

ompp_export_csv.py

Parsa Abadi

January 19, 2025

1 Introduction

`ompp_export_csv.py` is a Python script designed to export all output tables from an OpenM++ SQLite scenario database into a series of CSV files. This allows users to analyze or share the scenario data in a common, lightweight format outside the SQLite environment.

2 Features

- Reads all output tables from a given SQLite database.
- Exports the contents of each table into a corresponding CSV file.
- Ensures a clean, textual representation of model output data, making it easy to integrate with spreadsheets or other analysis tools.

3 Prerequisites

To use `ompp_export_csv.py`, you need:

- A valid SQLite database file produced by an OpenM++ model run.
- Python 3.x installed on your system.
- The `common.py` module and any other dependencies located in the same directory as `ompp_export_csv.py`.

4 Usage

Run the script from the command line as follows:

```
python ompp_export_csv.py <source-sqlite> <destination-folder>
```

Arguments

<source-sqlite> The path to the SQLite database file containing the scenario's model outputs.

<destination-folder> The path to a directory where the CSV files will be created.

Notes on the Destination Folder

Important: The output directory you provide must not already exist. The script requires a non-existing directory so it can create a fresh location where CSV files are placed. If the directory already exists, the script will refuse to run to prevent accidental overwriting of existing files.

5 Examples

Example 1

Suppose you have a scenario database named `Scenario.sqlite` and you want to export the tables into a new directory called `results`. Run:

```
python ompp_export_csv.py Scenario.sqlite results
```

If `results` does not exist yet, the script will create it and place all extracted CSV files inside.

Example 2

If the directory `results` already exists, you will see an error message:

```
ERROR =====> ompp_export_csv: Output directory for csv files 'results' must not already exist
```

In this case, choose a different directory name or remove/rename the existing one.

6 Error Handling

If the script cannot find the specified SQLite database, or if the database is empty, it will log an error and exit. Similarly, if the output directory already exists, it will not proceed, helping to ensure that no existing data is accidentally overwritten.