## Clone the repository from git hub, branch = regi

The sandbox is a fork from Influxdb sandbox where Grafana and all customization were applied on branch **regi**

SSH or HTTP methods can be used to clone it:

* git clone [git@github.com:rfcosta/sandbox.git](mailto:git@github.com:rfcosta/sandbox.git)
* git clone <https://github.com/rfcosta/sandbox.git>

One OR the other.

To start the stack go to the repository directory and do ./sabdbox up

To shut down do ./sandbox down

To enter any console do ./sandbox enter <docker name>

examples:

./sandbox enter grafana

./sandbox enter influxdb

The following directories are exposed to Grafana so you can have all scripts available to the container

- ./grafana/data/:/var/lib/grafana/

- ./grafana/config/:/usr/share/grafana/conf

- ./grafana/opt/aed:/opt/aed

- ./grafana/home:/opt/home

# Mount for influxdb data directory

- ./influxdb/data:/var/lib/influxdb

# Mount for influxdb configuration

- ./influxdb/config/:/etc/influxdb/

# Mount for telegraf configuration

- ./telegraf/:/etc/telegraf/

# Mount for Docker API access

- /var/run/docker.sock:/var/run/docker.sock

# Mount for kapacitor data directory

- ./kapacitor/data/:/var/lib/kapacitor

# Mount for kapacitor configuration

- ./kapacitor/config/:/etc/kapacitor/

## Initializing the database (once):

Enter influxdb container and execute the script on /etc/influxdb /etc/influxdb/initKPI-DB.sh

The source script is listed:

[JAMB17GHV2Q] /Users/sg0549743/\_\_github/influxdata> ./sandbox enter influxdb

Using latest, stable releases

Entering /bin/bash session in the influxdb container...

root@21b172cef3b1:/# cat /etc/influxdb/initKPI-DB.sh

#!/bin/sh

# INIT kpi database

alias ifx='influx -host 127.0.0.1 -port 8086 -execute '

ifx "SHOW DATABASES"

ifx "CREATE DATABASE \"kpi\" WITH DURATION 32d REPLICATION 2 NAME \"days\""

ifx "SHOW DATABASES"

ifx "CREATE RETENTION POLICY \"months\" ON \"kpi\" DURATION 56w REPLICATION 2"

ifx "CREATE RETENTION POLICY \"years\" ON \"kpi\" DURATION 520w REPLICATION 2"

ifx "CREATE CONTINUOUS QUERY \"kpi\_metric\_cq\_months\" ON \"kpi\" BEGIN SELECT mean(avg\_processing\_time) AS avg\_processing\_time, mean(error\_count) AS error\_count, mean(error\_rate) AS error\_rate, mean(transaction\_count) AS transaction\_count, mean(threshold\_dynamic\_lower) AS threshold\_dynamic\_lower, mean(threshold\_dynamic\_upper) AS threshold\_dynamic\_upper, mean(threshold\_static\_lower) AS threshold\_static\_lower, mean(threshold\_static\_upper) AS threshold\_static\_upper INTO \"kpi\".\"months\".\"metric\" FROM \"kpi\".\"days\".\"metric\" WHERE time > now() - 1h GROUP BY time(5m), \* END"

ifx "CREATE CONTINUOUS QUERY \"kpi\_threshold\_cq\_months\" ON \"kpi\" BEGIN SELECT mean(avg\_processing\_time\_crit\_lower) AS avg\_processing\_time\_crit\_lower, mean(avg\_processing\_time\_crit\_upper) AS avg\_processing\_time\_crit\_upper, mean(error\_count\_crit\_lower) AS error\_count\_crit\_lower, mean(error\_count\_crit\_upper) AS error\_count\_crit\_upper, mean(transaction\_count\_crit\_lower) AS transaction\_count\_crit\_lower, mean(transaction\_count\_crit\_upper) AS transaction\_count\_crit\_upper INTO \"kpi\".\"months\".\"thresholds\" FROM \"kpi\".\"days\".\"thresholds\" WHERE time > now() - 1h GROUP BY time(5m), \* END"

ifx "CREATE CONTINUOUS QUERY \"kpi\_metric\_cq\_years\" ON \"kpi\" BEGIN SELECT mean(avg\_processing\_time) AS avg\_processing\_time, mean(error\_count) AS error\_count, mean(error\_rate) AS error\_rate, mean(transaction\_count) AS transaction\_count, mean(threshold\_dynamic\_lower) AS threshold\_dynamic\_lower, mean(threshold\_dynamic\_upper) AS threshold\_dynamic\_upper, mean(threshold\_static\_lower) AS threshold\_static\_lower, mean(threshold\_static\_upper) AS threshold\_static\_upper INTO \"kpi\".\"years\".\"metric\" FROM \"kpi\".\"days\".\"metric\" WHERE time > now() - 1h GROUP BY time(15m), \* END"

ifx "CREATE CONTINUOUS QUERY \"kpi\_threshold\_cq\_years\" ON \"kpi\" BEGIN SELECT mean(avg\_processing\_time\_crit\_lower) AS avg\_processing\_time\_crit\_lower, mean(avg\_processing\_time\_crit\_upper) AS avg\_processing\_time\_crit\_upper, mean(error\_count\_crit\_lower) AS error\_count\_crit\_lower, mean(error\_count\_crit\_upper) AS error\_count\_crit\_upper, mean(transaction\_count\_crit\_lower) AS transaction\_count\_crit\_lower, mean(transaction\_count\_crit\_upper) AS transaction\_count\_crit\_upper INTO \"kpi\".\"years\".\"thresholds\" FROM \"kpi\".\"days\".\"thresholds\" WHERE time > now() - 1h GROUP BY time(5m), \* END"

ifx "CREATE DATABASE \"non\_kpi\" WITH DURATION 32d REPLICATION 1 NAME \"days\""

ifx "CREATE DATABASE \"\_kapacitor\" WITH DURATION 32d REPLICATION 1 NAME \"monitor\""

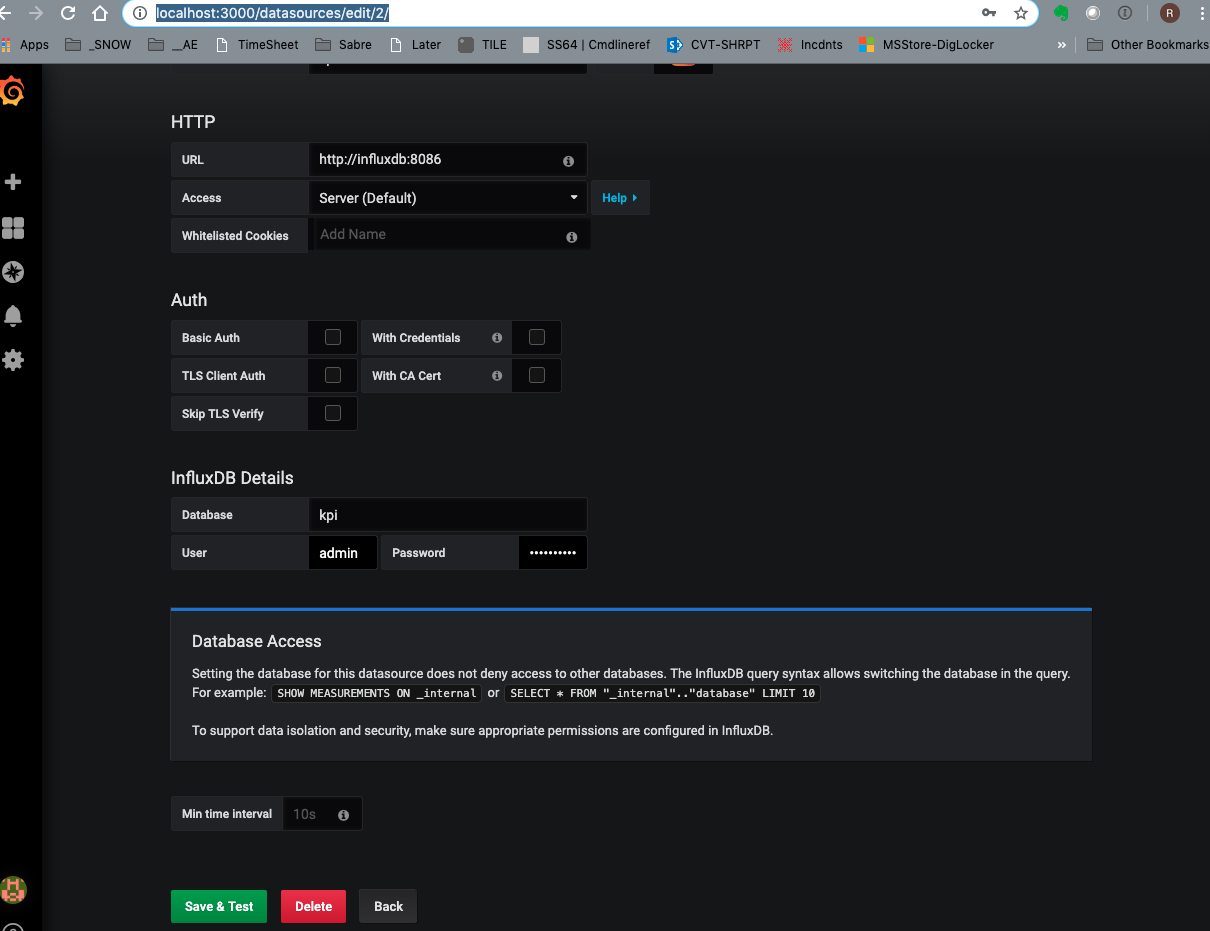
ifx "SHOW DATABASES"

root@21b172cef3b1:/#

## Create the datasource on Grafana for the newly created KPI database.

Use Grafana on <http://localhost:3000/datasources/edit/2/>

to define Data Source / kpi



## More details look at docker-compose.yml and sandbox script