Loading Musicbrainz in Elasticsearch

I stumpled upon the INSERT INTO LOGSTASH SELECT DATA FROM DATABASE article and decided to play around.

Installing PostgreSQL and loading the database and the data into Elasticsearch was just as described; the manual for MBSlave is very good.

Refer to the https://musicbrainz.org/doc/MusicBrainz_Database for more

✓ I decided to use elkserver3 and a new logstash on that one, to avoid messing elkserver1 up.

Is possible to have a different Logstash config on each server in the cluster. Also - I decided to name the Index "musicbrainz-%{+YYYY.MM.dd}" to have some control over the load and a possible cleanup afterwards.

My Input file:

10-musicbrainz.conf

```
input {
    jdbc {
        jdbc_driver_library => "/etc/logstash/postgresql-9.4.1212.jre6.jar"
        jdbc_driver_class => "org.postgresql.Driver"
        jdbc_connection_string => "jdbc:postgresql://localhost:5432/musicbrainz?
user=musicbrainz&password=*******"
        jdbc_user => "musicbrainz"
        statement_filepath => "/etc/logstash/query.sql"
        schedule => "0 15 * * *"
    }
}
```

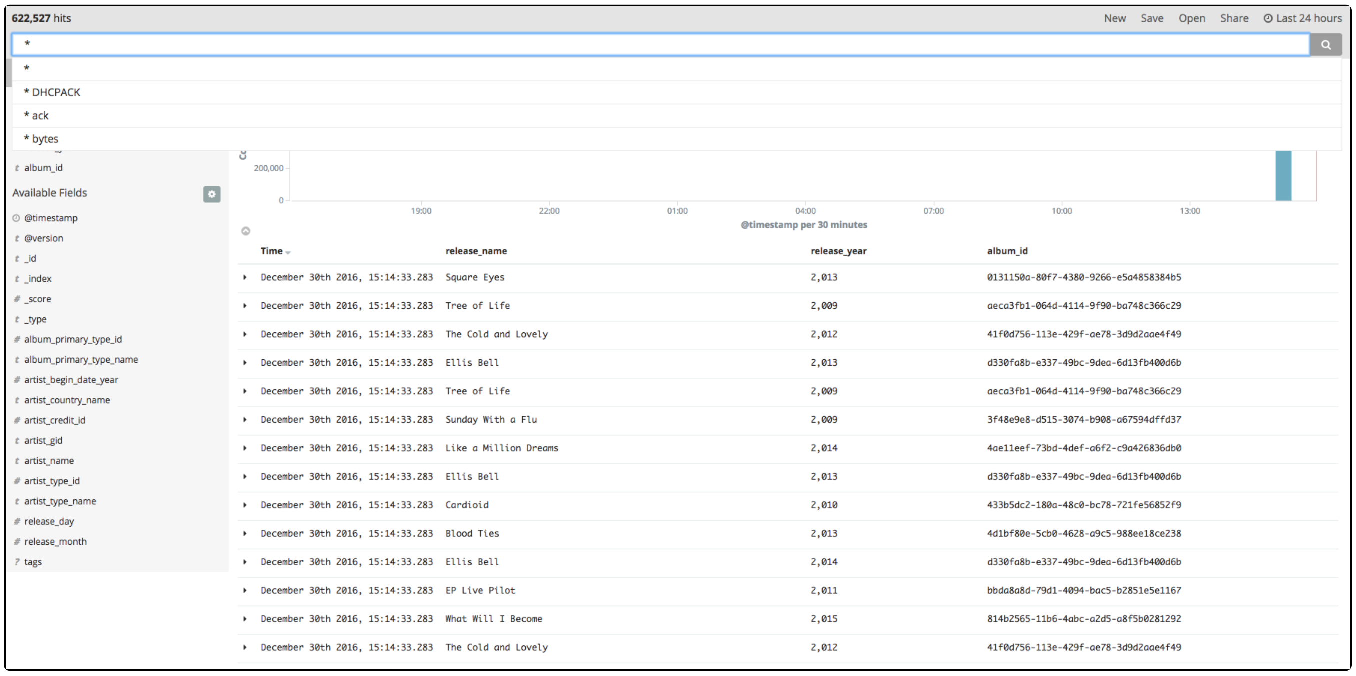
Notice the Schedule - the Query runs one time each day at 15:00 - but the data is static, so its not nessesary....But the schedule makes sure I know when the Query is runned one time (only)

My Output file:

```
output {

  elasticsearch
  {
    hosts => "localhost:9200"
    sniffing => false
    manage_template => false
    index => "musicbrainz-%{+YYYY.MM.dd}"
  }
}
```

The load gave this in Kibana:



Time	release_name	release_year	album_id
December 30th 2016, 15:14:33.283	Square Eyes	2,013	0131150a-80f7-4380-9266-e5a4858384b5
December 30th 2016, 15:14:33.283	Tree of Life	2,009	aeca3fb1-064d-4114-9f90-ba748c366c29
December 30th 2016, 15:14:33.283	The Cold and Lovely	2,012	41f0d756-113e-429f-ae78-3d9d2aae4f49
December 30th 2016, 15:14:33.283	Ellis Bell	2,013	d330fa8b-e337-49bc-9dea-6d13fb400d6b
December 30th 2016, 15:14:33.283	Tree of Life	2,009	aeca3fb1-064d-4114-9f90-ba748c366c29
December 30th 2016, 15:14:33.283	Sunday With a Flu	2,009	3f48e9e8-d515-3074-b908-a67594dffd37
December 30th 2016, 15:14:33.283	Like a Million Dreams	2,014	4ae11eef-73bd-4def-a6f2-c9a426836db0
December 30th 2016, 15:14:33.283	Ellis Bell	2,013	d330fa8b-e337-49bc-9dea-6d13fb400d6b
December 30th 2016, 15:14:33.283	Cardioid	2,010	433b5dc2-180a-48c0-bc78-721fe56852f9
December 30th 2016, 15:14:33.283	Blood Ties	2,013	4d1bf80e-5cb0-4628-a9c5-988ee18ce238
December 30th 2016, 15:14:33.283	Ellis Bell	2,014	d330fa8b-e337-49bc-9dea-6d13fb400d6b
December 30th 2016, 15:14:33.283	EP Live Pilot	2,011	bbda8a8d-79d1-4094-bac5-b2851e5e1167
December 30th 2016, 15:14:33.283	What Will I Become	2,015	814b2565-11b6-4abc-a2d5-a8f5b0281292
December 30th 2016, 15:14:33.283	The Cold and Lovely	2,012	41f0d756-113e-429f-ae78-3d9d2aae4f49

And verifying the row count in PostgreSQL with:

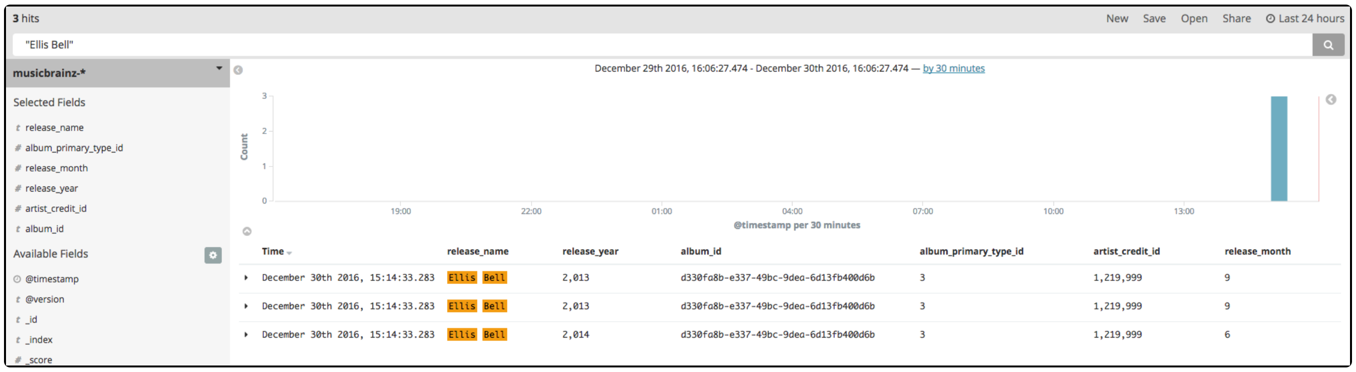
```
SELECT count(*) AS Dummy FROM (
SELECT
    release_group.gid AS album_id,
    release_group.type AS album_primary_type_id,
    release_group_primary_type.name AS album_primary_type_name,
    release.name AS release_name,
    artist.name AS artist_name,
    artist.gid AS artist_gid,
    artist_credit.id AS artist_credit_id,
    artist.type AS artist_type_id,
    artist_type.name AS artist_type_name,
    artist.begin_date_year artist_begin_date_year,
    area.name AS artist_country_name,
    release_country.date_year AS release_year,
    release_country.date_month AS release_month,
    release_country.date_day AS release_day
FROM
    musicbrainz.artist
INNER JOIN musicbrainz.artist_credit_name
    ON artist_credit_name.artist = artist.id
INNER JOIN musicbrainz.artist_credit
    ON artist_credit.id = artist_credit_name.artist_credit
INNER JOIN musicbrainz.release_group
    ON release_group.artist_credit = artist_credit.id
INNER JOIN musicbrainz.release
    ON release.release_group = release_group.id
INNER JOIN musicbrainz.release_country
    ON release.id = release_country.release
INNER JOIN musicbrainz.artist_type
    ON artist.type = artist_type.id
INNER JOIN musicbrainz.area
    ON artist.area = area.id
INNER JOIN musicbrainz.release_group_primary_type
    ON release_group_primary_type.id = release_group.type
WHERE
    ((release_country.date_year IS NOT NULL) AND
    (release_country.date_month IS NOT NULL) AND
    (release_country.date_day IS NOT NULL))
    ) As Dummy2
```

Gave:

```
 dummy
--------
 622527
(1 row)
```

Success - same row count 🙂

I do notice that some rows seems to be the same:



"Ellis Bell"

Time	release_name	release_year	album_id	album_primary_type_id	artist_credit_id	release_month
December 30th 2016, 15:14:33.283	Ellis Bell	2,013	d330fa8b-e337-49bc-9dea-6d13fb400d6b	3	1,219,999	9
December 30th 2016, 15:14:33.283	Ellis Bell	2,013	d330fa8b-e337-49bc-9dea-6d13fb400d6b	3	1,219,999	9
December 30th 2016, 15:14:33.283	Ellis Bell	2,014	d330fa8b-e337-49bc-9dea-6d13fb400d6b	3	1,219,999	6

Or not? The "album_id" is the same, but in one row the "release_year" differs from the two others....

Running the SQL

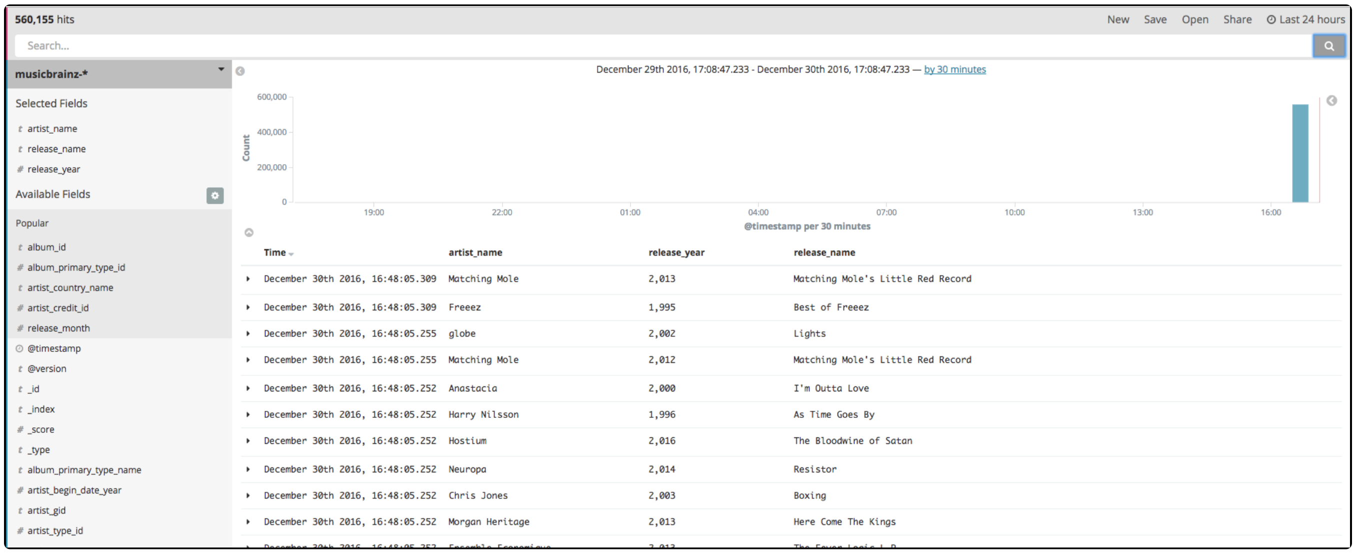
```
SELECT Distinct * FROM (
SELECT
    release_group.gid AS album_id,
    release_group.type AS album_primary_type_id,
    release_group_primary_type.name AS album_primary_type_name,
    release.name AS release_name,
    artist.name AS artist_name,
    artist.gid AS artist_gid,
    artist_credit.id AS artist_credit_id,
    artist.type AS artist_type_id,
    artist_type.name AS artist_type_name,
    artist.begin_date_year artist_begin_date_year,
    area.name AS artist_country_name,
    release_country.date_year AS release_year,
    release_country.date_month AS release_month,
    release_country.date_day AS release_day
FROM
    musicbrainz.artist
INNER JOIN musicbrainz.artist_credit_name
    ON artist_credit_name.artist = artist.id
INNER JOIN musicbrainz.artist_credit
    ON artist_credit.id = artist_credit_name.artist_credit
INNER JOIN musicbrainz.release_group
    ON release_group.artist_credit = artist_credit.id
INNER JOIN musicbrainz.release
    ON release.release_group = release_group.id
INNER JOIN musicbrainz.release_country
    ON release.id = release_country.release
INNER JOIN musicbrainz.artist_type
    ON artist.type = artist_type.id
INNER JOIN musicbrainz.area
    ON artist.area = area.id
INNER JOIN musicbrainz.release_group_primary_type
    ON release_group_primary_type.id = release_group.type
WHERE
    ((release_country.date_year IS NOT NULL) AND
    (release_country.date_month IS NOT NULL) AND
    (release_country.date_day IS NOT NULL))
    ) As Dummy2
```

Gave

```
560155 rows
```

So, there is a possible redundancy in the SQL provided from https://www.elastic.co/blog/logstash-jdbc-input-plugin

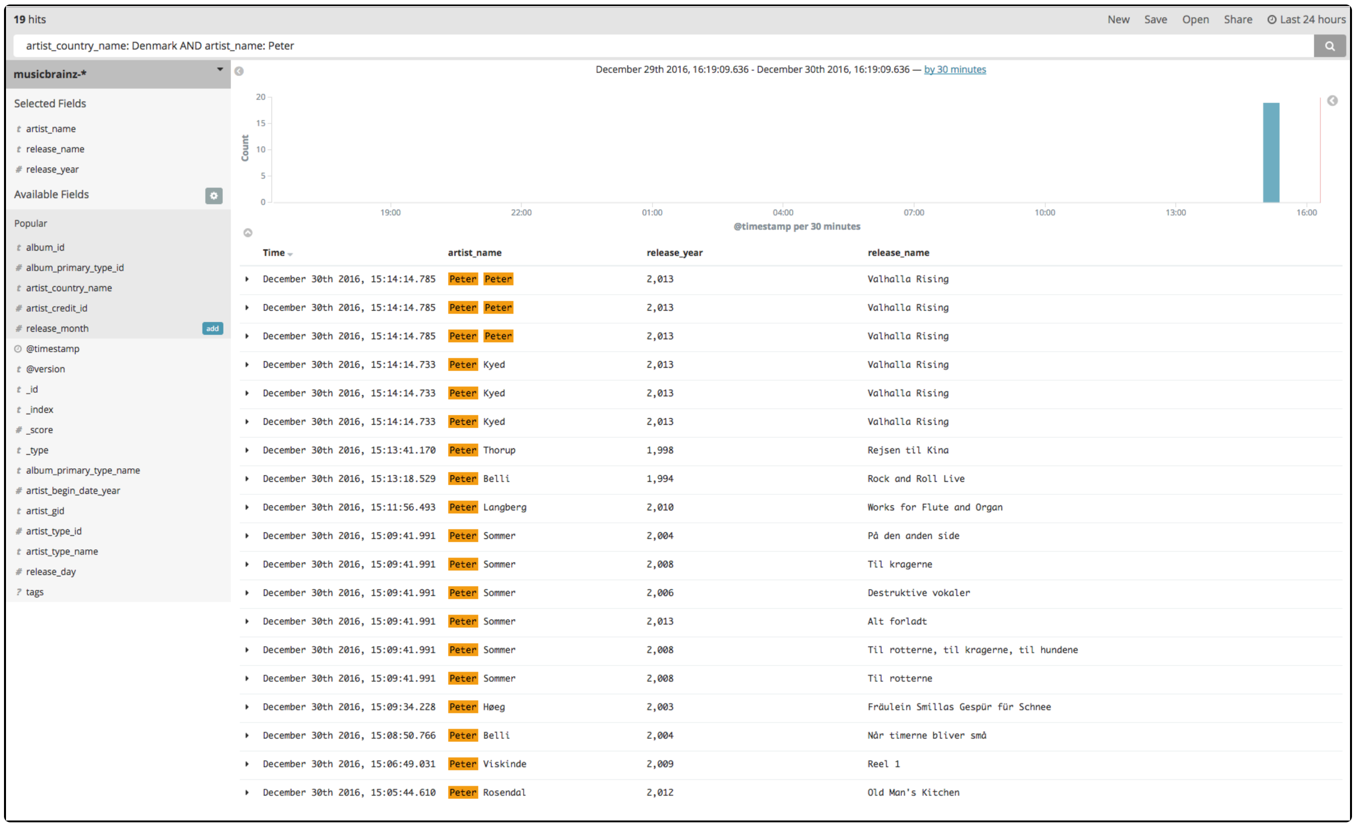
Reloading the data (after deleting the Index) gives:



Time	artist_name	release_year	release_name
December 30th 2016, 16:48:05.309	Matching Mole	2,013	Matching Mole's Little Red Record
December 30th 2016, 16:48:05.309	Freeez	1,995	Best of Freeez
December 30th 2016, 16:48:05.255	globe	2,002	Lights
December 30th 2016, 16:48:05.255	Matching Mole	2,012	Matching Mole's Little Red Record
December 30th 2016, 16:48:05.252	Anastacia	2,000	I'm Outta Love
December 30th 2016, 16:48:05.252	Harry Nilsson	1,996	As Time Goes By
December 30th 2016, 16:48:05.252	Hostium	2,016	The Bloodwine of Satan
December 30th 2016, 16:48:05.252	Neuropa	2,014	Resistor
December 30th 2016, 16:48:05.252	Chris Jones	2,003	Boxing
December 30th 2016, 16:48:05.252	Morgan Heritage	2,013	Here Come The Kings
December 30th 2016, 16:48:05.252	Ensemble Economique	2,013	The Fever Logic L.P

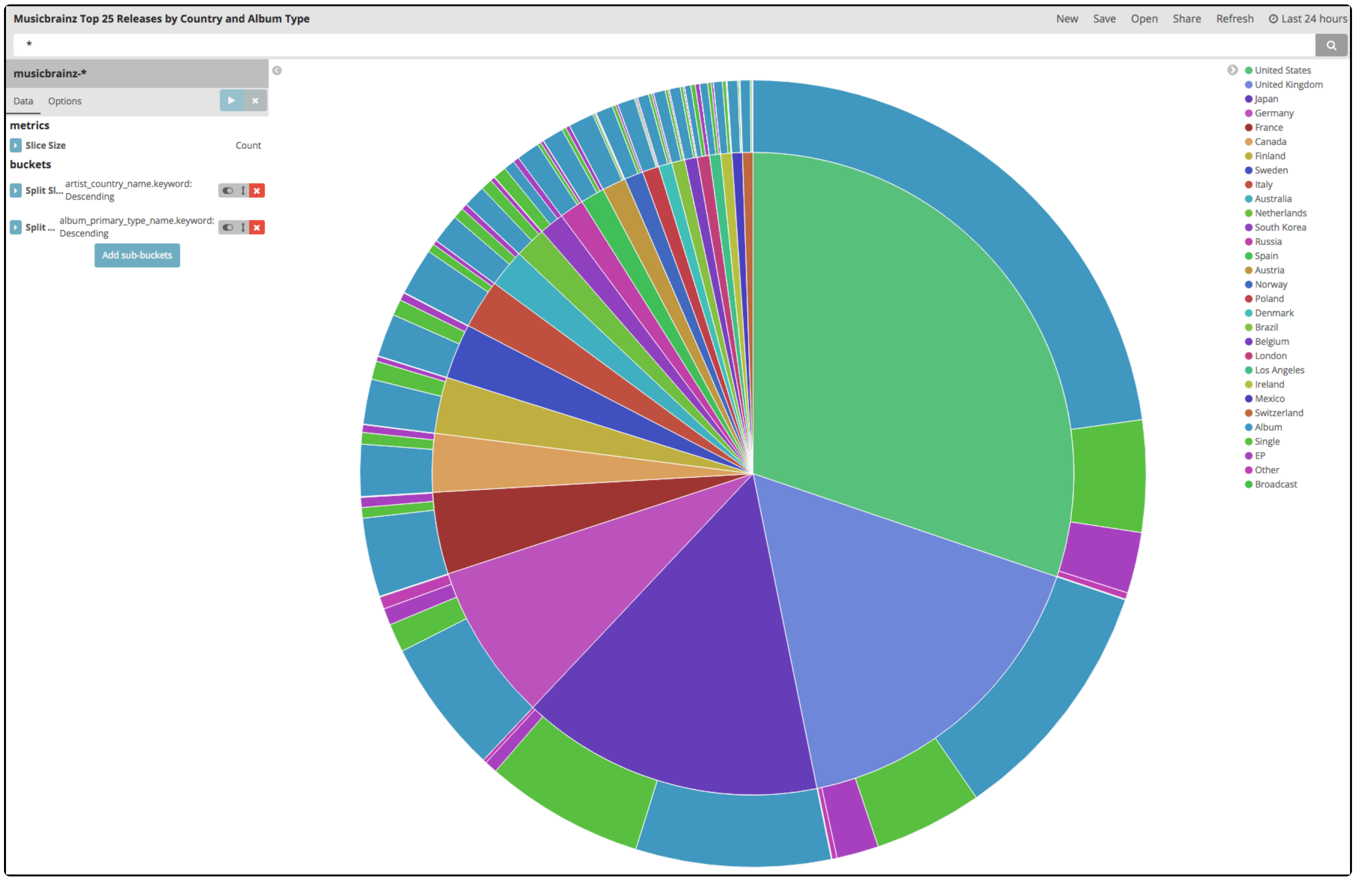
Samples

Here is a sample of all albums from "Denmark" with "Peter" in the Artist name:



Time	artist_name	release_year	release_name
December 30th 2016, 15:14:14.785	Peter Peter	2,013	Valhalla Rising
December 30th 2016, 15:14:14.785	Peter Peter	2,013	Valhalla Rising
December 30th 2016, 15:14:14.785	Peter Peter	2,013	Valhalla Rising
December 30th 2016, 15:14:14.733	Peter Kyed	2,013	Valhalla Rising
December 30th 2016, 15:14:14.733	Peter Kyed	2,013	Valhalla Rising
December 30th 2016, 15:14:14.733	Peter Kyed	2,013	Valhalla Rising
December 30th 2016, 15:13:41.170	Peter Thorup	1,998	Rejsen til Kina
December 30th 2016, 15:13:18.529	Peter Belli	1,994	Rock and Roll Live
December 30th 2016, 15:11:56.493	Peter Langberg	2,010	Works for Flute and Organ
December 30th 2016, 15:09:41.991	Peter Sommer	2,004	På den anden side
December 30th 2016, 15:09:41.991	Peter Sommer	2,008	Til kragerne
December 30th 2016, 15:09:41.991	Peter Sommer	2,006	Destruktive vokaler
December 30th 2016, 15:09:41.991	Peter Sommer	2,013	Alt forladt
December 30th 2016, 15:09:41.991	Peter Sommer	2,008	Til rotterne, til kragerne, til hundene
December 30th 2016, 15:09:41.991	Peter Sommer	2,008	Til rotterne
December 30th 2016, 15:09:34.228	Peter Høeg	2,003	Fräulein Smillas Gespür für Schnee
December 30th 2016, 15:08:50.766	Peter Belli	2,004	Når timerne bliver små
December 30th 2016, 15:06:49.031	Peter Viskinde	2,009	Reel 1
December 30th 2016, 15:05:44.610	Peter Rosendal	2,012	Old Man's Kitchen

A few Visualizations..


Musicbrainz Top 25 Releases by Country and Album Type

musicbrainz-*

metrics

Slice Size Count

buckets

Split Sl… artist_country_name.keyword: Descending

Split … album_primary_type_name.keyword: Descending

Whats Next..

Well, this is unfinished business...there are so much more data to combine......

And another project could be parsing IMDB data ... http://www.imdb.com/interfaces