

---

# Python ISCC library

*Release 0.1.0*

Oct 31, 2018



---

## Contents

---

<b>1</b>	<b>PyISCC Lib - International Standard Content Codes</b>	<b>1</b>
1.1	DEPRECATED . . . . .	1
1.2	Introduction . . . . .	1
1.3	Documentation . . . . .	1
1.4	Development . . . . .	2
<b>2</b>	<b>Installation</b>	<b>3</b>
<b>3</b>	<b>Usage</b>	<b>5</b>
<b>4</b>	<b>Reference</b>	<b>7</b>
4.1	iscclib . . . . .	7
<b>5</b>	<b>Contributing</b>	<b>9</b>
5.1	Bug reports . . . . .	9
5.2	Documentation improvements . . . . .	9
5.3	Feature requests and feedback . . . . .	9
5.4	Development . . . . .	10
<b>6</b>	<b>Authors</b>	<b>11</b>
<b>7</b>	<b>Changelog</b>	<b>13</b>
7.1	0.1.0 (2017-03-16) . . . . .	13
<b>8</b>	<b>Indices and tables</b>	<b>15</b>
	<b>Python Module Index</b>	<b>17</b>



---

## PyISCC Lib - International Standard Content Codes

---

### 1.1 DEPRECATED

This repository is deprecated and only left here for reference. **Do not use in production**

Current development of ISCC code and specifications is at: <https://github.com/coblo/iscc-specs> . See <https://iscc.codes> for more information.

### 1.2 Introduction

ISCC is a generic content identifier optimized for the economy of blockchains. The major innovation of the ISCC: Identifiers are generated algorithmically from a basic set of metadata and the content itself.

#### 1.2.1 ISCC Goals

- cross-media: for text, image, audio, and video content
- cross-community: for news, ebooks, journals, science, music, film content
- cross-layer: identify multiples content layers from abstract to concrete

ISCC is part of the [Content Blockchain Project](#)

### 1.3 Documentation

<https://pyiscclib.rtfd.io/>

## 1.4 Development

To run the all tests run:

```
tox
```

Note, to combine the coverage data from all the tox environments run:

Windows	<pre>set PYTEST_ADDOPTS=--cov-append tox</pre>
Other	<pre>PYTEST_ADDOPTS=--cov-append tox</pre>

## CHAPTER 2

---

### Installation

---

At the command line:

```
pip install iscclib
```





## CHAPTER 3

---

### Usage

---

To use Python ISCC library in a project:

```
import iscclib
```



## CHAPTER 4

---

### Reference

---

#### 4.1 iscclib



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

### 5.1 Bug reports

When [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.

### 5.2 Documentation improvements

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

### 5.3 Feature requests and feedback

The best way to send feedback is to file an issue at <https://github.com/coblo/pyiscclib/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 code contributions are welcome :)

## 5.4 Development

To set up *pyiscclib* for local development:

1. Fork [pyiscclib](#) (look for the “Fork” button).
2. Clone your fork locally:

```
git clone git@github.com:your_name_here/pyiscclib.git
```

3. Create a branch for local development:

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

Now you can make your changes locally.

4. When you’re done making changes, run all the checks, doc builder and spell checker with `tox` one command:

```
tox
```

5. 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
```

6. Submit a pull request through the GitHub website.

### 5.4.1 Pull Request Guidelines

If you need some code review or feedback while you’re developing the code just make the pull request.

For merging, you should:

1. Include passing tests (run `tox`)<sup>1</sup>.
2. Update documentation when there’s new API, functionality etc.
3. Add a note to `CHANGELOG.rst` about the changes.
4. Add yourself to `AUTHORS.rst`.

### 5.4.2 Tips

To run a subset of tests:

```
tox -e envname -- py.test -k test_myfeature
```

To run all the test environments in *parallel* (you need to `pip install detox`):

```
detox
```

---

<sup>1</sup> If you don’t have all the necessary python versions available locally you can rely on Travis - it will [run the tests](#) for each change you add in the pull request.  
It will be slower though ...

## CHAPTER 6

---

### Authors

---

- Titusz Pan - <https://content-blockchain.org>
- Patricia Schinke - <https://content-blockchain.org>





## CHAPTER 7

---

### Changelog

---

#### 7.1 0.1.0 (2017-03-16)

- Project setup



## CHAPTER 8

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



**i**

iscclib, [7](#)



I

iscclib (module), [7](#)