Adding a View to Apache Airflow UI

A new page in Airflow UI

The View

Apache Airflow allows you to add you own views making the UI customizable as much as you want. For example, you could show or hide links based on the current user according to your permission settings or you could add a view to expose current metrics about one of your external systems.

In this part of the section, we are going to create a specific view just to display some information about our Elasticsearch instance.

Let’s do this!

Instructions

- We are going to use 4 components to modify the Apache Airflow UI: a Blueprint, a View, a Menu Link and a Template.
- A quick reminder:
  - A view is derived from the base class flask_admin.BaseView. The view returns data that Flask turns into an ongoing response when a request is made on a specific URL.
  - A template is rendered with specific data coming from the associated view in order to produce a final html document. Templates use the Jinja template library.
  - A blueprint is the glue between the view, the template and the static folder.
  - A menu link is nothing more than a link into the header of the Apache Airflow UI.
- Now you remember the basics, let's continue to see how can we combine everything to customize the UI.

Instructions

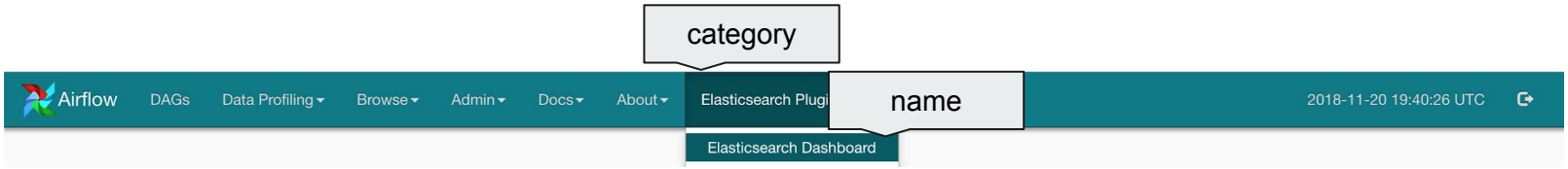
- `vim ~/airflow/plugins/elasticsearch_plugin/views/elasticsearch_view.py`
- If you take a look to the code you will notice that we actually use the hook we made earlier to interact with Elasticsearch directly from the view.
- Here, we fetch information about the Elasticsearch instance and we pass them as parameter to the function `render()` in order to make those data available from the template `elasticsearch_plugin.html`

Instructions

- `vim ~/airflow/plugins/elasticsearch_plugin/views/__init__.py`
- `__init__.py` is a special file in Python allowing you to mark directories as Python packages directories (`from airflow.elasticsearch_plugin.views`) and define variables at the package level.
- Unlike the operator and the hook we made previously, the view, blueprint and menu link need to be instantiated directly from the plugin (and not in a DAG file).
- For simplicity and clarity, we instantiate the view directly into the variable `ELASTICSEARCH_PLUGIN_VIEWS` in order to be used later into the class `ElasticsearchPlugin`describing our plugin.

Instructions

- Notice the two parameters category and name we have when we instantiate the view.
- The parameter “category” corresponds to the name of the top-level menu item. Basically, it will create a drop down menu in the header bar.
- The parameter “name” corresponds to the name of the menu item displayed into the header bar of Apache Airflow UI.



Instructions

- `vim ~/airflow/plugins/elasticsearch_plugin/templates/elasticsearch_plugin.html`
- This html page contains special codes in order to indicate to Jinja template engine where to apply the substitutions corresponding to the data given as parameter during the view instantiation.
- This page will be rendered by the view.

Instructions

- `vim ~/airflow/plugins/elasticsearch_plugin/blueprints/elasticsearch_blueprint.py`
- The parameter “template_folder” indicates where to find the associated html page to the view we made.
- Parameters “static_folder” and “static_url_path” give the name and the path to link static files such as images and scripts to the view.
- As we did for the view, the blueprint is instantiated from the `__init__.py`file into the variable `ELASTICSEARCH_PLUGIN_BLUEPRINTS`

Instructions

- `vim ~/airflow/plugins/elasticsearch_plugin/menu_links/elasticsearch_link.py`
- You actually don't need to create a Menu Link to add a new view into Airflow UI but if you want to add a menu item into the header it is the way to go.
- You have to specify a name corresponding to text of the menu item and if you want to add a drop down menu you just need to set a category value.
- The url is the link where you will be redirected to by clicking on the menu item.
- Again, like we did for the view and the blueprint, the menu link is instantiated from the `__init__.py` into the variable `ELASTICSEARCH_PLUGIN_LINKS`.

Instructions

- From the Airflow UI, if you take a look at the top bar, you will see a new menu item called “Elasticsearch Plugin”.
- If you click on it you will see two items
    - The first one called “Elasticsearch Dashboard” is the actual view we made
    - The second one called “More Info” is the menu link where you will be redirected on my website by clicking on it.

Instructions

- Now click on “Elasticsearch Dashboard” and you will see the following screen: