

#MDBlocal



PIPELINE POWER

DOING MORE WITH MONGODB AGGREGATION FRAMEWORK

ASYA KAMSKY

LEAD KNOW-IT-ALL

MONGODB, INC

@asya999 #askAsya

APPLICATIONS & DATA



find()

&

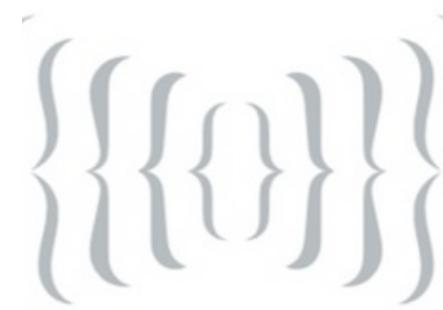
aggregate()



OPTIONS FOR ANALYTICS



pre-aggregate



aggregate
in MongoDB



aggregate elsewhere



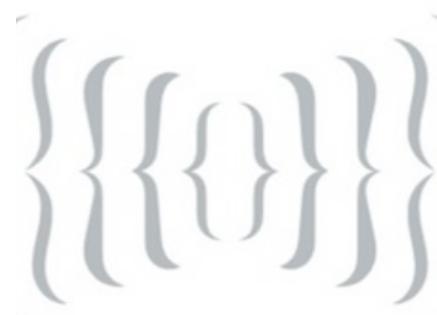
MONGODB ON TOUR
SAN FRANCISCO

#MDBTOUR

OPTIONS FOR ANALYTICS



pre-aggregate

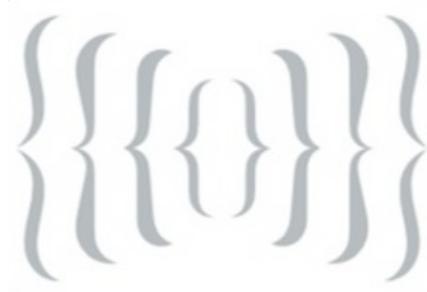


aggregate
in MongoDB



aggregate elsewhere

OPTIONS FOR ANALYTICS



aggregate
in MongoDB



aggregate elsewhere

OPTIONS FOR ANALYTICS



Show only

Available nearby

New items

Price

Up to \$60

\$60 - \$150

\$150 - \$450

\$450 - \$1,000

Over \$1,000

\$ 10

Category

Saw Blades

Drill, Driver & Screwdriver

Accessories

Brand

bailleigh

bosch

dewalt

festool

milwaukee hand trucks

More

Seller

Acme Tools

eBay

HardwareAndTools.co

Home Depot

Neobits.com

More

SKIL 3550-01 7 In. Wet Tile Saw with HydroLock System
\$999.99 from 25+ stores
★★★★★ 43 product reviews
SKIL
SKIL has the unique tools you need to deliver exceptional results. Skilsaws not only build projects, they build confidence.

17.25" SPECIAL 650 DI Semi Automatic Band Saw, Roller Table Support
\$38,995.00 from Circular Cold Saw Blades
17.25" SPECIAL, 650 DI Semi Automatic Band Saw with Roller Table Support

Baileigh CNC Table Saw - STS-14120-CNC
\$24,245.00 from 5+ stores
Baileigh - Cabinet - 14 inch - 7.5 horsepower
The STS-14120-CNC table saw from Baileigh Industrial is based from our other sliding table saw designs, but adds a CNC twist. The CNC function ...

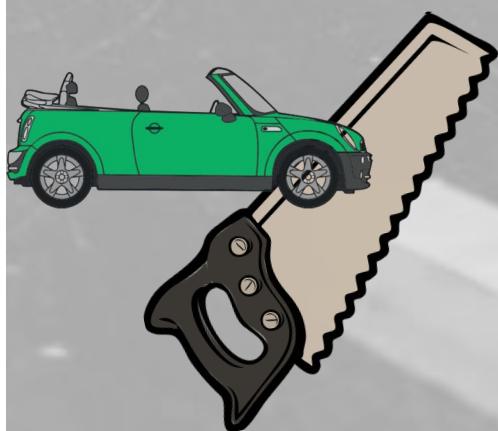
Safety Speed Cut Safety Speed ES5210M Panel Saw
\$19,949.00 from 3 stores
The ES5210M is the only panel saw in our lineup that features a carriage that traverses the full length of the machine frame. The traveling beam ...

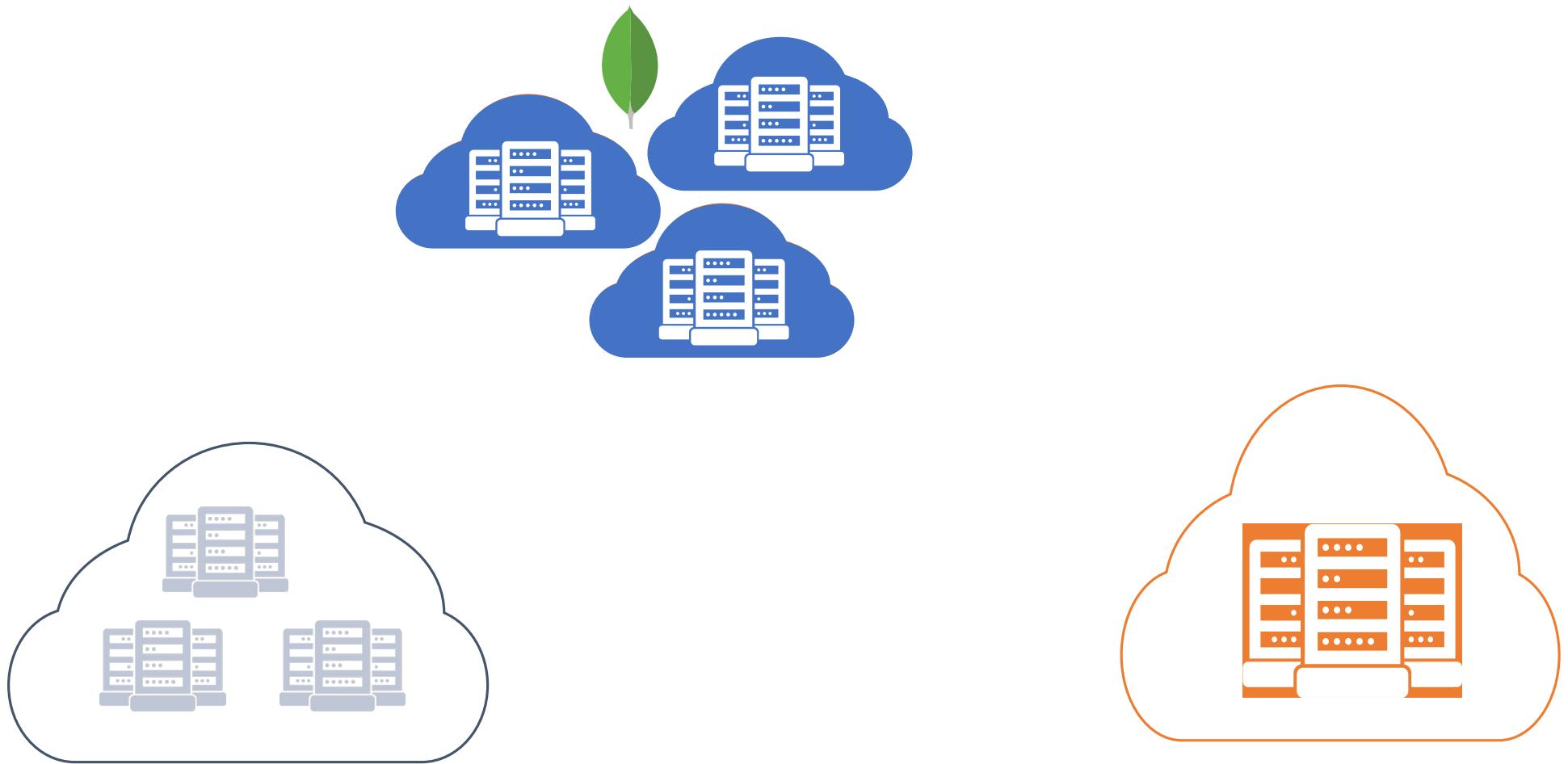
Hobart 6614-1 126" Vertical Meat Saw - 3 HP, 200-230/60/3V
\$8,481.55 from 4 stores
For precise cuts, simple operation and outstanding durability, look no further than this Hobart 6614-1 126" vertical meat saw! The 5/8" x 126 ...
Other size options: 142" (\$9,941)

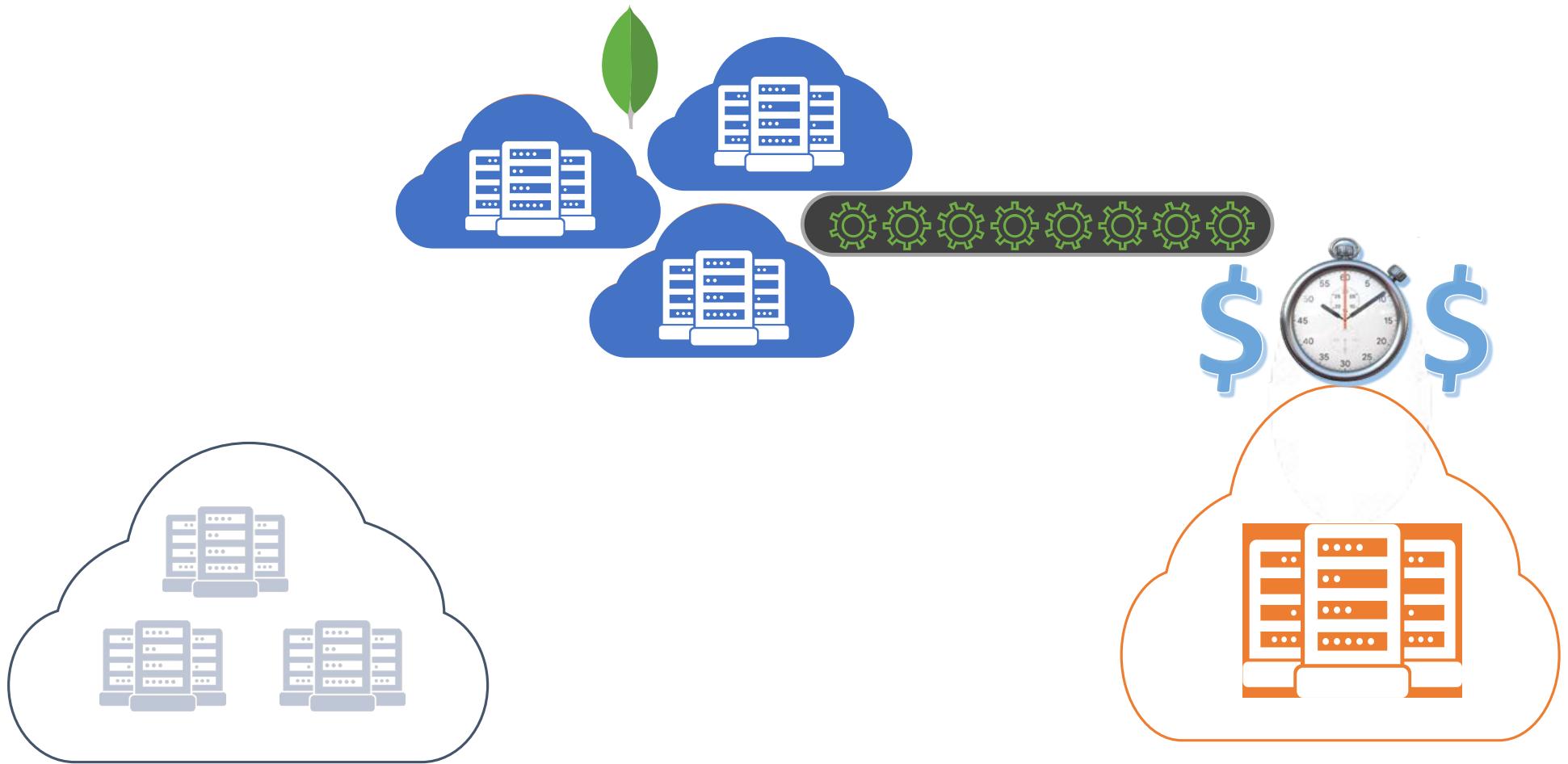
Baileigh 21 IN x 60 IN Combination Woodworking Machine MF-3005
\$8,295.00 from 3 stores
Baileigh - 12 inch
The MF-3005 from Baileigh Industrial takes care of that problem. This 5 in 1 combination woodworking machine combines all of the most popular ...

Powermatic PM2000 Table Saw with Accu-Fence System, 10", 3 HP
\$7,179.98 from 20+ stores
Powermatic - Cabinet - 10 inch - 3 horsepower
The PM2000 table saw has an array of features and massive design. It has the heaviest cast iron base that eliminates table vibration. It has a ...

Skid Steer Tree Saw Attachment Fixed Boom High Flow 20-30 gpm
\$6,995.00 from Spartan Equipment







OPTIONS FOR ANALYTICS



ANALYTICS = AGGREGATION



PIPELINE



PIPELINE

*nix command line pipe

```
ps ax | grep mongod | head 1
```



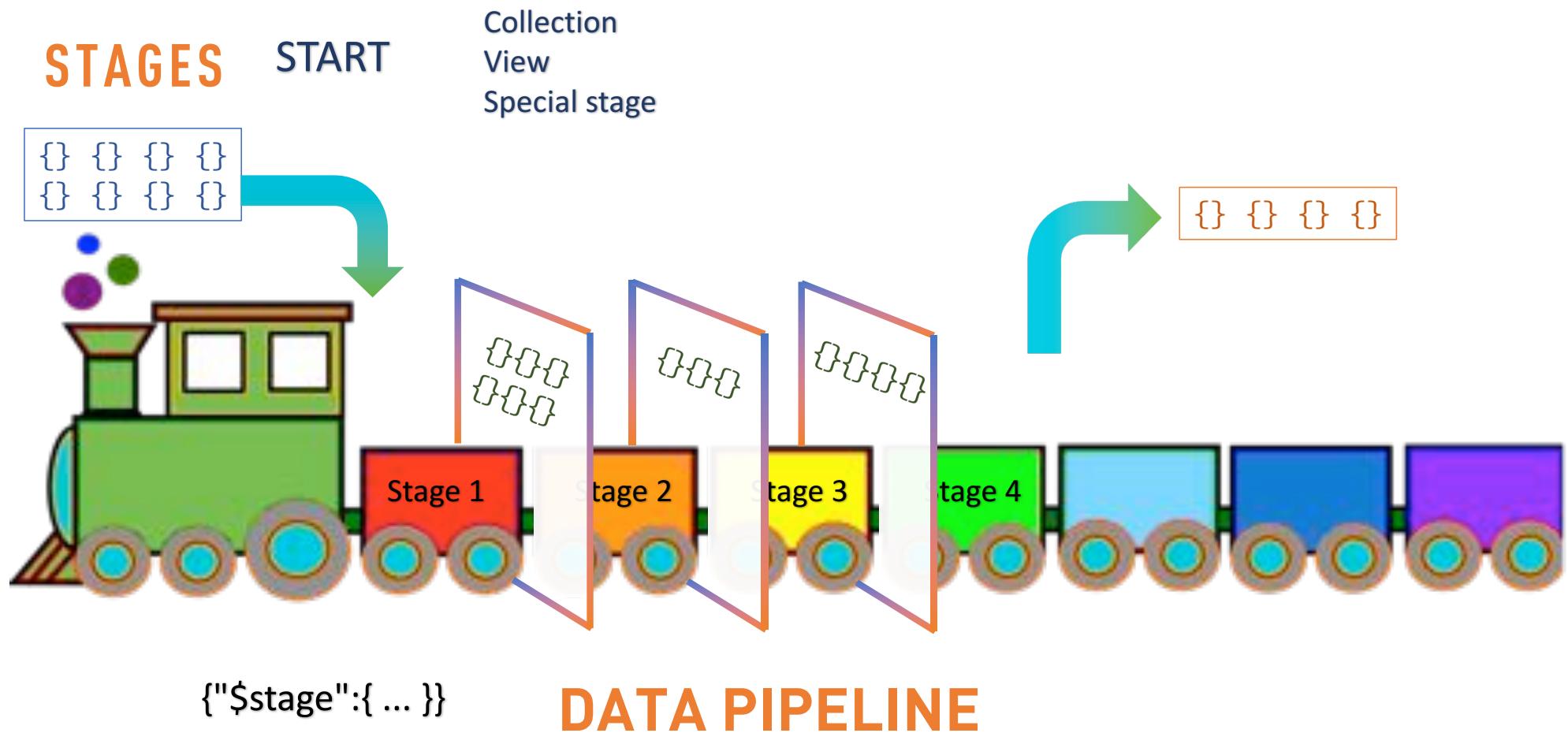
PIPELINE

MongoDB document pipeline

`$match | $group | $sort`



Input stream {} {} {} {} → Result {} {} ...



```
{
  title: "The Great Gatsby",
  language: "English",
  subjects: [
    "Long Island",
    "New York",
    "1920s" ],
  {
    title: "War and Peace",
    language: "Russian",
    subjects: [
      "Russia",
      "War of 1812",
      "Napoleon" ],
  {
    title: "Open City",
    language: "English",
    subjects: [
      "New York",
      "Harlem" ] },
  {
    title: "The Great Gatsby",
    language: "English",
    subjects: [
      "Long Island",
      "New York",
      "1920s" ],
  {
    title: "Open City",
    language: "English",
    subjects: [
      "New York"
      "Harlem" ] },
  {
    title: "The Great Gatsby",
    language: "English",
    subjects: [
      "Long Island" ]
  {
    title: "The Great Gatsby",
    language: "English",
    subjects: [
      "New York" ]
  {
    title: "The Great Gatsby",
    language: "English",
    subjects: [
      "1920s" ]
  {
    title: "Open City",
    language: "English",
    subjects: [
      "New York" ]
  {
    title: "Open City",
    language: "English",
    subjects: [
      "Harlem" ] }
```

```
{
  title: "The Great Gatsby",
  language: "English",
  subjects: [
    "Long Island",
    "New York",
    "1920s" ],
  {
    title: "Open City",
    language: "English",
    subjects: [
      "New York"
      "Harlem" ] }
```

```
{
  title: "The Great Gatsby",
  language: "English",
  subjects: [
    "Long Island" ]
  {
    title: "The Great Gatsby",
    language: "English",
    subjects: [
      "New York" ]
  {
    title: "The Great Gatsby",
    language: "English",
    subjects: [
      "1920s" ]
  {
    title: "Open City",
    language: "English",
    subjects: [
      "New York" ]
  {
    title: "Open City",
    language: "English",
    subjects: [
      "Harlem" ] }
```

```
{
  _id: "Long Island",
  count: 1 },
  {
  _id: "New York",
  count: 2 },
  {
  _id: "1920s",
  count: 1 },
  {
  _id: "Harlem",
  count: 1 },
```

```
{
  _id: "New York",
  count: 2 },
  {
  _id: "Long Island",
  count: 1 },
  {
  _id: "1920s",
  count: 1 },
  {
  _id: "Harlem",
  count: 1 },
```

```

  {"$unwind": "$subjects"}           {"$sort": {"count": -1}}          {"$limit": 3}
{"$match": {"language": "English"} }   {"$group": {"_id": "$subjects", "count": {"$sum": 1}}} {"$project": ...}
```

\$match

\$unwind

\$group

\$sort

\$limit

\$project

\$skip

STAGES

A circular diagram illustrating the stages of a MongoDB pipeline. The stages are represented by green text labels arranged in a circle, connected by arrows pointing clockwise. The stages are: \$sample, \$replaceRoot, \$out, \$skip, \$match, \$limit, \$sort, \$group, \$unwind, \$project, \$lookup, \$joinWith, \$count, and \$redact.

\$sample
\$replaceRoot
\$out
\$skip
\$match
\$limit
\$sort
\$group
\$unwind
\$project
\$joinWith
\$lookup
\$count
\$redact

\$match
\$geoNear

\$limit

\$sample

\$skip

\$project

\$addFields

\$replaceRoot
\$lookup

\$graphLookup

\$unwind

\$sortByCount

\$sample

\$replaceRoot

\$out

\$skip

\$match

\$limit

\$addFields

\$sort

\$group

\$count

\$indexStats

\$redact

\$bucket

\$geoNear

\$graphLookup

\$facet

\$collStats

\$bucketAuto

\$indexStats

\$redact

\$out

\$indexStats

\$collStats

\$group

\$count

\$sortByCount

\$bucket

\$bucketAuto

\$facet

\$redact

\$indexStats

\$out

\$collStats

\$match
\$geoNear

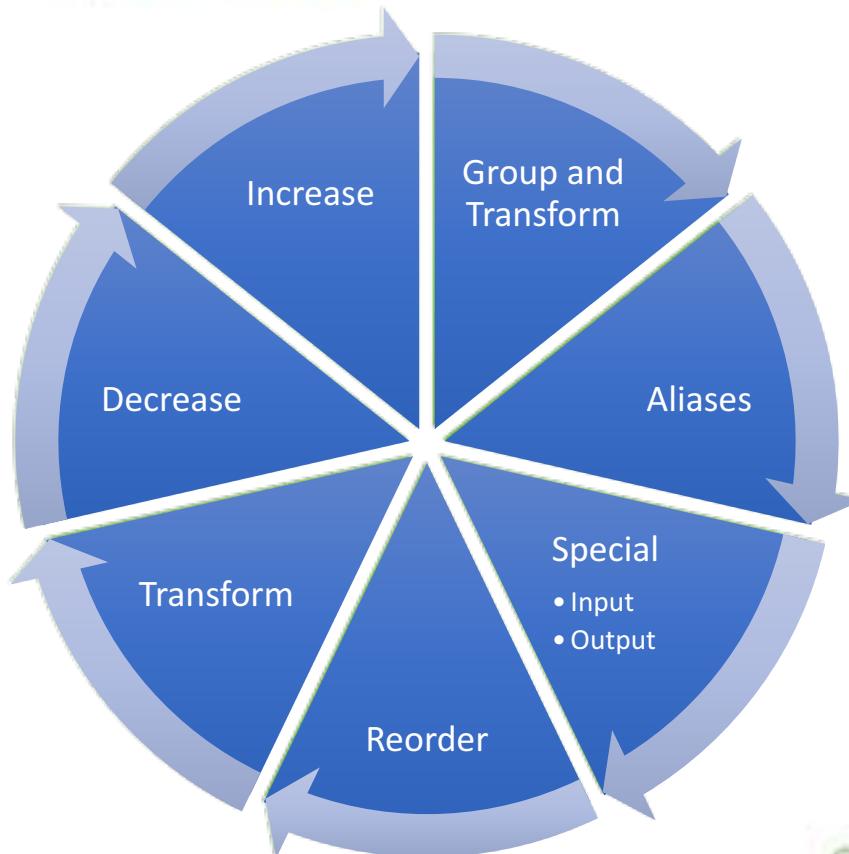
\$limit
\$sample
\$skip

\$project
\$addFields

\$replaceRoot
\$lookup

\$graphLookup

\$unwind



\$sort

\$group

\$count
\$sortByCount

\$bucket
\$bucketAuto

Sfacet \$redact

SindexStats

ScollStats

\$out

```
{
  title: "The Great Gatsby",
  language: "English",
  subjects: [
    "Long Island",
    "New York",
    "1920s" ],
  {
    title: "War and Peace",
    language: "Russian",
    subjects: [
      "Russia",
      "War of 1812",
      "Napoleon" ],
  {
    title: "Open City",
    language: "English",
    subjects: [
      "New York",
      "Harlem" ] },
  {
    title: "The Great Gatsby",
    language: "English",
    subjects: [
      "Long Island",
      "New York",
      "1920s" ],
  {
    title: "Open City",
    language: "English",
    subjects: [
      "New York"
      "Harlem" ] },
  {
    title: "The Great Gatsby",
    language: "English",
    subjects: [
      "Long Island" ]
  {
    title: "The Great Gatsby",
    language: "English",
    subjects: [
      "New York" ]
  {
    title: "The Great Gatsby",
    language: "English",
    subjects: [
      "1920s" ]
  {
    title: "Open City",
    language: "English",
    subjects: [
      "New York" ]
  {
    title: "Open City",
    language: "English",
    subjects: [
      "Harlem" ] }
```

```
{
  title: "The Great Gatsby",
  language: "English",
  subjects: [
    "Long Island",
    "New York",
    "1920s" ],
  {
    title: "Open City",
    language: "English",
    subjects: [
      "New York"
      "Harlem" ] }
```

```
{
  title: "The Great Gatsby",
  language: "English",
  subjects: [
    "Long Island" ]
  {
    title: "The Great Gatsby",
    language: "English",
    subjects: [
      "New York" ]
  {
    title: "The Great Gatsby",
    language: "English",
    subjects: [
      "1920s" ]
  {
    title: "Open City",
    language: "English",
    subjects: [
      "New York" ]
  {
    title: "Open City",
    language: "English",
    subjects: [
      "Harlem" ] }
```

```
{
  _id: "Long Island",
  count: 1 },
  {
  _id: "New York",
  count: 2 },
  {
  _id: "1920s",
  count: 1 },
  {
  _id: "Harlem",
  count: 1 },
```

```
{
  _id: "New York",
  count: 2 },
  {
  _id: "Long Island",
  count: 1 },
  {
  _id: "1920s",
  count: 1 },
  {
  _id: "Harlem",
  count: 1 },
```

```

  {"$unwind": "$subjects"}           {"$sort": {"count": -1}}          {"$limit": 3}
{"$match": {"language": "English"} }   {"$group": {"_id": "$subjects", "count": {"$sum": 1}}} {"$project": ...}
```

\$match

\$unwind

\$group

\$sort

\$limit

\$project

\$skip

LET ME EXPLAIN...

```
db.books.aggregate([
  {$match:{"language":"English"}},
  {$unwind:"$subjects"},
  {$group:{_id:"$subjects",count:{$sum:1}}},
  {$sort:{count:-1}},
  {$limit:3}
],{explain:true})
```

```
{"stages" : [
  {"$cursor" : {"query" : { "language" : "English"},
                "fields" : { "subjects" : 1,"_id" : 0} ...},
  {"$unwind" : {"path" : "$subjects"}},
  {"$group" : {"_id" : "$subjects","count" : {"$sum" : {"$const" : 1}}}},
  {"$sort" : {
    "sortKey" : {"count" : -1},
    "limit" : NumberLong(3)
  }}
]} ] }
```

```
db.books.aggregate([
    {$match:{"language":"English"}},
    {$unwind:"$subjects"},
    {$group:{_id:"$subjects",count:{$sum:1}}},
    {$sort:{count:-1}},
    {$limit:3}
],{explain:true})

{
    "stages": [
        {
            "$cursor": {
                "query": { "language": "English" },
                "fields": { "subjects": 1, "_id": 0 } ...
            }
        },
        {
            "$unwind": { "path": "$subjects" },
            {
                "$group": { "_id": "$subjects", "count": { "$sum": { "$const": 1 } } },
                {
                    "$sort": {
                        "sortKey": { "count": -1 },
                        "limit": NumberLong(3)
                    }
                }
            }
        }
    ]
}
```



```
db.books.aggregate([
    {$unwind:"$subjects"},
    {$group:{_id:"$subjects",count:{$sum:1}}},
    {$sort:{count:-1}},
    {$limit:3}
],{explain:true})

{"stages": [
    {"$cursor" : {"query" : { },
                  "fields" : { "subjects" : 1,"_id" : 0} ...
                },
    {"$unwind" : {"path" : "$subjects"}},
    {"$group" : {"_id" : "$subjects","count" : {"$sum" : {"$const" : 1}} } },
    {"$sort" : {
        "sortKey" : {"count" : -1},
        "limit" : NumberLong(3)
      }
  }
]} ] }
```

```
db.books.aggregate([
  {$unwind:"$subjects"},

  {$group:{_id:"$subjects",count:{$sum:1}}},
  {$sort:{count:-1}},
  {$limit:3}
],{explain:true})
```

```
db.books.aggregate([
  {$unwind:"$subjects"},
  {$match:{language:"English"}},
  {$group:{_id:"$subjects",count:{$sum:1}}},
  {$sort:{count:-1}},
  {$limit:3}
],{explain:true})

{
  "stages": [
    {
      "$cursor": {
        "query": { "language": "English" },
        "fields": { "subjects": 1, "_id": 0 } ...
      }
    },
    {
      "$unwind": { "path": "$subjects" },
      "$group": { "_id": "$subjects", "count": { "$sum": { "$const": 1 } } },
      "$sort": {
        "sortKey": { "count": -1 },
        "limit": NumberLong(3)
      }
    }
  ]
}
```

```
db.books.aggregate([
  {$unwind:"$subjects"},  

  {$match:{language:"English", "subjects":/^ABC/}},
  {$group:{_id:"$subjects",count:{$sum:1}}},
  {$sort:{count:-1}},
  {$limit:3}
],{explain:true})  
  
{"stages" : [  
    {"$cursor" : {"query" : { "language" : "English"},  
                 "fields" : { "subjects" : 1, "id" : 0} ...  
               },  
    {"$unwind" : {"path" : "$subjects"}},  
    {"$match" : {"subjects" : {"$regex" : "ABC"} }},  
    {"$group" : {"_id" : "$subjects", "count" : {"$sum" : {"$const" : 1}}}},  
    {"$sort" : {  
        "sortKey" : {"count" : -1},  
        "limit" : NumberLong(3)  
      }}  
] }
```

```
db.books.aggregate([
  {$unwind:"$subjects"},  

  {$match:{language:"English",subjects:/^ABC/}},
  {$group:{_id:$subjects,count:{$sum:1}}},
  {$sort:{count:-1}},
  {$limit:3}
],{explain:true})
```

```
{"stages": [  

  {"$cursor": {"query": { "language": "English"},  

    "fields": { "subjects": 1,"_id": 0} ...  

  },  

  {"$unwind": {"path": "$subjects"}},  

  {"$match": {"subjects": {"$regex": "ABC" }}}},  

  {"$group": {"_id": "$subjects","count": {"$sum": {"$const": 1}}}},  

  {"$sort": {  

    "sortKey": {"count": -1},  

    "limit": NumberLong(3)
  }}  

]} ] }
```

STREAMING VS BLOCKING

```
{ title: "The Great Gatsby",  
  language: "English",  
  subjects: [  
    "Long Island",  
    "New York",  
    "1920s" ] },  
{ title: "War and Peace",  
  language: "Russian",  
  subjects: [  
    "Russia",  
    "War of 1812",  
    "Napoleon" ] },  
{ title: "Open City",  
  language: "English",  
  subjects: [  
    "New York",  
    "Harlem" ] },
```

```
{title: "The Great Gatsby",  
language: "English",  
subjects: [  
  "Long Island",  
  "New York",  
  "1920s" ] },  
  
{title: "Open City",  
language: "English",  
subjects: [  
  "New York"  
  "Harlem" ] }
```

```
{title: "The Great Gatsby",  
language: "English".  
subjects: "Long Island"}  
  
{title: "The Great Gatsby",  
language: "English",  
subjects: "New York"}  
  
{title: "The Great Gatsby",  
language: "English",  
subjects: "1920s"}  
  
{title: "Open City",  
language: "English",  
subjects: "New York"}  
  
{title: "Open City",  
language: "English",  
subjects: "Harlem"}
```

```
{ _id: "Long Island",  
count: 1 },  
  
{ _id: "New York",  
count: 1 },  
  
{ _id: "1920s",  
count: 1 },  
  
{ _id: "Harlem",  
count: 1 },
```

```
{ _id: "New York",  
count: 2 },  
  
{ _id:"Long Island",  
count: 1 },  
  
{ _id: "1920s",  
count: 1 },  
  
{ _id: "Harlem",  
count: 1 },
```

`{"$unwind": "$subjects"}`

`{"$sort": {"count": -1} {"$limit": 3}}`

```
{"$match": {"language": "English"} }  {"$group": {"_id": "$subjects", "count": {"$sum": 1}} } {"$project": ...}
```

\$match

\$unwind

\$group

\$sort

\$limit

\$project

\$skip

\$match
\$geoNear

\$limit
\$sample

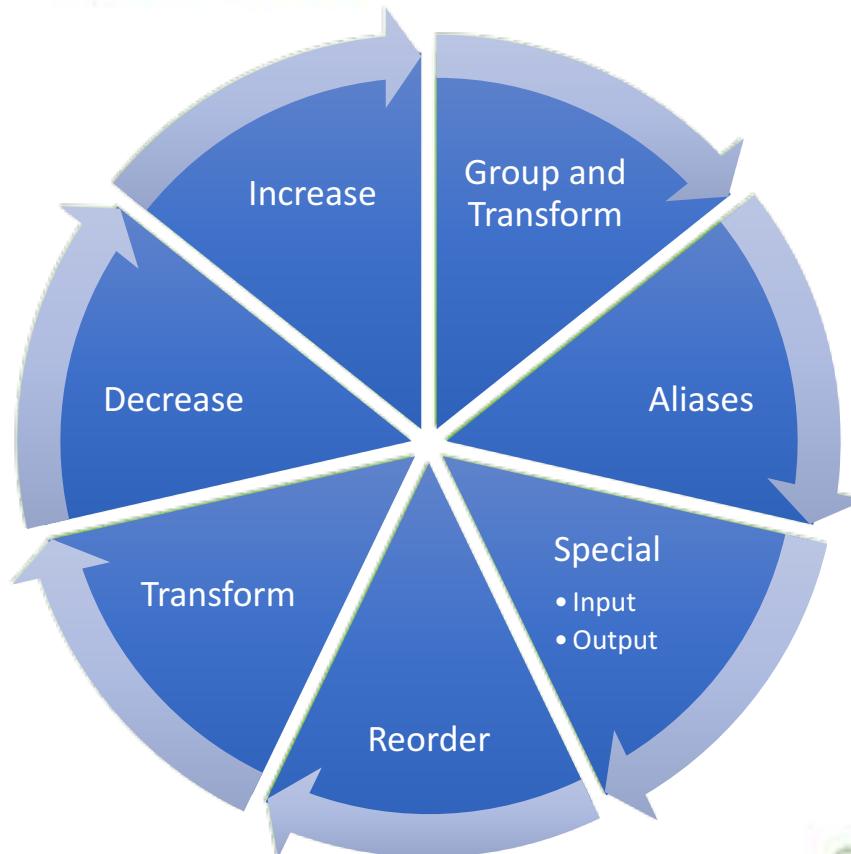
\$skip

\$project
\$addFields

\$replaceRoot
\$lookup

\$graphLookup

\$unwind



\$sort

\$group

\$count
\$sortByCount

\$bucket
\$bucketAuto

\$facet \$redact

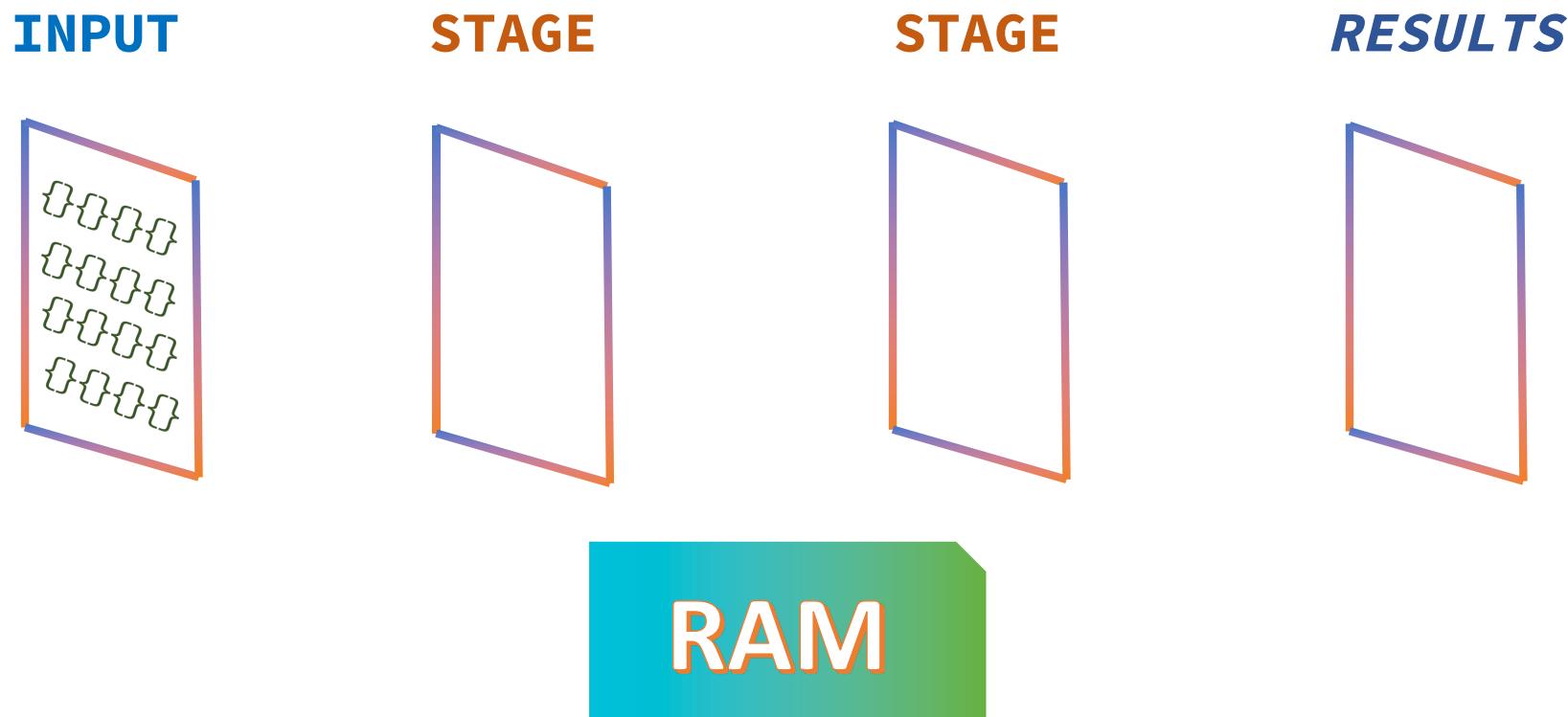
SindexStats

ScollStats

\$out

STREAMING VS BLOCKING RESOURCE USE

STREAMING RESOURCE USE

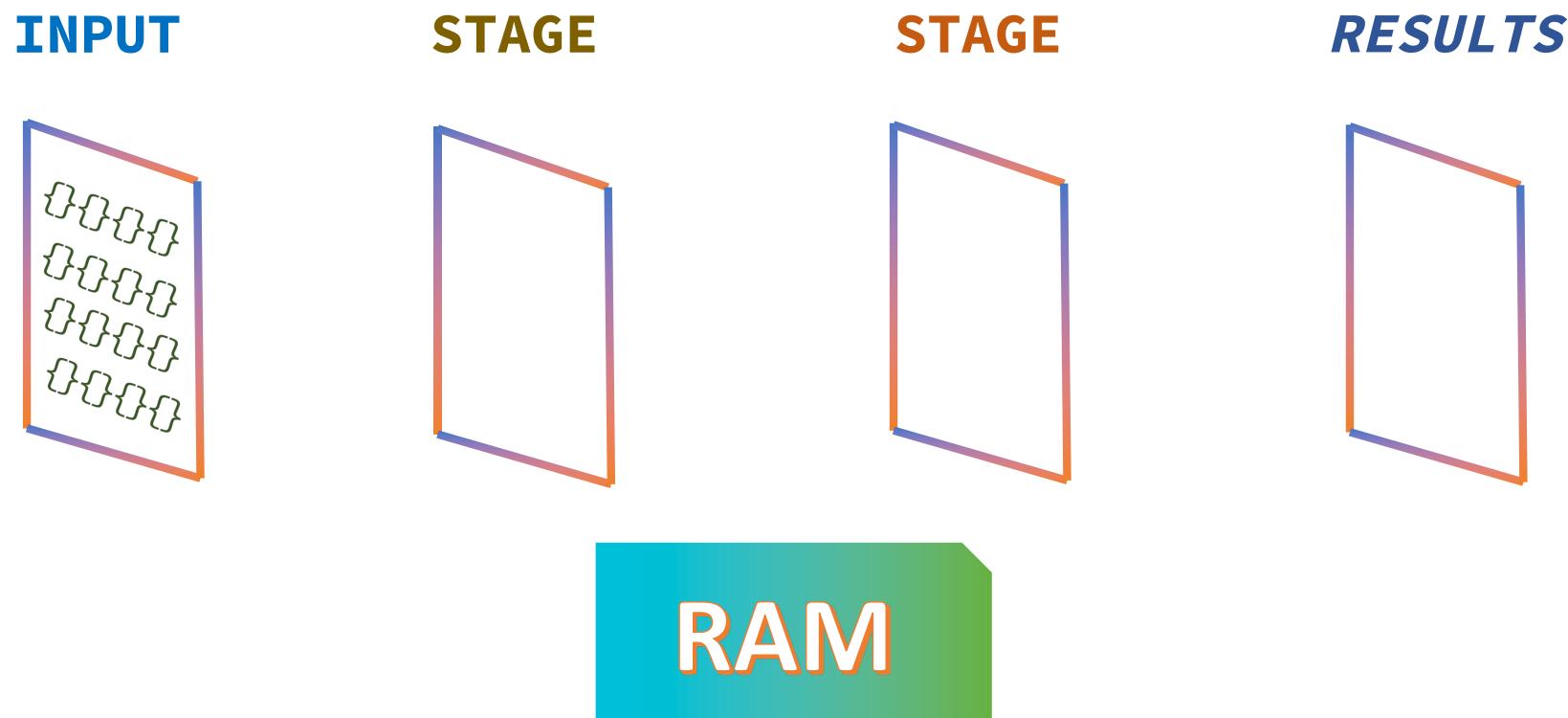


STREAMING RESOURCE USE



Each document is streamed through in RAM

BLOCKING RESOURCE USE



BLOCKING RESOURCE USE



Sort with Limit:

uses $\text{sizeOfDoc} * \text{Limit} + \text{sizeOfDoc}$

Sort without Limit:

uses sizeOfAllDocs

Group:

uses $\text{sizeOfDoc} * \text{numberOfGroups} + \text{sizeOfDoc}$

STAGES, EXPRESSIONS, OPERATORS, ACCUMULATORS, OH MY!

EXPRESSIONS

`$zip`
`$sum`
`$min`
`$max`
`$cmp`
`$hour`
`$split`
`$size`
`$mod`
`$avg`
`$isoWeek`
`$switch`
`$range`
`$ceil`
`$sqnrs`
`$stoLower`
`$month`
`$monthnsqnr`
`$stoUpper`
`$add`
`$divide`
`$arrayElemAt`
`$setDiffCP`
`$substrCP`
`$isoDayOfWeek`
`$millisecond`
`$abs`
`$type`
`$setEquals`
`$mergeObjects`
`$subtract`
`$first`
`$dayOfMonth`
`$second`
`$log10`
`$sdayOfYear`
`$sstrLenBytes`
`$reverseArray`
`$objectToArray`
`$let`
`$arrayToObject`
`$dateFromString`
`$filter`
`$setIntersection`
`$floor`
`$allElementsTrue`
`$indexOfCP`
`$setDifference`
`$pow`
`$strcasecmp`
`$dateFromParts`
`$subStrBytes`
`$indexOfArray`
`$anyElementTrue`
`$concat`
`$sindexOfBytes`
`$dayOfWeek`
`$minute`
`$year`
`$sqrt`
`$concat`
`$lte`
`$multiply`
`$toUpperCase`
`$meta`
`$cond`
`$week`
`$reduce`
`$runc`
`$dateToParts`
`$setIsSubset`
`$sconcat`
`$meta`

\$substr
\$toUpperCase
\$indexOfCP
\$substrBytes
\$indexOfBytes
\$split
\$strLenBytes
\$concat
\$strLenCP
\$substrCP
\$toLowerCase
\$cmp
\$isNumber
\$isoWeek
\$dayOfWeek
\$dateToParts
\$year
\$week
\$isoWeekYear
\$dateFromParts
\$week
\$dateFromString
\$isoDayOfWeek
\$millisecond
\$minute
\$dayOfMonth
\$second
\$dateToString

\$setIsSubset
\$arrayElemAt
\$indexOfArray
\$anyElementTrue
\$allElementsTrue
\$setIntersection
\$concatArrays
\$map
\$reverseArray
\$setUnion
\$size
\$setDifference
\$slice
\$zip
\$filter
\$mod
\$add
\$ceil
\$trunc
\$log
\$divide
\$abs
\$subtract
\$multiply
\$exp
\$floor
\$sqrt
\$avg
\$pow

\$type
\$lt
\$ifNull
\$let
\$mergeObjects
\$arrayToObject
\$objectToArray
\$switch
\$gte
\$cond
\$meta
\$max
\$last
\$addToSet
\$stdDevSamp
\$stdDevPop
\$min
\$push

\$literal

\$setIsSubset \$setEquals
\$arrayElemAt
\$indexOfArray
\$size
\$anyElementTrue
\$allElementsTrue
\$setIntersection
\$concatArrays
\$slice
\$zip
\$isArrayOf
\$map
\$reverseArray
\$setDifference
\$filter
\$setUnion

ARRAY EXPRESSIONS

ARRAY EXPRESSIONS

`$arrayElemAt`

`$concatArrays`

`$indexOfArray`

`$isArray`

`$size`

`$range`

`$reverseArray`

`$map`

`$reduce`

`$filter`

`$slice`

`$zip`

`$in`

... plus all the set expressions

ARRAY EXPRESSIONS

`$arrayElemAt`

`$concatArrays`

`$indexOfArray`

`$isArray`

`$size`

`$range`

`$reverseArray`

`$map`

`$reduce`

`$filter`

`$slice`

`$zip`

`$in`

... plus all the set expressions

ARRAY EXPRESSIONS

\$map

input: array

output: array

\$filter

input: array

output: subset of array

\$reduce

input: array

output: anything

you

want



MONGODB.local
SOCIAL

#MDBlocal

ARRAY EXPRESSIONS

```
"arr": [{"a":1}, {"a":99}, {"a":5}, {"a":3}]
```

```
{
  "$map": {
    "input": "$arr",
    "as": "var",
    "in": "$$this"
  }
}

{
  "$map": {
    "input": "$arr",
    "as": "eachElem",
    "in": {"b": "$eachElem.a"}
  }
}

{
  "$map": {
    "input": {"$range": [0, {"$size": "$arr"}]},
    "as": "index",
    "in": {"c": {"$arrayElemAt": ["$arr.a", "$$index"]}}
}
```

\$map

"a"	1
"a"	99
"a"	5
"a"	3

"a"	1
"a"	99
"a"	5
"a"	3

"a"	1
"a"	99
"a"	5
"a"	3

"b"	1
"b"	99
"b"	5
"b"	3

0	
1	
2	
3	

"c"	1
"c"	99
"c"	5
"c"	3

ARRAY EXPRESSIONS

\$filter

```
"arr": [{"a":1}, {"a":99}, {"a":5}, {"a":3}]
```

```
{"$filter": {  
    "input": "$arr",  
    "cond": {"$lt": ["$$this.a", 10]}  
}}
```

"a"	1
"a"	99
"a"	5
"a"	3



"a"	1
"a"	5
"a"	3

```
{"$filter": {  
    "input": "$arr",  
    "as": "elem",  
    "cond": {"$lt": ["$$elem.a", 10]}  
} }
```

ARRAY EXPRESSIONS

```
"arr": [{"a":1}, {"a":99}, {"a":5}, {"a":3}]
```

```
{"$reduce": {  
    "input": "$arr",  
    "initialValue": 0,  
    "in": {$add: ["$$value", "$$this.a"]}  
} }
```

\$reduce

"a"	1
"a"	99
"a"	5
"a"	3



108

ARRAY EXPRESSIONS

```
"arr": [{"a":1}, {"a":99}, {"a":5}, {"a":3}]
```

```
{"$reduce": {  
    "input": "$arr",  
    "initialValue": 0,  
    "in": {$add: ["$$value", "$$this.a"]}  
} }  
  
{"$reduce":{  
    "input": "$arr",  
    "intialValue": [],  
    "in": {"$concatArrays": [  
        [ $$this ],  
        "$$value"  
    ]}  
} }
```

\$reduce

"a"	1
"a"	99
"a"	5
"a"	3



108

"a"	1
"a"	99
"a"	5
"a"	3

ARRAY EXPRESSIONS

```
"arr": [{"a":1}, {"a":99}, {"a":5}, {"a":3}]
```

```
{"$reduce": {  
    "input": "$arr",  
    "initialValue": 0,  
    "in": {$add: ["$$value", "$$this.a"]}  
} }
```

```
{"$reduce":{  
    "input": "$arr",  
    "intialValue": [],  
    "in": {"$concatArrays": [  
        [ $$this ],  
        $$value  
    ]}  
} }
```

\$reduce

"a"	1
"a"	99
"a"	5
"a"	3



108

"a"	1
"a"	99
"a"	5
"a"	3



"a"	3
"a"	5
"a"	99
"a"	1

READABILITY TIP

ENCAPSULATE COMPLEXITY

Functions for expressions

```
reverseArray = function(input) {  
    return {"$reduce":{  
        "input": input,  
        "intialValue": [],  
        "in": {"$concatArrays": [  
            ["$$this"],  
            "$$value"  
        ]}  
    }};  
};  
  
db.c.aggregate([ {"$addFields":{  
    "revArray":reverseArray("$origArray")  
} } ])
```

ENCAPSULATE COMPLEXITY

Functions for expressions

```
db.c.aggregate([ {$addFields:{  
    "sortedArray":sortArray("$origArray")  
}}]  
function( inputArray, sortField="", asc=false)  
{  
    var suffix = "";  
    var maxF = MaxKey;  
    var minF = MinKey;  
    if (sortField != "") {  
        if (asc) {  
            maxF = MinKey;  
            minF = MaxKey;  
        }  
        sortField += suffix;  
        if (suffix == "")  
            sortField += "  
";  
        if (sortField[sortField.length - 1] != ",")  
            sortField += ",";  
    }  
    return sortField;  
}  
# M D B l o c a l
```



EXPRESSIONS

AGGREGATION

FIND QUERY

UPDATE QUERY

DOCUMENT VALIDATION



MONGODB.local
SOCAL

```
{children: [  
    {name:"Max", dob:"1994-12-01", dep:true},  
    {name:"Sam", dob:"1997-09-28", dep:true},  
    {name:"Kim", dob:"2000-02-29", dep:true}  
]}
```

#MDBlocal

EXPRESSIONS

AGGREGATION

FIND QUERY

UPDATE QUERY

DOCUMENT VALIDATION



MONGODB.local
SOCAL

```
{children: [
  {name:"Max", dob:"1994-12-01", dep:false},
  {name:"Sam", dob:"1997-09-28", dep:true},
  {name:"Kim", dob:"2000-02-29", dep:true}
]}

db.c.aggregate([
{$addFields:{
  numChildren:{$size:"$children"},
  numDependents:{$size:{
    $filter:{
      input:"$children.dep",
      cond: "$$this"
    }
  }}
}}, ...
])
```

#MDBlocal

EXPRESSIONS

new in 3.6!

AGGREGATION

FIND QUERY

UPDATE QUERY

DOCUMENT VALIDATION



MONGODB.local
SOCAL

```
{children: [  
    {name:"Max", dob:"1994-12-01", dep:false},  
    {name:"Sam", dob:"1997-09-28", dep:true},  
    {name:"Kim", dob:"2000-02-29", dep:true}  
]}  
  
db.c.find({$expr:{$gt:[ "$a", "$b" ]}})
```

#MDBlocal

EXPRESSIONS

new in 3.6!

AGGREGATION

FIND QUERY

UPDATE QUERY

DOCUMENT VALIDATION



```
{children: [
  {name:"Max", dob:"1994-12-01", dep:false},
  {name:"Sam", dob:"1997-09-28", dep:true},
  {name:"Kim", dob:"2000-02-29", dep:true}
]}

db.c.find({$expr: {
  $lt:[ {$size:{$filter:{ input:'$children.dep',
    cond:'$$this' }}}, 2
]}})
```

M D B l o c a l

EXPRESSIONS

new in 3.6!

AGGREGATION

FIND QUERY

UPDATE QUERY

DOCUMENT VALIDATION



MONGODB.local
SOCAL

```
{children: [
  {name:"Max", dob:"1994-12-01", dep:false},
  {name:"Sam", dob:"1997-09-28", dep:true},
  {name:"Kim", dob:"2000-02-29", dep:true}
]}

db.c.find({$expr:{$gt:[{$let:{vars:{dobs:{$map:{input:'$children.dob'
in: {$year:{$dateFromString:{dateString:'$$this'
}}}}}}},{$subtract:[{$max:'$$dobs'},{$min:'$$dobs'}]}]}},10
}])
```

M D B l o c a l

EXPRESSIONS

new in 3.6!

AGGREGATION

FIND QUERY

UPDATE QUERY

DOCUMENT VALIDATION



```
{children: [
  {name:"Max", dob:"1994-12-01", dep:false},
  {name:"Sam", dob:"1997-09-28", dep:true},
  {name:"Kim", dob:"2000-02-29", dep:true}
]}

db.c.update({$expr: {$anyElementTrue:{$map:{input:"$children", in: {$and:[{$lt:[{$$this.dob}, "1997-01-22"]}], "$$this.dep"}]} }}}, {$set:{ audit:true }})
```

M D B l o c a l

EXPRESSIONS

new in 3.6!

AGGREGATION

FIND QUERY

UPDATE QUERY

DOCUMENT VALIDATION

```
{children: [  
  {name:"Max", dob:"1994-12-01", dep:false},  
  {name:"Sam", dob:"1997-09-28", dep:true},  
  {name:"Kim", dob:"2000-02-29", dep:true}  
]}
```

```
db.createCollection("c", validator:  
{$expr: {  
  <anything you can express>  
}})
```



#MDBlocal

NETWORK SUSPECT ACTIVITY DETECTION

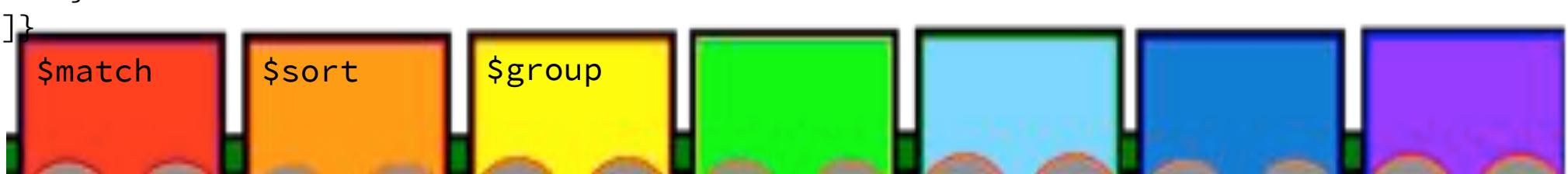


#MDBlocal

```
start=ISODate("...")  
end=ISODate("...")  
{  
    user: "303900",  
    ipaddr: "71.56.112.56",  
    ts:ISODate("2017-05-08T05:28:13Z")  
}
```



```
start=ISODate("...")  
end=ISODate("...")  
{  
    user: "303900",  
    ipaddr: "71.56.112.56",  
    ts:ISODate("2017-05-08T...")  
}  
  
{ _id: "303900",  
  ips: [  
    {ip:"71.56.112.56",  
     ts:ISODate("2017-05-08T08:54:04Z")  
    },  
    {ip:"71.56.112.56",  
     ts:ISODate("2017-05-09T09:01:11Z")  
    },  
    {ip:"12.130.117.87",  
     ts:ISODate("2017-05-09T09:04:59Z")  
    }  
]  
}
```



```

start=ISODate("...")
end=ISODate("...")
{
  user: "303900",
  ipaddr: "71.56.112.56",
  ts:ISODate("2017-05-08T...")  

}
  

{ _id: "303900",
  ips: [
    {ip:"71.56.112.56",
     ts:ISODate("2017-05-08T...")},
    {ip:"71.56.112.56",
     ts:ISODate("2017-05-09T...")},
    {ip:"12.130.117.87",
     ts:ISODate("2017-05-09T...")}
  ]
}  

{$match:{ts:{$gte:start,$lt:end}}},
{$sort:{ts:1}},
{$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}}}},
{$addFields:{diffs: {$filter:{  

  input:{$map:{  

    input: {$range:[0,{$subtract:[{$size:"$ips"},1]}]}, as:"i",  

    in:{$let:{vars:{ip1:{$arrayElemAt:["$ips","$$i"]},  

      ip2:{$arrayElemAt:["$ips",{$add:[$$i,1]}]}},  

      in:{  

        diff:{$divide:[{$subtract:[$$ip2.ts,"$$ip1.ts"]},60000]},  

        ip1:"$$ip1.ip", t1:"$$ip1.ts",  

        ip2:"$$ip2.ip", t2:"$$ip2.ts"  

      }}}},  

      cond:{$and:[{$lt:[$$this.diff,10]},{$ne:[$$this.ip1,"$$this.ip2"]}]}}}},  

    }]}

```

\$match

\$sort

\$group

\$addFields

\$match

\$project

```

start=ISODate("...")
end=ISODate("...")
{
  user: "303900",
  ipaddr: "71.56.112.56",
  ts:ISODate("2017-05-08T...")  

}  

{ _id: "303900",
  ips: [
    {ip:"71.56.112.56",
     ts:ISODate("2017-05-08T...")},
    {ip:"71.56.112.56",
     ts:ISODate("2017-05-09T...")},
    {ip:"12.130.117.87",
     ts:ISODate("2017-05-09T...")}
  ]
}  

{$match:{ts:{$gte:start,$lt:end}}},
{$sort:{ts:1}},
{$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}}}},
{$addFields:{diffs: {$filter:{  

  input:{$map:{  

    input: {$range:[0,{$subtract:[{$size:"$ips"},1]}]}, as:"i",  

    in:{$let:{vars:{ip1:{$arrayElemAt:["$ips","$$i"]},  

      ip2:{$arrayElemAt:["$ips",{$add:[$$i,1]}]}},  

      in:{  

        diff:{$divide:[{$subtract:[$$ip2.ts,"$$ip1.ts"]},60000]},  

        ip1:"$$ip1.ip", t1:"$$ip1.ts",  

        ip2:"$$ip2.ip", t2:"$$ip2.ts"  

      }}}},  

      cond:{$and:[{$lt:[$$this.diff,10]},{$ne:[$$this.ip1,"$$this.ip2"]}]}}},  

      {$match:{"diffs":{$ne:[]}}},  

      {$project:{_id:0, user:"$_id", suspectLogins:"$diffs"}}


```



```

start=ISODate("...")
end=ISODate("...")
{
  user: "303900",
  ipaddr: "71.56.112.56",
  ts:ISODate("2017-05-08T..."))
}

{ "user" : "35237073",
  "suspectLogins" : [
    {"diff": 4.8333333333,
     "ip1": "106.220.151.16",
     "t1":"2017-05-08T06:58",
     "ip2": "223.182.113.15"
     "t2":"2017-05-08T07:03"
    },
    {"diff": 8.3,
     "ip1": "223.182.113.15",
     "t1":"2017-05-08T07:03",
     "ip2": "49.206.217.26",
     "t2":"2017-05-08T07:11"
    }
  ]
}

{$match:{ts:{$gte:start,$lt:end}}},
{$sort:{ts:1}},
{$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}}}},
{$addFields:{diffs: {$filter:{  

  input:{$map:{  

    input: {$range:[0,{$subtract:[{$size:"$ips"},1]}]}, as:"i",  

    in:{$let:{vars:{ip1:{$arrayElemAt:["$ips","$$i"]},  

      ip2:{$arrayElemAt:["$ips",{$add:[["$$i",1]}]}},  

      in:{  

        diff:{$divide:[{$subtract:[ "$$ip2.ts", "$$ip1.ts"]},60000]},  

        ip1:"$$ip1.ip", t1:"$$ip1.ts",  

        ip2:"$$ip2.ip", t2:"$$ip2.ts"  

      }}}},  

      cond:{$and:[{$lt:[ "$$this.diff",10]},{$ne:[ "$$this.ip1", "$$this.ip2"]]}  

    }]}},  

    {$match:{"diffs":{$ne:[]}}},  

    {$project:{_id:0, user:"$_id", suspectLogins:"$diffs"}}
  ]
}

```

}

\$match

\$sort

\$group

\$addFields

\$match

\$project

```

start=ISODate("...")           {$match:{ts:{$gte:start,$lt:end}}},  

end=ISODate("...")             {$sort:{ts:1}},  

{                             {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}}}},  

  user: "303900",  

  ipaddr: "71.56.112.56",  

  ts:ISODate("2017-05-08T...")  

}  

{ _id: "303900",  

  ips: [  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-08T...")  

    },  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-09T...")  

    },  

    {ip:"12.130.117.87",  

     ts:ISODate("2017-05-09T...")  

    }  

]
}                                {$addFields:{diffs: {$filter:{  

  input:{'$map:{  

    input: {$range:[0,{'$subtract:[{$size:"$ips"},1]}]}, as:"i",  

    in:{$let:{vars:{ip1:{$arrayElemAt:["$ips","$$i"]},  

                  ip2:{$arrayElemAt:["$ips",{$add:["$$i",1]}]}},  

    in:{  

      diff:{$divide:[{$subtract:["$$ip2.ts","$$ip1.ts"]},60000]},  

      ip1:"$$ip1.ip", t1:"$$ip1.ts",  

      ip2:"$$ip2.ip", t2:"$$ip2.ts"  

    }}}},  

    cond:{$and:[{$lt:["$$this.diff",10]},{'$ne:["$$this.ip1","$$this.ip2"]}]}}},  

  {$match:{"diffs":{$ne:[]}}},  

  {$project:{_id:0, user:"$_id", suspectLogins:"$diffs"}}
}

```



```

start=ISODate("...")           {$match:{ts:{$gte:start,$lt:end}}},  

end=ISODate("...")             {$sort:{ts:1}},  

{                                {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}}}},  

  user: "303900",               {$addFields:{diffIpNum:{$size:{$setUnion:"$ips.ip"}}}},  

  ipaddr: "71.56.112.56",      {$match:{diffIpNum:{$gt:1}}},  

  ts:ISODate("2017-05-08T...")   {$addFields:{diffs: {$filter:{  

    input:{$map:{  

      input: {$range:[0,{$subtract:[{$size:"$ips"},1]}]}, as:"i"},  

      in:{$let:{vars:{ip1:{$arrayElemAt:["$ips","$$i"]},  

                     ip2:{$arrayElemAt:["$ips",{$add:["$$i",1]}]}},  

                     in:{  

                       diff:{$divide:[{$subtract:["$$ip2.ts","$$ip1.ts"]},60000]},  

                       ip1:"$$ip1.ip", t1:"$$ip1.ts",  

                       ip2:"$$ip2.ip", t2:"$$ip2.ts"  

                     }}}},  

      cond:{$and:[{$lt:["$$this.diff",10]},{$ne:["$$this.ip1","$$this.ip2"]}]}}},  

      {$match:{"diffs":{$ne:[]}}},  

      {$project:{_id:0, user:"$_id", suspectLogins:"$diffs"}}
}

```



```

start=ISODate("...")           {$match:{ts:{$gte:start,$lt:end}}},  

end=ISODate("...")             {$sort:{ts:1}},  

{                                {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}},  

                                         diffIps:{$addToSet:"$ipaddr"}}},  

  user: "303900",  

  ipaddr: "71.56.112.56",  

  ts:ISODate("2017-05-08T...")  

}  

{ _id: "303900",  

  ips: [  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-08T...")  

    },  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-09T...")  

    },  

    {ip:"12.130.117.87",  

     ts:ISODate("2017-05-09T...")  

    },  

    diffIps: [  

      "71.56.112.56",  

      "12.130.117.87"  

    ]  

}  

  {$match:{ts:{$gte:start,$lt:end}}},  

  {$sort:{ts:1}},  

  {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}},  

                                         diffIps:{$addToSet:"$ipaddr"}}},  

  {$match:{$expr:{$gt:[{$size:"$diffIps"},1]}}},  

  {$addFields:{diffs: {$filter:{  

    input:{$map:{  

      input: {$range:[0,{$subtract:[{$size:"$ips"},1]}]}, as:"i",  

      in:{$let:{vars:{ip1:{$arrayElemAt:["$ips","$$i"]},  

                           ip2:{$arrayElemAt:["$ips",{$add:[["$$i",1]}]}},  

                           in:{  

                             diff:{$divide:[{$subtract:["$$ip2.ts","$$ip1.ts"]},60000]},  

                             ip1:"$$ip1.ip", t1:"$$ip1.ts",  

                             ip2:"$$ip2.ip", t2:"$$ip2.ts"  

                           }}}},  

    cond:{$and:[{$lt:[{$this.diff},10]},{$ne:[{$this.ip1},{$this.ip2}]}]}  

}}}},  

  {$match:{"diffs":{$ne:[]}}},  

  {$project:{_id:0, user:"$_id", suspectLogins:"$diffs"}}

```



```

start=ISODate("...")           {$match:{ts:{$gte:start,$lt:end}}},  

end=ISODate("...")             {$sort:{ts:1}},  

{                            {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}},  

                                         diffIps:{$addToSet:"$ipaddr"}}},  

  user: "303900",  

  ipaddr: "71.56.112.56",  

  ts:ISODate("2017-05-08T...")  

}  

{ _id: "303900",  

  ips: [  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-08T...")  

    },  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-09T...")  

    },  

    {ip:"12.130.117.87",  

     ts:ISODate("2017-05-09T...")  

    },  

    diffIps: [  

      "71.56.112.56",  

      "12.130.117.87"  

    ]  

}  

  {$match:{$gt:{$size:ips}},  

   {$sort:{ts:-1}},  

   {$group:{_id:"$user",  

             ips:{$push:{ip:"$ipaddr", ts:"$ts"}},  

             diffIps:{$addToSet:"$ipaddr"}},  

   {$addFields:{diffs: {$filter:  

      {$map:  

        {$input:{$range:[0,{$subtract:[{$size:$ips},1]}]},  

         as:"i"},  

        {$let:{vars:{ip1:{$arrayElemAt:[$ips,"$$i"]},  

                     ip2:{$arrayElemAt:[$ips,{+$add:[$$i,1]}]}},  

        {$in:  

          {$divide:{$subtract:[$$ip2.ts,"$$ip1.ts"]},60000},  

          ip1:"$$ip1.ip", t1:"$$ip1.ts",  

          ip2:"$$ip2.ip", t2:"$$ip2.ts"}  

        }}}},  

      cond:{$and:[{$lt:[{$this.diff},10]},{$ne:[{$this.ip1},{$this.ip2}]}]}  

    }},  

    {$match:{diffs:{$ne:[]}}},  

    {$project:{_id:0, user:"$_id", suspectLogins:"$diffs"}}
  ]
}

```



```

start=ISODate("...")           {$match:{ts:{$gte:start,$lt:end}}},  

end=ISODate("...")             {$sort:{ts:1}},  

{                                {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}},  

          diffIps:{$addToSet:"$ipaddr"}}},  

  user: "303900",  

  ipaddr: "71.56.112.56",  

  ts:ISODate("2017-05-08T...")  

}  

{ _id: "303900",  

  ips: [  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-08T...")  

    },  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-09T...")  

    },  

    {ip:"12.130.117.87",  

     ts:ISODate("2017-05-09T...")  

    },  

    diffIps: [  

      "71.56.112.56",  

      "12.130.117.87"
    ]
}

```

\$match

\$sort

\$group

\$match

\$addFields

\$match

\$project



```

start=ISODate("...")           {$match:{ts:{$gte:start,$lt:end}}},  

end=ISODate("...")             {$sort:{ts:1}},  

{                             {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}},  

      diffIps:{$addToSet:"$ipaddr"}}},  

  user: "303900",  

  ipaddr: "71.56.112.56",  

  ts:ISODate("2017-05-08T...")  

}  

{ _id: "303900",  

  ips: [  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-08T...")  

    },  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-09T...")  

    },  

    {ip:"12.130.117.87",  

     ts:ISODate("2017-05-09T...")  

    },  

    diffIps: [  

      "71.56.112.56",  

      "12.130.117.87"
    ]
}
  {$match:{"diff":{$lt:10}},{$ne:["$ip1","$ip2"]}}
  {$match:{"diffs":{$ne:[]}}}
  {$project:{_id:0, user:"$id", suspectLogins:"$diffs"}}

```



```

start=ISODate("...")           {$match:{ts:{$gte:start,$lt:end}}},  

end=ISODate("...")             {$sort:{ts:1}},  

{                                {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}},  

                                         diffIps:{$addToSet:"$ipaddr"}}},  

  user: "303900",  

  ipaddr: "71.56.112.56",  

  ts:ISODate("2017-05-08T...")  

}  

{ _id: "303900",  

  ips: [  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-08T...")  

    },  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-09T...")  

    },  

    {ip:"12.130.117.87",  

     ts:ISODate("2017-05-09T...")  

    },  

    diffIps: [  

      "71.56.112.56",  

      "12.130.117.87"  

    ]  

}

```

```

  {$match:{diffIps.1:{$exists:true}}},  

  {$addFields:{diffs: {$filter:{  

    input:{$map:{  

      input: {$range:[0,{$subtract:[{$size:"$ips"},1]}]}, as:"i",  

      in:{$let:{vars:{ip1:{$arrayElemAt:["$ips","$$i"]},  

                     ip2:{$arrayElemAt:["$ips",{$add:[["$$i",1]}]}},  

                     in:{  

                       diff:{$cond:{  

                         if:{$ne:[ "$$ip1.ip", "$$ip2.ip"]},  

                         then:{$divide:[{$subtract:[ "$$ip2.ts", "$$ip1.ts"]},60000]},  

                         else: 9999 },  

                         ip1:"$$ip1.ip", t1:"$$ip1.ts",  

                         ip2:"$$ip2.ip", t2:"$$ip2.ts"  

                       }}}},  

                       cond:{$and:[{$lt:[ "$$this.diff",10]},{$ne:[ "$$this.ip1", "$$this.ip2"]]}  

                     }}}}},  

  {$match:{diffs:{$ne:[]}}},  

  {$project:{_id:0, user:"$_id", suspectLogins:"$diffs"}}

```



```

start=ISODate("...")           {$match:{ts:{$gte:start,$lt:end}}},  

end=ISODate("...")             {$sort:{ts:1}},  

{                                {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}},  

          diffIps:{$addToSet:"$ipaddr"}}},  

  user: "303900",  

  ipaddr: "71.56.112.56",  

  ts:ISODate("2017-05-08T...")  

}  

{ _id: "303900",  

  ips: [  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-08T...")  

    },  

    {ip:"71.56.112.56",  

     ts:ISODate("2017-05-09T...")  

    },  

    {ip:"12.130.117.87",  

     ts:ISODate("2017-05-09T...")  

    },  

    diffIps: [  

      "71.56.112.56",  

      "12.130.117.87"  

    ]  

}

```

```

  {$match:{diffs:{$ne:[]}}},  

  {$project:{_id:0, user:"$_id", suspectLogins:"$diffs"}}

```



```

start=ISODate("...")           {$match:{ts:{$gte:start,$lt:end}}},  

end=ISODate("...")             {$sort:{ts:1}},  

{                                {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}},  

          diffIps:{$addToSet:"$ipaddr"}}},  

  user: "303900",  

  ipaddr: "71.56.112.56",  

  ts:ISODate("2017-05-08T...")  

}  

{ "user" : "35237073",  

  "suspectLogins" : [  

    {"diff": 4.8333333333,  

     "ip1": "106.220.151.16",  

     "t1":"2017-05-08T06:58",  

     "ip2": "223.182.113.15"  

     "t2":"2017-05-08T07:03"  

    },  

    {"diff": 8.3,  

     "ip1": "223.182.113.15",  

     "t1":"2017-05-08T07:03",  

     "ip2": "49.206.217.26",  

     "t2":"2017-05-08T07:11"
    }
  ]
}  

] $match $sort $group $match $addFields $match $project

```

```

start=ISODate("...")           {$match:{ts:{$gte:start,$lt:end}}},  

end=ISODate("...")             {$sort:{ts:1}},  

{                                {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}},  

    user: "303900",               diffIps:{$addToSet:"$ipaddr"}}},  

    ipaddr: "71.56.112.56",  

    ts:ISODate("2017-05-08T...")  

}  

{ "user" : "35237073",  

  "suspectLogins" : [  

    {"diff": 4.8333333333,  

     "ip1": "106.220.151.16",  

     "t1":"2017-05-08T06:58",  

     "ip2": "223.182.113.15"  

     "t2":"2017-05-08T07:03"  

    },  

    {"diff": 8.3,  

     "ip1": "223.182.113.15",  

     "t1":"2017-05-08T07:03",  

     "ip2": "49.206.217.26",  

     "t2":"2017-05-08T07:11"
    }
  ]
}  

] $match $sort $group $match $addFields $match $project

```

 {\$match:{diffs:{\$ne:[]}}},
 {\$project:{_id:0, user:"\$_id", suspectLogins:"\$diffs"}}

```

start=ISODate("...")           {$match:{ts:{$gte:start,$lt:end}}},  

end=ISODate("...")             {$sort:{ts:1}},  

{                                {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}},  

    user: "303900",               diffIps:{$addToSet:"$ipaddr"}}},  

    ipaddr: "71.56.112.56",  

    ts:ISODate("2017-05-08T...")  

}  

{ "user" : "35237073",  

  "suspectLogins" : [  

    {"diff": 4.8333333333,  

     "ip1": "106.220.151.16",  

     "t1":"2017-05-08T06:58",  

     "ip2": "223.182.113.15"  

     "t2":"2017-05-08T07:03"  

    },  

    {"diff": 8.3,  

     "ip1": "223.182.113.15",  

     "t1":"2017-05-08T07:03",  

     "ip2": "49.206.217.26",  

     "t2":"2017-05-08T07:11"
    }
  ]
}  

] $match $sort $group $match $addFields $match $project

```

 {\$match:{diffs:{\$ne:[]}}},
 {\$project:{_id:0, user:"\$_id", suspectLogins:"\$diffs"}}

```

start=ISODate("...")           {$match:{ts:{$gte:start,$lt:end}}},  

end=ISODate("...")             {$sort:{ts:1}},  

{                                {$group:{_id:"$user",ips:{$push:{ip:"$ipaddr", ts:"$ts"}},  

    user: "303900",               diffIps:{$addToSet:"$ipaddr"}}},  

    ipaddr: "71.56.112.56",  

    ts:ISODate("2017-05-08T...")  

}  

{ "user" : "35237073",  

  "suspectLogins" : [  

    {"diff": 4.8333333333,  

     "ip1": "106.220.151.16",  

     "t1":"2017-05-08T06:58",  

     "ip2": "223.182.113.15"  

     "t2":"2017-05-08T07:03"  

    },  

    {"diff": 8.3,  

     "ip1": "223.182.113.15",  

     "t1":"2017-05-08T07:03",  

     "ip2": "49.206.217.26",  

     "t2":"2017-05-08T07:11"
    }
  ]
}  

] $match $sort $group $match $addFields $match $project

```

 {\$match:{diffs:{\$ne:[]}}},
 {\$project:{_id:0, user:"\$_id", suspectLogins:"\$diffs"}}

POWERFUL AGGREGATIONS

understand stages

- Best order for performance
- Avoid unnecessary "blocking"
 - keep "streaming"
- Maximize use of indexes
 - early stages get the index!
- Liberally check `explain()` output

understand expressions

- Schema manipulation
- Array transformation
- Use in find query filter, ...

use functions

- Readable, debug-able, reusable



THE FUTURE OF AGGREGATION

Better performance & optimizations

More stages & expressions

More options for output

Compass helper for aggregate

Unify different languages



<https://github.com/asya999/mdbw17>

AGGREGATE is the new FIND

<https://github.com/asya999/mbw17>

#MDBlocal



THANK YOU!

<https://github.com/asya999/mbw17>