Test Case Run | TCR756 InfluxDB connector

Status: ✔ Pass

Test Plan Run: TPR78 dashboard_tactics_pi_playback_input_TC105only_InfluxDB_connect

Test Plan: TP6 dashboard_tactics_pi_playback_input

Test Case: TC105 InfluxDB connector

Assigned To: Petri Makijarvi

Time Spent: 1,080.000

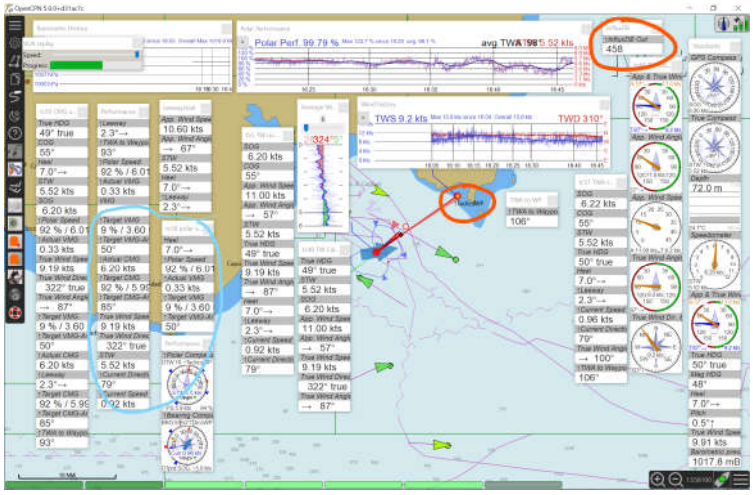
Precondition

It is required to have a HTTP-access (no https:// but http://) to a InfluxDB 2.0 server (alpha 14 or superior).

This data NMEA-0183 play-back file is used:
tc105_nmeastream.zip
A dashboard with all performance instruments and practically all other instruments is needed:
opencpn_ini_tc105.zip

Steps

1	Clean out the data directory of the Dashboard-Tactics plugin from all streaming data files and the streamout.json configuration file, if they exist. If there are no other files in the folder anymore, remove also the folder. In Windows C:\ProgramData\opencpn\plugins\dashoard_tactics_pi - in Linux ~/opencpnplugins/dashboard_tactics_pi	✔ Pass	30
2		✔ Pass	60



Launch the test case NMEA streaming (south of Sardegna). Add one single instrument "InfluxDB Out" instrument in specific window-pane. Start streaming the test NMEA-file. Place a Tactics WP ahead of the boat. The situation should look like the above. Note the need of Tactics instruments which are generating the calculated values which are to be also transmitted to the database.

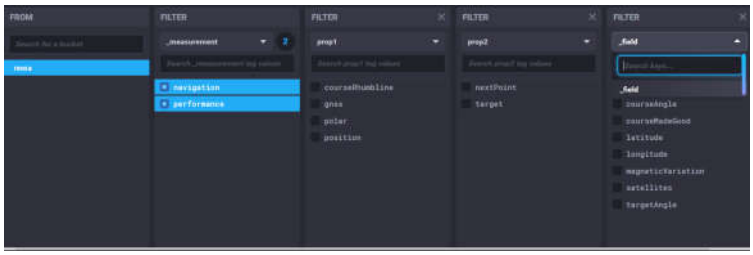
3	Observe that the above data directory has been created (if it was removed) and that both datafile *streamout* and the configuration file *streamout.json* have appeared. and that the counter is increasing in the *InfluxDB Out* "instrument" pane - it indicates lines written out per 100 lines. PASS or FAIL	✔ Pass ✖	20
4	Wait until the counter has reached minimum 500 write-outs.	✔ Pass ✖	360
5	Stop streaming. Observe the last written out number appearing in in *InfluxDB Out* pane. If it continues to slowly (every few seconds or so) to increase, this is an indication that the buffered data is still going out (there is place for 20 000 lines) - however the thread is getting slower when no more new data is coming out. This is a known bug caused by the thread being somewhat being event driven. For the time being, this is PASS, but it can change. PASS	✔ Pass ✖	30
6	Quit OpenCPN to definitely closing the streamout file.	✔ Pass ✖	

7 In InfluxDB2.0 UI, select a bucket (like *nmea* if you want to create one) and read the *streamout* data file in it, selecting first *milliseconds* as timestamp base. There should be no errors.

✔ Pass ✖ 30

PASS or FAIL

8 Go the data data exploring and select the bucket into which you uploaded the data. You should be able to see the following field names and tag properties:



✔ Pass ✖ 30

PASS or FAIL

9 Select now a wider range in time, and then select (*bucket - nmea*) -> performance -> polar -> target -> velocityMadeGood. The following graph should be able to be reproduced (graph with a single value):

✔ Pass ✖ 030

PASS or FAIL

10 Copy configuration template *streamout_template_http.json* from plug-ins program data directory, into the plug-in's data directory (above) and replace the actual *streamout.json* with it (name must be s*treamout.json* or you can, alternatively, change it in the *opencpn.ini* or *opencpn.conf* file.

Verify that in the file the server name is the same as your test InfluxDB HTTP port and that the organization name, bucket name and the Token are set according to the settings of your InfluxDB 2.0 server - you may need to set a new Token for this test unless you have a previous one. In a Docker based installation on a same machine, the modifications required

✔ Pass ✖ 360

may look like this:

```
   "influxdb" : {
      "target"    : "localhost:9999", // set to InfluxDB HTTP (no
https) API
      "api"       : "v2",           // HTTP: currently only v2 is
supported
      "org"       : "nmea",         // HTTP: Influx DB organization
name
      "bucket"    : "ocarina",      // HTTP: Influx DB bucket to
write to
      "precision" : "ms",           // HTTP: timestamps "local" ->
"ms"
      "token"     : "K4UnicxKbIgW6YpMEzjrtga2juy6BWjkntY-
EcxyaJxG6-rtOIN5AienkRnBE8FKNO0oAwIr9jVa-7-
SSIRVIQ=="
   }, // "token" only HTTP: InfluxAPI UI generates, allows
writing in "bucket"
```

The token is always unique to your installation and to your InfluxDB 2.0 bucket.

If you have issues with this step and with your server in the next step, please come back to this step, increase the debug level in the *streamout.json* to 3 and observe the tail of the *opencpn.log* - most often, the issues are with the bad token or other settings which do not allow the data to be accepted, the higher debug level will show information like this:

$ *tail -f /c/ProgramData/opencpn/opencpn.log | grep dashboard_tactics_pi*

5:41:36 PM: dashboard_tactics_pi: DB Streamer : Data Rejected (HTTP/1.1 404 Not Found

What went wrong? In the above I should have said:

```
      "org"       : "ocarina",      // HTTP: Influx DB organization
name
      "bucket"    : "nmea",         // HTTP: Influx DB bucket to
write to
```

Other errors may occur, such as not-authorized reply, in this case it is usually the Token which is wrong or which does not give you the access to the bucket for writing.

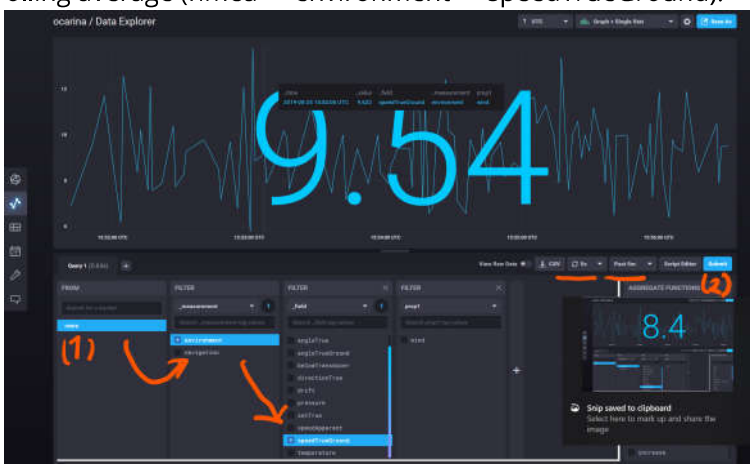
11 ✔ Pass ✖

Restart the test just like above, with the data file writing: you will notice that the counter in *InfluxDB Out* is now moving much slower, that is because there is less data to be streamed out - the performance of the HTTP connection is lesser than that of the local file system and the some selection of the data is made: the objective is to make to make some live histograms in the InfluxDB 2.0 and eventually using its Dashboard feature for those near-real-time-stream analysis.

The increasing counter does not guarantee that the data is actually accepted to be stored, only the disconnection from the server makes it stop to avoid buffer filling between plugin and the output thread. Next step is to verify that the data is actually streaming into InfluxDB 2.0.

12

In InfluxDB 2.0 UI, you should set the data refresh rate to 5 seconds loop in the data explorer, and show data from the past five minutes. In order to populate the filter selection, you may have still items from the previous test. Quite the data explorer and come back into it to select the data filters as indicated below, to see the last five minutes of TWS and its rolling average (*nmea* -> environment -> speedTrueGround):



PASS or FAIL

✔ Pass ✖ 30

13

✔ Pass ✖ 30

Stop now the streaming and observe the lines out indicator in *InfluxDB Out* pane: it should stop and not continue to roll - if it does, there is a performance issue, the HTTP-template file is throttled to a quite modes output to support low speed socket connection. The main issue is with the slow replies from the server, therefore one can attempt to adjust the number of lines to something bigger to reduce the number of writes/reads (there is a timeout of 5 seconds in HTTP stack so please progress in increments of 50 or so).
Nevertheless, if the default setting makes the test system to be sluggish, it is FAIL for this test.

PASS or FAIL

Reason For Status

Defects

Key	ID	Summary	Created↑	Assigned	Status	
D27	76	Provide data connect to InfluxDB 2.0 [D27]	3 minutes ago	canne	closed	

ADD DEFECT

Test Case History

RESULTS | DEFECTS | REQUIREMENTS

Status	Test Plan Run	Assigned To	Updated At↑	Actions
✔ Pass	TPR78 dashboard_tactics_pi_pl...	Petri Makijarvi	8 minutes ago	

ACTIVITY | HISTORY | COMMENTS

TODAY

Petri Makijarvi Updated Run TPR78 dashboard_tactics_pi_playback_input_TC105only_InfluxDB_connect
less than a minute ago / Aug 25, 2019

Petri Makijarvi Updated RunResultStep SR1415
3 minutes ago / Aug 25, 2019

Petri Makijarvi Updated RunResultStep SR1414
3 minutes ago / Aug 25, 2019

Petri Makijarvi Updated RunResultStep SR1413
3 minutes ago / Aug 25, 2019

Petri Makijarvi Updated RunResultStep SR1412
3 minutes ago / Aug 25, 2019