between Fires and MTBS table.
9. Every fire may or may not have ICS\_209\_INCIDENT\_NUMBER and one ICS\_209\_INCIDENT\_NUMBER can be used for multiple fires. So there exists zero to many relationship between Fires and ICS\_209 table.
10. For every fire, there is SOURCE\_SYSTEM and corresponding SOURCE\_SYSTEM\_TYPE which is defined in Source\_System table.
11. For every fire, there is SOURCE\_REPORTING\_UNIT and corresponding SOURCE\_REPORTING\_UNIT\_NAME that is defined in SOURCE\_REPORTING\_UNIT table.
12. For every Location where the fire occurred, there is a corresponding Owner which is defined in Owner table.
13. To define cost of each fire, there is a corresponding stat\_cause\_code which is defined in Statistic\_Cause table.
14. Every NWCG\_REPORTING\_ID is assigned to one or more fires and there exist corresponding NWCG\_REPORTING\_AGENCY and NWCG\_REPORTING\_UNIT\_NAME.
15. DISCOVERY\_DOY can be derived from DISCOVERY\_DATE. Therefore we can remove DISCOVERY\_DOY column from the Fires table.
16. Fire\_Year can be derived from DISCOVERY\_DATE. Therefore, we can remove the Fire\_Year column from Fires table.
17. CONT\_DOY can be derived from CONT\_DATE. Therefore, we can remove the CONT\_DOY column from Fires table.
18. Every spatial\_ref\_sys have an authority name which is defined in Authority table. One authority can be assigned to many srid.
19. There is one to many relationships between spatial\_ref\_sys\_aux and spatial\_ref\_sys table.
20. Every GACC is assigned to one or more NWCG\_Agency\_Unit, which is defined in Geographic table.
21. We can remove redundant idx\_Fires\_Shape\_rowid table, since it only contains information on node numbers. This column can be added to the main fires table.

## 

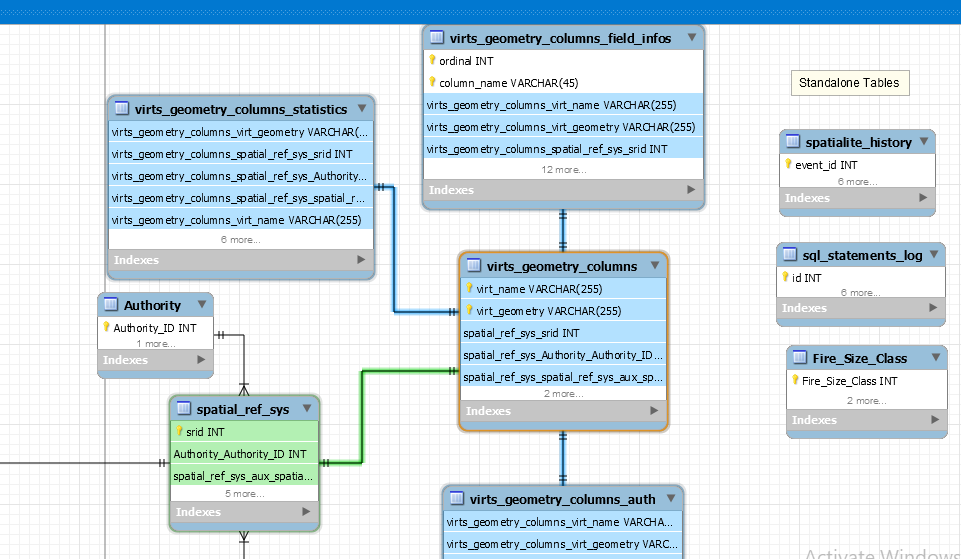
## Entity-Relationship Diagram

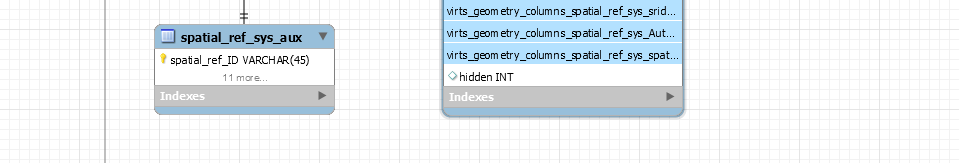
Geometric Columns –



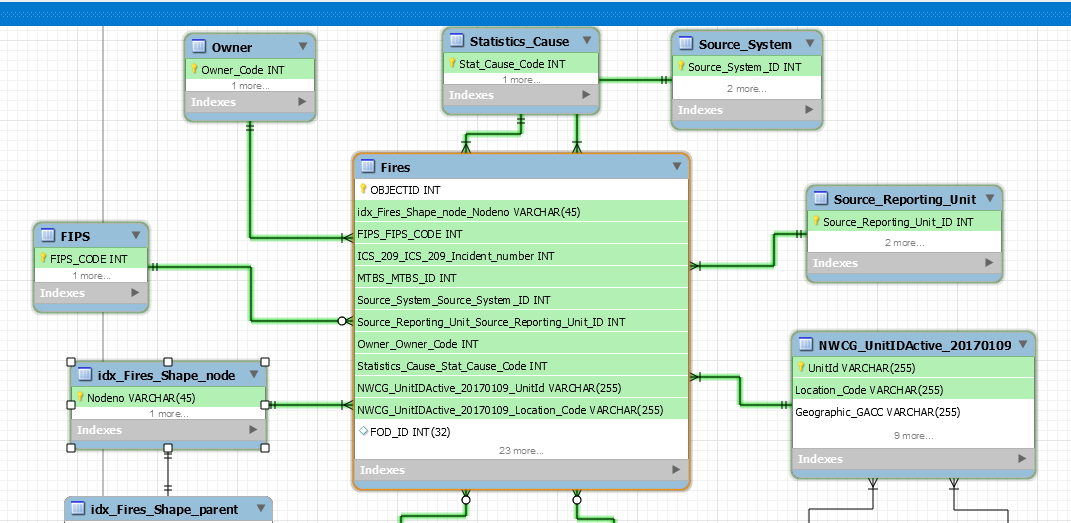


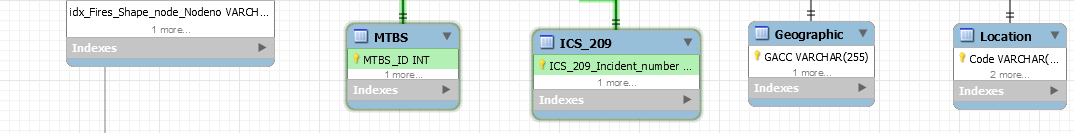
Virts\_geometry –





Fires –





Physical Database

## Assumptions/Notes About Data Set

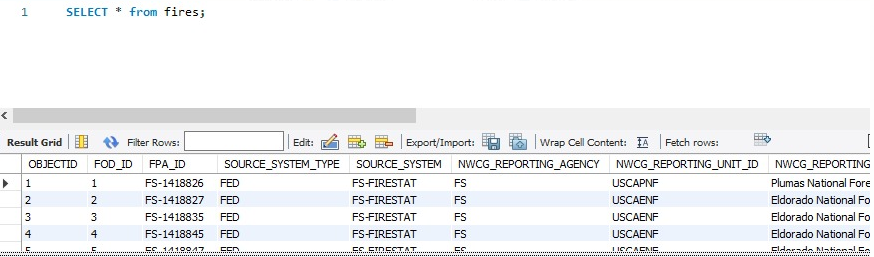
Include any assumptions made about data such as empty fields, sparse data, bad data, etc.

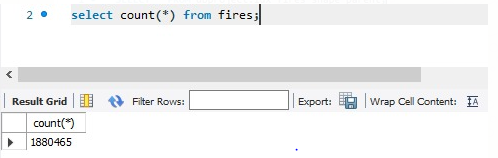
## Screen shot of Physical Database objects

## Data in the Database

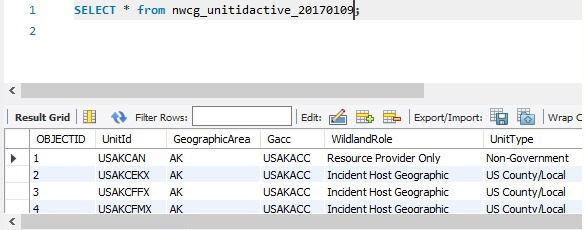
|  |  |  |  |
| --- | --- | --- | --- |
| **Table Name** | **Primary Key** | **Foreign Key** | **# of Rows in Table** |
| Fires | Column (OBJECTID) |  | 1880465 |
| geometry\_columns | pk\_geom\_cols (f\_table\_name, f\_geometry\_column) | fk\_gc\_srs (srid) REFERENCES spatial\_ref\_sys (srid) | 1 |
| geometry\_columns\_auth | pk\_gc\_aut (f\_table\_name, f\_geometry\_column) | fk\_gc\_auth (f\_table\_name, f\_geometry\_column) REFERENCES geometry\_columns (f\_table\_name, f\_geometry\_column) ON DELETE CASCADE | 1 |
| geometry\_columns\_field\_infos | pk\_gcfld\_infos (f\_table\_name, f\_geometry\_column, ordinal, column\_name) | fk\_gcfld\_infos (f\_table\_name, f\_geometry\_column) REFERENCES geometry\_columns (f\_table\_name, f\_geometry\_column) ON DELETE CASCADE | 0 |
| geometry\_columns\_statistics | pk\_gc\_statistics (f\_table\_name, f\_geometry\_column) | fk\_gc\_statistics (f\_table\_name, f\_geometry\_column) REFERENCES geometry\_columns (f\_table\_name, f\_geometry\_column) ON DELETE CASCADE | 1 |
| geometry\_columns\_time | pk\_gc\_time (f\_table\_name, f\_geometry\_column) | fk\_gc\_time (f\_table\_name, f\_geometry\_column) REFERENCES geometry\_columns (f\_table\_name, f\_geometry\_column) ON DELETE CASCADE | 1 |
| idx\_Fires\_Shape\_node | nodeno |  | 72263 |
| idx\_Fires\_Shape\_parent | nodeno |  | 72262 |
| idx\_Fires\_Shape\_rowid | rowid |  | 1880465 |
| NWCG\_UnitIDActive\_20170109 | OBJECTID |  | 5867 |
| spatial\_ref\_sys | srid |  | 4924 |
| spatial\_ref\_sys\_aux | srid | fk\_sprefsys (srid) REFERENCES spatial\_ref\_sys (srid) | 4873 |
| spatialite\_history | event\_id |  | 16 |
| sql\_statements\_log | id |  | 0 |
| views\_geometry\_columns | pk\_geom\_cols\_views (view\_name, view\_geometry) | fk\_views\_geom\_cols (f\_table\_name, f\_geometry\_column) REFERENCES geometry\_columns (f\_table\_name, f\_geometry\_column) ON DELETE CASCADE | 0 |
| views\_geometry\_columns\_auth | pk\_vwgc\_auth (view\_name, view\_geometry) | fk\_vwgc\_auth (view\_name, view\_geometry) REFERENCES views\_geometry\_columns (view\_name, view\_geometry) ON DELETE CASCADE | 0 |
| views\_geometry\_columns\_field\_infos | pk\_vwgcfld\_infos (view\_name, view\_geometry, ordinal, column\_name) | fk\_vwgcfld\_infos (view\_name, view\_geometry) REFERENCES views\_geometry\_columns (view\_name, view\_geometry) ON DELETE CASCADE | 0 |
| views\_geometry\_columns\_statistics | pk\_vwgc\_statistics (view\_name, view\_geometry) | fk\_vwgc\_statistics (view\_name, view\_geometry) REFERENCES views\_geometry\_columns (view\_name, view\_geometry) ON DELETE CASCADE | 0 |
| virts\_geometry\_columns | pk\_geom\_cols\_virts (virt\_name, virt\_geometry) | fk\_vgc\_srid (srid) REFERENCES spatial\_ref\_sys (srid) | 0 |
| virts\_geometry\_columns\_auth | pk\_vrtgc\_auth (virt\_name, virt\_geometry) | fk\_vrtgc\_auth (virt\_name, virt\_geometry) REFERENCES virts\_geometry\_columns (virt\_name, virt\_geometry) ON DELETE CASCADE | 0 |
| virts\_geometry\_columns\_field\_infos | pk\_vrtgcfld\_infos (virt\_name, virt\_geometry, ordinal, column\_name) | fk\_vrtgcfld\_infos (virt\_name, virt\_geometry) REFERENCES virts\_geometry\_columns (virt\_name, virt\_geometry) ON DELETE CASCADE | 0 |
| virts\_geometry\_columns\_statistics | pk\_vrtgc\_statistics  (virt\_name, virt\_geometry) | fk\_vrtgc\_statistics (virt\_name, virt\_geometry) REFERENCES virts\_geometry\_columns (virt\_name, virt\_geometry) ON DELETE CASCADE | 0 |

Fires table loaded in the database:



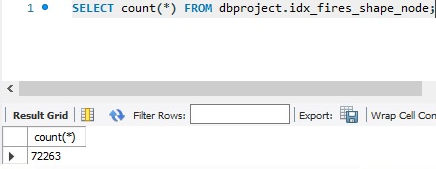


NWCG table loaded in the database :

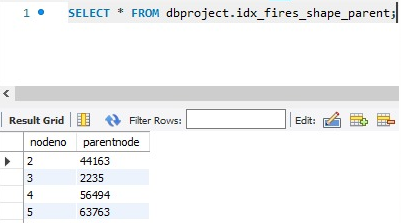


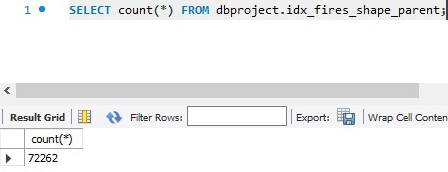
## 

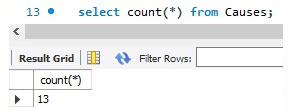
IDX\_FIRES\_SHAPE\_NODE table loaded in the database :

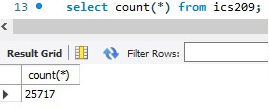


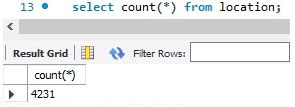
IDX\_FIRES\_SHAPE\_PARENT table loaded in the database :











SQL Queries

## **Query 1**

### A leading beverage company has announced a billion-dollar fund for removing debris from forests, rivers and mountains in the US. All states are interested. Which 2 states have the least chance to win a share of the fund?

### **Notes/Comments About SQL Query and Results (Include # of Rows in Result)**

### SELECT Count(f.fod\_ID) , f.state FROM fires f join NWCG\_UnitIDActive\_20170109 N ON f.NWCG\_REPORTING\_UNIT\_ID = N.UnitId where f.stat\_cause\_code = "5.0" and N.country = "US" group by f.state order by count(f.fod\_ID) ASC LIMIT 4

#4 number of records

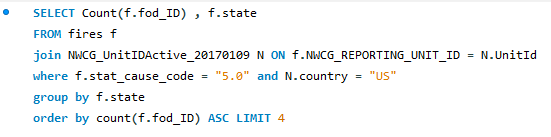
### **Translation**

### Obtain the count of FOD\_ID’s grouped on state from Fires table joined with NWCG\_UnitIDActive\_20170109 table on NWCG\_REPORTING\_UNIT\_ID where stat\_cause\_code is 5 and country is US. Apply limit to filter out the two states which has the least reported cases of debris.

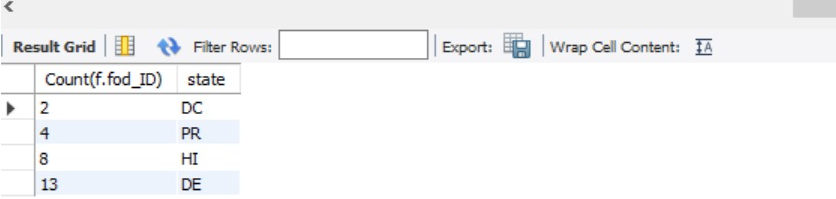
### **Cleanup**

Select count of FOD\_ID,state from fires JOIN NWCG\_UnitIDActive\_20170109 on NWCG\_REPORTING\_UNIT\_ID where f.stat\_cause\_code = "5.0" and N.country = "US group by state order by count(f.fod\_ID) ascending LIMIT 4

### **Screenshot of SQL Query**



### **Screenshot of Results**



## **Query 2**

### One of the reporting agencies has suggested that children be banned from its forests unless there is one adult for every 4 children in a group visiting a forest. Name top 5 forests where this would be the least appropriate.

### 

### **Notes/Comments About SQL Query and Results (Include # of Rows in Result)**

### SELECT Count(f.fod\_id), fs.source\_reporting\_unit\_name from Fires f join Statistics\_Cause s on f.stat\_cause\_code = s.stat\_cause\_code join Source\_Reporting\_Unit fs on f.Source\_Reporting\_Unit\_ID = fs.Source\_Reporting\_Unit\_ID where s.stat\_cause\_descr = "Children" and fs.source\_reporting\_unit\_name like "%Forest" group by fs.source\_reporting\_unit\_name order by count(f.fod\_id) ASC LIMIT 5;

#5 number of records

### **Translation**

Select count of FOD\_ID and source\_reporting\_unit\_name from Fires table joined with Statistics\_Cause table on stat\_cause\_code and joined with Source\_Reporting\_Unit on Source\_Reporting\_Unit\_ID where stat\_cause\_descr is "Children" and source\_reporting\_unit\_name contains like "%Forest" group by source\_reporting\_unit\_name order by count(f.fod\_id) ascending LIMIT 5

**Clean up**

Select count of fod\_id and source\_reporting\_unit\_name from Fires JOIN Statistics\_Cause ON stat\_cause\_code JOIN Source\_Reporting\_Unit ON Source\_Reporting\_Unit\_ID where stat\_cause\_descr="Children" and source\_reporting\_unit\_name like "%Forest" group by fs. source\_reporting\_unit\_name order by count(f.fod\_id) ASC LIMIT 5

### **Screenshot of SQL Query**

### 

### **Screenshot of Results**

## 

## **Query 3**

One advocacy group says human actions and nature are equally to blame for most wildfires. Write a query that can help determine the truth of this statement.

### **Notes/Comments About SQL Query and Results (Include # of Rows in Result)**

select "Human" as "Reason", count(f.fod\_id) as Count from fires f join Statistics\_Cause s on f.STAT\_CAUSE\_CODE = s.stat\_cause\_code where s.Stat\_Cause\_Desc in ("Fireworks", "Powerline", "Structure", "Equipment Use", "Smoking", "Campfire", "Debris Burning", "Railroad", "Arson", "Children")

union

select "Nature" as "Reason", count(f.fod\_id) as Count from fires f join Statistics\_Cause s on f.STAT\_CAUSE\_CODE = s.stat\_cause\_code where s.Stat\_Cause\_Desc in ("Lightning")

### **Translation**

Select "Human" as "Reason" and obtain count of fod\_id as Count from Fires table joined with Statistics\_Cause on stat\_cause\_code where Stat\_Cause\_Desc contains values in ("Fireworks", "Powerline", "Structure", "Equipment Use", "Smoking", "Campfire", "Debris Burning", "Railroad", "Arson", "Children").

Select "Nature" as "Reason" and obtain count of fod\_id as Count from Fires table joined with Statistics\_Cause on stat\_cause\_code where Stat\_Cause\_Desc contains values in ("Lightning"). Union both the queries for combined output.

**Clean up**

Select "Human" as "Reason" and count of f.fod\_id as Count from fires JOIN Statistics\_Cause on stat\_cause\_code where Stat\_Cause\_Desc in ("Fireworks", "Powerline", "Structure", "Equipment Use", "Smoking", "Campfire", "Debris Burning", "Railroad", "Arson", "Children")

Union

select "Nature" as "Reason" and count of f.fod\_id as Count from fires f JOIN Stat\_Cause\_Desc s on f.STAT\_CAUSE\_CODE = s.stat\_cause\_code where s.STAT\_CAUSE\_DESCR in ("Lightning")

### **Screenshot of SQL Query**

### **Screenshot of Results**

## 

## **Query 4**

### 

### Which state had fires only in the second half of the calendar years?

### 

### **Notes/Comments About SQL Query and Results (Include # of Rows in Result)**

SELECT distinct state from fires where state not in (select distinct state from fires where month(discovery\_date) <= 6)

#0 number of rows

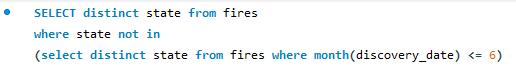
### **Translation**

Select all distinct state from Fires table which are not in the list of distinct states where month of discovery\_date is less than equal to 6

**Clean up**

Select distinct(state) from Fires table where state not in (select distinct state from fires where month of discovery\_date <= 6

### **Screenshot of SQL Query**



### **Screenshot of Results**

No rows found

## **Query 5**

What were the forests that had only one fire that lasted more than two days?

### **Notes/Comments About SQL Query and Results (Include # of Rows in Result)**

SELECT fs.source\_reporting\_unit\_name, Count(f.fod\_id) as "Count", (f.CONT\_DATE - f.DISCOVERY\_DATE) as "Duration" FROM fires f join Source\_Reporting\_Unit fs on f. Source\_Reporting\_Unit\_ID= fs. Source\_Reporting\_Unit\_ID where fs.source\_reporting\_unit\_name like "%Forest" and (f.CONT\_DATE - f.DISCOVERY\_DATE > 2) group by fs.source\_reporting\_unit\_name having Count(f.fod\_id) = 1

#1 number of row

### **Translation**

Select all source\_reporting\_unit\_name, Count all fod\_id’s and calculate duration by subtracting cont\_date with discovery\_date from fires table joined with Source\_Reporting\_Unit table on Source\_Reporting\_Unit\_ID where source\_reporting\_unit\_name contains "%Forest" and cont\_date minus discovery\_date is less than 2 group by source\_reporting\_unit\_name which is having Count of fod\_id equal to 1

**Clean up**

SELECT source\_reporting\_unit\_name, Count of fod\_id as "Count", CONT\_DATE minus DISCOVERY\_DATE as "Duration" FROM fires join Source\_Reporting\_Unit on Source\_Reporting\_Unit\_ID where source\_reporting\_unit\_name like "%Forest" and (CONT\_DATE minus DISCOVERY\_DATE > 2) group by source\_reporting\_unit\_name having Count(fod\_id) = 1

### **Screenshot of SQL Query**

### 

### **Screenshot of Results**

### **Query 6**

What are the top two unit types that reported wildfires in each state in the US?

### **Notes/Comments About SQL Query and Results (Include # of Rows in Result)**

select state,agency,cnt from ( select state,agency,cnt,ROW\_NUMBER()OVER(PARTITION BY state ORDER BY cnt DESC) AS Ranking from (select f.state, n.Agency, Count(1) AS cnt from fires f join NWCG\_UnitIDActive\_20170109 n on f.NWCG\_REPORTING\_UNIT\_ID = n.NWCG\_REPORTING\_UNIT\_ID where n.Agency is not Null group by f.state, n.Agency )a group by state,agency,cnt )k where k.Ranking<=2;

#103 rows returned

### **Translation**

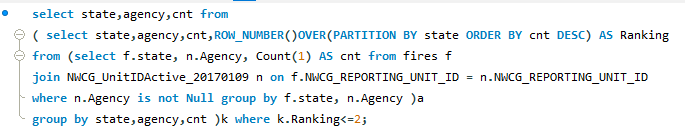
Select all states and agencies and the count from the subquery where we are filtering the state, agency, ranking by calculating row\_number by performing partition by state and order by cnt and sort descending grouped by state, agency and cnt where ranking is less that equal to 2

Obtain cnt column from fires table joined on NWCG\_UnitIDActive\_20170109 table on NWCG\_REPORTING\_UNIT\_ID where agency is not null grouped by state and agency.

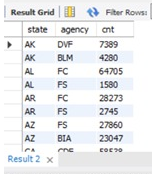
**Clean up**

select state,agency,cnt from ( select state,agency,cnt,ROW\_NUMBER()OVER(PARTITION BY state ORDER BY cnt DESC) AS Ranking from (select state, Agency, Count(1) AS cnt from fires f join NWCG\_UnitIDActive\_20170109 n on f.NWCG\_REPORTING\_UNIT\_ID = n.NWCG\_REPORTING\_UNIT\_ID where n.Agency is not Null group by f.state, n.Agency )a group by state,agency,cnt )k where k.Ranking is less that equal to 2;

### **Screenshot of SQL Query**



### **Screenshot of Results**



Data Review for MongoDB

## Assumptions/Notes About Data Collections, Attributes and Relationships between Collections

All collections in MongoDB are uploaded using JSON files which are exported from Mysql Normalized tables.

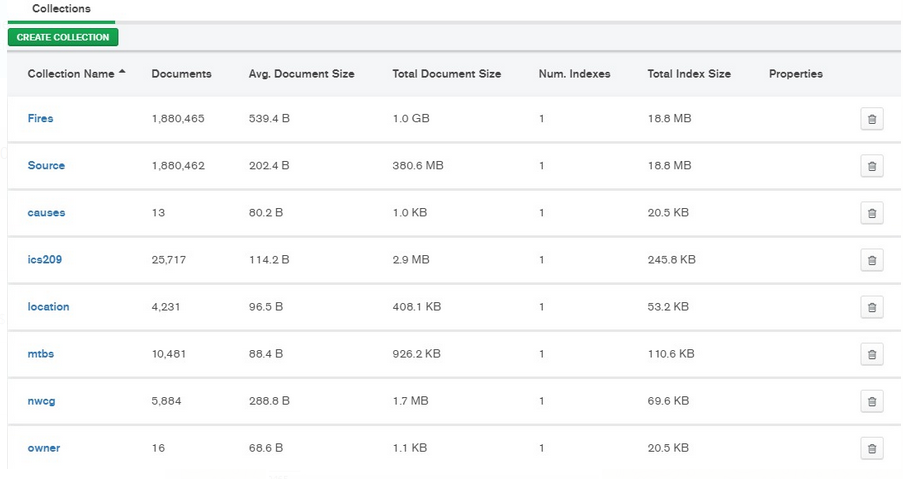
Physical Mongo Database

## Assumptions/Notes About Data Set

## Screen shot of Physical Database objects (Database, Collections and Attributes)

## Data in the Database:

Data is uploaded in MongoDB from json files and below collections are created -



|  |  |  |
| --- | --- | --- |
| **Collection Name** | **Relationships With Other Collections (if any)** | **# of Documents in Collection** |
| Fires |  | 1,880,465 |
| Source |  | 1,880,462 |
| causes |  | 13 |
| ics209 |  | 25,717 |
| location |  | 4,231 |
| mtbs |  | 10,481 |
| nwcg |  | 5,884 |
| owner |  | 16 |

MongoDB Queries/Code

## Query 1

### Question

A leading beverage company has announced a billion-dollar fund for removing debris from forests, rivers and mountains in the US. All states are interested. Which 2 states have the least chance to win a share of the fund?

db.fires.aggregate([

{

$lookup: {

from: "NWCG",

localField: "State",

foreignField: "State",

as: "fromStates"

}

},

{

$replaceRoot: { newRoot: { $mergeObjects: [ { $arrayElemAt: [ "$fromStates", 0 ] }, "$$ROOT" ] } }

},

{ $match : { $and: [ { "Country": "US" }, { "STAT\_CAUSE\_CODE": "5.0" } ] } },

{ $group : {\_id : {State: "$State"}, Fcount: { $sum: 1}}},

{$sort :{Fcount:1}},

{$limit:4}

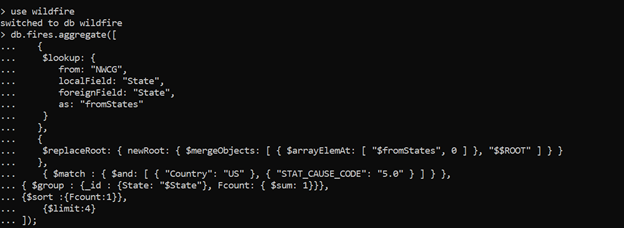
]);

### Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)

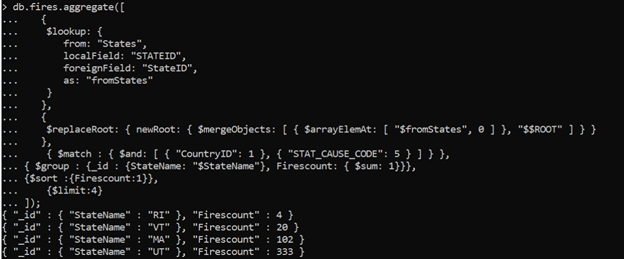
In this question, STAT\_CAUSE\_CODE for debris is 5.0 and we need to check for country USA. We checked the count state wise and found 2 states which have the least chance to win a share of the fund. Here we have shown 4 rows since the least 2 states i.e. DC and PR.

### Translation

### Screen Shot of MongoDB Query/Code and Results



Output Snapshot:



# of documents returned: 4

## Query 2

### Question

One of the reporting agencies has suggested that children be banned from its forests unless there is one adult for every 4 children in a group visiting a forest. Name the top 5 forests where this would be the least appropriate.

### Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)

db.fires.aggregate([

{

$lookup: {

from: "causes",

localField: "STAT\_CAUSE\_CODE",

foreignField: "STAT\_CAUSE\_CODE",

as: "fromCause"

}

},

{

$replaceRoot: { newRoot: { $mergeObjects: [ { $arrayElemAt: [ "$fromCause", 0 ] }, "$$ROOT" ] } }

},

{

$lookup: {

from: "Source",

localField: "FPA\_ID",

foreignField: "FPA\_ID",

as: "fromSource"

}

},

{

$replaceRoot: { newRoot: { $mergeObjects: [ { $arrayElemAt: [ "$fromSource", 0 ] }, "$$ROOT" ] } }

},

{ $match : { $and: [ { "STAT\_CAUSE\_DESCR": "Children" }, { "SOURCE\_REPORTING\_UNIT\_NAME": /$Forest$/ } ] } },

{ $group : {\_id : {SOURCE\_REPORTING\_UNIT\_NAME: "$SOURCE\_REPORTING\_UNIT\_NAME"}, Fcount: { $sum: 1}}},

{$sort :{Fcount:1}},

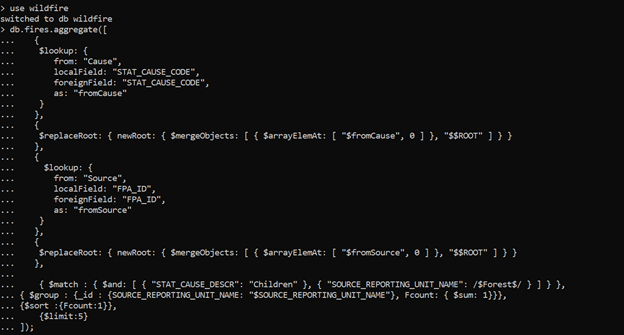
{$limit:5}

]);

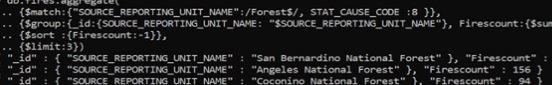
### Translation

In this question, we checked STAT\_CAUSE\_DESCR for Children and we need to check for country USA. We checked the count state wise and found 5 states which have SOURCE\_REPORTING\_UNIT\_NAME like forests where this would be the least appropriate.

### Screen Shot of MongoDB Query/Code and Results



Output:



# of documents returned: 5

## Query 3

### Question

One advocacy group says human actions and nature are equally to blame for most wildfires. Write a query that can help determine the truth of this statement.

### Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)

db.fires.aggregate([

{

$lookup: {

from: "causes",

localField: "STAT\_CAUSE\_CODE",

foreignField: "STAT\_CAUSE\_CODE",

as: "fromCause"

}},

{

$replaceRoot: { newRoot: { $mergeObjects: [ { $arrayElemAt: [ "$fromCause", 0 ] }, "$$ROOT" ] } }

},

{

$match: { "STAT\_CAUSE\_DESCR": {$in: ["Fireworks","Powerline","Structure", "Equipment Use", "Smoking", "Campfire", "Debris Burning", "Railroad", "Arson", "Children"]}}

},

{ $group : {\_id : {STAT\_CAUSE\_DESCR: "$STAT\_CAUSE\_DESCR"}, Fcount: { $sum: 1}}},

]).pretty();

db.fires.aggregate([

{

$lookup: {

from: "Cause",

localField: "STAT\_CAUSE\_CODE",

foreignField: "STAT\_CAUSE\_CODE",

as: "fromCause"

}},

{

$replaceRoot: { newRoot: { $mergeObjects: [ { $arrayElemAt: [ "$fromCause", 0 ] }, "$$ROOT" ] } }

},

{

$match: { "STAT\_CAUSE\_DESCR": ["Lightning"]}}

},

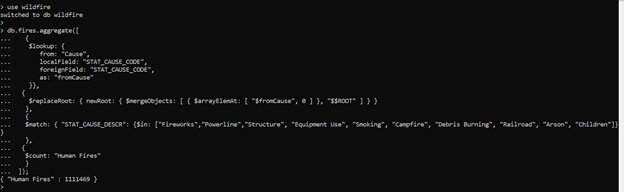
{ $group : {\_id : {STAT\_CAUSE\_DESCR: "$STAT\_CAUSE\_DESCR"}, Fcount: { $sum: 1}}},

]).pretty();

### Translation

In this scenario, here we found first what are STAT\_CAUSE\_DESCR which are caused by nature i.e. Lightning and STAT\_CAUSE\_DESCR which are caused by human i.e. Fireworks, Powerline, Structure, Equipment Use, Smoking, Campfire, Debris Burning, Railroad, Arson and Children. Then we compared number of fires caused by both. And then after comparing the count we found that human actions are more to blame for most wildfires as compared to nature.

### Screen Shot of MongoDB Query/Code and Results





# of documents returned: For Human Fires: 1111469 , For Natural fires: 278468

## Query 4

### Question

How many wildfires were reported by at least two units/agencies?

### Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)

db.fires.aggregate([

{$project:{FIRE\_NAME:1,SOURCE\_REPORTING\_UNIT:1}},

{$group: {\_id:"$FIRE\_NAME",No\_of\_units\_reported: {$sum: "$SOURCE\_REPORTING\_UNIT"}}},

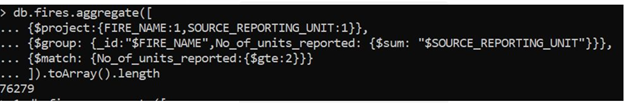
{$match: {No\_of\_units\_reported:{$gte:2}}}

]).toArray().length;

### Translation

In this scenario, we checked that No\_of\_units\_reported which are greater than 2 for each fire and we grouped it by FIRE\_NAME and we found that the number of wildfires reported by at least two agencies.

### Screen Shot of MongoDB Query/Code and Results



# of documents returned: 76279

## Query 5

### Question

Which state had fires only in the second half of the calendar years?

### Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)

### Translation

In this scenario, we are checking those states which have discovery day of year is less than 180 days.

so in this case we found all 52 states are there where fires occurred in both the first half and second half of the year so there is no row returned for states in only the second half of the year.

### Screen Shot of MongoDB Query/Code and Results

db.fires.aggregate([

{ "$match": {

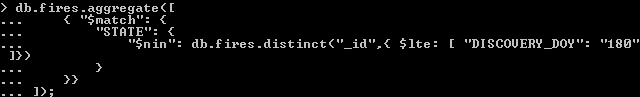
"STATE": {

"$nin": db.fires.distinct("\_id",{ $lte: [ "DISCOVERY\_DOY": "180" ]})

}

}}

]);



# of documents returned: 0

## Query 6

### Question

What were the forests that had only one fire that lasted more than two days?

### Notes/Comments About MongoDB Query/Code and Results (Include # of Documents in Result)

db.fires.aggregate([

{$match : { $and: [ { "SOURCE\_REPORTING\_UNIT\_NAME":/Forest$/}, { "CONT\_DOY": { $not: / / } }, { "CONT\_DOY-DISCOVERY\_DOY": { $not: /0/ } }, { "CONT\_DOY-DISCOVERY\_DOY": { $not: /1/ } },

{"CONT\_DOY-DISCOVERY\_DOY": { $not: /2/ } },{"CONT\_DOY-DISCOVERY\_DOY": { $not: /-363/ } },{"CONT\_DOY-DISCOVERY\_DOY": { $not: /-364/ } } ] } },

{ $group: {\_id: {SOURCE\_REPORTING\_UNIT\_NAME: "$SOURCE\_REPORTING\_UNIT\_NAME"}, Firescount: { $sum: 1}}},

{ $match: { Firescount: { $gte: 2} } },

{$sort :{Firescount:-1}}

]);

### Translation

In this scenario, we checked that SOURCE\_REPORTING\_UNIT\_NAME like Forest and where CONT\_DOY is not NULL, and difference between CONT\_DOY and DISCOVERY\_DOY should not be equal to 0, 1, 2, -363 and -364 and grouped it by SOURCE\_REPORTING\_UNIT\_NAME and such Firescount should be greater than equal to one.

### Screen Shot of MongoDB Query/Code and Results

