



# Creating a Transfer Operator PostgresqlToElasticsearch

Add a custom operator



# The Operator

In this lesson we are going to create an Operator to move data from PostgreSQL to Elasticsearch using the Hook we made previously.

Again, the goal here is not to make the perfect operator, but just to show you a good example of how to implement a useful transfer operator.

Let's do it!



# Instructions

- `vim`  
`~/airflow/plugins/elasticsearch_plugin/operators/elasticsearch_operator.py`
- `vim ~/airflow/plugins/elasticsearch_plugin/__init__.py`

# Instructions

- From the Airflow UI, click on “Admin”, “Connections” and “Create”.
- We are going to create the connection to Elasticsearch used by our Hook.
- Fill the fields with the same values and click on save.

Connection [create]

List Create

Conn Id	elasticsearch_default
Conn Type	
Host	localhost
Schema	
Login	
Password	
Port	9200
Extra	

Save Save and Add Another Save and Continue Editing Cancel

# Instructions

- From the Airflow UI, click on “Admin”, “Connections” and “Create”.
- We are going to edit the connection to PostgreSQL used by our Operator.
- Fill the fields with the same values and click on save.

## Connection [edit]

List Create Edit

Conn Id	postgres_default
Conn Type	Postgres
Host	localhost
Schema	airflow_mdb
Login	airflow
Password	
Port	5432
Extra	

Save Save and Add Another Save and Continue Editing Cancel

# Instructions

- `cp ~/airflow_files/plugin_operator_dag.py ~/airflow/dags/`
- `vim ~/airflow/dags/plugin_operator_dag.py`
- On the Airflow UI, trigger the DAG “plugin\_operator\_dag” as follow:

First, click here



Then, click here to trigger the DAG

- Then, go to the “Graph View” and wait for the DAG to finish.

# Instructions

- Once the DAG has finished executing you should have the following DAG from the “Graph View”:



- If you click on the task “postgres\_to\_es” and “View Log”, you will notice that many POST requests have been sent to Elasticsearch corresponding to the number of records we fetched from PostgreSQL.

```
[2018-11-15 03:35:35,757] {elasticsearch_operator.py:41} INFO - Extracting data from PostgreSQL: SELECT * FROM course.source
[2018-11-15 03:35:37,109] {logging_mixin.py:95} INFO - [2018-11-15 03:35:37,108] {base.py:83} INFO - POST http://localhost:9200/sources/external [status:201 request:1.350s]
[2018-11-15 03:35:37,207] {logging_mixin.py:95} INFO - [2018-11-15 03:35:37,207] {base.py:83} INFO - POST http://localhost:9200/sources/external [status:201 request:0.098s]
[2018-11-15 03:35:37,232] {logging_mixin.py:95} INFO - [2018-11-15 03:35:37,232] {base.py:83} INFO - POST http://localhost:9200/sources/external [status:201 request:0.024s]
[2018-11-15 03:35:37,244] {logging_mixin.py:95} INFO - [2018-11-15 03:35:37,244] {base.py:83} INFO - POST http://localhost:9200/sources/external [status:201 request:0.011s]
```



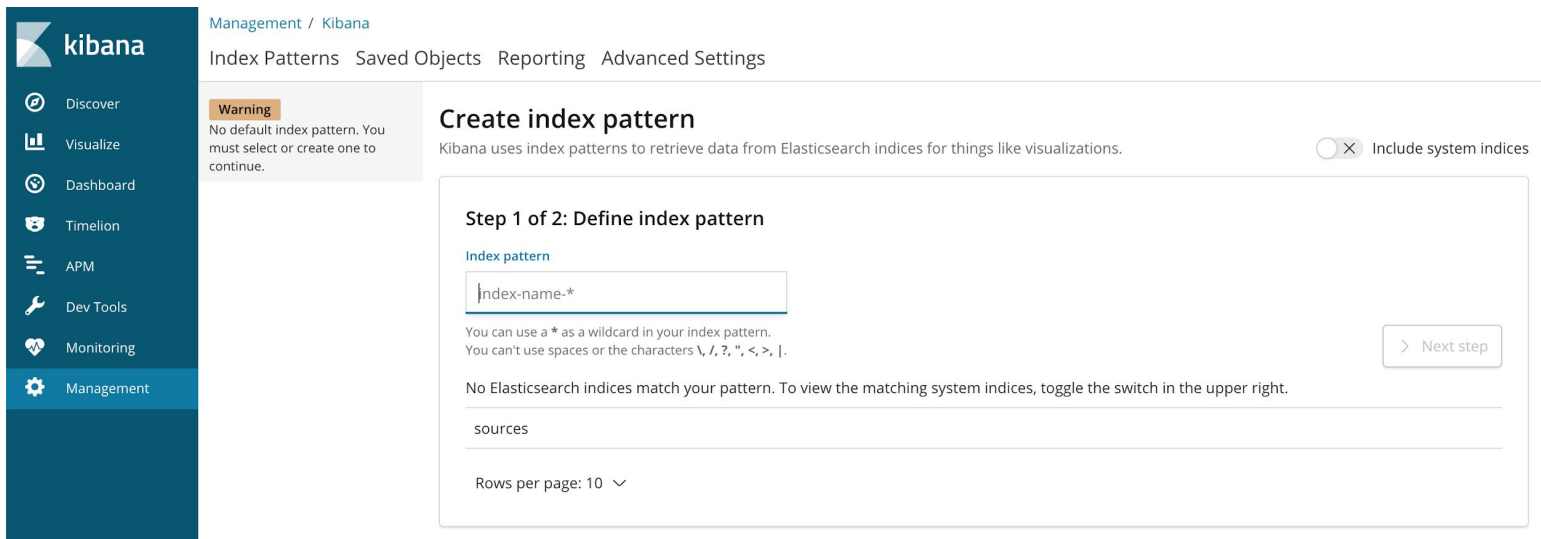
# Instructions

- Now, let's use Kibana ( the interface of Elasticsearch ) to see if all the data have been saved into Elasticsearch.
- I'm not going to explain how Elasticsearch works here, but just keep in mind that data are saved into an “index” representing a collection of documents.
- In our case, each document corresponds to a record saved from PostgreSQL.



# Instructions

- Go to <http://localhost:5601> and click on “Management” on the left sidebar,



The screenshot shows the Kibana Management interface. On the left is a dark blue sidebar with the Kibana logo and navigation links: Discover, Visualize, Dashboard, Timelion, APM, Dev Tools, Monitoring, and Management (which is highlighted). The main content area has a breadcrumb 'Management / Kibana' and a sub-header 'Index Patterns | Saved Objects | Reporting | Advanced Settings'. A warning box states: 'Warning: No default index pattern. You must select or create one to continue.' The main heading is 'Create index pattern', with a sub-note: 'Kibana uses index patterns to retrieve data from Elasticsearch indices for things like visualizations.' There is a toggle switch for 'Include system indices' which is currently turned off. The 'Step 1 of 2: Define index pattern' section contains an 'Index pattern' input field with the text 'index-name-\*'. Below this, explanatory text says: 'You can use a \* as a wildcard in your index pattern. You can't use spaces or the characters \, /, ?, ", <, >, |.' A 'Next step' button is to the right. Further down, it says 'No Elasticsearch indices match your pattern. To view the matching system indices, toggle the switch in the upper right.' Below this is a 'sources' section and a 'Rows per page: 10' dropdown menu.

kibana

Management / Kibana

Index Patterns | Saved Objects | Reporting | Advanced Settings

**Warning**  
No default index pattern. You must select or create one to continue.

## Create index pattern

Kibana uses index patterns to retrieve data from Elasticsearch indices for things like visualizations.

☐ Include system indices

### Step 1 of 2: Define index pattern

Index pattern

index-name-\*

You can use a \* as a wildcard in your index pattern.  
You can't use spaces or the characters \, /, ?, ", <, >, |.

Next step

No Elasticsearch indices match your pattern. To view the matching system indices, toggle the switch in the upper right.

sources

Rows per page: 10

# Instructions

- In the text field below “Index Pattern” type “sources”,

**kibana**

Management / Kibana

Index Patterns Saved Objects Reporting Advanced Settings

**Warning**  
No default index pattern. You must select or create one to continue.

## Create index pattern

Kibana uses index patterns to retrieve data from Elasticsearch indices for things like visualizations.

☐ Include system indices

### Step 1 of 2: Define index pattern

**Index pattern**

index-name-\*

You can use a \* as a wildcard in your index pattern.  
You can't use spaces or the characters \, /, ?, \*, <, >, |.

> Next step

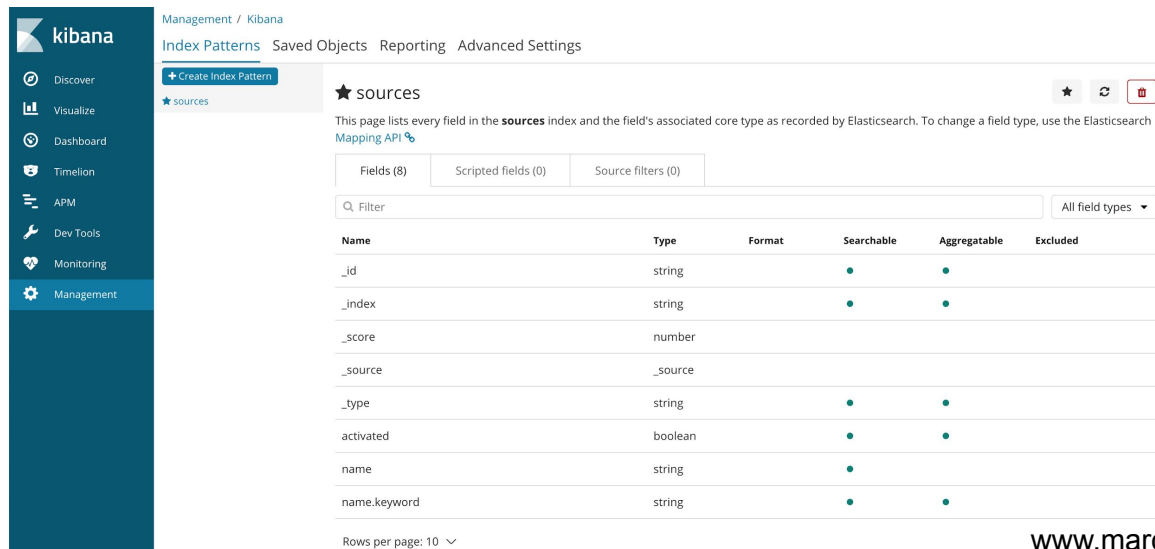
No Elasticsearch indices match your pattern. To view the matching system indices, toggle the switch in the upper right.

sources

Rows per page: 10

# Instructions

- Click on “Next Step” and “Create Index Pattern”. Once you have done that you should see the following screen:



The screenshot shows the Kibana Management interface. On the left is a dark blue sidebar with the Kibana logo and navigation links: Discover, Visualize, Dashboard, Timelion, APM, Dev Tools, Monitoring, and Management (which is highlighted). The main content area has a breadcrumb trail 'Management / Kibana' and tabs for 'Index Patterns', 'Saved Objects', 'Reporting', and 'Advanced Settings'. The 'Index Patterns' tab is active, showing a '+ Create Index Pattern' button and a list of index patterns, with 'sources' selected. The 'sources' page title is '★ sources'. Below the title is a description: 'This page lists every field in the **sources** index and the field's associated core type as recorded by Elasticsearch. To change a field type, use the Elasticsearch Mapping API'. There are three tabs: 'Fields (8)', 'Scripted fields (0)', and 'Source filters (0)'. The 'Fields (8)' tab is active, showing a search filter and a dropdown for 'All field types'. Below is a table with 6 columns: Name, Type, Format, Searchable, Aggregatable, and Excluded. The table lists 8 fields: \_id (string), \_index (string), \_score (number), \_source (string), \_type (string), activated (boolean), name (string), and name.keyword (string). Each field has green dots in the Searchable and Aggregatable columns. At the bottom left, it says 'Rows per page: 10'. At the bottom right, the URL 'www.marclamberti.com' is visible.

Management / Kibana

Index Patterns Saved Objects Reporting Advanced Settings

+ Create Index Pattern

★ sources

★ sources

This page lists every field in the **sources** index and the field's associated core type as recorded by Elasticsearch. To change a field type, use the Elasticsearch Mapping API.

Fields (8) Scripted fields (0) Source filters (0)

Filter All field types

Name	Type	Format	Searchable	Aggregatable	Excluded
_id	string		•	•	
_index	string		•	•	
_score	number				
_source	string				
_type	string		•	•	
activated	boolean		•	•	
name	string		•		
name.keyword	string		•	•	


Rows per page: 10

www.marclamberti.com



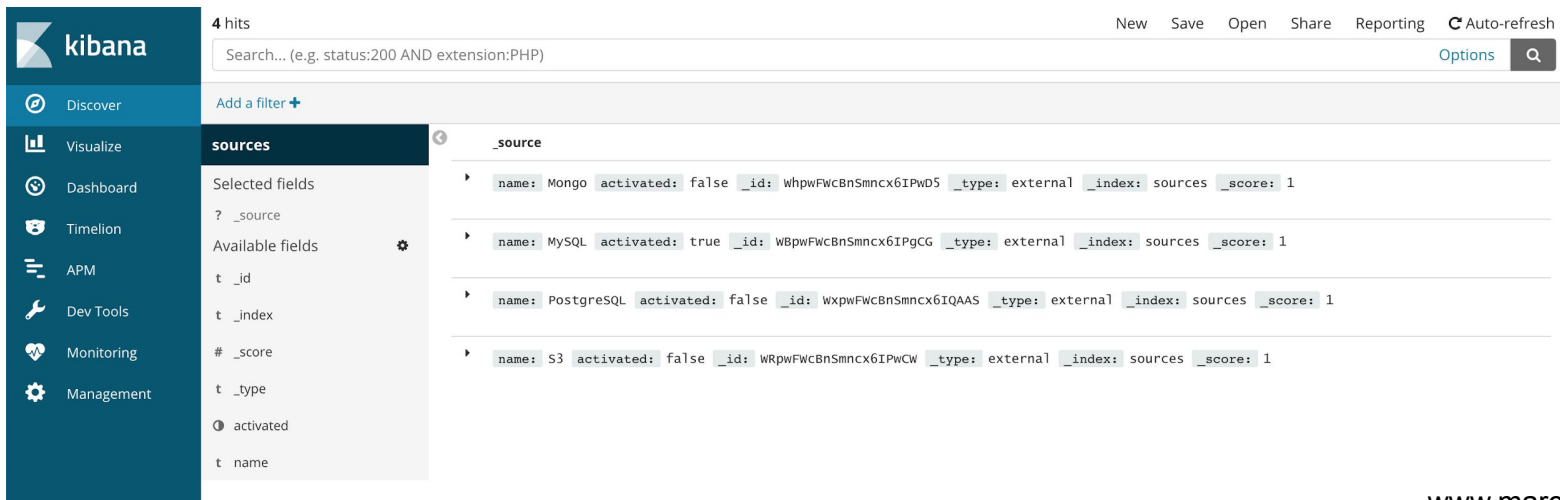
# Instructions

- If you take a look at the field names, you should notice that we get the same names corresponding to the column we have from PostgreSQL as you can see below,

Name	Type	Format	Searchable	Aggregatable	Excluded
_id	string		●	●	
_index	string		●	●	
_score	number				
_source	_source				
_type	string		●	●	
→ activated	boolean		●	●	
→ name	string		●		
name.keyword	string		●	●	

# Instructions

- Finally, click on “Discover” on the left sidebar to see the same data we have from PostgreSQL transferred to Elasticsearch.



The screenshot shows the Kibana Discover interface. The left sidebar contains navigation links: Discover, Visualize, Dashboard, Timelion, APM, Dev Tools, Monitoring, and Management. The 'Discover' tab is active. The main area displays search results for the query 'Search... (e.g. status:200 AND extension:PHP)'. The results are shown in a table with columns: name, activated, \_id, \_type, \_index, and \_score. The results are as follows:

name	activated	_id	_type	_index	_score
Mongo	false	whpwFwCBnSmncx6IPwD5	external	sources	1
MySQL	true	WBpwFwCBnSmncx6IPgCG	external	sources	1
PostgreSQL	false	wxpwFwCBnSmncx6IQAAS	external	sources	1
S3	false	WRpwFwCBnSmncx6IPwCW	external	sources	1



# Important Notes

- When you create an Operator, you must implement the method `execute(self, context)`