ArchMap Documentation

Release

James Fawcus-Robinson, Johannes Löthberg

Oct 31, 2017

Contents

The ArchMap project creates a map of Arch Linux users all over the world.

CHAPTER 1

About

In this section:

ArchMap

archmap generates *GeoJSON* and *KML* files which can be used to display a map of Arch Linux users, it does this by parsing data from the ArchWiki.

One rendering of the data is on a map over at mapbox.com - This is updated manually so it may be out of date.

Have a look at the ArchMap page on the ArchWiki for more information about this project.

The documentation is hosted by readthedocs.org.

Installation

You can install `archmap` from PyPi by running `pip3 install archmap` or from the AUR by installing the archmap-git package.

Synopsis

By default, running `archmap` will output three files to /tmp, `archmap_users.txt`, `archmap.geojson` and `archmap.kml`, this can be overridden by either using the config file or by the command line switches.

The config file should be placed in `/etc/archmap.conf`, this can be overridden by using `--config <path-to-config-file>`

Use

Running `archmap --help` will display this help message:

```
archmap [-h] [-v] [--config FILE] [--url URL] [--file FILE] [--users FILE] [--geojson␣
↪FILE] [--kml FILE] [--csv FILE]

optional arguments:
-h, --help      show this help message and exit
-v, --verbose   Show info messages
--config FILE   Use an alternative configuration file instead of /etc/archmap.conf
--url URL       Use an alternative URL to parse the wiki list from
--file FILE     Use a file to parse the wiki list from
--users FILE    Output the list of users to FILE, use 'no' to disable output
--geojson FILE  Output the GeoJSON to FILE, use 'no' to disable output
--kml FILE      Output the KML to FILE, use 'no' to disable output
--csv FILE      Output the CSV to FILE, use 'no' to disable output
```

License

Everything in the ArchMap repo is unlicensed.

All of the files that this script can generate (`archmap_users.txt`, `archmap.geojson`, `archmap.kml`, and `archmap.csv`) will contain text from the ArchWiki which puts them under the GNU Free Documentation License 1.3 or later.

Unlicense

This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or distribute this software, either in source code form or as a compiled binary, for any purpose, commercial or non-commercial, and by any means.

In jurisdictions that recognize copyright laws, the author or authors of this software dedicate any and all copyright interest in the software to the public domain. We make this dedication for the benefit of the public at large and to the detriment of our heirs and successors. We intend this dedication to be an overt act of relinquishment in perpetuity of all present and future rights to this software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For more information, please refer to <http://unlicense.org/>

External links

Protect info

- Project page on the ArchWiki - https://wiki.archlinux.org/index.php/ArchMap
- Project talk on the Arch forums - https://bbs.archlinux.org/viewtopic.php?id=22518
- Arch Women wiki page - https://archwomen.org/wiki/aw-tech:archmap

Data

- The list of user data - https://wiki.archlinux.org/index.php/ArchMap/List
- Pre-generated *GeoJSON*, *KML* and *CSV* files - https://archwomen.org/media/archmap/

Code

- Repository on GitHub - https://github.com/guyfawcus/ArchMap
- Stats on Ohloh - https://www.ohloh.net/p/ArchMap
- Code search on sourcegraph.com https://sourcegraph.com/github.com/guyfawcus/ArchMap

CHAPTER 2

Install

In this section:

System Requirements

Python 3.4 - If your running Arch, this shouldn't be a problem!

- geojson
- simplekml

How-to

See also: *Packaging*

Manual git install

This will make a directory `ArchMap/` with a link to `archmap` in it. It uses *git* and *pip* to resolve the dependencies.

Download `this script`:

```
echo -e "==> Make and cd to ./ArchMap\n"
mkdir ArchMap && cd ArchMap

echo -e "\n\n==> Download the ArchMap repo from GitHub\n"
git clone https://github.com/guyfawcus/ArchMap.git ArchMap-git

echo -e "\n\n==> Install the required packages\n"
pip3 install -r ArchMap-git/requirements.txt

echo -e "\n\n==> Make an easy link to archmap.py\n"
cd ../ && ln -s ArchMap-git/archmap.py ./archmap

echo -e "\n\n==> Test by printing the help message\n"
./archmap --help
```

Support

External links

Release Notes

See also: Releases on GitHub

CHAPTER 3

Use

If you would just like to get up and running, have a look at:

Examples

All of these example assume you have installed **archmap** via pacman or pip, if you would like to use the script directly, just use `./archmap.py`.

Help

The **–help** flag will output a help message with all of the available options:

```
archmap --help
```

Basic use

By default, running **archmap** will output three files to /tmp, **archmap_users.txt**, **archmap.geojson** and **archmap.kml**, this can be overridden by either using the config file or by the following command line switches.

Using the **–verbose** flag will print information on what the script is doing:

```
archmap --verbose
```

You can specify the output location for the user list text, GeoJSON, KML and CSV:

```
archmap --users /tmp/archmap_users.txt --geojson /tmp/archmap.geojson --kml /tmp/
↪archmap.kml --csv /tmp/archmap.csv
```

If you would like to parse an alternate copy of the wiki list, simply pass either the –url or –file flags:

```
archmap --url https://wiki.archlinux.org/index.php?title=ArchMap/List&oldid=131196
```

or

```
archmap --file "$HOME/Downloads/ArchMap_List - ArchWiki.html"
```

Logging

If the script is run on a system that uses systemd, it will log to it using the syslog identifier - "archmap".

You can review all logs generated by **archmap** by using:

```
journalctl SYSLOG_IDENTIFIER=archmap
```

If you would like to use any of this code, have a look at:

Use the code

Getting and parsing user data

`archmap.`**`get_users`**(*url='https://wiki.archlinux.org/index.php/ArchMap/List'*, *local=''*)

This funtion parses the list of users from the ArchWiki and returns it as a string.

Parameters

- **`url`** (*`string`*) – Link to a URL that points to a ArchWiki ArchMap list (default)
- **`local`** (*`string`*) – Path to a local copy of the ArchWiki ArchMap source

Returns The raw text list of users

Return type string

`archmap.`**`parse_users`**(*users*)

This function parses the wiki text from `users` into it's components.

Parameters **`users`** (*`string`*) – Raw user data from the ArchWiki

Returns A list of lists, each sub_list has 4 elements: `[latitude, longitude, name, comment]`

Return type list

Output generators

`archmap.`**`make_users`**(*parsed_users*, *output_file*)

This function reads the raw text supplied by `users`, it then writes it to `output_file`.

Parameters

- **`parsed_users`** (*`list`*) – A list of lists, each sub_list should have 4 elements: `[latitude, longitude, name, comment]`
- **`output_file`** (*`open`*) – Location to save the text output

`archmap.`**`make_geojson`**(*parsed_users*, *output_file*)

This function reads the user data supplied by `parsed_users`, it then generates GeoJSON output and writes it to `output_file`.

Parameters

- **`parsed_users`** (*`list`*) – A list of lists, each sub_list should have 4 elements: `[latitude, longitude, name, comment]`
- **`output_file`** (*`open`*) – Location to save the GeoJSON output

`archmap.`**`make_kml`**(*parsed_users*, *output_file*)

This function reads the user data supplied by `parsed_users`, it then generates KML output and writes it to `output_file`.

Parameters

- **`parsed_users`** (*`list`*) – A list of lists, each sub_list should have 4 elements: `[latitude, longitude, name, comment]`
- **`output_file`** (*`open`*) – Location to save the KML output

`archmap.`**`make_csv`**(*parsed_users*, *output_file*)

This function reads the user data supplied by `parsed_users`, it then generates CSV output and writes it to `output_file`.

Parameters

- **`parsed_users`** (*`list`*) – A list of lists, each sub_list should have 4 elements: `[latitude, longitude, name, comment]`
- **`output_file`** (*`open`*) – Location to save the CSV output

CHAPTER 4

Contribute

In this section:

Roadmap

- Add more tests
- Work on packaging
- Use GitHub pages to build a homepage
  - Use Leaflet to get and display coords on a ...
  - MapBox map

Contributing

Contributions are always welcome! Here are a few ways you could contribute:

- Bug fixes
- New tests
- New features
- Testing on different platforms
- Documentation

Support: *External links*

Development

All of the following commands assume you are are starting in the root `ArchMap` directory.

System Requirements

In addition to the *System Requirements* for the install, the following packages are required:

- To generate these docs:
  - sphinx
- For packaging:
  - setuptools
  - wheel (optional) - for building *wheels*

Documentation

Sphinx can be used to build a variety of formats.

First, make sure you're in the docs directory:

```
cd docs/
```

Make the preferred output:

```
make html
```

Open the the index page in your browser:

```
firefox _build/html/index.html
```

Testing

`unittest` is used for testing:

```
python setup.py test
```

This will search the `tests` directory for tests.

To check your commits before submitting, it is advisable to set up pre-commit first. Install it with:

```
pip3 install pre-commit
```

Then install the hooks so that they automatically run before each commit:

```
pre-commit install
```

Make sure the hooks are up to date:

```
pre-commit autoupdate
```

To run the hooks before a commit use:

```
pre-commit run --all-files
```

For further information, have a look at the pre-commit advanced features page or the `.pre-commit-config.yaml` config file to see what is run.

See also:

- unittest - Python docs

Packaging

ArchMap is currently packaged in two forms.

Arch Linux package

Packages are built using the `PKGBUILD` and `archmap.install` for settings.

To build package using the PKGBUILD:

```
cd pkgbuild
makepkg PKGBUILD
```

Related issues:

- #3 PKGBUILD - **Closed**
- #9 PKGBUILD: Update pkgbuild with new deps and manpage - **Closed**

See also:

- Creating packages
- Python Package Guidelines

Python package

Packages are built using `setup.py` and `setup.cfg` for settings.

To build a source distribution:

```
python setup.py sdist
```

To build a wheel:

```
python setup.py bdist_wheel
```

Related issues:

- #8 Build a python package - **Open**

See also:

- Installation & Packaging Tutorial

CHAPTER 5

Indices and tables

- genindex
- modindex
- search

Index

G

M

P