Moja Reporting Tool

OPERATION GUIDE

02 December 2020



# 

# Introduction

This document is a fourth installment in a series of guides detailing how to set up and operate the Reporting Tool. Its role is to lay out a step by step guide on how to onboard the FLINT’s Flux Output Database into the Reporting Tool and have it processed and aggregated to give you Emissions Graphs and Reports. It assumes that you have already installed the Reporting Tool as detailed in the third guide: [Installing The Reporting Tool](https://docs.google.com/document/d/1xvTpzxOyuHvqk8YZN7-z0hm3infCWyBKHQxIlwsW1S8/edit).

# Prerequisites

We need to meet two preconditions in order to successfully operate the Reporting Tool:

|  | Prerequirement | |
| --- | --- | --- |
| 1 | Ascertaining that we the right to log in to the server and perform the operational actions | |
| 2. | Ascertaining that we have a compatible Flux Output Database Format to work with | |
| 3. | A shell Terminal to work from (the default in Linux or Mac and Powershell in Windows) | |

## 

# Outline

There are four key steps to successfully operating the Reporting Tool:

|  | Step | |
| --- | --- | --- |
| 1. | Check that all Reporting Tool Services are Up and Running | |
| 2. | Upload a Flux Output Database Backup to the Reporting Tool Application Server | |
| 3. | Onboard the Flux Output Database Backup | |
| 4. | Monitoring the Database Processing Event | |

We will explore each of these steps in detail in the sections that follow.

## 

# Operating The Reporting Tool

## 1. Check that all Reporting Tool Services are Up & Running

In order to check if all Reporting Tool services are up and running, we need to:

| 1.1. Open a shell Terminal on our local computer. | |
| --- | --- |

| 1.2. SSH into the Reporting Tool Server: | |
| --- | --- |
| ssh <username>@<reporting.moja.global> | |
| Please replace the username / Reporting Tool Server domain details with the correct ones | |

| 1.3. Invoke the following kubectl command: | |
| --- | --- |
| kubectl get pods −−all −namespaces | |

| 1.4. Check that all services have a status of Running | |
| --- | --- |

## 2. Upload Flux Output Database to the Reporting Tool Server

In order to upload a Flux Output Database to the Reporting Tool Server, we need to:

| 2.1. Open a shell Terminal on our local computer. | |
| --- | --- |

| 2.2. Change the working directory to the local folder with the Flux Output Database Backup: | |
| --- | --- |
| cd <local-flux-outputs-backups-directory> | |
| Please replace the <local-flux-outputs-backup-directory> with the correct directory | |

| 2.3. Invoke the following ‘shell copy’ command: | |
| --- | --- |
| scp <flux.out><username>@<reporting.moja.global>:~/data | |
| Please replace the flux backup file name, system username and Reporting Tool Server Domain details with the correct ones | |

## 3. Onboard The Flux Output Database into the Reporting Tool

In order to onboard the Flux Output Database into the Reporting, we need to:

| 3.1. Open a shell Terminal on our local computer. | |
| --- | --- |

| 3.2. SSH into the Reporting Tool Server: | |
| --- | --- |
| ssh <username>@<reporting.moja.global> | |
| Please replace the username / Reporting Tool Server domain details with the correct ones | |

| 3.3. Navigate to the data operations scripts directory: | |
| --- | --- |
| cd ~/scripts/operations/data | |

| 3.4. Invoke the database onboarding script: | |
| --- | --- |
| ./onboard\_database.sh | |

| 3.5. Follow the onboarding instructions: | |
| --- | --- |
| The onboarding script will ask us to enter the name of the Flux Output Database that we want to upload and the administrative Password of the PostgreSQL database that we want to upload it to to successfully upload and process the database | |

## 4. Monitor The Flux Output Database Processing Event

In order to monitor the Flux Output Database Processing Progress, we need to:

| 4.1. Open a shell Terminal on our local computer. | |
| --- | --- |

| 4.2. SSH into the Reporting Tool Server: | |
| --- | --- |
| ssh <username>@<reporting.moja.global> | |
| Please replace the username / Reporting Tool Server domain details with the correct ones | |

| 4.3. Invoke the following kubectl command: | |
| --- | --- |
| kubectl get pods −−all −namespaces | |

| 4.4. Get the name name of the database microservice pod: | |
| --- | --- |
| It name should look something akin to the following: database-xxxxxxxxx-xxxxx, where the x’s represent a unique combination of alphanumeric values | |

| 4.5. Inspect the database service logs: | |
| --- | --- |
| kubectl logs database-xxxxxxxxx-xxxxx | |

| 💡 | Please note that it is all possible to monitor the database processing status by examining the database event logs in a similar manner as above. It is also possible to make a curl request to the database event logs service to monitor the processing event. The curl request should look like this: curl [http://<reporting.moja.global>/databases\_events\_logs/all](http://reporting.moja.global/databases_events_logs/all) . Please replace [<reporting.moja.global>](http://reporting.moja.global/databases_events_logs/all) with the correct server name |
| --- | --- |