
ENTRINSICON
LEVEL UP!
September 26th-29th, 2023
Raleigh, NC

Working with Large Datasets

Creation, Maintenance and Modification

Andrew Morovati, CIO, Entrinsik Inc.

Agenda

1. How large can we get?
2. Initial Ingest
3. Updates
4. Adding columns
5. Questions?

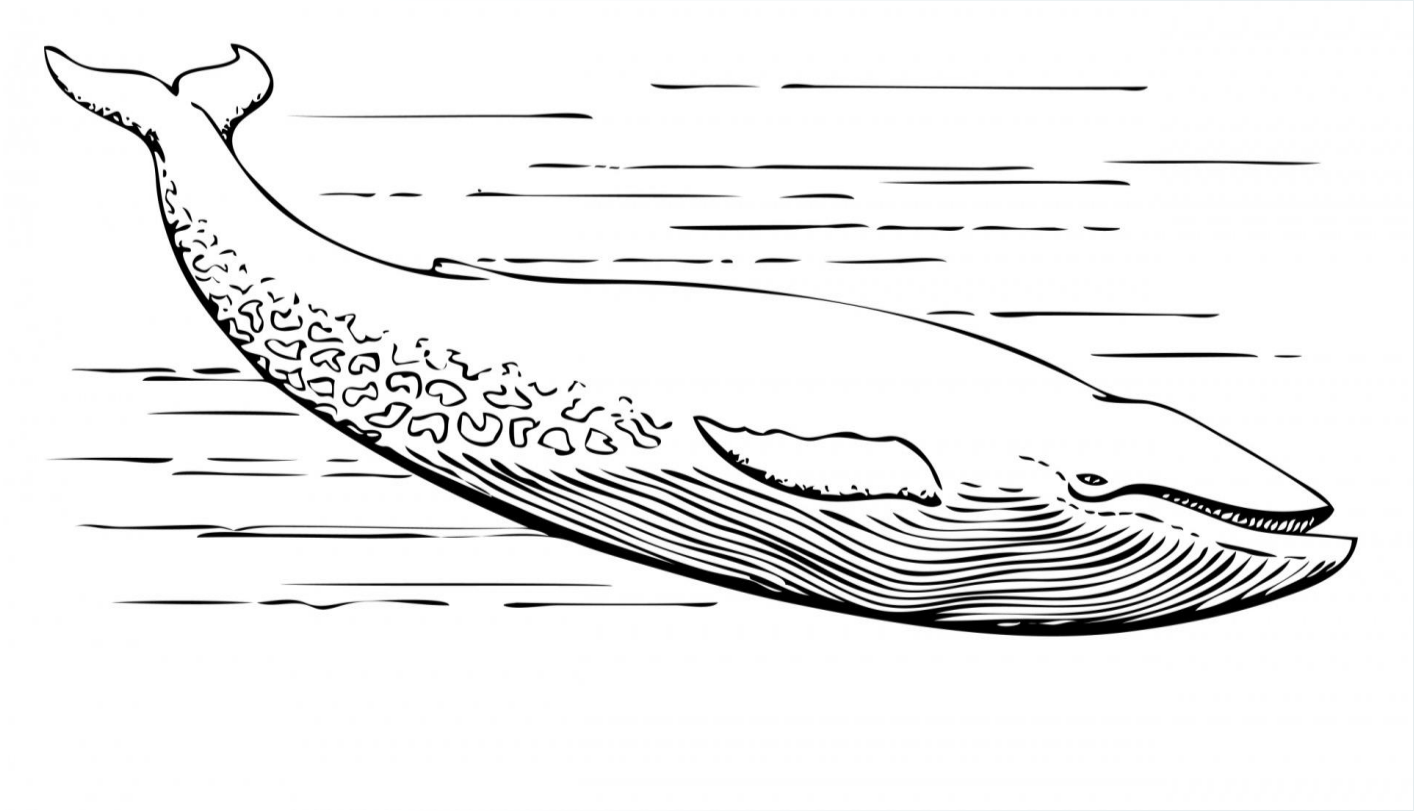
ENTRINSICON
LEVEL UP!

Andrew Morovati

Entrinsik Inc.

How large is large?

- Big as you want!
- Demo today is the Iowa Liquor Sales publicly available dataset.
- Filtering easier than database queries (and quicker).



ENTRINSICON
LEVEL UP!

Creating a large dataset

- A huge query running in one thread has a limit for Records per Second (RPS)
- A single query against our 27 million row database table would simply not return any data
- Break up the query into smaller queries, to leverage multiple Informer threads, and database capabilities.

Entrinsik IR tool

- May be run on any computer that can log into Informer.
- Can stage many queries to run in parallel
- Works by filtering on date fields
- Installs with `npm install -g @entrinsik/ir`

Set up your dataset

- Create your dataset
- Add "on or after" and "on or before" conditions, with inputs
- Set your dataset to upsert or append mode. Upsert requires a column with a unique value

Prepare the ingest process with IR

- Clear any old tasks
- Edit the plan configuration file
- Load the ingest plan for staging
- Start your ingest server
- Start the ingest process

Ingest Plan – plan.json

```
{
  "defaults": {
    "tenant":["entrinsicon"]
  },
  "datasets": {
    "andrew:all-iowa-liquor-sales-2012": {
      "params": {
        "start": "${start.format('YYYY-MM-DD')}",
        "end": "${end.format('YYYY-MM-DD')}"
      },
      "interval": "P1M",
      "start": "2012-01-01",
      "end": "2012-02-28"
    }
  }
}
```

ENTRINSICON
LEVEL UP!

INGEST DEMO

ENTRINSICON
LEVEL UP!

Maintaining your large dataset

Query should now be adjusted to retrieve all the records since the last refresh data, on a schedule:

- Date on or after LAST_QUERIED_AT
- Remove inputs
- Put on schedule of your choosing
- Because we are in upsert mode, any invoice will be either updated or added

Adding fields - Elasticsearch Script

If you want to add a calculated field but don't want to reindex all 27M + rows, you might be able to use an Elasticsearch script field to your data. These are values derived from fields in each ES document (row). Today we'll add two fields: a arithmetic field to calculate profits per liter sold, and a location field, so we can make use of a Google maps visual to show the distribution of various products.

The script language is called "painless" and is proprietary to Elastic. Chat GPT helps tremendously with these.

Elasticsearch Script – Profit per liter

```
if(doc['bottleVolumeMl'].size()==0) return null;

// need 1000.00 because 1000 would result in a rounded integer
def multiplier = 1000.00 / doc['bottleVolumeMl'].value;

def retail = doc['stateBottleRetail'].size()==0 ? 0 :
doc['stateBottleRetail'].value;

def cost = doc['stateBottleCost'].size()==0 ? 0 :
doc['stateBottleCost'].value;
return (retail - cost) * multiplier;
```

ENTRINSICON
LEVEL UP!

Elasticsearch Script – Location code

```
if(doc['storeLocation'].size() == 0) return null;

// the storeLocation looks like POINT (-95.144439 43.13837)

def pattern = /^POINT \(([0-9\.-]+) ([0-9\.-]+)\)$/;

def matcher = pattern.matcher(doc['storeLocation'].value);
if(matcher.matches()) {
      return [ 'lon' : Double.parseDouble(matcher.group(1)),
                  'lat' : Double.parseDouble(matcher.group(2))];
}
return null;
```

ENTRINSICON
LEVEL UP!

ENTRINSICON

LEVEL UP!

September 26th-29th, 2023

Raleigh, NC

Q & A

ENTRINSICON

LEVEL UP!

ENTRINSICON

LEVEL UP!

September 26th-29th, 2023

Raleigh, NC

Thank You

ENTRINSICON

LEVEL UP!