

---

# **pybtex-docutils Documentation**

***Release 0.2.0***

**Matthias C. M. Troffaes**

August 08, 2013



# CONTENTS



**Release** 0.2.0

**Date** August 08, 2013



# CONTENTS

## 1.1 Getting Started

### 1.1.1 Overview

A docutils backend for pybtex.

- Download: <http://pypi.python.org/pypi/pybtex-docutils/#downloads>
- Documentation: <http://pybtex-docutils.readthedocs.org/>
- Development: <http://github.com/mcmtroffaes/pybtex-docutils/>

### 1.1.2 Installation

Install the module with `pip install pybtex-docutils`, or from source using `python setup.py install`.

### 1.1.3 Minimal Example

```
import StringIO
import pybtex.database.input.bibtex
import pybtex.plugin

style = pybtex.plugin.find_plugin('pybtex.style.formatting', 'plain')()
backend = pybtex.plugin.find_plugin('pybtex.backends', 'docutils')()
parser = pybtex.database.input.bibtex.Parser()
data = parser.parse_stream(StringIO.StringIO(u"""
@Book{1985:lindley,
  author = {D. Lindley},
  title = {Making Decisions},
  publisher = {Wiley},
  year = {1985},
  edition = {2nd},
}
"""))
for entry in style.format_entries(data.entries.itervalues()):
    print(backend.paragraph(entry))
```

would produce:

```
<paragraph>
  D. Lindley. <emphasis>Making Decisions</emphasis>.
  Wiley, 2nd edition, 1985.
</paragraph>
```

## 1.2 API

The backend renders `pybtex.richtext.Text` instances into a list of `docutils.nodes.Text` and `docutils.nodes.inline` instances (or subclasses of these). For typical use cases, all you need to care about are the methods `Backend.paragraph()`, `Backend.citation()`, and `Backend.citation_reference()` which are to be called on *formatted* entries, as in the *minimal example*.

Note that you should not import the `pybtex_docutils` module directly. Instead, use pybtex's plugin system to get the `Backend` class, again, as in the *minimal example*.

```
class pybtex_docutils.Backend(encoding=None)
    Bases: pybtex.backends.BaseBackend

    RenderType
        alias of list

    paragraph(entry)
        Return a docutils.nodes.paragraph containing the rendered text for entry (without label). New in version 0.2.0.

    citation(entry, document, use_key_as_label=True)
        Return citation node, with key as name, label as first child, and paragraph with entry text as second child. The citation is expected to be inserted into document prior to any docutils transforms.

    citation_reference(entry, document, use_key_as_label=True)
        Return citation_reference node to the given citation. The citation_reference is expected to be inserted into document prior to any docutils transforms.
```

## 1.3 Changes

### 1.3.1 0.2.0 (8 August 2013)

- **BACKWARD INCOMPATIBLE** The backend now renders into a list of docutils nodes instead of a single docutils node.
- New `paragraph()` method to render an entry into a single docutils paragraph.
- The `<inline>` wrapper nodes are no more, leading to much simpler generated code.
- Full test coverage.
- Generated citation nodes now contain text inside a paragraph.
- Minimal example.

### 1.3.2 0.1.0 (7 August 2013)

- Copied the backend from pybtex.
- Initial documentation.



- Initial tests and travis.ci integration.

## 1.4 License

pybtex-docutils is a docutils backend for pybtex  
Copyright (c) 2013 by Matthias C. M. Troffaes

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

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 OR COPYRIGHT HOLDERS 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.



# INDICES AND TABLES

- *genindex*
- *modindex*
- *search*



# PYTHON MODULE INDEX

p

pybtex\_docutils, ??