How to Guide:

Connect Grafana to ArangoDB

In this tutorial we will use Grafana with the Grafana Simple JSON connector for ArangoDB using ArangoDB 3.4.

In this tutorial we will use Grafana with the Grafana Simple JSON connector and ArangoDB 3.4.

This tutorial assumes you have access to a running instance of ArangoDB that already contains the data you want to export to Grafana. Note that the Grafana Simple JSON connector for ArangoDB only supports exporting aggregate data from a single collection.

Installing the connector

The Grafana connector can be installed as a Foxx service using the ArangoDB web interface or the Foxx CLI.

To install the service using the web interface:

1. Open the ArangoDB web interface in your browser (e.g. `http://localhost:8529` if ArangoDB is running locally).
2. Enter your ArangoDB credentials and select the database from which you want to export data to Grafana.
3. Select the *Services* tab on the right, press *Add Service* and select the *grafana-connector* service.
4. Enter a mount point (e.g. `/grafana`) and press *Install*.

To install the service using the Foxx CLI use the following command (assuming user `root`, database `_system` and mount point `/grafana`):

```
foxx install -u root -P -H http://localhost:8529 -D _system /grafana \
https://github.com/arangodb-foxx/grafana-connector/archive/master.zip
```

Configuring the connector

Before the Grafana connector is ready to use it needs to be configured:

- **Collections**: the names of collections in the current database the connector should have access to (multiple values can be separated by comma but only one collection can be imported at a time).

- **Aggregation** (default: `SUM`): name of the AQL aggregation function that will be used to aggregate results for Grafana. Should be one of `SUM`, `AVG` (or `AVERAGE`), `MIN` or `MAX`.

- **Date field** (default: `date`): name of the field on each document that will be used to find the documents relevant for each time range. The value of this field should be expressed in milliseconds since the start of the UNIX epoch.

- **Value field** (default: `value`): name of the field on each document that will be used to determine the value for that document. The value of this field must be numeric and may be aggregated when Grafana requests data for long time ranges.

- **Username** and **password**: credentials that will be used to protect the connector against unauthorized access. Note that these are different from the credentials used to access ArangoDB itself and will only be used by Grafana to authenticate against the connector.

Grafana always uses aggregates rather than individual data points. Depending on the type of data you may wish to use a different aggregation function. You may have to experiment a little to determine which aggregation function suits your data best.

Keep in mind that the Grafana connector only supports one aggregation function for all exposed collections. If you need to use different aggregation functions for different collections you can install multiple copies of the connector with different configurations on different mount points.

To configure the service from the web interface:

1. Select the *Services* tab and select the mount point where the Grafana connector service was installed.
2. Select the *Settings* tab from the top bar.
3. Fill in the configuration values and press the *Apply* button to save.

To configure the service using the Foxx CLI use the following command (assuming collection `data`, value field `latency`, username `grafana`, password `grafana123` and defaults for everything else):

```
foxx config -u root -P -H http://localhost:8529 -D _system /grafana \
collections=data valueField=latency username=grafana password=grafana123
```

Preparing Grafana

In order to be able to use Simple JSON data sources in Grafana you need to install the Simple JSON Data Source plugin using the `grafana-cli` on the machine on which Grafana is installed:

```
grafana-cli plugins install grafana-simple-json-datasource
```

You may have to restart the Grafana service for the new data source to become available.

Using the connector in Grafana

We'll start out by connecting a new data source by navigating to *Configuration > Data Sources* in the sidebar and clicking the *Add data source* button. Select the *SimpleJson* data source type. You can also find the connector more easily by entering "Json" in the search box.
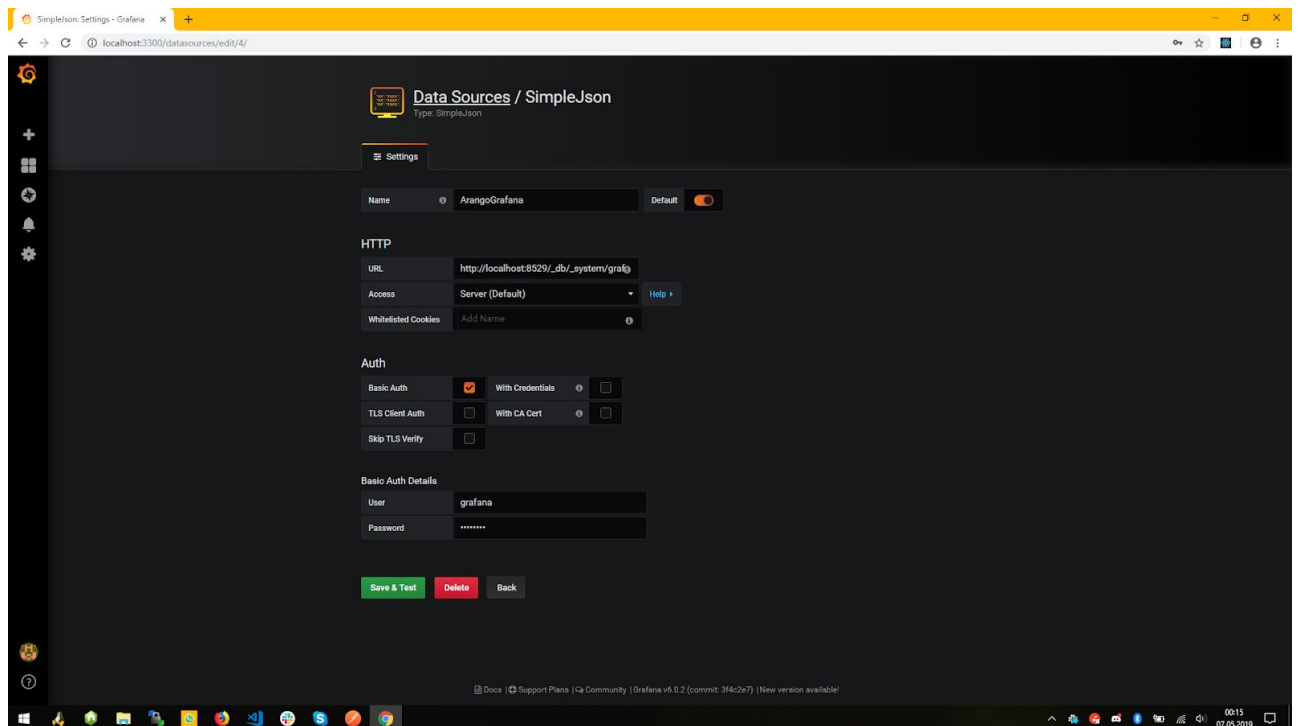
The **Name** can be anything but we will go with "ArangoGrafana" for this tutorial.

The **URL** should be the full path of the Grafana connector. For example, if we have a local copy of ArangoDB running on our machine and the Grafana connector was installed at `/grafana`, our URL looks like this:
`http://localhost:8529/_db/_system/grafana`

Note that the URL of the Grafana connector needs to be accessible from the Grafana server. You can alternatively set the **Access** to `Browser` but this requires every user to be able to access the Grafana connector URL from their machine.

You need to enable **Basic Auth**. The **User** and **Password** need to be taken from the configuration of our Grafana connector. Remember that these are not the credentials used to connect to ArangoDB itself. They merely protect the Grafana connector against unauthorized access.

Click the *Save & Test* button to ensure the data source is configured correctly, then click the *Back* button to proceed to the next step.

Data selection

We'll create a new dashboard by navigating to *Create > Dashboard* in the sidebar, which brings up the *New Panel* dialog.

Click on *Add Query* to switch to the query view. If you have a different default data source configured you may need to switch *Queries to* from the default to the newly created data source.
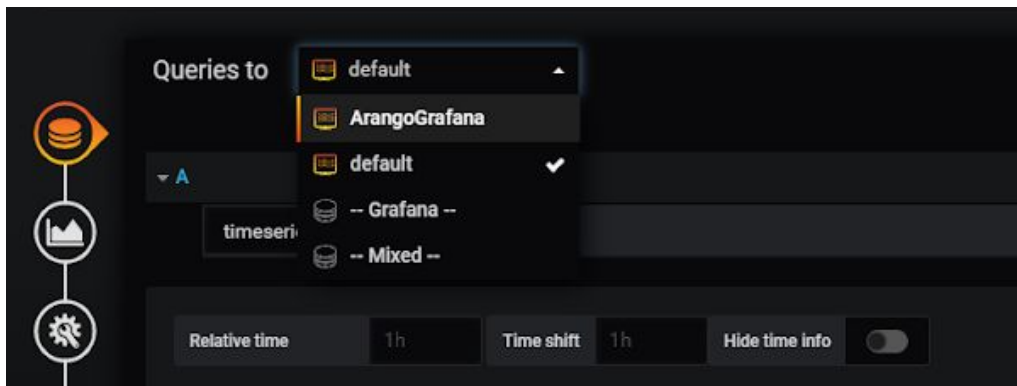
Click on *select metric* to see the available collections exposed by the Grafana connector and select a collection from the dropdown.

If the panel does not indicate an error but no data appears you may have to adjust the date range to a range that actually contains data.

Once you've ensured the data is being loaded correctly, proceed by clicking on the *Visualization* step in the sidebar.

Visualization

With the data properly loading into Grafana, you can now experiment with the different visualizations.