

---

# **sphinxcontrib-relativeinclude**

***Release 0.0.1a1***

**Fynn Freyer**

**Sep 20, 2023**

---

## Contents:

---

<b>1</b>	<b>About</b>	<b>2</b>
<b>2</b>	<b>Installation</b>	<b>3</b>
<b>3</b>	<b>Motivation</b>	<b>4</b>
<b>4</b>	<b>Usage</b>	<b>6</b>
<b>5</b>	<b>TODOs</b>	<b>7</b>
<b>6</b>	<b>License</b>	<b>8</b>
<b>7</b>	<b>Module Documentation</b>	<b>9</b>
	<b>Python Module Index</b>	<b>11</b>
	<b>Index</b>	<b>12</b>

build passing

docs passing

pypi v0.0.1a1

python 3.8 | 3.9 | 3.10 | 3.11

license MIT

# CHAPTER 1

---

## About

---

This package implements a new reST directive to include files and translate paths included in those files.

## CHAPTER 2

---

### Installation

---

The project is hosted on [PyPI](#), and can be installed via `pip`.

```
pip install sphinxcontrib-relativeinclude
```

You can find the contents of the README and the module documentation for the latest release [online](#).

## CHAPTER 3

---

### Motivation

---

Let's assume you want to transclude the `README.rst` sitting in your repository root, in your `docs/index.rst`, so it automatically shows up in your generated documentation. You can just use reST's standard `include` directive.

```
.. include: ../README.rst
```

**Sidenote:** If you're using a parser, like `MyST`, you could also easily include markdown files, of course.

This will insert the contents of your `README` in the appropriate place, and even take care of heading levels for you.

The problem arises when you have images or other files included in there. Sphinx won't resolve those links properly, i.e., relative to the `README`, but instead relative to your Sphinx index document. That means those pictures won't show up, which, needless to say, is not optimal. If you use your valuable time to create visual resources for your documentation these should also be included in the documentation output.



Picture of a honey badger by [Sumeetmoghe on Wikimedia Commons](#) (CC-BY-SA-4.0)

This picture uses a relative path to a file in the **docs/assets** directory, and would happily show up in your Git repo, but not in your documentation.

This is what this extension is supposed to solve. It defines a new `relativeinclude` directive, that takes relative paths in included files, and resolves them into absolute ones. This way your images show up in your documentation output, but you don't have to hardcode absolute paths in your documentation. (Cf. [this awesome honey badger here](#))

**Caveat emptor:** At this point in time, nested includes are unfortunately not supported. (See *TODOs*)

## CHAPTER 4

---

### Usage

---

You have to register the directive in the `setup` function in your `conf.py` file. (This is a bug, see *TODOs*)

```
def setup(app):  
    """Register the directive."""  
    from sphinxcontrib_relativeinclude import RelativeInclude  
    app.add_directive("relativeinclude", RelativeInclude)
```

Then you can use it in your documentation index under the registered name.

```
.. relativeinclude: ../README.md  
   :parser: myst_parser.docutils_
```

It supports the same options as the standard `include` directive. If not, you've found a bug, and I'd be happy if you reported it on the [issue tracker](#). Please provide thorough description, and a minimal reproducible example, i.e., (abbreviated) reST files you used, potentially your `conf.py` contents and maybe other relevant info.

If you want to see some real code, check out this repositories [docs/index.rst](#).



## CHAPTER 5

---

### TODOs

---

- support multiple levels of indirection
- properly register directive on install

## CHAPTER 6

---

### License

---

`sphinxcontrib-relativeinclude` is distributed under the terms of the [MIT](#) license.

---

## Module Documentation

---

The `sphinxcontrib-relativeinclude.relativeinclude` module implements the *RelativeInclude* class, which provides a new `relativeinclude` directive for reST. This directive works just like a normal `include`, but it will translate relative paths in the included document, so that they're relative to the document we're including from.

**Warning:** This does not support nested includes yet!

**class** `LinkTranslator`(*document: document, absolute\_base: Path, relative\_base: Path*)

Translates relative links in included documents with respect to the source document.

**resolve\_attrs:** `ClassVar[Collection[str]] = ('reftarget', 'uri')`

Attributes, that need changing.

**ignore\_schemes:** `ClassVar[Collection[str]] = ('http', 'https', 'data')`

If we encounter these prefixes in a link, we leave it alone.

**default\_visit**(*node: Node*)

When visiting a node, we change all attrs with relative paths, defined in `change_attrs`.

**unknown\_visit**(*node: Node*)

If we don't know a Node, we ignore it.

**default\_departure**(*node: Node*)

Override abstract method for completeness.

**class** `RelativeInclude`(*name, arguments, options, content, lineno, content\_offset, block\_text, state, state\_machine*)

This class is supposed to implement a `relativeinclude` directive in Sphinx. It should automatically translate relative links in included documents and apart from that works just like `.. include::`.

**resolve\_attrs:** `ClassVar[Collection[str]] = ('reftarget', 'uri')`

Attributes, that need changing.

**property include\_source:** `Path`

The absolute path of the document we're including from.

**property include\_target:** `Path`

The absolute path of the document we're including.

**run()** `→ List[Node]`

Include a file as part of the content of this reST file, and translate any relative paths in the included file.

**translate**(*node: Node, absolute\_base: Path, relative\_base: Path*)

Recursively translate relative nodes.

#### Parameters

- **node** – The node we're currently working on.
- **absolute\_base** – The absolute path of the file from which we are including.
- **relative\_base** – The absolute path of the file we are including.

**setup**(*app*)

Register the directive.

---

## Python Module Index

---

### S

`sphinxcontrib_relativeinclude.relativeinclude,`

9

## D

`default_departure()` (*LinkTranslator method*), 9

`default_visit()` (*LinkTranslator method*), 9

## I

`ignore_schemes` (*LinkTranslator attribute*), 9

`include_source` (*RelativeInclude property*), 10

`include_target` (*RelativeInclude property*), 10

## L

`LinkTranslator` (*class in sphinxcontrib\_relativeinclude.relativeinclude*), 9

## M

module  
    `sphinxcontrib_relativeinclude.relativeinclude`, 8

## R

`RelativeInclude` (*class in sphinxcontrib\_relativeinclude.relativeinclude*), 9

`resolve_attrs` (*LinkTranslator attribute*), 9

`resolve_attrs` (*RelativeInclude attribute*), 10

`run()` (*RelativeInclude method*), 10

## S

`setup()` (*in module sphinxcontrib\_relativeinclude.relativeinclude*), 10

`sphinxcontrib_relativeinclude.relativeinclude` module, 8

## T

`translate()` (*RelativeInclude method*), 10

## U

`unknown_visit()` (*LinkTranslator method*), 9