Developer Manual

GLIMPSE is a graph-based desktop application to visualize and update GridLAB-D power grid models. GLIMPSE is a cross-platform desktop application developed using modern web standards technologies.

Libraries and Languages Used

- JavaScript
  - Electron.js
  - React.js
  - Vis.js
  - Node.js
  - Webpack
- Python
  - Netowrkx
  - Flask
  - Flask-SocketIO
  - glm

Build Guide

Latest stable release is available as an installer on github release page Release Additionally, user can also clone and build the tool locally as highlighted below.

Begin by downloading Node and Nim

- Node.js
- Nim (Only if planning to export glm files updated with GLIMPSE tool)
- In a directory of your choice clone the repository :

```
git clone http://github.com/pnnl/GLIMPSE
```

- Then in `GLIMPSE/glimpse` :

```
npm install
```

- After all node modules are installed run the following command to bundle the source code using webpack:

```
npm run webpack
```

- After that in `GLIMPSE/glimpse/local-server` create a python environment:

```
python -m venv glimpse-server
```

- Once the environment is created activate it using one of the following command for your system in the table below:

Platform	Shell	Command to activate virtual environment
POSIX	bash/zsh	`$ source glimpse-server/bin/activate`
-	fish	`$ source glimpse-server/bin/activate.fish`
-	csh/tcsh	`$ source glimpse-server/bin/activate.csh`
-	PowerShell	`$ glimpse-server/bin/Activate.ps1`
Windows	cmd.exe	`C:\> glimpse-server\Scripts\activate.bat`
-	PowerShell	`PS C:\> glimpse-server\Scripts\Activate.ps1`

- You will know if the environment activation worked if there is a `(glimpse-server)` indicator at the start of your command line.
- Next install the server's requirements:

```
pip install -r requirements.txt
```

Additional Instructions for MacOS with Apple Silicon

In `glimpse/local-server/` clone the glm parser repository.

```
git clone https://github.com/NREL/glm.git
```

Then in `glm/` you will then build the glm parser. For this you need to make sure that `nim` is installed and added to your computers `PATH`.

```
nim c -d:release --opt:size --cpu:arm64 --passC:"-flto -target arm64-
apple-macos11" --passL:"-flto -target arm64-apple-macos11" --app:lib --
```

```
out:lib/_glm.so src/glm.nim
```

Next run the following command to create the glm python library wheel

```
python setup.py bdist_wheel --plat-name=macosx_11_0_arm64
```

Once that is done, in `glm/dist/` there is a `.whl` archive that you are able to install using pip to the local `glimpse-server` python environment

```
python -m pip install ./dist/glm-0.4.4-py2.py3-none-macosx_11_0_arm64.whl
```

Start GLIMPSE

In `GLIMPSE/glimpse/` start the application with the following command:

```
npm run start
```

Supported Files

JSON

GLIMPSE supports two different JSON file structures:

1. GLIMPSE's data structure which is based on the glm2json parser used by GLIMPSE.
   Examples:
   - example 1
   - example 2
   - example 3
   - example 4
2. Networkx's node_link_data JSON dump function
   Example:
   - fishing example

GLM (GridLAB-D Model)

Examples:

We provide few examples of exploring standard IEEE bus models using GLIMPSE. From the home page, upload all the `.glm` files from `GLIMPSE/glimpse/data/123-bus-model/`.