esis Documentation

Release 0.1.0

Javier Collado

May 13, 2015

Contents

Contents:

esis

Elasticsearch Index & Search

- Free software: MIT license

- Documentation: https://esis.readthedocs.org.

1.1 Features

- Index content for every SQLite database row in Elasticsearch

- Search indexed content

1.2 Usage

- Index every SQLite database row under a given directory (recursively)

```
esis index <directory>
```

- Search for a given string in the indexed data

```
esis search <query>
```

- Get information about the number of indexed documents

```
esis count
```

- Delete all indexed documents

```
esis clean
```

1.3 Docker containers

Docker files are included in the source code to run esis and elasticsearch in their own containers. To build/pull the images needed to run esis and start the elasticsearch server, use the following commands:

```
docker-compose build
docker-compose start
```

After that, to launch esis in a container run:

```
docker-compose run esis <subcommand>
```

where *<subcommand>* is any of the subcommands in the previous section (*index*, *search*, *count* or *clean*).

Note:

- If *docker-compose run* is executed too quickly, then a connection error might be returning meaning that elasticsearch is still initializing.

- The entry point in the esis container uses the *–host* command line option to connect to the linked container where elasticsearch is running.

- The user home directory is mounted in the esis container as */data*. This must be taken into account when passing a directory to the *index* subcommand using a path in the container, not in the host machine.

Installation

At the command line:

```
$ easy_install esis
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv esis
$ pip install esis
```

Usage

To use esis in a project:

```
import esis
```

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

You can contribute in many ways:

4.1 Types of Contributions

4.1.1 Report Bugs

Report bugs at https://github.com/jcollado/esis/issues.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

4.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with "bug" is open to whoever wants to implement it.

4.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with "feature" is open to whoever wants to implement it.

4.1.4 Write Documentation

esis could always use more documentation, whether as part of the official esis docs, in docstrings, or even on the web in blog posts, articles, and such.

4.1.5 Submit Feedback

The best way to send feedback is to file an issue at https://github.com/jcollado/esis/issues.

If you are proposing a feature:

- Explain in detail how it would work.

- Keep the scope as narrow as possible, to make it easier to implement.

- Remember that this is a volunteer-driven project, and that contributions are welcome :)

4.2 Get Started!

Ready to contribute? Here's how to set up *esis* for local development.

1. Fork the *esis* repo on GitHub.

2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/esis.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv esis
$ cd esis/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 esis tests
$ python setup.py test
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

4.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.

2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.

3. The pull request should work for Python 2.7. Check https://travis-ci.org/jcollado/esis/pull_requests and make sure that the tests pass for all supported Python versions.

4.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_esis
```

Credits

5.1 Development Lead

- Javier Collado <jcollado@nowsecure.com>

5.2 Contributors

None yet. Why not be the first?

History

0.1.0 (2015-03-23)

- First release on PyPI.

0.2.0 (2015-05-14)

- All documents indexed under the same index name.

- Docker files allow using the tool in a container.

Indices and tables

- genindex
- modindex
- search