Spectrify Documentation

Release 0.2.1

The Narrativ Company, Inc.

Sep 28, 2017

Contents

Contents:

Spectrify

A simple yet powerful tool to move your data from Redshift to Redshift Spectrum.

- Free software: MIT license
- Documentation: https://spectrify.readthedocs.io.

Features

One-liners to:

- Export a Redshift table to S3 (CSV)
- Convert exported CSVs to Parquet files in parallel
- Create the Spectrum table on your Redshift cluster
- **Perform all 3 steps in sequence**, essentially "copying" a Redshift table Spectrum in one command.

S3 credentials are specified using boto3. See http://boto3.readthedocs.io/en/latest/guide/configuration.html

Redshift credentials are supplied via environment variables, command-line parameters, or interactive prompt.

Install

```
$ pip install spectrify
```

Command-line Usage

Export Redshift table *my_table* to a folder of CSV files on S3:

Convert exported CSVs to Parquet:

Create Spectrum table from S3 folder:

Transform Redshift table by performing all 3 steps in sequence:

Python Usage

Currently, you'll have to supply your own SQL Alchemy engine to each of the below commands (pull requests welcome to make this eaiser).

Export to S3:

```
from spectrify.export import export_to_csv
export_to_csv(sa_engine, table_name, s3_csv_dir)
```

Convert exported CSVs to Parquet:

```
from spectrify.convert import convert_redshift_manifest_to_parquet
from spectrify.utils.schema import get_table_schema
sa_table = get_table_schema(sa_engine, source_table_name)
convert_redshift_manifest_to_parquet(s3_csv_manifest_path, sa_table, s3_spectrum_dir)
```

Create Spectrum table from S3 parquet folder:

```
from spectrify.create import create_external_table
from spectrify.utils.schema import get_table_schema
sa_table = get_table_schema(sa_engine, source_table_name)
create_external_table(sa_engine, dest_schema, dest_table_name, sa_table, s3_spectrum_
→path)
```

Transform Redshift table by performing all 3 steps in sequence:

```
from spectrify.transform import transform_table
transform_table(sa_engine, table_name, s3_base_path, dest_schema, dest_table, num_
→workers)
```

Contribute

Contributions always welcome! Read our guide on contributing here: http://spectrify.readthedocs.io/en/latest/contributing.html

License

MIT License. Copyright (c) 2017, The Narrativ Company, Inc.

Installation

Stable release

To install Spectrify, run this command in your terminal:

```
$ pip install spectrify
```

This is the preferred method to install Spectrify, as it will always install the most recent stable release.

If you don't have pip installed, this Python installation guide can guide you through the process.

From sources

The sources for Spectrify can be downloaded from the Github repo.

You can either clone the public repository:

```
$ git clone git://github.com/hellonarrativ/spectrify
```

Or download the tarball:

```
$ curl  -OL https://github.com/hellonarrativ/spectrify/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```

Usage

To use Spectrify in a project:

```python
import spectrify
```

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:

Types of Contributions

Report Bugs

Report bugs at https://github.com/hellonarrativ/spectrify/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.

Fix Bugs

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

Implement Features

Look through the GitHub issues for features. Anything tagged with "enhancement" and "help wanted" is open to whoever wants to implement it.

Write Documentation

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

Submit Feedback

The best way to send feedback is to file an issue at https://github.com/hellonarrativ/spectrify/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 :)

Get Started!

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

1. Fork the *spectrify* repo on GitHub.

2. Clone your fork locally:

   ```
   $ git clone git@github.com:your_name_here/spectrify.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 spectrify -p `which python3`   # or python2, if you prefer
   $ cd spectrify/
   $ pip install -e .
   ```

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 spectrify tests
   $ python setup.py test or 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.

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, 3.4 and 3.5, and 3.6. Check https://travis-ci.org/hellonarrativ/ spectrify/pull_requests and make sure that the tests pass for all supported Python versions.

Tips

To run a subset of tests:

```
$ py.test tests.test_spectrify
```

Credits

Development Lead

- Colin Nichols <engineering@narrativ.com>

Contributors

None yet. Why not be the first?

History

0.2.0 (2017-09-27)

- First release on PyPI.

0.1.0 (2017-09-13)

- Didn't even make it to PyPI.

CHAPTER 7

Indices and tables

- genindex
- modindex
- search