

---

# **sphinxcontrib-bibtex Documentation**

*Release 0.2.3*

**Matthias C. M. Troffaes**

July 30, 2012



# CONTENTS



**Release** 0.2.3

**Date** July 30, 2012



# CONTENTS

## 1.1 Getting Started

### 1.1.1 Overview

sphinxcontrib-bibtex is a Sphinx extension for BibTeX style citations.

It allows BibTeX citations to be inserted into documentation generated by Sphinx, via a `bibliography` directive, and a `cite` role, which work similarly to LaTeX's `\bibliography` and `\cite` commands.

Inspired by `bibstuff.sphinxext.bibref` by Matthew Brett.

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

### 1.1.2 Installation

Install the module with `pip install sphinxcontrib-bibtex`, or from source using `python setup.py install`. Then add:

```
extensions = ['sphinxcontrib.bibtex']
```

to your project's Sphinx configuration file `conf.py`.

### 1.1.3 Minimal Example

In your project's documentation, you can then write for instance:

```
See :cite:'1987:nelson' for an introduction to non-standard analysis.
```

```
.. bibliography:: refs.bib
```

where `refs.bib` would contain an entry:

```
@Book{1987:nelson,  
  author = {Edward Nelson},  
  title = {Radically Elementary Probability Theory},  
  publisher = {Princeton University Press},  
  year = {1977},
```

```
series = {Annals of Mathematical Studies}
}
```

## 1.2 Usage

### 1.2.1 Roles and Directives

#### **:cite:**

Create a citation to a bibliographic entry. For example:

See `:cite:'1987:nelson'` for an introduction to non-standard analysis.

which would be equivalent to the following LaTeX code:

See `\cite{1987:nelson}` for an introduction to non-standard analysis.

#### **.. bibliography:: refs.bib [...]**

Create bibliography for all cited references. The `all` flag forces all references to be included (equivalent to `\nocite{*}` in LaTeX). The `notcited` flag causes all references that were not cited to be included. The `cited` flag is recognized as well but is entirely optional. For example:

```
.. rubric:: References

.. bibliography:: refs.bib
   :cited:

.. rubric:: Further reading

.. bibliography:: refs.bib
   :notcited:
```

**Warning:** Sphinx will attempt to resolve references to the bibliography across all documents, so you must take care that no citation key is included more than once.

You can also pick a bibliography style, using the `style` option. This is not yet quite as useful, as only `unsrt` is supported, which is also the default.

```
.. bibliography:: refs.bib
   :style: unsrt
```

All citations have numbered labels, as in the `plain` LaTeX bibliography style, regardless of the style chosen. This limitation might be lifted in a future version.

You can also set the encoding of the bibliography files, using the `encoding` option.

```
.. bibliography:: refs.bib
   :encoding: latin
```

### 1.2.2 Known Issues and Workarounds

#### Tinkerer

To use the bibtex extension with `Tinkerer`, be sure to specify the bibtex extension first in your `conf.py` file:



```
extensions = ['sphinxcontrib.bibtex', 'tinkerer.ext.blog', 'tinkerer.ext.disqus']
```

## 1.3 Extension API

### 1.3.1 Sphinx Interface

`sphinxcontrib.bibtex.setup(app)`

Set up the bibtex extension:

- register directives
- register nodes
- register roles
- register transforms
- connect events to functions

**Parameters** `app` (`sphinx.application.Sphinx`) – The sphinx application.

`sphinxcontrib.bibtex.init_bibtex_cache(app)`

Create `app.env.bibtex_cache` if it does not exist yet. Reset citation label dictionary.

**Parameters** `app` (`sphinx.application.Sphinx`) – The sphinx application.

`sphinxcontrib.bibtex.purge_bibtex_cache(app, env, docname)`

Remove all information related to `docname` from the cache.

**Parameters**

- `app` (`sphinx.application.Sphinx`) – The sphinx application.
- `env` (`sphinx.environment.BuildEnvironment`) – The sphinx build environment.

`sphinxcontrib.bibtex.process_citations(app, doctree, docname)`

Replace labels of citation nodes by numbers.

**Parameters**

- `app` (`sphinx.application.Sphinx`) – The sphinx application.
- `doctree` (`docutils.nodes.document`) – The document tree.
- `docname` (`str`) – The document name.

`sphinxcontrib.bibtex.process_citation_references(app, doctree, docname)`

Replace text of citation reference nodes by numbers.

**Parameters**

- `app` (`sphinx.application.Sphinx`) – The sphinx application.
- `doctree` (`docutils.nodes.document`) – The document tree.
- `docname` (`str`) – The document name.

### 1.3.2 New Doctree Roles

**class** sphinxcontrib.bibtex.roles.**CiteRole** (*fix\_parens=False*, *lowercase=False*,  
*nodeclass=None*, *innernodeclass=None*,  
*warn\_dangling=False*)

Bases: sphinx.roles.XRefRole

Class for processing the `cite` role.

**result\_nodes** (*document, env, node, is\_ref*)

Transform reference node into a citation reference, and note that the reference was cited.

### 1.3.3 New Doctree Nodes

**class** sphinxcontrib.bibtex.nodes.**bibliography** (*rawsource='', \*children, \*\*attributes*)  
Node for representing a bibliography. Replaced by a list of citations by `BibliographyTransform`.

### 1.3.4 New Doctree Directives

**class** sphinxcontrib.bibtex.directives.**BibliographyDirective** (*name*, *arguments*,  
*options*, *content*,  
*lineno*, *content\_offset*,  
*block\_text*, *state*,  
*state\_machine*)

Class for processing the `bibliography` directive.

Parses the bibliography files, and produces a `bibliography` node.

**See Also:**

Further processing of the resulting `bibliography` node is done by `BibliographyTransform`.

**run()**

Process `.bib` files, set file dependencies, and create a nodes for all entries of the bibliography.

**process\_bibfile** (*bibfile, encoding*)

Check if `env.bibtex_cache.bibfiles[bibfile]` is still up to date. If not, parse the `bibfile` (see `update_bibfile_cache()`), and store parsed data in the bibtex cache.

**Parameters** `bibfile` (`str`) – The bib file name.

**Returns** The parsed bibliography data.

**Return type** `pybtex.database.BibliographyData`

**update\_bibfile\_cache** (*bibfile, mtime, encoding*)

Parse `bibfile` (see `parse_bibfile()`), and store the parsed data, along with modification time `mtime`, in the bibtex cache.

**Parameters**

- `bibfile` (`str`) – The bib file name.
- `mtime` (`float`) – The bib file's modification time.

**Returns** The parsed bibliography data.

**Return type** `pybtex.database.BibliographyData`

**parse\_bibfile** (*bibfile, encoding*)

Parse `bibfile`, and return parsed data.

**Parameters** `bibfile` (`str`) – The bib file name.

**Returns** The parsed bibliography data.

**Return type** `pybtex.database.BibliographyData`

### 1.3.5 New Doctree Transforms

`class sphinxcontrib.bibtex.transforms.BibliographyTransform` (`document`, `startnode=None`)

**default\_priority** = 10

Priority of the transform. See <http://docutils.sourceforge.net/docs/ref/transforms.html>

**apply** ()

Transform each `bibliography` node into a list of citations.

### 1.3.6 Cached Information

Classes and methods to maintain any information that is stored outside the doctree.

`class sphinxcontrib.bibtex.cache.Cache`

Global bibtex extension information cache. Stored in `app.env.bibtex_cache`, so must be picklable.

**bibfiles**

A `dict` mapping `.bib` file names (relative to the top source folder) to `BibfileCache` instances.

**bibliographies**

Each bibliography directive is assigned an id of the form `bibtex-bibliography-xxx`. This `dict` maps each such id to information about the bibliography directive, `BibliographyCache`. We need to store this extra information separately because it cannot be stored in the `bibliography` nodes themselves.

**purge** (`docname`)

Remove all information related to `docname`.

**Parameters** `docname` (`str`) – The document name.

`class sphinxcontrib.bibtex.cache.BibfileCache` (`mtime=None`, `data=None`)

Contains information about a parsed `.bib` file.

**mtime**

A `float` representing the modification time of the `.bib` file when it was last parsed.

**data**

A `pybtex.database.BibliographyData` containing the parsed `.bib` file.

`class sphinxcontrib.bibtex.cache.BibliographyCache` (`docname=None`, `bibfiles=None`, `cite='cited'`, `style=None`, `encoding=None`, `curly_bracket_strip=True`)

Contains information about a bibliography directive.

**docname**

A `str` containing the name of the document in which the directive occurs. We need this information during the Sphinx event `env-purge-doc`.

**bibfiles**

A list of `strs` containing the `.bib` file names (relative to the top source folder) that contain the references.

**cite**

A `str`. Should be one of:

"**cited**" Only generate cited references.

"**notcited**" Only generated non-cited references.

"**all**" Generate all references from the `.bib` files.

**style**

The bibtex style.

## 1.4 Changes

### 1.4.1 0.2.3 (30 July 2012)

- Document workaround for Tinkerer (issue #1).
- Use `tox` for testing.
- Full 2to3 compatibility.
- Document supported versions of Python (2.6, 2.7, 3.1, and 3.2).

### 1.4.2 0.2.2 (6 July 2012)

- Documentation and manifest fixes.

### 1.4.3 0.2.1 (19 June 2012)

- First public release.

## 1.5 License

sphinxcontrib-bibtex is a Sphinx extension for BibTeX style citations

Copyright (c) 2011-2012 by Matthias C. M. Troffaes

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED

TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.



# INDICES AND TABLES

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





# PYTHON MODULE INDEX

## S

- `sphinxcontrib.bibtex`, ??
- `sphinxcontrib.bibtex.cache`, ??
- `sphinxcontrib.bibtex.directives`, ??
- `sphinxcontrib.bibtex.nodes`, ??
- `sphinxcontrib.bibtex.roles`, ??
- `sphinxcontrib.bibtex.transforms`, ??