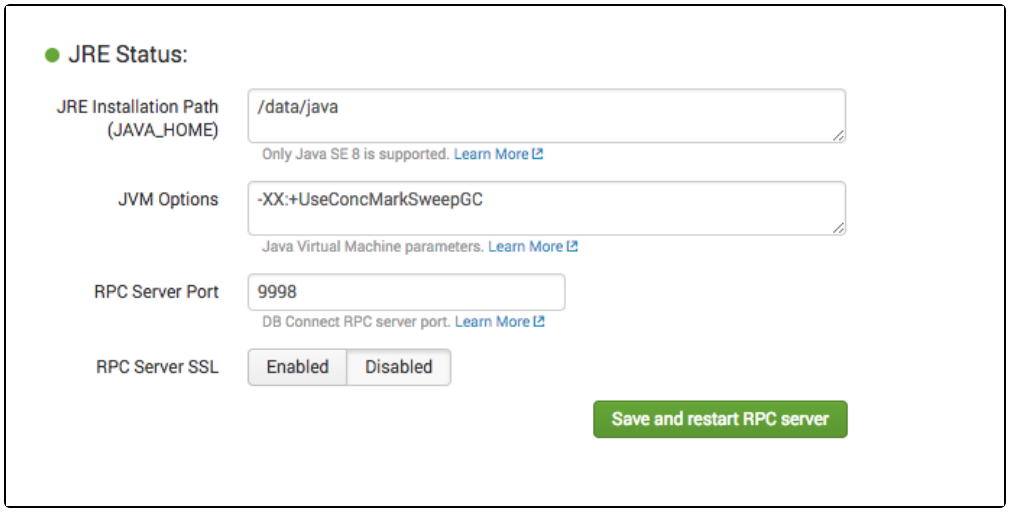
Loading Musicbrainz in Splunk

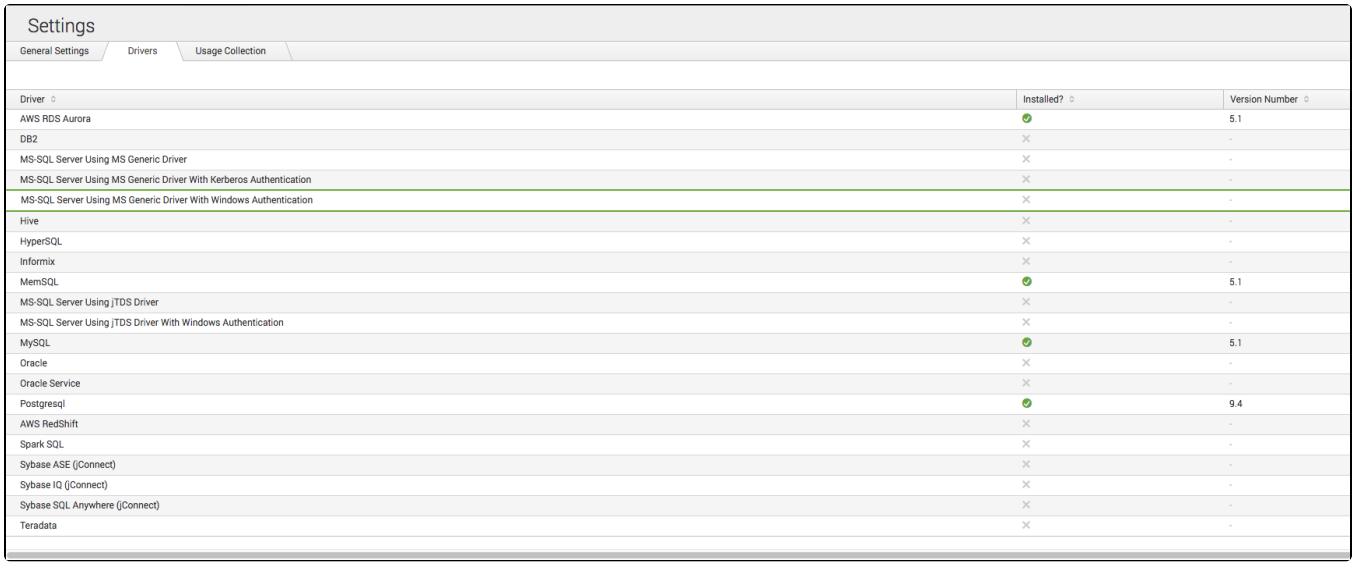
Refer to the https://musicbrainz.org/doc/MusicBrainz_Database for the Original setup

First part - Install and setup "Splunk DB Connect"

Afterwards, make sure JAVA is configed:



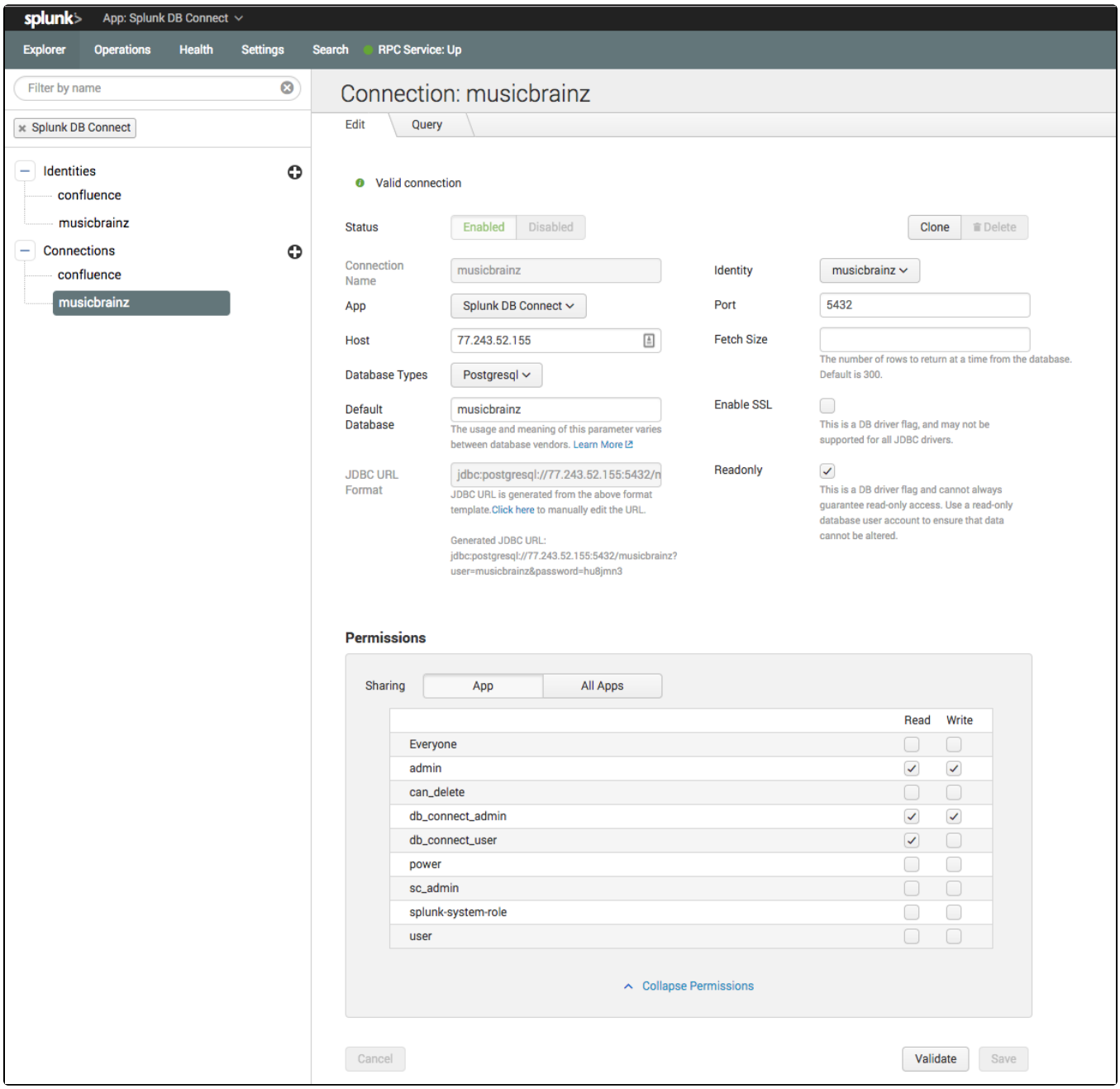
add the Postgres JDBC driver and check the Config:



Now, the core part is done.

Then setup Identities and Connection - this is pretty basic

> Remember to make sure the Databases You are connecting to are listening on 0.0.0.0 and NOT 127.0.0.1 - The 127.0.0.1 is typically default for Postgres and MySQL



Reffering to the previous ELK setup, we use the same query:

```
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
```

To get a preview:

Editor Mode | Max Rows 100 | Format SQL | Save As | Execute

```
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
```

✓ 100 rows

10 per Page | « prev 1 2 3 4 5 6 7 8 9 10 next »

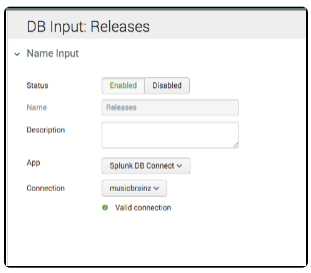
	album_id	album_primary_type_id	album_primary_type_name	release_name	artist_name	artist_gid	artist_credit_id	artist_type_id	artist_type_name	artist_begin_date_year
1	00000519-f708-35a3-be4e-cbe552ece701	1	Album	So Alone	Johnny Thunders	3b14b1a7-2235-4fcf-a718-383beb79e856	4645	1	Person	195
2	00000519-f708-35a3-be4e-cbe552ece701	1	Album	So Alone	Johnny Thunders	3b14b1a7-2235-4fcf-a718-383beb79e856	4645	1	Person	195
3	00007f9f-f34c-3713-97c0-b22ca5010087	2	Single	The X Files Theme (remix)	Mark Snow	a1d3685a-c61d-49aa-a294-bcfc25e516b5	2483	1	Person	194
4	00007f9f-f34c-3713-97c0-b22ca5010087	2	Single	The X-Files Theme	Mark Snow	a1d3685a-c61d-49aa-a294-bcfc25e516b5	2483	1	Person	194
5	00007f9f-f34c-3713-97c0-b22ca5010087	2	Single	The X-Files Theme	Mark Snow	a1d3685a-c61d-49aa-a294-bcfc25e516b5	2483	1	Person	194
6	0000aae9-737f-31bf-a9f0-6b7c3297ab1d	1	Album	Hier kommt Ärger!	Betontod	9ae757bc-89bb-45b7-800c-00959ea371dd	169642	2	Group	199
7	0000bbd3-8d64-4edc-88ac-f4c7bf01849f	2	Single	Sea of Flags	Jessica Mauboy	18f719e7-e9b4-4216-8869-9083ebc23f7d	571185	1	Person	198
8	0000d1ef-cbe5-398e-846f-7eff1a364cef	2	Single	It's My Life	No Doubt	fbd2a255-1d57-4d31-ac11-65b671c19958	464	2	Group	198
9	0000d1ef-cbe5-398e-846f-7eff1a364cef	2	Single	It's My Life / Bathwater (The Remixes)	No Doubt	fbd2a255-1d57-4d31-ac11-65b671c19958	464	2	Group	198
10	0000d1ef-cbe5-398e-846f-7eff1a364cef	2	Single	It's My Life	No Doubt	fbd2a255-1d57-4d31-ac11-65b671c19958	464	2	Group	198

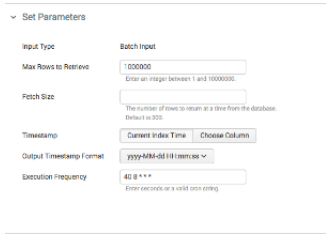
Where Logstash only gave us the possibility to "bulk" upload the Query result to Elasticsearch - "Splunk DB Connect" gives us 3 options:

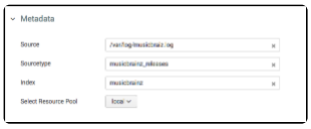
> This is where Splunk in my Opinion "runs over" the ELK stack - There are more GUI and both inputs (as Logstash) and on-the-fly lookup 🙂

DB Inputs

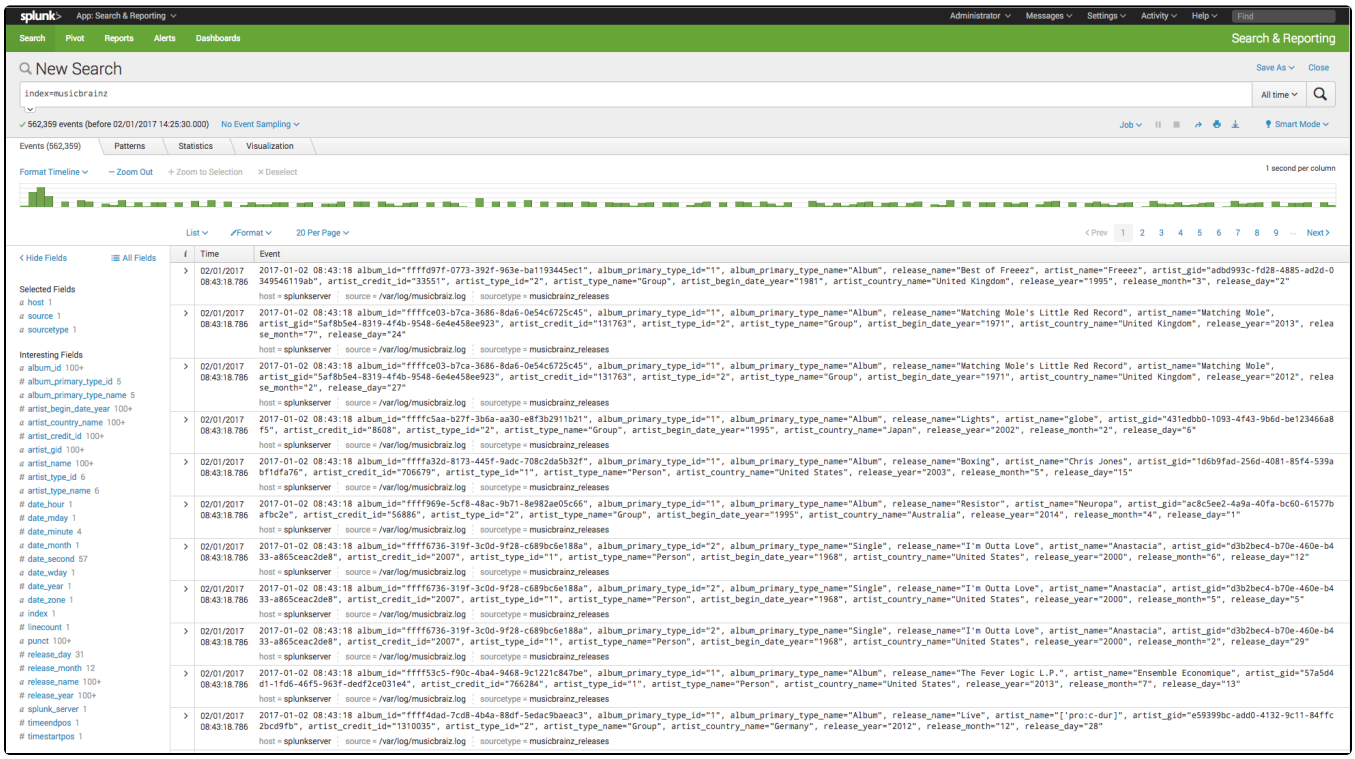
DB Inputs are "equal" to the Logstash approach, load the Query result into Splunk as "log lines":







Gives the loglines:



DB Outputs

DB Lookups