# **Linkage of OnSSET to the OpenMod4Africa Platform**

The Open Source Spatial Electrification Tool (OnSSET) can be installed on a local machine and requires a series of GIS input data and parameters (<https://onsset.readthedocs.io/en/latest/data_preparation.html>) Most of the of the open source spatial data available for several African countries can be downloaded from <https://energydata.info/dataset> and other sources are available at <https://onsset.readthedocs.io/en/latest/data_acquisition.html>.

Uploading the scenarios of the model requires summarizing the values from different OnSSET output files for each country according to the variables and scenario defined in OpenModAfrica.

The scenarios of the OnSSET model for the different countries can be downloaded by using the Global Electrification Platform (GEP) web site at <https://electrifynow.energydata.info/>.

The descriptions of the files generated by OnSSET and examples of the implementation of the OnSSET model can be found at <https://github.com/global-electrification-platform> and <https://global-electrification-platform.github.io/docs/preparing-the-data/scenario-results/>.

OnSSET can also be used to create new scenarios by using different parameters and input data, by following the specifications and documentation published at <http://www.onsset.org/get-started.html>.

**Scenarios File naming convention**

OnSSET generates one file for each possible scenario and the names of the output files include the information necessary to identify and describe the different scenarios.

The file name matches the pattern **CC-V\_T\_S\_G\_P\_I\_R.csv**, in which the different descriptive components of the scenarios are separated by the underscore character “\_” according to the following specifications.

***CC***

The first two letter are the two letters country codes followed by “-“: <https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2>

***V***

The integer number is the progressive version of the GEP model (implementation of OnSSET). The most recent version is 2, the version 1 files are superseded and no longer used.

***T***

Target electricity consumption level (0,1,2). Definition: Examines the electricity demand target expressed in kWh/capita/year. The top-down scenarios introduce consistent Tier based demand targets across all settlements based on their urban (U) and rural (R) status. In the low demand scenario, the urban demand target is informed by the average consumption observed in currently electrified areas in the country, translated into the nearest access Tier (e.g. U4 – Urban Tier 4). Low rural demand target is always set to Tier 1 (e.g. R1 – Rural Tier 1). The high demand target scenario reflects more aggressive goals. Urban demand target is increased by 1 Tier, unless already Tier 5 while rural demand target is increased, while rural demand target is increased by 2 Tiers. The bottom-up scenario assigns a unique demand target (kWh/cap/year) in each settlement, based on local poverty rate and GDP level.

0: Bottom-up demand target (Poverty GDP)

1: Top-down demand target- Low (example U4R1)

2: Top-down demand target – High (example U5R3)

***S***

Social & Commercial uses

0: Social and productive uses demand included

1: Residential demand only

***G***

Grid generating cost of electricity.

0: Estimated on grid cost (example 0.052 $/kWh)

1: High on grid cost (example 0.052 $/kWh)

***P***

PV System cost

0: Expected PV cost

1: Low PV cost (-50%)

***I***

Intermediate investment plan

0: Not capped

1: Capped annual connections

Definition: Indicates the electrification rate to be achieved in the first few years of the analysis (2025). In GEP it is assumed that final electrification rate is 100%, analysis considers 10 years (2020-2030) with an intermediate time step of 5 years. Not capped scenario assumes that the electrification rate increases linearly over the modelling years; no restrictions are set in terms of feasible grid connections per year. Capped growth scenario assumes that the electrification rate increases as in the first scenario but is subject to grid connections limitation, equal to 2.5% of population per year. In this case, grid electrified population slowly ramps up over the years of analysis following often an S-like curve.

***R***

Rollout plan

0: Nationwide Least Cost approach

1: Grid connection within 2 km

More information related to the different input parameters can be found in the documentation of OnSSET at <https://onsset.readthedocs.io/en/latest/index.html>.

## **Workflow for Scenario Explorer**

The OnSSET files are available in CSV format and can be optionally converted by using a PowerShell script, that allows to summarize the results by using MS Excel formulas and Pivot tables. The values are then copied and pasted in the corresponding OnSSET data in MS Excel. The script is available at the linkage GitHub repository: <https://github.com/openENTRANCE/linkages/>.

OnSSET generates 96 files for each country, with different spatial resolution, according to the distribution of the human population. A Python script is available to analyze the data stored in the OnSSET CSV files and insert the values in the OpenMod4Africa scenario MS Excel file, for each of the 96 scenarios considered. The script is available in the openENTRANCE linkage GitHhub repository.

*Step 1*

Generate the scenarios by using the OnSSET software or download the OnSSET output files from <https://electrifynow.energydata.info/> and <https://energydata.info/dataset>. Please note the size of the output file for each country is often more than 5 GB.

*Step 2*

Unzip di scenario output files in one new folder. Optionally, the files can be converted to MS Excel by using the script “csv\_to\_xls.ps1” provided.

*Step 3*

Run the Python code. The script will read all the scenarios files (up to 96 files) in the folder and generate an Excel file with all the data summarized by country according to the nomenclature specified in OpenEntrance. The Excel file will be named by default result.xlsx, but it can be renamed. Note that summarizing the information stored in more than 5GB of more than 90 csv files could take a considerable amount of time according to the performance of your hardware.

The file that will be generated is compatible with the IAMC format and can be uploaded to the Scenario Explorer. It is suggested to validate the output file by using the validate function from the nomenclature before proceeding to the last step and upload the Excel file by using the Scenario Explorer web interface.

### *Step 4*

### *Uploading OnSSET results to the OpenMod4Africa Scenario Explorer.*

The nomenclature for the variables is available in the OnSSET OpenENTRANCE repository on GitHub. The variables, regions and scenarios of the OnSSET data MS Excel file must be present in GitHub to allow the upload to complete successfully. The workflow explained above explains the steps to ensure that the format and the content of the file is appropriate for the Scenario Explorer platform.

The OnSSET data MS Excel file can be uploaded by using the upload tools of the Scenario Explorer platform.

## **How to install and run the Python script**

The tool requires Python 3.12.x. If Python is not installed, the user should download Python (<https://www.python.org/downloads/>) and install it for the current user.

The environment variables can be created by typing the following instructions:

<Installation path>\python.exe -m venv env

python3.12 -m venv env

source env/bin/activate (for Linux)

.\env\Scripts\activate.bat (for Windows)

(env) python -m pip install -U pip setuptools wheel

(env) python -m pip install -r requirements.txt

(env) cd src

(env) python main.py --help

A file .env must be stored in the same folder of the main.py file and must include the specify the years of reference for the OnSSET scenario and a local folder, as explained in the readme.md file, according to the pattern key-value.

CURRENT\_YEARS='[2025,2030]'

DELETE\_TEMPORARY\_FILE=0|1

LOCAL\_WORKING\_FOLDER=...

LOG\_LEVEL=DEBUG|INFO|WARNING

The script can be started by typing the following command:

(env) python main.py parse-csv-files

Please note that OnSSET can generate around 20 GB of csv files for each country and therefore the scripts can require a considerable amount of time to complete the generation of the Excel file.