
sphinx-nbexamples Documentation

Release 0.3.2

Philipp Sommer

May 17, 2019

1	Content	3
1.1	Installation	3
1.1.1	How to install	3
	Installation using pip	3
	Installation from source	3
1.1.2	Usage on readthedocs.org	4
1.1.3	Running the tests	4
1.1.4	Building the docs	4
1.2	Getting started	4
1.2.1	Structure of the notebooks	5
1.2.2	Choosing the examples	5
1.2.3	Preprocessing the examples or not	5
1.2.4	Choosing the thumbnail	5
1.2.5	Providing supplementary files	6
1.2.6	Including a link to the nbviewer	6
1.2.7	Including a link to the binder	6
1.2.8	Including bokeh	7
1.2.9	Removing cells	7
1.3	Linking to other galleries	8
1.3.1	Linked gallery example	8
1.4	Examples for the sphinx-nbexamples package	10
1.4.1	Bash example	10
1.4.2	Basic example	11
1.4.3	Examples in a subfolder	12
	Bokeh example	12
1.5	API Reference	13
1.6	Changelog	21
1.6.1	v0.3.1	21
	Changed	21
1.6.2	v0.3.0	21
	Added	21
1.6.3	v0.2.2	21
	Added	21
	Changed	21
1.6.4	v0.2.1	21
	Changed	21

1.6.5	v0.2.0	22
	Added	22
	Changed	22
1.7	Installation	22
1.8	Requirements	22
1.9	Indices and tables	22

Python Module Index	23
----------------------------	-----------

Welcome! Similarly to Oscar Najeras [sphinx-gallery](#) module, this module intends to create an example gallery for your documentation. However, we don't use python scripts, instead we create the example gallery out of a bunch of jupyter notebooks using nbconvert.

This package can be used to

1. Put all the examples you prepared in different notebooks in an pictured gallery
2. use the same html (sphinx) scheme for your examples that you are using for your documentation
3. Include the example notebooks in an offline (pdf) documentation
4. Include not only the code, but also the link to required supplementary files
5. Include a link to the [Jupyter nbviewer](#)

1.1 Installation

1.1.1 How to install

Installation using pip

If you do not want to use conda for managing your python packages, you can also use the python package manager `pip` and install via:

```
$ pip install sphinx-nbexamples
```

If you want to preprocess your notebooks before including them in the documentation, you might also have to install the `ipykernel` module via:

```
$ pip install ipykernel
```

and register the kernel depending on the kernel name in your notebooks via:

```
$ python -m ipykernel install --user --name <kernel-name> --display-name <kernel-name>
```

where the `<kernel-name>` should be replaced by the kernel name as it is used in the examples.

Note: If your examples require additional packages, you of course have to install them by yourself

Installation from source

You can as well install the package from the [github](#) via:

```
$ python setup.py install
```

1.1.2 Usage on readthedocs.org

When building your documentation on readthedocs.org, you can either disable the preprocessing of the notebooks via the `process_examples` configuration value, e.g. via:

```
on_rtd = os.environ.get('READTHEDOCS', None) == 'True'
process_examples = not on_rtd
```

or:

```
example_gallery_config['dont_preprocess'] = on_rtd
```

or you make sure that the virtual environment installs ipykernel and all the other necessary packages for your examples and include:

```
on_rtd = os.environ.get('READTHEDOCS', None) == 'True'
if on_rtd:
    import subprocess as spr
    spr.call([sys.executable] +
             ('-m ipykernel install --user --name python3 '
              '--display-name python3').split())
```

in your `conf.py` of your sphinx documentation. Change `'python3'` to the kernel name you are using in your examples.

1.1.3 Running the tests

We use `pytest` for our testing, so simply install it and run:

```
$ py.test
```

or in the downloaded directory from [github](https://github.com) run

```
$ python setup.py pytest
```

1.1.4 Building the docs

To build the docs, check out the [github](https://github.com) repository and install the requirements in `'docs/environment.yml'`. The easiest way to do this is via `anaconda` by typing:

```
$ conda env create -f docs/environment.yml
$ source activate sphinx_nbexamples_docs
$ conda install ipykernel sphinx_rtd_theme
```

Then build the docs via:

```
$ cd docs
$ make html
```

1.2 Getting started

The module provides 2 additional configuration values.

process_examples

If `True`, (the default), then the notebook files are converted to `rst`.

example_gallery_config

A dictionary with the parameters of the `Gallery` class

By default, the `sphinx-nbexamples` package converts your jupyter notebook in a specific directory into `rst` files to include it into your documentation. The `process_examples` configuration value controls this conversion. If switched off, no new files will be created.

The second configuration value, `example_gallery_config`, can be used to control which examples are converted and how. They are simply the keyword arguments for the `Gallery` class, but we will go in more detail in the next sections. But look into the *Examples for the sphinx-nbexamples package* section to see the outcome of the gallery.

1.2.1 Structure of the notebooks

You are free to format your notebooks however you want. There are only 2 important features since we convert the notebook to a single html page:

1. The first cell must be a Markdown cell containing a title
2. The first cell should include a short summary which will then be shown as the tooltip in the summary

1.2.2 Choosing the examples

The three keywords `'examples_dirs'`, `'gallery_dirs'`, and `'pattern'` can be used to select which notebooks shall be converted. The value for `'examples_dirs'` is the path to the directory where your raw jupyter notebooks are located. The `'gallery_dirs'` key on the other hand will point to the directories where the converted notebooks will be. You can also provide a list of example directories to create multiple galleries.

Finally the `'pattern'` corresponds to the filename pattern for the example notebooks. Using the default pattern (`'example_*.ipynb'`) implies, that all your notebooks in the `'examples_dirs'` starts with `'example_'`

1.2.3 Preprocessing the examples or not

When converting the examples, the default behaviour is to process the examples as well. This is a good possibility if you have an automatic building of the docs (e.g. using readthedocs.org) to check that all your examples really work. However, you might not want this for all your notebooks, because it eventually takes a lot of time to process all the notebooks or it requires additional libraries. Therefore you can use the `'preprocess'` and `'dont_preprocess'` keys so select which examples are processed.

1.2.4 Choosing the thumbnail

As you see in our *example gallery*, little thumbnails are created for each notebook. They can be chosen via

1. the `'code_examples'` key in the `example_gallery_config`
2. the `'code_example'` key in the meta data of the notebook
3. the `'thumbnail_figures'` key in the `example_gallery_config`
4. the key `'thumbnail_figures'` in the meta data of the notebook
5. automatically from the last matplotlib figure in the example notebook

Hence, if you do not specify either 'code_examples' nor 'thumbnail_figure' (which is the default), it looks for a matplotlib plot in the notebook and uses this one.

Otherwise, you have the possibility to give a small code sample via the 'code_examples' or use the 'thumbnail_figure'. The latter can be the path to a picture (relative to the notebook) or a number to specify which figure of the matplotlib figures to use.

1.2.5 Providing supplementary files

Sphinx-nbexamples automatically inserts links to download the jupyter notebook and the converted python file. However, often your example requires additional data files, etc. Here, you have two possibilities:

1. Specify the external data in the metadata of your notebook (see the *Basic example*)
2. Specify the external data in the 'supplementary_files' key of your *example_gallery_config* specific for each notebook

1.2.6 Including a link to the nbviewer

If your notebooks are also published online, you can embed a link to the wonderful [jupyter nbviewer](#) in the documentation. You have multiple options here. You can either

1. specify the url for each notebook separately providing a mapping from notebook file to url in the 'urls' option of the *example_gallery_config*
2. include a url item in the metadata of your notebook that points to the url of the notebook
3. specify one single url in the 'urls' option of the *example_gallery_config* that will then be extended to the corresponding notebook path. For sphinx-nbexamples, this looks like:

```
example_gallery_config = {
    urls='https://github.com/Chilipp/sphinx-nbexamples/blob/master/examples',
}
```

1.2.7 Including a link to the binder

[Jupyter's binderhub](#) allows to run the example notebooks of your repository (see for example [mybinder.org](#)). If your notebooks are also published online, sphinx-nbexamples can add a badge like in the documentation.

You have multiple options here. You can either

1. specify the url for each notebook separately providing a mapping from notebook file to url in the 'binder_urls' option of the *example_gallery_config*
2. include a binder_url item in the metadata of your notebook that points to the url of the notebook where it can be run interactively
3. specify one single url in the 'binder_urls' option of the *example_gallery_config* that will then be extended to the corresponding notebook path. For sphinx-nbexamples at [mybinder.org](#), this looks like:

```
example_gallery_config = {
    'binder_url': 'https://mybinder.org/v2/gh/Chilipp/sphinx-nbexamples/master?
↪filepath=examples',
}
```

or for the [pangeo binder](#):

```
example_gallery_config = {
    'binder_url': 'https://binder.pangeo.io/v2/gh/Chilipp/sphinx-nbexamples/
↳master?filepath=examples',
}
```

See the binderhub service you use (e.g. mybinder.org) for how to get this url for your repository. This will, e.g. for the *example_basic.ipynb* notebook, translate into:

```
.. image:: https://mybinder.org/badge_logo.svg
    :target: https://mybinder.org/v2/gh/Chilipp/sphinx-nbexamples/master?
↳filepath=examples/example_basic.ipynb
```

1.2.8 Including bokeh

Warning: Bokeh is not working for the latest version (see #10). PR's welcomed!

Note that bokeh needs a special treatment, especially when using the scheme from readthedocs.org, because it requires additional style sheets and javascript files. So, if you have bokeh plots in your documentation, we recommend to

1. use the `bokeh.io.output_notebook()` function in your examples
2. disable the preprocessing for this notebook using the `'dont_preprocess'` keyword
3. Give the bokeh version via the `'insert_bokeh'` keyword

If you furthermore use widgets from bokeh, use the `'insert_bokeh_widgets'` keyword, too.

Note: We cannot extract a thumbnail figure for bokeh notebooks. Hence, you should provide it by yourself (see *Choosing the thumbnail*).

1.2.9 Removing cells

Using notebook 5.0 and nbconvert 5.3 and higher, you can also tag cells and specify them for removal in the converted rst file.

In the jupyter notebook click on *View* → *Cell Toolbar* → *Tags* and assign a tag to the cell you want to remove. You can then use one or more of the keywords

remove_all_outputs_tags removes all outputs

remove_cell_tags removes the entire cell

remove_input_tags removes the input and only keeps the output

remove_single_output_tags Removes an individual output

in the *example_gallery_config*. See the *Gallery* and `nbconvert.preprocessors.Preprocessor` documentation for more information.

To remove the entire cell, for example, set

```
example_gallery_config = {
    'remove_cell_tags': ['the-tag-of-the-cell-you-want-to-remove'],
}
```

in the `conf.py` of your docs.

1.3 Linking to other galleries

You can insert the links to the example galleries in other projects using the `linkgalleries` directive. This will insert all the thumbnails and the titles of the examples in a list. You can find an example [below](#).

.. linkgalleries::

Insert links to other example galleries generated with the sphinx-nbexamples extension.

The directive takes no arguments and the options are the same as for the `figure` directive. By default, we use a width of 160px and the `align` parameter is set to `'left'`.

Each line of the content for this package must be the name of a package as it is registered in the `intersphinx_mapping` configuration value by the `sphinx.ext.intersphinx` extension. Optionally you can also provide the folder for the examples.

Warning: This directive only works well for examples that have a thumbnail associated with them, i.e. not with code examples (see [Choosing the thumbnail](#)).

Examples

To insert links to the examples of the *sphinx-nbexamples gallery* you can either use

```
.. linkgalleries::  
  
    sphinx_nbexamples
```

or more explicit

```
.. linkgalleries::  
  
    sphinx_nbexamples examples
```

1.3.1 Linked gallery example

The outputs of

```
.. linkgalleries::  
  
    psyplot
```

are links to the examples in `psyplots` example gallery.

This then transforms to

In the `conf.py` script, the `intersphinx_mapping` configuration value then looks like

```
intersphinx_mapping = {  
    'psyplot': ('http://psyplot.readthedocs.io/en/latest/', None),  
}
```

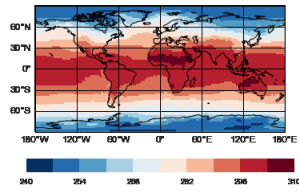


Fig. 1: Usage of Climate Data Operators

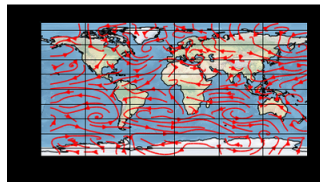


Fig. 2: Applying your own post processing

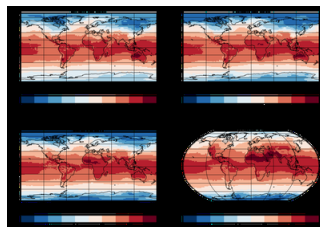


Fig. 3: Sharing formatoptions

We can also insert links into the library of the current project by just inserting the name of the project.

In our case (`sphinx-nbexamples`) this then looks like



Fig. 4: *Bash example*



Fig. 5: *Basic example*

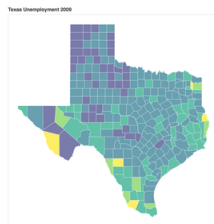


Fig. 6: *Bokeh example*

The `linkgalleries` directive also accepts multiple packages, e.g.

```
.. linkgalleries::  
  
    pyplot  
    sphinx-nbexamples
```

1.4 Examples for the sphinx-nbexamples package

Those examples shall just show you how you might use the sphinx-nbexamples package. All those examples are generated automatically from the `examples` directory of the package. But you can also download them from the given notebook.

1.4.1 Bash example

This example notebook uses a Bash kernel.

```
# A basic hello, world in Bash
```

```
function hello {
    echo hello, world
}
```

```
hello
```

```
hello, world
```

1.4.2 Basic example

This example notebook demonstrates as a basic example how sphinx-nbexamples works.

You can edit the metadata of the notebook as shown in the screenshot below and include supplementary files, select an image for displaying it in the gallery or provide a code example.

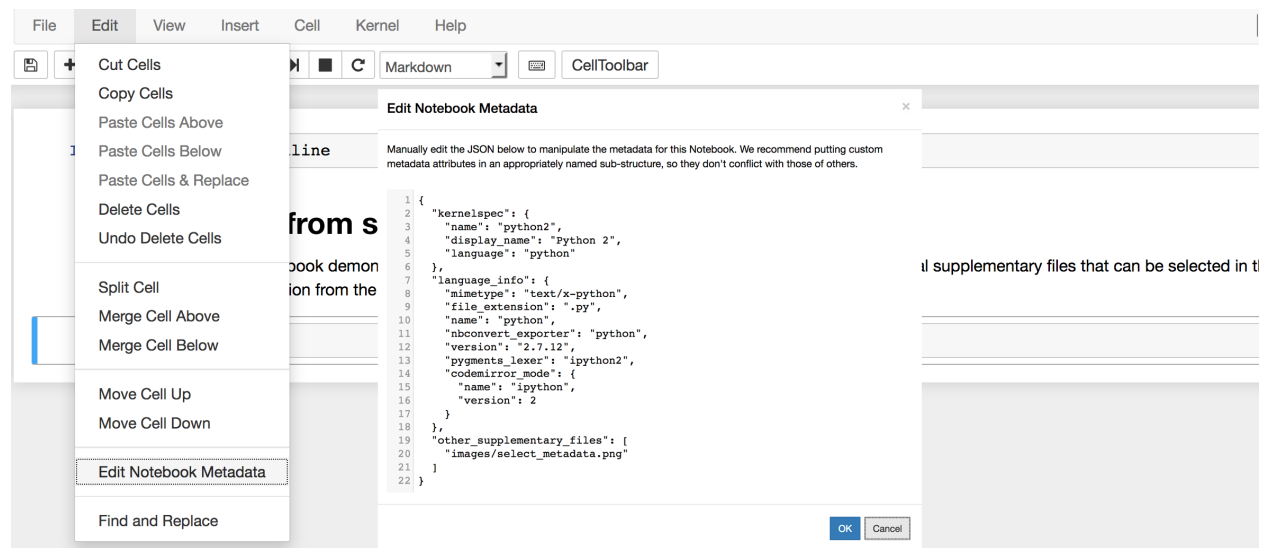


Fig. 7: Edit metadata

Otherwise you can do whatever you want. You can also include ipython magic commands, e.g `%matplotlib inline`

which then will be (as well as all the warnings) removed in the rendered html file of your example.

By default, the sphinx-nbexamples looks for pictures you created with matplotlib and saved it in the notebook. If such a picture exists, it uses the last created picture as a thumbnail in the gallery. For this example, we modified the `thumbnail_figure` key of the notebook metadata to use our own picture (the one you see above).

```
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
```

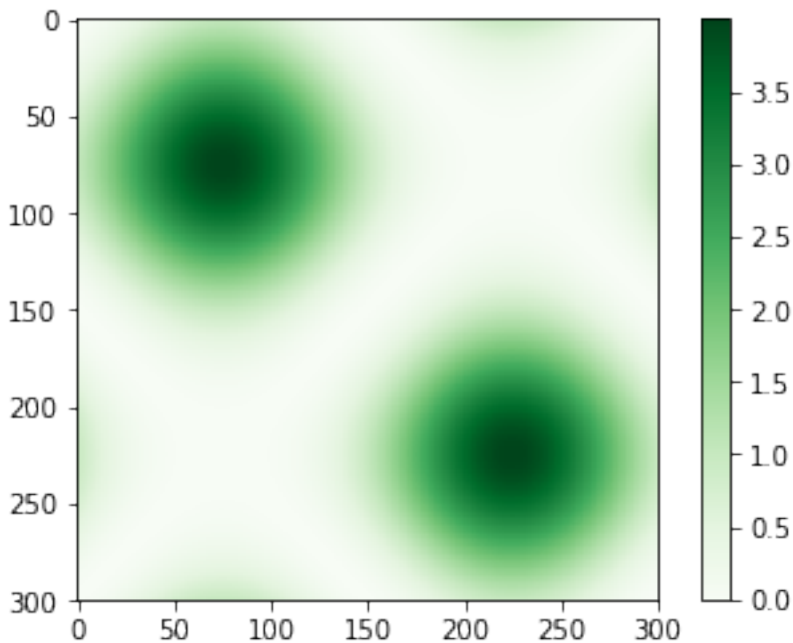
```
x = np.linspace(-np.pi, np.pi, 300)
xx, yy = np.meshgrid(x, x)
```

(continues on next page)

(continued from previous page)

```
z = (np.sin(xx) + np.sin(yy)) ** 2
plt.figure()
plt.imshow(z, cmap='Greens')
plt.colorbar()
```

```
<matplotlib.colorbar.Colorbar at 0x7fac60d2f438>
```



1.4.3 Examples in a subfolder

Just some examples in a subfolder. Important is, that you have a `README.rst` file in this folder. You can also have more subfolders with examples in it.

Bokeh example

This example demonstrates the use of bokeh in the sphinx-nbgallery.

The example is taken from the bokeh-notebook gallery to demonstrate the use bokeh. For the sphinx-nbgallery, you have to do 3 additional modifications in the `example_gallery_config` of your `conf.py`:

1. You have to set the `insert_bokeh` configuration value because we need additional style sheets and javascript files
2. Note, that, since the `output_notebook` function needs some time, we recommend to use the `dont_preprocess` configuration value for this notebook.
3. We cannot estimate a thumbnail figure for a notebook not using matplotlib. So you should provide it using the `thumbnail_figure` metadata key (as it has been done for this notebook)


```

from bokeh.models import HoverTool, ColumnDataSource
from bokeh.palettes import Viridis6
from bokeh.plotting import figure, show, output_notebook
from bokeh.sampledata.us_counties import data as counties
from bokeh.sampledata.unemployment import data as unemployment

```

```

counties = {
    code: county for code, county in counties.items() if county["state"] == "tx"
}

county_xs = [county["lons"] for county in counties.values()]
county_ys = [county["lats"] for county in counties.values()]

```

```

county_names = [county['name'] for county in counties.values()]
county_rates = [unemployment[county_id] for county_id in counties]
county_colors = [Viridis6[int(rate/3)] for rate in county_rates]

source = ColumnDataSource(data=dict(
    x=county_xs,
    y=county_ys,
    color=county_colors,
    name=county_names,
    rate=county_rates,
))

```

```
output_notebook()
```

```

TOOLS="pan,wheel_zoom,box_zoom,reset,hover,save"

p = figure(title="Texas Unemployment 2009", tools=TOOLS,
           x_axis_location=None, y_axis_location=None)
p.grid.grid_line_color = None

p.patches('x', 'y', source=source,
          fill_color='color', fill_alpha=0.7,
          line_color="white", line_width=0.5)

hover = p.select_one(HoverTool)
hover.point_policy = "follow_mouse"
hover.tooltips = [
    ("Name", "@name"),
    ("Unemployment rate", "@rate%"),
    ("Long, Lat", "($x, $y)"),
]

```

```
show(p)
```

1.5 API Reference

Create an example gallery out of a bunch of ipython notebooks

This sphinx extension extracts the ipython notebooks in a given folder to create an example gallery. It provides the following configuration values for your sphinx configuration file 'conf.py':

<code>process_examples</code>	Boolean controlling whether the rst files shall created and examples processed
<code>gallery_config</code>	dictionary containing the configuration of the example gallery.

Notes

This module was motivated by the `sphinx-gallery` module by Oscar Najera and in fact uses parts of it's html template for including the thumbnails and the download containers

Classes

<code>Gallery([examples_dirs, gallery_dirs, ...])</code>	Class to create one or more example gallerys
<code>LinkGalleriesDirective(name, arguments, ...)</code>	
<code>NotebookProcessor(infile, outfile[, ...])</code>	Class to run process one ipython notebook and create the necessary files

Functions

<code>align(argument)</code>	Conversion function for the "align" option.
<code>create_dirs(*dirs)</code>	
<code>isstring(s)</code>	
<code>nbviewer_link(url)</code>	Return the link to the Jupyter nbviewer for the given notebook url
<code>setup(app)</code>	

Data

<code>gallery_config</code>	dictionary containing the configuration of the example gallery.
<code>process_examples</code>	Boolean controlling whether the rst files shall created and examples

```
class Gallery (examples_dirs=['../examples'], gallery_dirs=None, pattern='example_+.ipynb',
               disable_warnings=True, dont_preprocess=[], preprocess=True, clear=True, dont_clear=[],
               code_examples={}, supplementary_files={}, other_supplementary_files={}, thumb-
               nail_figures={}, urls=None, insert_bokeh=False, insert_bokeh_widgets=False,
               remove_all_outputs_tags={}, remove_cell_tags={}, remove_input_tags={}, re-
               move_single_output_tags={}, toctree_depth=-1, binder_url=None)
```

Bases: `object`

Class to create one or more example gallerys

Parameters

- **examples_dirs** (*list of str*) – list containing the directories to loop through. Default: `['../examples']`
- **gallerys_dirs** (*list of str*) – None or list of directories where the rst files shall be created. If None, the current working directory and the name of the corresponding directory in the `examples_dirs` is used. Default: `None`

- **pattern** (*list of str*) – str. The pattern to use to find the ipython notebooks. Default: 'example_+.ipynb'
- **disable_warnings** (*bool*) – Boolean controlling whether warnings shall be disabled when processing the examples. Default: True
- **preprocess** (*bool or list of str*) – If True, all examples (except those specified in the *dont_preprocess* item) will be preprocessed when creating the rst files. Otherwise it might be a list of files that shall be preprocessed.
- **dont_preprocess** (*bool or list of str*) – If True, no example will be preprocessed when creating the rst files. Otherwise it might be a list of files that shall not be preprocessed
- **clear** (*bool or list of str*) – If True, the output in all notebooks to download will be cleared. Otherwise it might be a list of notebook files of whom to clear the output
- **dont_clear** (*bool or list of str*) – If True, the output in all notebooks to download will not be cleared. Otherwise it might be a list of notebook files of whom not to clear the output
- **code_examples** (*dict*) – A mapping from filename to code samples that shall be used instead of a thumbnail figure in the gallery. Note that you can also include a 'code_example' key in the metadata of the notebook
- **supplementary_files** (*dict*) – A mapping from filename to a list of supplementary data files that shall copied to the documentation directory and can be downloaded. Note that you can also include a 'supplementary_files' key in the metadata of the notebook
- **other_supplementary_files** (*dict*) – A mapping from filename to a list of other supplementary data files that shall copied to the documentation directory but can not be downloaded (e.g. pictures). Note that you can also include a 'other_supplementary_files' key in the metadata of the notebook
- **thumbnail_figures** (*dict*) – A mapping from filename to an integer or the path of a file to use for the thumbnail
- **urls** (*str or dict*) – The urls where to download the notebook. Necessary to provide a link to the jupyter nbviewer. If string, the path to the notebook will be appended for each example notebook. Otherwise it should be a dictionary with links for the given notebook
- **insert_bokeh** (*bool or str*) – If True, the bokeh js¹ and the stylesheet² are inserted in the notebooks that have bokeh loaded (using the installed or specified bokeh version)
- **insert_bokeh_widgets** (*bool or str*) – If True, the bokeh widget js³ is inserted in the notebooks that have bokeh loaded (using the installed or specified bokeh version)
- **remove_all_outputs_tags** (*set*) – Tags indicating cells for which the outputs are to be removed, matches tags in cell.metadata.tags.
- **remove_cell_tags** (*set*) – Tags indicating which cells are to be removed, matches tags in cell.metadata.tags.
- **remove_input_tags** (*set*) – Tags indicating cells for which input is to be removed, matches tags in cell.metadata.tags.
- **remove_single_output_tags** (*set*) – Tags indicating which individual outputs are to be removed, matches output i tags in cell.outputs[i].metadata.tags.

¹ <http://cdn.pydata.org/bokeh/release/bokeh-0.12.0.min.js>

² <http://cdn.pydata.org/bokeh/release/bokeh-0.12.0.min.css>

³ <http://cdn.pydata.org/bokeh/release/bokeh-widgets-0.12.0.min.js>

- **tocdepth** (*int*) – Depth to expand table-of-contents trees to. To disable, set to 0. For automatic depth, set to -1. Default: -1
- **binder_url** (*str*) – Link to the notebook on mybinder.org or equivalent

Attributes

<i>binder_urls</i>	
<i>in_dir</i>	The input directories
<i>out_dir</i>	The output directories
<i>urls</i>	

Methods

<i>from_sphinx</i> (app)	Class method to create a <i>Gallery</i> instance from the
<i>get_binder_url</i> (nbfile)	Return the url corresponding to the given notebook file
<i>get_url</i> (nbfile)	Return the url corresponding to the given notebook file
<i>process_directories</i> ()	Create the rst files from the input directories in the
<i>recursive_processing</i> (base_dir, target_dir, it)	Method to recursively process the notebooks in the <i>base_dir</i>

References

binder_urls

classmethod from_sphinx (*app*)

Class method to create a *Gallery* instance from the configuration of a sphinx application

get_binder_url (*nbfile*)

Return the url corresponding to the given notebook file

Parameters **nbfile** (*str*) – The path of the notebook relative to the corresponding :attr:in_dir

Returns The url or None if no url has been specified

Return type *str* or *None*

get_url (*nbfile*)

Return the url corresponding to the given notebook file

Parameters **nbfile** (*str*) – The path of the notebook relative to the corresponding :attr:in_dir

Returns The url or None if no url has been specified

Return type *str* or *None*

in_dir

The input directories

out_dir

The output directories

process_directories ()

Create the rst files from the input directories in the *in_dir* attribute

recursive_processing (*base_dir, target_dir, it*)

Method to recursively process the notebooks in the *base_dir*

Parameters

- **base_dir** (*str*) – Path to the base example directory (see the *examples_dir* parameter for the *Gallery* class)
- **target_dir** (*str*) – Path to the output directory for the rst files (see the *gallery_dirs* parameter for the *Gallery* class)
- **it** (*iterable*) – The iterator over the subdirectories and files in *base_dir* generated by the `os.walk()` function

urls

class LinkGalleriesDirective (*name, arguments, options, content, lineno, content_offset, block_text, state, state_machine*)

Bases: `docutils.parsers.rst.Directive` **Methods**

`create_image_nodes`(*header, thumb_url, key* [,

...])

`get_outdirs`()

`run`()

Attributes

`has_content`

`bool(x) -> bool`

`option_spec`

`dict() -> new empty dictionary`

create_image_nodes (*header, thumb_url, key, link_url=None*)

get_outdirs ()

has_content

option_spec

run ()

class NotebookProcessor (*infile, outfile, disable_warnings=True, preprocess=True, clear=True, code_example=None, supplementary_files=None, other_supplementary_files=None, thumbnail_figure=None, url=None, insert_bokeh=False, insert_bokeh_widgets=False, tag_options={}, binder_url=None*)

Bases: `object`

Class to run process one ipython notebook and create the necessary files

Parameters

- **infile** (*str*) – path to the existing notebook
- **outfile** (*str*) – path to the new notebook
- **disable_warnings** (*bool*) – Boolean to control whether warnings shall be included in the rst file or not
- **preprocess** (*bool*) – If True, the notebook is processed when generating the rst file
- **clear** (*bool*) – If True, the output in the download notebook is cleared

- **code_example** (*str*) – A python code sample that shall be used instead of a thumbnail figure in the gallery. Note that you can also include a 'code_example' key in the metadata of the notebook
- **supplementary_files** (*list of str*) – Supplementary data files that shall be copied to the output directory and inserted in the rst file for download
- **other_supplementary_files** (*list of str*) – Other supplementary data files that shall be copied but not inserted for download
- **thumbnail_figure** (*int*) – The number of the figure that shall be used for download or a path to a file
- **url** (*str*) – The url where to download the notebook
- **insert_bokeh** (*False or str*) – The version string for bokeh to use for the style sheet
- **insert_bokeh_widgets** (*bool or str*) – The version string for bokeh to use for the widgets style sheet
- **tag_options** (*dict*) – A dictionary with traitlets for the `nbconvert.preprocessors.TagRemovePreprocessor`
- **binder_url** (*str*) – Link to the repository on mybinder.org or equivalent

Attributes

<code>BOKEH_JS</code>	<code>str(object='') -> str</code>
<code>BOKEH_STYLE_SHEET</code>	<code>str(object='') -> str</code>
<code>BOKEH_TEMPLATE</code>	<code>str(object='') -> str</code>
<code>BOKEH_WIDGETS_JS</code>	<code>str(object='') -> str</code>
<code>BOKEH_WIDGETS_STYLE_SHEET</code>	<code>str(object='') -> str</code>
<code>BOKEH_WIDGETS_TEMPLATE</code>	<code>str(object='') -> str</code>
<code>CODE_DOWNLOAD</code>	base string for downloading the python file and ipython notebook
<code>CODE_DOWNLOAD_NBVIEWER</code>	base string for viewing the notebook in the jupyter nbviewer
<code>CODE_RUN_BINDER</code>	base string for viewing the notebook in the binder
<code>CODE_TEMPLATE</code>	<code>str(object='') -> str</code>
<code>DATA_DOWNLOAD</code>	base string for downloading supplementary data
<code>THUMBNAIL_TEMPLATE</code>	base string for creating the thumbnail
<code>code_div</code>	The string for creating a code example for the gallery
<code>code_example</code>	The code example out of the notebook metadata
<code>other_supplementary_files</code>	The supplementary files of this notebook
<code>pictures</code>	Paths to the pictures of this notebook
<code>reference</code>	The rst label of this example
<code>remove_tags</code>	
<code>supplementary_files</code>	The supplementary files of this notebook
<code>thumb_file</code>	Path to the thumbnail image
<code>thumbnail_div</code>	The string for creating the thumbnail of this example
<code>url</code>	The url on jupyter nbviewer for this notebook or None if unknown

Methods

<code>copy_thumbnail_figure()</code>	The integer of the thumbnail figure
<code>create_py(nb[, force])</code>	Create the python script from the notebook node
<code>create_rst(nb, in_dir, odir)</code>	Create the rst file from the notebook node
<code>create_thumb()</code>	Create the thumbnail for html output
<code>data_download(files)</code>	Create the rst string to download supplementary data
<code>get_description()</code>	Get summary and description of this notebook
<code>get_out_file([ending])</code>	get the output file with the specified <i>ending</i>
<code>get_thumb_path(base_dir)</code>	Get the relative path to the thumb nail of this notebook
<code>process_notebook([disable_warnings])</code>	Process the notebook and create all the pictures and files
<code>save_thumbnail(image_path)</code>	Save the thumbnail image
<code>scale_image(in_fname, out_fname, max_width, ...)</code>	Scales an image with the same aspect ratio centered in an

BOKEH_JS

BOKEH_STYLE_SHEET

BOKEH_TEMPLATE

BOKEH_WIDGETS_JS

BOKEH_WIDGETS_STYLE_SHEET

BOKEH_WIDGETS_TEMPLATE

CODE_DOWNLOAD

base string for downloading the python file and ipython notebook

CODE_DOWNLOAD_NBVIEWER

base string for viewing the notebook in the jupyter nbviewer

CODE_RUN_BINDER

base string for viewing the notebook in the binder

CODE_TEMPLATE

DATA_DOWNLOAD

base string for downloading supplementary data

THUMBNAIL_TEMPLATE

base string for creating the thumbnail

code_div

The string for creating a code example for the gallery

code_example

The code example out of the notebook metadata

copy_thumbnail_figure ()

The integer of the thumbnail figure

create_py (nb, force=False)

Create the python script from the notebook node

create_rst (nb, in_dir, odir)

Create the rst file from the notebook node

create_thumb ()

Create the thumbnail for html output

data_download (*files*)

Create the rst string to download supplementary data

get_description ()

Get summary and description of this notebook

get_out_file (*ending='rst'*)

get the output file with the specified *ending*

get_thumb_path (*base_dir*)

Get the relative path to the thumb nail of this notebook

other_supplementary_files

The supplementary files of this notebook

pictures

Paths to the pictures of this notebook

process_notebook (*disable_warnings=True*)

Process the notebook and create all the pictures and files

This method runs the notebook using the `nbconvert` and `nbformat` modules. It creates the `outfile` notebook, a python and a rst file

reference

The rst label of this example

remove_tags

save_thumbnail (*image_path*)

Save the thumbnail image

scale_image (*in_fname, out_fname, max_width, max_height*)

Scales an image with the same aspect ratio centered in an image with a given `max_width` and `max_height` if `in_fname == out_fname` the image can only be scaled down

supplementary_files

The supplementary files of this notebook

thumb_file

Path to the thumbnail image

thumbnail_div

The string for creating the thumbnail of this example

url

The url on jupyter nbviewer for this notebook or None if unknown

align (*argument*)

Conversion function for the “align” option.

create_dirs (**dirs*)

gallery_config

dictionary containing the configuration of the example gallery.

Possible keys for the dictionary are the initialization keys of the `Gallery` class

isstring (*s*)

nbviewer_link (*url*)

Return the link to the Jupyter nbviewer for the given notebook url

process_examples

Boolean controlling whether the rst files shall created and examples processed

`setup` (*app*)

1.6 Changelog

1.6.1 v0.3.1

This patch fixes some minor logging issues with sphinx >1.7.6

Changed

- Minor compatibility fix for using the logger with Sphinx
- Corrected typos [see PR #1](#)

1.6.2 v0.3.0

Added

- The removal of tags for the converted rst file. With [nbconvert 5.3](#) we have the `nbconvert.preprocessors.TagRemovePreprocessor` available which gave the motivation to 4 new gallery configuration values, namely

remove_all_outputs_tags: set Tags indicating cells for which the outputs are to be removed, matches tags in `cell.metadata.tags`.

remove_cell_tags: set Tags indicating which cells are to be removed, matches tags in `cell.metadata.tags`.

remove_input_tags: set Tags indicating cells for which input is to be removed, matches tags in `cell.metadata.tags`.

remove_single_output_tags: set Tags indicating which individual outputs are to be removed, matches output `i` tags in `cell.outputs[i].metadata.tags`.

The tags specified by these configuration values will be removed in the rst file.

1.6.3 v0.2.2

Added

- The `linkgalleries` directive now can also insert links to the current sphinx project that is build

Changed

- the `linkgalleries` directive uses the styles from the `example_gallery_styles.css`, i.e. the same style as it is used in the processed example gallery.

1.6.4 v0.2.1

Changed

- Minor bug fix in `option_spec` of `LinkGalleriesDirective`

1.6.5 v0.2.0

Added

- Added changelog
- Added linkgalleries directive

Changed

- The name of a thumbnail is now `reference + '_thumb.png'` where `reference` is the section label of the rst file
- Reference labels are now all lower case

1.7 Installation

Simply install it via `pip`:

```
$ pip install sphinx-nbexamples
```

Or you install it via:

```
$ python setup.py install
```

from the [source on GitHub](#).

1.8 Requirements

The package requires

- `Sphinx`>=1.3: The python library for generating automated documentation
- `jupyter`: The jupyter framework for jupyter notebooks. `sphinx-nbexamples` explicitly depends on
 - `nbconvert`: For converting jupyter notebooks to RST
 - `jupyter_client`: For managing the kernels
 - `ipykernel`: For installing an ipython kernel and run the notebooks

1.9 Indices and tables

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

S

`sphinx_nbexamples`, 13

A

`align()` (in module *sphinx_nbexamples*), 20

B

`binder_urls` (*Gallery attribute*), 16

`BOKEH_JS` (*NotebookProcessor attribute*), 19

`BOKEH_STYLE_SHEET` (*NotebookProcessor attribute*), 19

`BOKEH_TEMPLATE` (*NotebookProcessor attribute*), 19

`BOKEH_WIDGETS_JS` (*NotebookProcessor attribute*), 19

`BOKEH_WIDGETS_STYLE_SHEET` (*NotebookProcessor attribute*), 19

`BOKEH_WIDGETS_TEMPLATE` (*NotebookProcessor attribute*), 19

C

`code_div` (*NotebookProcessor attribute*), 19

`CODE_DOWNLOAD` (*NotebookProcessor attribute*), 19

`CODE_DOWNLOAD_NBVIEWER` (*NotebookProcessor attribute*), 19

`code_example` (*NotebookProcessor attribute*), 19

`CODE_RUN_BINDER` (*NotebookProcessor attribute*), 19

`CODE_TEMPLATE` (*NotebookProcessor attribute*), 19

configuration value

`example_gallery_config`, 5

`process_examples`, 4

`copy_thumbnail_figure()` (*NotebookProcessor method*), 19

`create_dirs()` (in module *sphinx_nbexamples*), 20

`create_image_nodes()` (*LinkGalleriesDirective method*), 17

`create_py()` (*NotebookProcessor method*), 19

`create_rst()` (*NotebookProcessor method*), 19

`create_thumb()` (*NotebookProcessor method*), 19

D

`DATA_DOWNLOAD` (*NotebookProcessor attribute*), 19

`data_download()` (*NotebookProcessor method*), 19

E

`example_gallery_config`
configuration value, 5

F

`from_sphinx()` (*sphinx_nbexamples.Gallery class method*), 16

G

`Gallery` (*class in sphinx_nbexamples*), 14

`gallery_config` (in module *sphinx_nbexamples*), 20

`get_binder_url()` (*Gallery method*), 16

`get_description()` (*NotebookProcessor method*), 20

`get_out_file()` (*NotebookProcessor method*), 20

`get_outdirs()` (*LinkGalleriesDirective method*), 17

`get_thumb_path()` (*NotebookProcessor method*), 20

`get_url()` (*Gallery method*), 16

H

`has_content` (*LinkGalleriesDirective attribute*), 17

I

`in_dir` (*Gallery attribute*), 16

`isstring()` (in module *sphinx_nbexamples*), 20

L

`linkgalleries` (*directive*), 8

`LinkGalleriesDirective` (*class in sphinx_nbexamples*), 17

N

`nbviewer_link()` (in module *sphinx_nbexamples*), 20

`NotebookProcessor` (*class in sphinx_nbexamples*), 17

O

`option_spec` (*LinkGalleriesDirective attribute*), 17

other_supplementary_files (*NotebookProcessor* attribute), 20
out_dir (*Gallery* attribute), 16

P

pictures (*NotebookProcessor* attribute), 20
process_directories() (*Gallery* method), 16
process_examples
 configuration value, 4
process_examples (in module *sphinx_nbexamples*),
 20
process_notebook() (*NotebookProcessor* method),
 20

R

recursive_processing() (*Gallery* method), 16
reference (*NotebookProcessor* attribute), 20
remove_tags (*NotebookProcessor* attribute), 20
run() (*LinkGalleriesDirective* method), 17

S

save_thumbnail() (*NotebookProcessor* method), 20
scale_image() (*NotebookProcessor* method), 20
setup() (in module *sphinx_nbexamples*), 20
sphinx_nbexamples (module), 13
supplementary_files (*NotebookProcessor* at-
tribute), 20

T

thumb_file (*NotebookProcessor* attribute), 20
thumbnail_div (*NotebookProcessor* attribute), 20
THUMBNAIL_TEMPLATE (*NotebookProcessor* at-
tribute), 19

U

url (*NotebookProcessor* attribute), 20
urls (*Gallery* attribute), 17