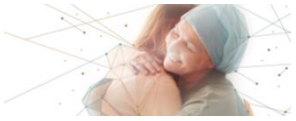


MOBYDQ

Testing tool - aims to automate Data Quality checks during data processing

MobyDQ does not provide any data profiling functionality. MobyDQ, previously termed "Data Quality Framework", by Alexis Rolland is a open DQ solution that aims to automate DQ checks during data processing, storing DQ measurements and metric results, and triggering alerts in case of anomaly.

We found this tool very difficult to install in a Windows environment and there is very little supporting documentation. After struggling to get the tool up and running, we found that the results were disappointing. Inspirata has provided instructions in the related directory provided.

Connectivity

Cloudera Hive, MariaDB, Microsoft SQL Server, MySQL, Oracle, PostgreSQL, SQLite, Teradata, Snowflake, Hortonworks Hive

UBISOFT MOBYDQ ANALYSIS

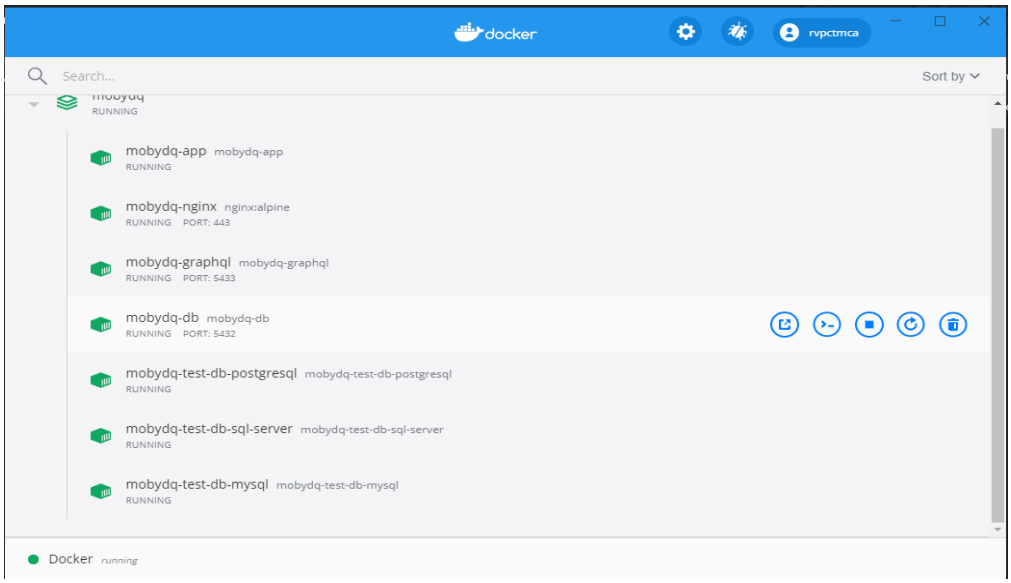
Prerequisites:

1. Download Windows Docker from"https://hub.docker.com/editions/community/docker-ce-desktop-windows" and install.
   Note: Ensure whether the WSL and Hyper V is enabled.
2. Download Ubisoft MobyDQ from the Github "https://github.com/ubisoft/mobydq" as a zip file and extract it.
   Note: If Github desktop is installed we can able to clone this repo also.
3. Download the Postgre SQL from the "https://www.enterprisedb.com/downloads/postgres-postgresql-downloads" and install.

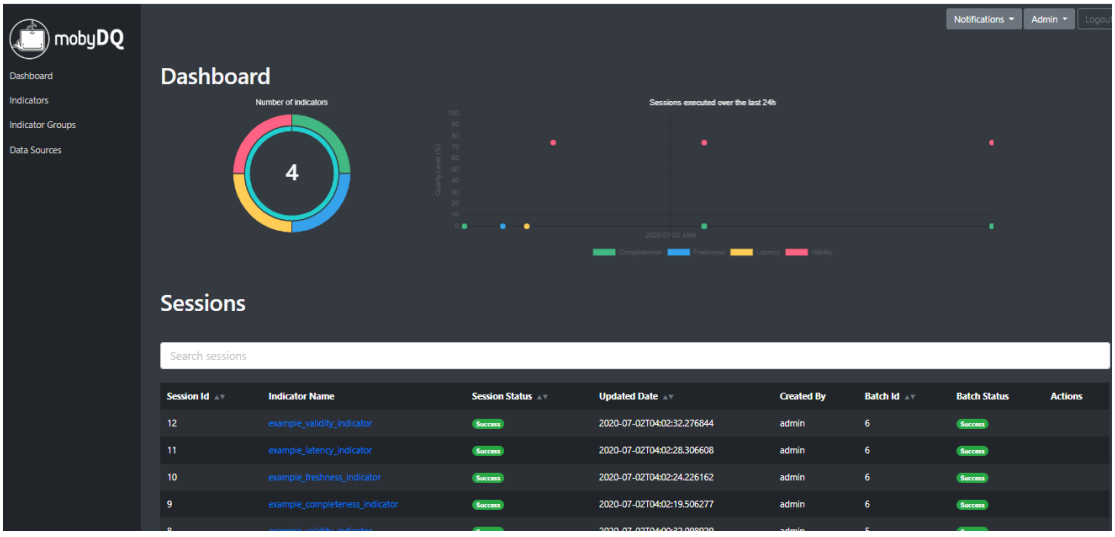
Note: Please note the password (master password) while installing which needs to be used in .env file.

4. Create an environment file (.env) file with the template as given in Moby DQ documentation. Give the Postgre SQL password and configure the Gmail settings.
5. Follow the Moby DQ documentation for creating and configuring the Nginx server.
6. Download the FreeTDS and place it in the MobyDQ root folder.
7. Follow the Moby DQ documentation for building and deploying the containers.

SAMPLE DOCKER:



SAMPLE MOBYDQ LANDING PAGE:



Note: Username: 'admin' and Password: 'admin'

COMPLETENESS:

gender	nb_people_source	nb_people_target	nb_people_delta	nb_people_delta_pe	Alert
female	19	19	0	0	TRUE
hermaphrodite	1	1	0	0	TRUE
male	62	62	0	0	TRUE
n/a	3	3	0	0	TRUE
none	2	2	0	0	TRUE

Freshness:

last_updated_date	current_timestamp	last_updated_date_c	last_updated_date_c	Alert
2014-12-20 21:17:57	2020-07-02 4:02:20	2909204	2020 days 06:44:23.	TRUE
2014-12-20 21:17:50	2020-07-02 4:02:20	2909205	2020 days 06:44:30.	TRUE
2014-12-20 21:17:50	2020-07-02 4:02:20	2909205	2020 days 06:44:30.	TRUE

Feedback from Neonatal Team:

Victor L Banda, a Data Analyst from the Neonatal Data Analysis Unit, Neonatal Medicine Research Group, Imperial College London, Chelsea and Westminster Hospital provided the following feedback:

Setup of MobyDQ on the windows platform was via the Docker Desktop for Windows. Though not successfully to fully run the app on our NNRD table, the two year follow up table, as with Orange and Data Cleaner, We explored it's setup via

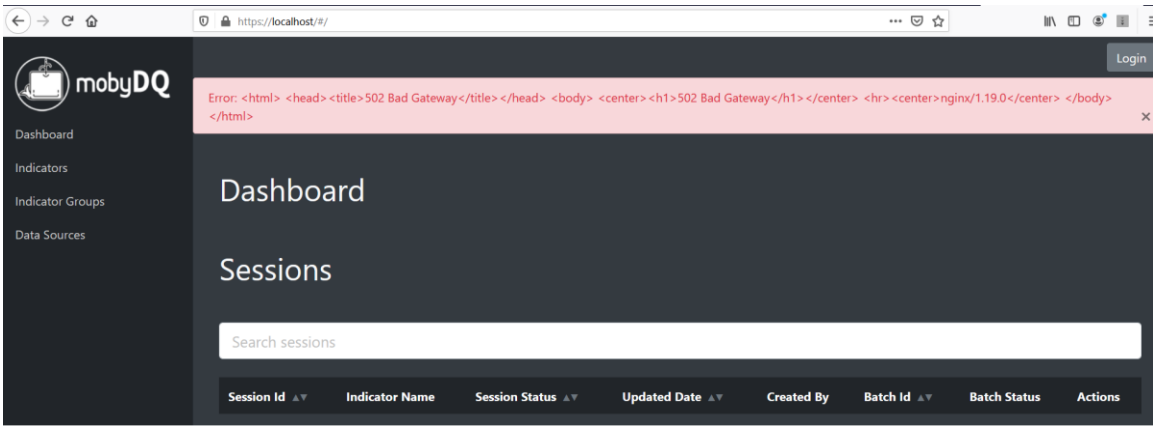
1. A pre-exisiting Postgres server:
   Creating the .env with database credentials of our pre existing postgres server encountered an error that seemed to be associated with private keys:

```
mobydq-graphql | ⚠ WARNING⚠  You requested to use schema 'base'; however we couldn't find
some of those! Missing schemas are: 'base'
mobydq-graphql | A serious error occurred when building the initial schema. Exiting because
`retryOnInitFail` is not set. Error details:
mobydq-graphql |
mobydq-graphql | Error: Could not find JWT type '"base"."token"'
mobydq-graphql |     at PgJWTPlugin/init/PgJWT (/home/node/app/node_modules/graphile-build-
pg/node8plus/plugins/PgJWTPlugin.js:47:13)
mobydq-graphql |       at SchemaBuilder.applyHooks (/home/node/app/node_modules/graphile-
build/node8plus/SchemaBuilder.js:252:20)
mobydq-graphql |      at SchemaBuilder.createBuild (/home/node/app/node_modules/graphile-
build/node8plus/SchemaBuilder.js:313:10)
mobydq-graphql |      at SchemaBuilder.buildSchema (/home/node/app/node_modules/graphile-
build/node8plus/SchemaBuilder.js:321:26)
mobydq-graphql     |                         at     Object.exports.createPostGraphileSchema
(/home/node/app/node_modules/postgraphile-core/node8plus/index.js:225:28)
mobydq-graphql |     at processTicksAndRejections (internal/process/task_queues.js:97:5)
mobydq-graphql      |                                  at      async      createGqlSchema
(/home/node/app/node_modules/postgraphile/build/postgraphile/postgraphile.js:77:33)
```

We explored with recreating the cert.pem and key.pem files in C:\mobydq\nginx\config, to no avail.
In the .env file, we also changed the POSTGRES_DB variable from mobydq to our custom database, but still more, the error showed up.
Finally, I ran the 00-database.sql script independently in Postgres, just to successfully create the base schema, though it was able to create the schema, for some reason this mobdy docker image could not interface with our Postgres instance, and therefore the no base schema error still showed up.

2. A Postgres docker Image:
With the above finding, it indicated as though our Postgres was not interacting with our mobydq docker image, and so we resorted to unistalling it and downloading and using a postgres docker image instead.
This presented issues of not being able to setup or identify postgres database credentials (username and password). Uisng the default ones, that were mentioned on https://hub.docker.com/_/postgres/, the username was obvious, but the POSTGRES_PASSWORD was tricky to identify.
The documentation stated that it is generated when the initdb script runs during initial container startup.
The location of the docker-entrypoint-initdb.d script was not found and this made it impossible to identify the correct password to specify in the .env file.
As a result we kept on getting a failed authentication for user postgres.

All in all, whether we run the app in production or development mode, we would only get a blank mobydql dashboard, through which we could not set parameters and test on our NNRD table.



.

The supporting document "**Data Quality Tool Assessment Request - Data Custodian Neonatal.docx**" contains all feedback provided by the Neonatal team.