
Sphinx Confluence Builder Documentation

Release

Anthony Shaw and Contributors

Dec 06, 2017

Contents

1	documentation contents
----------	-------------------------------

3

`Sphinx` extension to build Confluence Wiki markup formatted files and optionally publish them to a Confluence server.

- License: BSD
- `Confluence` 4.0+ and `Python` 2.7, 3.3+
- Home: <https://github.com/tonybaloney/sphinxcontrib-confluencebuilder>
- Documentation: <http://sphinxcontrib-confluencebuilder.readthedocs.io>

1.1 installing

Atlassian Confluence Builder for Sphinx depends on:

- Python 2.7 or 3.3+
- Requests
- Sphinx 1.0 or later
- Confluence 4.0 or later

The recommended method of installation is using `pip`.

```
pip install --user sphinxcontrib-confluencebuilder
```

To verify the package has been installed, if desired, the following command can be used:

```
python -m sphinxcontrib.confluencebuilder --version
```

1.1.1 fresh quick-start

The following provides a series of steps to assist in preparing a new environment to use this package. This quick-start will aim to use the most recent version of Python.

Note: If the installation process fails with the following error “AttributeError: ‘_NamespacePath’ object has no attribute ‘sort’”, try upgrading your `setuptools` module:

```
python -m pip install --upgrade setuptools
```

linux

While the use of [Python/pip](#) is almost consistent between Linux distributions, the following are a series of helpful steps to install this package under specific distributions of Linux. From a terminal, invoke the following commands:

centos

```
$ sudo yum install epel-release
$ sudo yum install python-pip
$ pip install --user sphinxcontrib-confluencebuilder
$ python -m sphinxcontrib.confluencebuilder --version
sphinxcontrib.confluencebuilder <version>
```

fedora

```
$ sudo dnf install python-pip
$ pip install --user sphinxcontrib-confluencebuilder
$ python -m sphinxcontrib.confluencebuilder --version
sphinxcontrib.confluencebuilder <version>
```

ubuntu

```
$ sudo apt-get install python-pip
$ pip install --user sphinxcontrib-confluencebuilder
$ python -m sphinxcontrib.confluencebuilder --version
sphinxcontrib.confluencebuilder <version>
```

os x

From a terminal, invoke the following commands:

```
$ sudo easy_install pip
$ pip install --user sphinxcontrib-confluencebuilder
$ python -m sphinxcontrib.confluencebuilder --version
sphinxcontrib.confluencebuilder <version>
```

windows

If not already installed, download the most recent version of [Python](#):

Python - Downloads

<https://www.python.org/downloads/>

When invoking the installer, it is recommended to select the option to “Add Python to PATH”; however, users can explicitly invoke Python from an absolute path. Open a Python command environment from the start-menu or, if Python was added to the system path, open a Windows command prompt. Invoke the following:

```
python -m pip install sphinxcontrib-confluencebuilder
python -m sphinxcontrib.confluencebuilder --version
```


1.1.2 master

To install the bleeding edge sources, the following `pip` command can be used:

```
pip install \
    git+https://github.com/tonybaloney/sphinxcontrib-confluencebuilder
```

1.2 tutorial

After *installing* Atlassian Confluence Builder for Sphinx, a Sphinx project can be configured to support a confluence builder. The following tutorial will provide a series of steps which will:

- Allow a builder to generate Confluence-compatible markup documentation.
- Allow a builder to publish to a Confluence instance.

Tip: If starting without an existing Sphinx-based documentation, open a terminal to the location where documentation should be generated and invoke the following to generate a new documentation project:

```
sphinx-quickstart
(or)
python -m sphinx.quickstart
```

Enable the new builder by adding the new extension to the target project's Sphinx configuration (`conf.py`):

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

Next, include a series of publish-related settings to the configuration file:

```
confluence_publish = True
confluence_space_name = 'TEST'
confluence_parent_page = 'Documentation'
confluence_server_url = 'https://intranet-wiki.example.com'
confluence_server_user = 'username'
confluence_server_pass = 'password'
```

Make appropriate changes to the above configuration for the environment being prepared. The configuration `confluence_parent_page` should be supplied with the name of the page to append published documents. If omitted, the builder will publish documents in the root of the space.

Tip: For additional configuration options, consult *all configuration options*.

Invoke Sphinx with the confluence builder to perform building/publishing:

```
make confluence
(or)
sphinx-build -b confluence . _build/confluence
(or)
python -m sphinx -b confluence . _build/confluence
```

Documentation of the project should now be published to the Confluence site.

1.3 configuration

The following is an example of simple configuration for Confluence generation and publishing:

```
extensions = ['sphinxcontrib.confluencebuilder']
confluence_publish = True
confluence_space_name = 'TEST'
confluence_parent_page = 'Documentation'
confluence_server_url = 'https://intranet-wiki.example.com'
confluence_server_user = 'username'
confluence_server_pass = 'password'
```

All Atlassian Confluence Builder configurations are prefixed with `confluence_`. View the entire list of configuration options below.

- *essential configuration*
 - *confluence_publish*
 - *confluence_server_pass*
 - *confluence_server_url*
 - *confluence_server_user*
 - *confluence_space_name*
- *general configuration*
 - *confluence_disable_notifications*
 - *confluence_header_file*
 - *confluence_footer_file*
 - *confluence_master_homepage*
 - *confluence_page_hierarchy*
 - *confluence_parent_page*
 - *confluence_publish_prefix*
 - *confluence_purge*
- *advanced configuration - processing*
 - *confluence_file_suffix*
 - *confluence_file_transform*
 - *confluence_link_suffix*
 - *confluence_link_transform*
- *advanced configuration - publishing*
 - *confluence_disable_rest*
 - *confluence_disable_ssl_validation*
 - *confluence_disable_xmlrpc*
 - *confluence_parent_page_id_check*

- *confluence_proxy*
- *confluence_timeout*

1.3.1 essential configuration

confluence_publish

A boolean that decides whether or not to allow publishing. This option must be explicitly set to `True` if one wishes to publish content. By default, the value is set to `False`.

```
confluence_publish = True
```

confluence_server_pass

The password used to authenticate with the Confluence server.

```
confluence_server_pass = 'myawesomepassword'
```

confluence_server_url

The URL for Confluence. The URL should be prefixed with `https://` or `http://`, depending on the URL target. The target API folder should not be included – for example, if the target Confluence server's REST API is `https://intranet-wiki.example.com/rest/api/` or XML-RPC API is at `https://intranet-wiki.example.com/rpc/xmlrpc`, the URL configuration provided should be as follows:

```
confluence_space_name = 'https://intranet-wiki.example.com'
```

confluence_server_user

The username used to authenticate with the Confluence server.

```
confluence_server_user = 'myawesomeuser'
```

confluence_space_name

Key of the space in Confluence to be used to publish generated documents to.

```
confluence_space_name = 'MyAwesomeSpace'
```

1.3.2 general configuration

confluence_disable_notifications

A boolean value which explicitly disable any page update notifications (i.e. treats page updates from a publish request as minor updates). By default, notifications are enabled with a value of `False`.

```
confluence_disable_notifications = True
```

confluence_header_file

The name of the file to use header data. If provided, the contents found inside the header file will be added to the start of all generated documents. The file path provided should be relative to the build environment's source directory. For example:

```
confluence_header_file = 'assets/header.tpl'
```

confluence_footer_file

The name of the file to use footer data. If provided, the contents found inside the footer file will be added at the end of all generated documents. The file path provided should be relative to the build environment's source directory. For example:

```
confluence_footer_file = 'assets/footer.tpl'
```

confluence_master_homepage

A boolean value to whether or not force the configured space's homepage to be set to the page defined by the Sphinx configuration's `master_doc`. By default, the `master_doc` configuration is ignored with a value of `False`.

```
confluence_master_homepage = False
```

confluence_page_hierarchy

A boolean value to whether or not nest pages in a hierarchical ordered. The root of all pages is typically the configured `master_doc`. If a `master_doc` instance contains a `toctree`, listed documents will become child pages of the `master_doc`. This cycle continues for child pages with their own `toctree` markups. By default, the hierarchy mode is disabled with a value of `False`.

```
confluence_page_hierarchy = False
```

Note that even if hierarchy mode is enabled, the configured `master_doc` page and other published pages that are not defined in the complete `toctree`, these documents will still be published based off the configured (or unconfigured) `confluence_parent_page` setting.

confluence_parent_page

The root page found inside the configured space (`confluence_space_name`) where published pages will be a descendant of. The parent page value is used to match with the title of an existing page. If this option is not provided, pages will be published to the root of the configured space. If the parent page cannot be found, the publish attempt will stop with an error message. For example, the following will publish documentation under the `MyAwesomeDocs` page:

```
confluence_parent_page = 'MyAwesomeDocs'
```

If a parent page is not set, consider using the `confluence_master_homepage` option as well.

confluence_publish_prefix

If set, the prefix value is added to the title of all published document. In Confluence, page names need to be unique for a space. A prefix can be set to either:

- Add a unique naming schema to generated/published documents in a space which has manually created pages; or,
- Allow multiple published sets of documentation, each each with their own prefix value.

An example publish prefix is as follows:

```
confluence_publish_prefix = 'prefix-'
```

confluence_purge

A boolean value to whether or not purge legacy pages detected in a space or parent page. By default, this value is set to `False` to indicate that no pages will be removed. If this configuration is set to `True`, detected pages in Confluence that do not match the set of published documents will be automatically removed. If the option `confluence_parent_page` is set, only pages which are a descendant of the configured parent page can be removed; otherwise, all pages in the configured space could be removed.

```
confluence_purge = False
```

1.3.3 advanced configuration - processing

confluence_file_suffix

The file name suffix to use for all generated files. By default, all generated files will use the extension `.conf` (see `confluence_file_transform`).

```
confluence_file_suffix = '.conf'
```

confluence_file_transform

A function to override the translation of a document name to a filename. The provided function is used to perform translations for both Sphinx’s `get_outdated_docs` and `write_doc` methods. The default translation will be the combination of “`docname + confluence_file_suffix`”.

confluence_link_suffix

The suffix name to use for for generated links to files. By default, all generated links will use the value defined by `confluence_file_suffix` (see `confluence_link_transform`).

```
confluence_link_suffix = '.conf'
```

confluence_link_transform

A function to override the translation of a document name to a (partial) URI. The provided function is used to perform translations for both Sphinx’s `get_relative_uri` method. The default translation will be the combination of “`docname + confluence_link_suffix`”.

1.3.4 advanced configuration - publishing

confluence_disable_rest

A boolean value to explicitly disable any REST API calls. This extension has the ability to publish using either Confluence's REST or XML-RPC API calls. When publishing, this extension will first attempt to publish using REST and fallback to using XML-RPC. If the target Confluence instance cannot use REST for publishing, it is recommended to set the option to `True` to always use XML-RPC instead. By default, this option is set to `False`.

```
confluence_disable_rest = False
```

confluence_disable_ssl_validation

A boolean value to explicitly disable any verification of SSL certificates when making a publish request. By default, this option is set to `False`.

```
confluence_disable_ssl_validation = False
```

confluence_disable_xmlrpc

A boolean value to explicitly disable any XML-RPC API calls. This extension has the ability to publish using either Confluence's REST or XML-RPC API calls. When publishing, this extension will first attempt to publish using REST and fallback to using XML-RPC. If the target Confluence instance supports REST or has XML-RPC explicitly disabled, it is recommended to set this option to `True`. By default, this option is set to `False`.

```
confluence_disable_xmlrpc = False
```

confluence_parent_page_id_check

The page identifier check for `confluence_parent_page`. By providing an identifier of the parent page, both the parent page's name and identifier must match before this extension will publish any content to a Confluence server. This serves as a sanity-check configuration for the cautious.

```
confluence_parent_page_id_check = 1
```

confluence_proxy

Provide the proxy needed to be used to interact with the Confluence server over the network. At this time, the proxy configuration only applies to XML-RPC calls (REST calls use the `Requests` library which will use system-defined proxy configuration).

```
confluence_proxy = 'myawesomeproxy:8080'
```

confluence_timeout

Force a timeout (in seconds) for network interaction. The timeout used by this extension is not explicitly configured (i.e. managed by `Requests` and other implementations). By default, assume that any network interaction will not timeout. Since the target Confluence server is most likely to be found on an external server, it is recommended to

explicitly configure a timeout value based on the environment being used. For example, to configure a timeout of ten seconds, the following can be used:

```
confluence_timeout = 10
```

1.4 documentation

Documentation for Atlassian Confluence Builder is made with [Sphinx](#) and distributed on [Read the Docs](#). Sources of the documentation can be found inside the package repository's `doc` folder:

Atlassian Confluence Builder for Confluence

<https://github.com/tonybaloney/sphinxcontrib-confluencebuilder>

The documentation theme used by default is set to `sphinx_rtd_theme`. If locally generating documentation, the theme can be installed on systems using the following command:

```
pip install --user sphinx_rtd_theme
```

1.5 changelog

1.5.1 master

- fix case where first-publish with 'confluence_master_homepage' fails to configure the space's homepage
- support page hierarchy

1.5.2 0.8.0.dev0 (2017-12-06)

- Improve PyPi cover notes

1.5.3 0.7.0 (2017-11-30)

- cap headers/sections to six levels for improved visualization
- fixed rest publishing for encoding issues and python 3.x (< 3.6) issues
- improve markup for:
 - body element lists
 - citations
 - definitions
 - footnotes
 - inline literals
 - literal block (code)
 - rubric
 - seealso
 - table

- versionmodified
- re-work generated document references/targets (reference to section names)
- sanitize output to prevent confluence errors for certain characters
- support indentations markup
- support master_doc option to configure space's homepage
- support removing document titles from page outputs
- support silent page updates

1.5.4 0.6.0 (2017-04-23)

- cleanup module's structure, versions and other minor files
- drop 'confluence' pypi package (embedded xml-rpc support added)
- improve hyperlink and cross-referencing arbitrary locations/documents support
- improve proxy support
- re-support python 3.x series
- support anonymous publishing
- support rest api

1.5.5 0.5.0 (2017-03-31)

- header/footer support
- purging support
- use macros for admonitions
- (note) known issues with python 3.3, 3.4, 3.5 or 3.6 (see [tonybaloney/sphinxcontrib-confluencebuilder#10](https://github.com/tonybaloney/sphinxcontrib-confluencebuilder#10))

1.5.6 0.4.0 (2017-02-21)

- move from 'confluence' pypi package to 'confluence' pypi package (required for publishing). see: <https://github.com/pycontribs/confluence>

1.5.7 0.3.0 (2017-01-22)

- adding travis ci, tox and initial unit testing
- module now depends on `future`
- providing initial support for python 3

1.5.8 0.2.0 (2016-07-13)

- moved configuration to the sphinx config

1.5.9 0.1.1 (2016-07-12)

- added table support
- fixed internal links

1.5.10 0.1.0 (2016-07-12)

- added lists, bullets, formatted text
- added headings and titles