



# **Li-Pro.Net Sphinx Primer**

***Version 0.0.3+RTDEXT***

**The LP/N Documentation Team**

**Sep 11, 2020**

**Li-Pro.Net  
Jena, Germany  
Email: [info@li-pro.net](mailto:info@li-pro.net)**

## Table of Contents

<b>1</b>	<b>Concepts</b>	<b>8</b>
<b>2</b>	<b>Extensions</b>	<b>9</b>
2.1	Spelling Checker . . . . .	9
2.1.1	Private Dictionaries . . . . .	9
2.2	BibTeX Citations . . . . .	10
2.3	LinuxDoc . . . . .	12
2.3.1	Flat list table . . . . .	12
2.4	Program Output . . . . .	15
2.4.1	Complete output . . . . .	15
2.4.2	Shortening the output . . . . .	16
2.4.3	Mimicking shell input . . . . .	16
2.4.4	Command execution and shell expansion . . . . .	17
2.4.5	Error handling . . . . .	18
2.5	Mathematical Plots . . . . .	19
2.5.1	Expressions . . . . .	19
2.5.2	Plots . . . . .	20
2.6	PGF/TikZ LaTeX Pictures . . . . .	24
2.7	Block Diagram Family . . . . .	27
2.7.1	Block Diagram . . . . .	28
2.7.2	Sequence Diagram . . . . .	31
2.7.3	Activity Diagram . . . . .	34
2.7.4	Network Diagram . . . . .	36
2.8	Tabbed Content . . . . .	42
2.9	Paneled Content . . . . .	45
2.10	Email Obfuscate . . . . .	46
<b>3</b>	<b>Themes</b>	<b>48</b>
3.1	Read the Docs Sphinx Theme . . . . .	48

<b>4</b>	<b>Cheat Sheet</b>	<b>49</b>
	<b>Appendices</b>	<b>53</b>
<b>A</b>	<b>Appendix</b>	<b>53</b>
A.1	License . . . . .	53
A.2	Credits . . . . .	61
<b>B</b>	<b>Glossary</b>	<b>62</b>
B.1	Terms . . . . .	62
B.1.1	Commons . . . . .	62
B.1.2	Technologies . . . . .	63
	<b>Listings</b>	<b>65</b>
	<b>List of Tables</b>	<b>67</b>
	<b>List of Figures</b>	<b>69</b>
	<b>List of Equations</b>	<b>71</b>
	<b>List of Downloads</b>	<b>73</b>
	<b>List of Issues (To-Do)</b>	<b>75</b>
	<b>Bibliography</b>	<b>78</b>
	<b>Index</b>	<b>79</b>

## Summary of Li-Pro.Net Sphinx Primer

**Abstract** This document has documentation of the Li-Pro.Net community which is a effort to improve the technical writing of any kind of documents and publishings. In short words: How to write Li-Pro.Net documentation with Sphinx.

### Involved Components

- *Sphinx*
- *Docutils*
- *reStructuredText*

### Audience

- Project members / maintainer
- Hard- and software developer
- Integrators and testers
- Technical writer / editor

**Status** preliminary (*some mature, much in progress*)

**Version** 0.0.3

**Release** 0.0.3+RTDEXT

**Date** Sep 11, 2020

**Authors** The LP/N Documentation Team

- Stephan Linz <[linz@li-pro.net](mailto:linz@li-pro.net)>

**Copyright** Copyright © 2020, Li-Pro.Net, The LP/N Documentation Team and individual contributors.—all rights reserved. This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License ([CC-BY-SA-3.0](https://creativecommons.org/licenses/by-sa/3.0/)). See the file LICENSE and CREDITS that comes with the documentation.

**Organization** Li-Pro.Net

**Contact** Stephan Linz <[linz@li-pro.net](mailto:linz@li-pro.net)>

**Address** Jena, Germany



## Legal Notice of Li-Pro.Net Sphinx Primer

Creative Commons Legal Code

Attribution-ShareAlike 3.0 Unported

For details of the terms and definitions, representations, warranties and disclaimer see the file LICENSE that comes with the documentation and/or read the online version ([CC-BY-SA-3.0](#)).

### You are free:

**to Share** —to copy, distribute and transmit the work

**to Remix** —to adapt the work to make commercial use of the work

### Under the following conditions:

**Attribution** —You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

**Share Alike** —If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.

### With the understanding that:

**Waiver** —Any of the above conditions can be waived if you get permission from the copyright holder.

**Public Domain** —Where the work or any of its elements is in the public domain under applicable law, that status is in no way affected by the license.

**Other Rights** —In no way are any of the following rights affected by the license:

- Your fair dealing or fair use rights, or other applicable copyright exceptions and limitations;
- The author's moral rights;
- Rights other persons may have either in the work itself or in how the work is used, such as publicity or privacy rights.



**Version history of Li-Pro.Net Sphinx Primer**

Table 1: Li-Pro.Net Sphinx Primer Document Revisions

<b>Version</b>	<b>Change</b>	<b>Passed</b>	<b>Date</b>
<b>0.0.3</b>	initial content in chapters “Extensions”	Stephan Linz	2020-09-11
<b>0.0.2</b>	initial content in chapters “Cheat Sheet” and “Themes”	Stephan Linz	2020-09-08
<b>0.0.1</b>	base document skeleton	Stephan Linz	2020-09-08
<b>0.0</b>	preliminary, project created	Stephan Linz	2020-09-05





## PREAMBLE

### How to write Li-Pro.Net documentation with Sphinx.

Excerpts from the [Sphinx Tutorial](#) by Eric Holscher and [Documentation Style Guide](#) by Bareos GmbH & Co. KG and others. See [\[juh2019swdocwsp\]](#) for an introduction to Sphinx.

This documentation is built using *Sphinx*, a static-site generator designed to create structured, semantic, and internally consistent documentation. Source documents are written in *reStructuredText*, a semantic, extensible markup syntax similar to Markdown.

- [reStructuredText Primer](#) - Introduction to reStructuredText
  - [reStructuredText Quick Reference](#)
  - [reStructuredText 1-page cheat sheet](#)
- [Sphinx Markup](#) - Detailed guide to Sphinx's markup concepts and reStructuredText extensions

---

**Note:** *Sphinx* and *reStructuredText* can be very flexible. For the sake of consistency and maintainability, this how to guide is *highly opinionated* about how documentation source files are organized and marked up.

---

*Section author: Stephan Linz <[linz@li-pro.net](mailto:linz@li-pro.net) >*

t.b.d.

Section author: Stephan Linz <[linz@li-pro.net](mailto:linz@li-pro.net)>

## 2.1 Spelling Checker

**PyPI Package** <https://pypi.org/project/sphinxcontrib-spelling/>

**Documentation** <https://sphinxcontrib-spelling.readthedocs.io/>

**Git Repository** <https://github.com/sphinx-contrib/spelling>

Spelling checker for Sphinx. It uses *PyEnchant* to produce a report showing misspelled words.

### Features

1. Supports multiple source languages using the standard enchant dictionaries.
2. Supports project-specific dictionaries for localized jargon and other terminology that may not appear in the global dictionaries.
3. Suggests alternatives to words not found in the dictionary, when possible.

It consists:

- [sphinxcontrib.spelling](#): spelling checker for Sphinx

### 2.1.1 Private Dictionaries

For more details, see [Configuration Options](#) section *Private Dictionaries*.

#### **.. spelling::**

The `.. spelling::` directive can be used to create a list of words known to be spelled correctly within a single file. For example, if a document refers to a person or project by name, the name can be added to the list of known words for just that single document.

When a more common list of words is needed, related to check multiple document at once, the `spelling_word_list_filename` (page 10) variable should be set properly.

### **spelling\_word\_list\_filename**

That is a list specifying files containing a list of words known to be spelled correctly but that do not appear in the referred language dictionary. The files should contain one word per line. Refer to the *PyEnchant* tutorial for details.

## **2.2 BibTeX Citations**

**PyPI Package** <https://pypi.org/project/sphinxcontrib-bibtex/>

**Documentation** <https://sphinxcontrib-bibtex.readthedocs.org/>

**Git Repository** <https://github.com/mcmtrroffaes/sphinxcontrib-bibtex>

Allowing *BibTeX* citations to be inserted into documentation via a `.. bibliography::` directive, and a `:cite:` role, which work similarly to *LaTeX*'s `\begin{thebibliography}` ... `\end{thebibliography}` environment and `\cite{cite_key}` command. It consists:

- `sphinxcontrib.bibtex`: Sphinx interface
- `sphinxcontrib.bibtex.roles`: Doctree roles
- `sphinxcontrib.bibtex.nodes`: Doctree nodes
- `sphinxcontrib.bibtex.directives`: Doctree directives
- `sphinxcontrib.bibtex.transforms`: Doctree transforms
- `sphinxcontrib.bibtex.cache`: Cached information

Create a citation to a bibliographic entry.

---

**Todo:** activate “BibTeX Citations” extension.

---

```
.. rst:role:: cite
```

```
For more details, see :rst:role:`scbibtex:cite` role.
```

```
:the example:
```

```
.. code-block:: rst
   :linenos:
```

```
See :cite:`juh2014swdocwspx` for an introduction to Sphinx.
```

```
:which gives:
```

(continues on next page)

(continued from previous page)

See `:cite:`juh2014swdocwsp`` for an introduction to Sphinx.

For this sample you will need a corresponding bibliography for all cited references.

```
.. rst:directive:: bibliography
```

For more details, see `:rst:dir:`scbibtex:bibliography`` directive.

:the example:

```
.. code-block:: rst
   :linenos:

   .. bibliography:: bibliography.bibtex
      :style: kcsalpha
      :encoding: utf
      :all:
```

:which gives:

```
.. only:: html or man or texinfo or text

   .. rubric:: Documentation with Sphinx

.. only:: latex
```

All entries in the central document bibliography list, mostly on the end of the document.

```
.. bibliography:: bibliography.bibtex
   :style: kcsalpha
   :encoding: utf
   :all:
```

:which needs:

The example above processed the following BibTeX file content:

```
.. literalinclude:: bibliography.bibtex
   :caption: BibTeX example file (bibliography.bibtex)
   :language: bibtex
   :emphasize-lines: 1
   :start-at: @book
   :linenos:
```

(continues on next page)

(continued from previous page)

```
.. spelling::
```

Hasecke

## 2.3 LinuxDoc

**Documentation** <https://return42.github.io/linuxdoc/>

**Git Repository** <https://github.com/return42/linuxdoc>

The LinuxDoc library with extensions of the Linux-Kernel documentation, you can use these extensions in common Sphinx projects. It consists:

- `linuxdoc.rstFlatTable`: the `.. flat-table::` reST-directive
- `linuxdoc.rstKernelDoc`: the `.. kernel-doc::` reST-directive
- `linuxdoc.kernel_include`: the `.. kernel-include::` reST-directive
- `linuxdoc.manKernelDoc`: the **kernel-doc-man** builder
- `linuxdoc.cdomain`: replacement for the sphinx C-domain
- `linuxdoc.kfigure`: implements scalable image handling

---

**Todo:** activate “LinuxDoc” extension.

---

### 2.3.1 Flat list table

```
.. flat-table::
```

**See also:**

[About tables](#): [flat-table](#)

The `.. flat-table::`` (FlatTable) is a double-stage list similar to the `.. list-table::`` with some additional features:

- *column-span*: with the role `:cspan: `num`` a cell can be extended through additional columns
- *row-span*: with the role `:rspan: `num`` a cell can be extended through additional rows

- *auto-span*: rightmost cell of a table row over the missing cells on the right side of that table-row. With Option `:fill-cells:` this behavior can be changed from auto span to auto fill, which automatically inserts (empty) cells instead of spanning the last cell.

### Options

- `:header-rows: (integer)`**  
count of header rows
- `:stub-columns: (integer)`**  
count of stub columns
- `:widths: (list of integer)`**  
widths of columns
- `:fill-cells:`**  
instead of auto-span missing cells, insert missing cells

### Roles

- `:cspan:`**  
**`(integer)`**: additional columns (*morecols*)
- `:rspan:`**  
**`(integer)`**: additional rows (*morerows*)

The example below shows how to use this markup. The first level of the staged list is the *table-row*. In the *table-row* there is only one markup allowed, the list of the cells in this *table-row*. Exception are *comments* ( `..` ) and *targets* (e.g. a ref to [row 2 of table's body](#)).

### the example

**Attention: line 2:** The option `:class: longtable` will not be interpreted from directive `.. flat-table::` and has no effects.

```

1 .. flat-table:: LinuxDoc :rst:`.. flat-table::` example (table title)
2   :class: longtable
3   :widths: 15 15 15 15 40
4   :header-rows: 2
5   :stub-columns: 1
6
7   * - :rspan:`1` head / stub
8     - :cspan:`3` head 1.1-4
9
10  * - head 2.1
11    - head 2.2
12    - head 2.3
13    - head 2.4

```

(continues on next page)



(continued from previous page)

```
14
15 * .. row body 1 / this is a comment
16
17 - row 1
18 - :rspan:`2` cell 1-3.1
19 - cell 1.2
20 - cell 1.3
21 - cell 1.4
22
23 * .. Comments and targets are allowed on *table-row* stage.
24 .. _`row body 2`:
25
26 - row 2
27 - cell 2.2
28 - :rspan:`1` :cspan:`1`
29   cell 2.3 with a span over
30
31   * col 3-4 &
32   * row 2-3
33
34 * - row 3
35   - cell 3.2
36
37 * - row 4
38   - cell 4.1
39   - cell 4.2
40   - cell 4.3
41   - cell 4.4
42
43 * - row 5
44   - cell 5.1 with automatic span to right end
45
46 * - row 6
47   - cell 6.1
48   - .. empty cell 6.2 with automatic span to right end
```

:which gives:

```
.. include:: linuxdoc-flat-table-example.rsti
```

## 2.4 Program Output

**PyPI Package** <https://pypi.org/project/sphinxcontrib-programoutput/>

**Documentation** <https://sphinxcontrib-programoutput.readthedocs.org/>

**Git Repository** <https://github.com/NextThought/sphinxcontrib-programoutput>

Literally insert the output of arbitrary commands into documents, helping you to keep your command examples up to date. It consists:

- `sphinxcontrib.programoutput`: insert command output

---

**Todo:** activate “Program Output” extension.

---

### 2.4.1 Complete output

To include the output of a command into your document, use the `.. program-output::` directive provided by this extension.

`.. program-output::`

For more details, see `program-output` directive.

#### The example

```
1 .. program-output:: python --version
```

:which gives:

```
.. program-output:: python --version
```

The whole output of `python --version`, including any messages on standard error, is inserted into the current document, formatted as literal text without any syntax highlighting. You can omit the content of the standard error stream with the `:nstderr:` option.

By default, commands are executed in the top-level source directory. You can choose an alternate working directory with the `:cwd:` option. The argument of this option is either a path relative to the current source file, or a absolute path which means that it is relative to the top level source directory.

## 2.4.2 Shortening the output

Lengthy output can be shortened with the `:ellipsis:` option. Its value denotes lines to omit when inserting the output of the command. Instead, a single ellipsis `...` is inserted.

**the example** If used with a single number, all lines after the specified line are omitted:

```
1 .. program-output:: python --help
2   :ellipsis: 2
```

:which gives:

The above omits all lines after the second one:

```
.. program-output:: python --help
:ellipsis: 2
```

Negative numbers count from the last line backwards, thus replacing 2 with -2 in the above example would only omit the last two lines.

**the example** If used with two comma-separated line numbers, all lines in between the specified lines are omitted. Again, a negative number counts from the last line backwards:

```
1 .. program-output:: python --help
2   :ellipsis: 2,-2
```

:which gives:

The above omits all lines except the first two and the last two lines:

```
.. program-output:: python --help
:ellipsis: 2,-2
```

## 2.4.3 Mimicking shell input

You can mimic shell input with the `.. command-output:: directive`<sup>1</sup>. This directive inserts the command along with its output into the document.

**.. command-output::**

For more details, see [command-output](#)  directive.

**The example**

---

<sup>1</sup>This directive is just an alias for the `.. program-output:: directive` with the `:prompt:` option set.

```
1 .. command-output:: python --version
```

:which gives:

```
.. command-output:: python --version
```

The appearance of this output can be configured with `programoutput_prompt_template`. When used in conjunction with `:ellipsis:`, the command itself and any additional text is *never* omitted. `:ellipsis:` always refers to the *immediate output* of the command.

#### the example

```
1 .. command-output:: python --help
2 :ellipsis: 2
```

:which gives:

```
.. command-output:: python --help
:ellipsis: 2
```

## 2.4.4 Command execution and shell expansion

Normally the command is splittet according to the POSIX shell syntax (see [shlex](#)), and executed directly. Thus special shell features like expansion of environment variables will not work.

#### the example

```
1 .. command-output:: echo "$USER"
```

:which gives:

```
.. command-output:: echo "$USER"
```

To enable these features, enable the `:shell:` option. With this option, the command is literally passed to the system shell.

#### the example

```
1 .. command-output:: echo "$USER"
2 :shell:
```

:which gives:

```
.. command-output:: echo "$USER"
:shell:
```

Other shell features like process expansion consequently work, too.

#### the example

```
1 .. command-output:: ls -l $(which grep)
2   :shell:
```

:which gives:

```
.. command-output:: ls -l $(which grep)
   :shell:
```

**Warning:** Remember to use `:shell:` carefully to avoid unintended interpretation of shell syntax and swallowing of fatal errors!

## 2.4.5 Error handling

If an unexpected exit code (also known as *return code*) is returned by a command, it is considered to have failed. In this case, a build warning is emitted to help you to detect misspelled commands or similar errors. By default, a command is expected to exit with an exit code of 0, all other codes indicate an error. In some cases however, it may be reasonable to demonstrate failed programs. To avoid a (superfluous) warning in such a case, you can specify the expected return code of a command with the `:returncode:` option.

#### the example

```
1 .. command-output:: python -c 'import sys, platform; print(sys.
   ↪version); sys.exit(1)'
2   :returncode: 1
```

:which gives:

```
.. command-output:: python -c 'import sys, platform; print(sys.version); sys.
   ↪exit(1)'
   :returncode: 1
```

The above command returns the exit code 1 (as given to `sys.exit()` [↗](#)), but no warning will be emitted. On the contrary, a warning will be emitted, should the command return 0!

---

**Note:** Upon fatal errors which even prevent the execution of the command neither return code nor command output are available. In this case an error message is inserted into the document instead.

If `:shell:` is set however, most of these fatal errors are handled by the system shell and turned into return codes instead. In this case the error message will only appear in the output of the shell. If you're

using `:shell:`, double-check the output for errors. Best avoid `:shell:`, if possible.

## 2.5 Mathematical Plots

**Attention:** Matplotlib does not support labels and auto-references. You can not refer to a equation and you will never see an entry to `.. mathmpl::` expressions in the list of equations.

**PyPI Package** <https://pypi.org/project/matplotlib/>

**Project Home** <https://matplotlib.org/>

**Documentation** <https://matplotlib.org/contents.html>

**Git Repository** <https://github.com/matplotlib/matplotlib>

**Documentation** <https://matplotlib.org/sampled/doc/index.html>

**Git Repository** <https://github.com/matplotlib/sampled>

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. It consists:

- `matplotlib.sphinxext.mathmpl`: Matplotlib math-text in a Sphinx document
- `matplotlib.sphinxext.plot_directive`: Matplotlib plot in a Sphinx document

**Todo:** activate “Mathematical Plots” extension.

### 2.5.1 Expressions

See the [Writing mathematical expressions](#) for lots more information how to writing mathematical expressions in matplotlib.

With matplotlib in Sphinx you can include inline math

```
:mathmpl: `(\alpha^{ic} > \beta_{ic})` (as role
:rst: `:mathmpl: `(\alpha^{ic} > \beta_{ic})` or display math:

.. mathmpl::

    \sum_{i=0}^{\infty} x_i
```

`.. mathmpl::`

**The example**

```
1 .. mathmpl::
2
3     \left(\frac{5 - \frac{1}{x}}{4}\right)
```

:which gives:

```
.. include:: matplotlib-mathmpl-example.rst
```

## 2.5.2 Plots

**.. plot::**

See the matplotlib [Pyplot tutorial](#) and the [Gallery](#) for lots of examples of matplotlib plots.

The source code for the plot may be included in one of three ways:

## inline content

### the example

```
1  .. plot::  
2     :align: center  
3     :scale: 75  
4  
5     import matplotlib.pyplot as plt  
6     import matplotlib.image as mpimg  
7     import numpy as np  
8     img = mpimg.imread('https://github.com/matplotlib/matplotlib/raw/  
→master/doc/_static/stinkbug.png')  
9     imgplot = plt.imshow(img)
```

:which gives:

```
.. include:: matplotlib-inline-example.rsti
```



## doctest content

### the example

```
1 .. plot::
2   :format: doctest
3   :align: center
4   :scale: 75
5
6   >>> import matplotlib.pyplot as plt
7   >>> plt.plot([1, 2, 3], [4, 5, 6]) # doctest: +ELLIPSIS
8   [<matplotlib.lines.Line2D object at 0x...>]
```

:which gives:

```
.. include:: matplotlib-doctest-example.rst
```

### source file content

When a path to a source file is given, the Sphinx configuration option `plot_basedir` will respect. It is the base directory, to which `.. plot::` file names are relative to. If **None or empty**, file names are **relative** to the directory where the file containing the directive is.

```
.. ifconfig:: not plot_basedir

:plot_basedir: **None or empty**, file names are **relative**

.. ifconfig:: plot_basedir

:plot_basedir: currently set to :file:`{plot_basedir}`.
```

### the example

```
1 .. plot:: ellipses.py
2   :include-source:
3   :encoding: utf
4   :format: python
5   :align: center
6   :scale: 75
```

:which gives:

```
.. include:: matplotlib-srcfile-example.rst
```

## 3D-Plots

See [mplot3d](#), [mplot3d FAQ](#), and [mplot3d API](#).

the example

```
1  .. plot::
2     :format: python
3     :align: center
4     :scale: 75
5
6     import matplotlib.pyplot as plt
7     from matplotlib import cm
8     from mpl_toolkits.mplot3d import axes3d
9
10    fig = plt.figure()
11    ax = fig.gca(projection='3d')
12    X, Y, Z = axes3d.get_test_data(0.005)
13    ax.plot_surface(X, Y, Z, rstride=8, cstride=8, alpha=0.3)
14    cset = ax.contourf(X, Y, Z, zdir='z', offset=-100, cmap=cm.coolwarm)
15    cset = ax.contourf(X, Y, Z, zdir='x', offset=-40, cmap=cm.coolwarm)
16    cset = ax.contourf(X, Y, Z, zdir='y', offset=40, cmap=cm.coolwarm)
17
18    ax.set_xlabel('X'); ax.set_xlim(-40, 40)
19    ax.set_ylabel('Y'); ax.set_ylim(-40, 40)
20    ax.set_zlabel('Z'); ax.set_zlim(-100, 100)
21
22    plt.show()
```

:which gives:

```
.. include:: matplotlib-mplot3d-example.rsti
```

## 2.6 PGF/TikZ LaTeX Pictures

**Attention:** Only practicable and usable for HTML and [LaTeX](#) builder.

**PyPI Package** <https://pypi.org/project/sphinxcontrib-tikz/>

**Documentation** <http://sphinxcontrib-tikz.readthedocs.io/>

**Git Repository** <https://bitbucket.org/philexander/tikz>

Sphinx extension, which enables the use of the [PGF/TikZ LaTeX](#) package to draw nice pictures.

This extension relies on two software packages being installed on your computer:

1. `latex` with the `tikz` and the `amsmath` packages
2. A software package that is able to convert a PDF to an image. Currently, four different ways of doing this conversion are supported, called conversion “suites”. Below is a list for each suite what must be installed on your computer. Only one such suite need to be installed:
  - *pdf2svg* suite: `pdf2svg` (preferred, default)
  - *Netpbm* suite: `pdftoppm` (part of the *Poppler* PDF library) and `pnmtopng` (part of the *Netpbm* package)
  - *ImageMagick* suite: `pdftoppm` (part of the *Poppler* PDF library) and `convert` (part of the *ImageMagick* package)
  - *GhostScript* suite: `ghostscript`

See [Configuration](#) in the extension documentation for more details.

---

**Todo:** activate “PGF/TikZ LaTeX Pictures” extension.

---

### **:tikz:**

For more details, see [Usage](#) in the extension documentation.

### **inline content**

#### **The example**

```
1 An example role :tikz:`[thick] \node[blue,draw] (a) {A};
2 \node[draw,dotted,right of=a] {B} edge[<-] (a);`
```

:which gives:

```
.. include:: tikz-inline-example.rsti
```

### **.. tikz::**

For more details, see [Usage](#) in the extension documentation.

## explicit markup

### The example

```
1 .. rst-class:: centered
2 .. tikz:: [>=latex',dotted,thick] \draw[->] (0,0) -- (1,1) -- (1,0)
3    -- (2,0);
4    :libs: arrows
```

:which gives:

```
.. include:: tikz-explicit-example.rsti
```

## from source file

### The example

```
1 .. rst-class:: centered
2 .. tikz:: Control system principles (PGF/TikZ example)
3    :include: tikz/ctrlloop.tikz
4    :libs: arrows,shapes
```

:which gives:

```
.. include:: tikz-srcfile-example.rsti
```

**Which needs** The example above comes from the [Control system principles](#) web page and processed the following TikZ file content:

Listing 2.1: TikZ example file (ctrlloop.tikz)

```
1 [auto, node distance=2cm,>=latex']
2 \tikzstyle{block} = [draw, fill=blue!20, rectangle,
3    minimum height=3em, minimum width=6em]
4 \tikzstyle{sum} = [draw, fill=blue!20, circle, node distance=1cm]
5 \tikzstyle{input} = [coordinate]
6 \tikzstyle{output} = [coordinate]
7 \tikzstyle{pinstyle} = [pin edge={to-,thin,black}]
8 % placing the blocks
9 \node [input, name=input] {};
10 \node [sum, right of=input] (sum) {};
11 \node [block, right of=sum] (controller) {Controller};
12 \node [block, right of=controller, pin=
    ↪{[pinstyle]above:Disturbances},
```

(continues on next page)

(continued from previous page)


```

13         node distance=3cm] (system) {System};
14 % draw an edge between the controller and system block to calculate
15 % the coordinate -- need it to place the measurement block
16 \draw [->] (controller) -- node[name=u] {$u$} (system);
17 \node [output, right of=system] (output) {};
18 \node [block, below of=u] (measurements) {Measurements};
19 % once the nodes are placed, connecting them is easy
20 \draw [draw,->] (input) -- node {$r$} (sum);
21 \draw [->] (sum) -- node {$e$} (controller);
22 \draw [->] (system) -- node [name=y] {$y$}(output);
23 \draw [->] (y) |- (measurements);
24 \draw [->] (measurements) -| node[pos=0.99] {$-$}
25                        node [near end] {$y_m$} (sum);
26
27 %Local variables:
28 % coding: utf-8
29 % mode: text
30 % mode: rst
31 % End:
32 % vim: fileencoding=utf-8 filetype=tex :

```

## 2.7 Block Diagram Family

**Todo:** activate “Block Diagram Family” extensions.

`blockdiag`  and its family generate diagram images from simple text files:

```


.. blockdiag::

blockdiag {
    blockdiag -> generates -> "block-diagrams";
    blockdiag -> is -> "very easy!";

    blockdiag [color = "greenyellow"];
    "block-diagrams" [color = "pink"];
    "very easy!" [color = "orange"];
}

```

### Features

1. Supports many types of diagrams
  - block diagram (w/ `blockdiag` )

- sequence diagram (w/ [seqdiag](#))
  - activity diagram (w/ [actdiag](#))
  - logical network diagram (w/ [nwdiag](#))
  - rack-structure diagram (w/ [rackdiag](#))
  - packet header diagram (w/ [packetdiag](#))
2. Generates beautiful diagram images from simple text format (similar to Graphviz's dot format)
  3. Layouts diagram elements automatically
  4. Embeds to many documentations; Sphinx, Trac, Redmine, and some Wikis

### 2.7.1 Block Diagram

[sphinxcontrib-blockdiag](#) is a Sphinx extension for embedding block diagrams. You can embed block diagrams with the `.. blockdiag::` directive.

**PyPI Package** <https://pypi.org/project/sphinxcontrib-blockdiag/>

**Documentation** <http://blockdiag.com/en/blockdiag/sphinxcontrib.html>

**Git Repository** <https://github.com/blockdiag/sphinxcontrib-blockdiag>

Sphinx extension for embedding block diagrams using [blockdiag](#).

#### Features

1. Generate block-diagram from dot like text (basic feature).
2. Multilingualism for node-label (utf-8 only).

---

**Todo:** activate “Block Diagram” extension.

---

### Directive Body Diagram

#### `.. blockdiag::`

For more details, see [sphinxcontrib-blockdiag](#) in the extension demonstration and the README .rst in the extension Git repository.

#### The example

```
1 .. blockdiag::
2   :align: center
3
4   blockdiag {
```

(continues on next page)

(continued from previous page)

```

5      A -> B -> C;
6      B -> D;
7  }
```

:which gives:

```
.. include:: blockdiag-directive-body-example.rsti
```

## Description Table

### the example

```

1  .. blockdiag::
2     :align: center
3     :desctable:
4
5     blockdiag {
6         A -> B -> C;
7         A [description = "browsers in each client"];
8         B [description = "web server"];
9         C [description = "database server"];
10    }
```

:which gives:

```
.. include:: blockdiag-description-table-example.rsti
```

## Include Diagram

### the example

```

1  .. blockdiag:: block.diag
2     :caption: Style attributes to nodes and edges (Block Diagram
3     ↪example)
4     :align: center
5     :width: 640
```

:which gives:

```

.. blockdiag:: block.diag
:caption: Style attributes to nodes and edges (Block Diagram example)
:align: center
:width: 640
```



**which needs** The example above comes from the original [Sample diagrams](#) web page and processed the following file content:

Listing 2.2: Block Diagram example file (block.diag)

```

1 blockdiag {
2     // Set border-style, background-color and text-color to nodes.
3     A [style = dotted];
4     B [style = dashed];
5     C [color = pink, style = "3,3,3,3,15,3"]; //dashed_array format
→style
6     D [shape = circle, color = "#888888", textcolor="#FFFFFF"];
7
8     // Set border-style and color to edges.
9     A -> B [style = dotted];
10    B -> C [style = dashed];
11    C -> D [color = "red", style = "3,3,3,3,15,3"]; //dashed_array
→format style
12
13    // Set numbered-badge to nodes.
14    E [numbered = 99];
15
16    // Set background image to nodes (and erase label).
17    F [label = "", background = "https://github.com/sphinx-doc/sphinx/
→raw/master/doc/_static/sphinx.png"];
18    G [label = "", background = "https://www.python.org/static/
→community_logos/python-logo-master-v3-TM.png"];
19    H [icon = "https://github.com/blockdiag/blockdiag.com/raw/master/
→sources/en/_static/help-browser.png"];
20    I [icon = "https://github.com/blockdiag/blockdiag.com/raw/master/
→sources/en/_static/internet-mail.png"];
21    J [shape = actor]
22
23    // Set arrow direction to edges.
24    E -> F [dir = none, label = edge];
25    F -> G [dir = forward];
26    G -> H [dir = back];
27
28    group {
29        orientation = portrait;
30        color = lightgray;
31        H -> I [dir = both];
32    }
33
34    // Set width and height to nodes.
35    K [width = 192]; // default value is 128
36    L [shape = square, height = 64]; // default value is 40

```

(continues on next page)

(continued from previous page)

```

37
38 // Use thick line
39 J -> K [thick]
40 K -> L;
41 }

```

## 2.7.2 Sequence Diagram

[sphinxcontrib-seqdiag](#) is a Sphinx extension for embedding sequence diagrams. You can embed sequence diagrams with the `.. seqdiag::` directive.

**PyPI Package** <https://pypi.org/project/sphinxcontrib-seqdiag/>

**Documentation** <http://blockdiag.com/en/seqdiag/sphinxcontrib.html>

**Git Repository** <https://github.com/blockdiag/sphinxcontrib-seqdiag>

Sphinx extension for embedding sequence diagrams using [seqdiag](#).

### Features

1. Generate sequence-diagram from dot like text (basic feature).
2. Multilingualism for node-label (utf-8 only).

---

**Todo:** activate “Sequence Diagram” extension.

---

## Directive Body Diagram

### `.. seqdiag::`

For more details, see [sphinxcontrib-seqdiag](#) in the extension demonstration and the README.rst in the extension Git repository.

### The example

```

1 .. seqdiag::
2   :align: center
3
4   seqdiag {
5     # define order of elements
6     # seqdiag sorts elements by order they appear
7     browser; database; webserver;
8
9     browser -> webserver [label = "GET /index.html"];
10    browser <-- webserver;

```

(continues on next page)

(continued from previous page)

```
11     browser  -> webserver [label = "POST /blog/comment"];
12     webserver -> database [label = "INSERT comment
↪"];
13     webserver <-- database;
14     browser <-- webserver;
15 }
```

:which gives:

```
.. include:: seqdiag-directive-body-example.rsti
```

## Description Table

### the example

```
1 .. seqdiag::
2   :align: center
3   :desctable:
4
5   seqdiag {
6     A -> B -> C;
7     A [description = "browsers in each client"];
8     B [description = "web server"];
9     C [description = "database server"];
10  }
```

:which gives:

```
.. include:: seqdiag-description-table-example.rsti
```

## Include Diagram

### the example

```
1 .. seqdiag:: seq.diag
2   :caption: Style attributes to diagram and edges (Sequence Diagram
↪example)
3   :align: center
4   :height: 640
```

:which gives:

(continues on next page)

(continued from previous page)

```

.. seqdiag:: seq.diag
   :caption: Style attributes to diagram and edges (Sequence Diagram example)
   :align: center
   :height: 640

```

**which needs** The example above comes from the original [Sample diagrams](#) web page and processed the following file content:

Listing 2.3: Sequence Diagram example file (seq.diag)

```

1  seqdiag {
2      // Set edge metrix.
3      edge_length = 300; // default value is 192
4      span_height = 80; // default value is 40
5
6      // Set fontsize.
7      default_fontsize = 16; // default value is 11
8
9      // Do not show activity line
10     activation = none;
11
12     // Numbering edges automatically
13     autonumber = True;
14
15     // Change note color
16     default_note_color = lightgreen;
17
18     browser -> webserver [label = "GET \n/index.html"];
19     browser <-- webserver [note = "Apache works!"];
20
21     // Separator
22     === Separator line ===
23
24     // color of edge
25     browser -> webserver [label = "misformatted", color = red];
26
27     // failed edge
28     browser -> webserver [label = "failed browser", failed];
29 }

```

## 2.7.3 Activity Diagram

`sphinxcontrib-actdiag` [↗](https://pypi.org/project/sphinxcontrib-actdiag/) is a Sphinx extension for embedding activity diagrams. You can embed activity diagrams with the `.. actdiag::`` directive.

**PyPI Package** <https://pypi.org/project/sphinxcontrib-actdiag/> [↗](#)

**Documentation** <http://blockdiag.com/en/actdiag/sphinxcontrib.html> [↗](#)

**Git Repository** <https://github.com/blockdiag/sphinxcontrib-actdiag> [↗](#)

Sphinx extension for embedding activity diagrams using `actdiag` [↗](#).

### Features

1. Generate activity-diagram from dot like text (basic feature).
2. Multilingualism for node-label (utf-8 only).

---

**Todo:** activate “Activity Diagram” extension.

---

## Directive Body Diagram

### `.. actdiag::`

For more details, see `sphinxcontrib-actdiag` [↗](#) in the extension demonstration and the README .rst in the extension Git repository.

### The example

```
1  .. actdiag::
2     :align: center
3     :scale: 75
4
5     actdiag {
6         A -> B -> C -> D;
7
8         lane foo {
9             A; B;
10        }
11        lane bar {
12            C; D;
13        }
14    }
```

:which gives:

```
.. include:: actdiag-directive-body-example.rst
```

## Description Table

### the example

```

1  .. actdiag::
2     :align: center
3     :scale: 75
4     :desctable:
5
6     actdiag {
7         A -> B -> C;
8         A [description = "browsers in each client"];
9         B [description = "web server"];
10        C [description = "database server"];
11    }
```

:which gives:

```
.. include:: actdiag-description-table-example.rsti
```

## Include Diagram

### the example

```

1  .. blockdiag:: act.diag
2     :caption: Style attributes to frames and nodes (Activity Diagram
3     ↪example)
4     :align: center
5     :scale: 75
6     :width: 640
```

:which gives:

```

.. actdiag:: act.diag
:caption: Style attributes to frames and nodes (Activity Diagram example)
:align: center
:scale: 75
:width: 640
```

**which needs** The example above comes from the original [Sample diagrams](#) web page and processed the following file content:

Listing 2.4: Activity Diagram example file (act.diag)

```
1 actdiag {
2     write -> convert -> image;
3
4     lane user {
5         label = "User";
6         write [label = "Writing reST"];
7         image [label = "Get diagram IMAGE"];
8     }
9     lane actdiag {
10        convert [label = "Convert reST to Image"];
11    }
12 }
```

## 2.7.4 Network Diagram

[sphinxcontrib-nwdiag](#) is a Sphinx extension for embedding network diagrams. You can embed network diagrams with the `.. nwdiag::`, `.. rackdiag::` and `.. packetdiag::` directives.

**PyPI Package** <https://pypi.org/project/sphinxcontrib-nwdiag/>

**Documentation** <http://blockdiag.com/en/nwdiag/sphinxcontrib.html>

**Git Repository** <https://github.com/blockdiag/sphinxcontrib-nwdiag>

Sphinx extension for embedding network diagrams using [nwdiag](#).

### Features

1. Generate network-diagram from dot like text (basic feature).
2. Multilingualism for node-label (utf-8 only).

---

**Todo:** activate “Network Diagram” extension.

---

## Directive Body Diagram

`.. nwdiag::`

For more details, see [sphinxcontrib-nwdiag](#) in the extension demonstration and the README.rst in the extension Git repository.

### The example

```

1  .. nwdiag::
2     :align: center
3     :scale: 75
4
5     nwdiag {
6         network dmz {
7             web01;
8             web02;
9         }
10    }

```

:which gives:

```
.. include:: nwdiag-directive-body-example.rsti
```

## Description Table

### the example

```

1  .. nwdiag::
2     :align: center
3     :scale: 75
4     :desctable:
5
6     nwdiag {
7         network dmz {
8             web01 [address = "192.168.0.1", description = "web server01
9             ↪"];
10            web02 [address = "192.168.0.2", description = "web server02
11            ↪"];
12        }
13        network internal {
14            web01 [address = "172.0.0.1"];
15            db01 [address = "172.0.0.2,172.0.0.20", description =
16            ↪"database server"];
17        }
18    }

```

:which gives:

```
.. include:: nwdiag-description-table-example.rsti
```



## Include Diagram

### Network

#### the example

```
1 .. nwdiag:: nw.diag
2   :caption: Peer networks and grouping nodes (Network Diagram example)
3   :align: center
4   :scale: 75
5   :width: 640
```

:which gives:

```
.. nwdiag:: nw.diag
   :caption: Peer networks and grouping nodes (Network Diagram example)
   :align: center
   :scale: 75
   :width: 640
```

**which needs** The example above comes from the original [Sample diagrams: nwdiag](#) web page and processed the following file content:

Listing 2.5: Network Diagram example file (nw.diag)

```

1  nwdiag {
2      inet [shape = cloud];
3      inet -- router;
4
5      network front {
6          address = "192.168.0.0/24";
7          router;
8          web01;
9          web02;
10
11         // define network using defined nodes
12         group db {
13             web01;
14             web02;
15         }
16     }
17 }

```

## Rack

### .. rack::

For more details, see [sphinxcontrib-nwdiag](#) in the extension demonstration and the README.rst in the extension Git repository.

#### The example

```

1  .. rackdiag:: rack.diag
2      :caption: Multiple racks with multiple and blocked units (Rack
3      ↪Diagram example)
4      :align: center
5      :height: 480

```

:which gives:

```

.. rackdiag:: rack.diag
   :caption: Multiple racks with multiple and blocked units (Rack Diagram
   ↪example)
   :align: center
   :height: 480

```

**Which needs** The example above comes from the original [Sample diagrams: rack-diag](#) web page and processed the following file content:

Listing 2.6: Rack Diagram example file (rack.diag)

```
1 rackdiag {
2     default_fontsize = 10;
3
4     // define 1st (height) rack
5     rack {
6         16U;
7
8         // define rack items
9         1: UPS [2U, fontsize = 14];
10        3: DB Server;
11        // put 2 units to rack-level 4
12        4: Web\nServer 1;
13        4: Web\nServer 2;
14        5: Web\nServer 3;
15        5: Web\nServer 4;
16        7: Load Balancer;
17        8: L3 Switch;
18    }
19
20    // define 2nd rack
21    rack {
22        12U;
23
24        // define rack items
25        1: UPS [2U, fontsize = 14];
26        3: DB Server;
27        4: Web Server;
28        5: Web Server;
29        6: Web Server;
30        7: Load Balancer;
31        8: L3 Switch;
32    }
33
34    // define 3rd rack (with not available units)
35    rack {
36        12U;
37
38        1: Server;
39        2: Server;
40        3: Server;
41        4: Server;
42        5: N/A [8U, fontsize = 14];
43    }
44 }
```

## Packet

### .. packet::

For more details, see [sphinxcontrib-nwdiag](#) in the extension demonstration and the README.rst in the extension Git repository.

#### The example

```

1 .. packetdiag:: packet.diag
2   :caption: Structure of TCP Header (Packet Diagram example)
3   :align: center
4   :width: 640

```

:which gives:

```

.. packetdiag:: packet.diag
   :caption: Structure of TCP Header (Packet Diagram example)
   :align: center
   :width: 640

```

**Which needs** The example above comes from the original [Sample diagrams: packet-diag](#) web page and processed the following file content:

Listing 2.7: Packet Diagram example file (packet.diag)

```

1 packetdiag {
2     colwidth = 32;
3     node_height = 40;
4     default_fontsize = 12;
5
6     0-15: Source Port;
7     16-31: Destination Port;
8     32-63: Sequence Number;
9     64-95: Acknowledgment Number;
10    96-99: Data\nOffset;
11    100-105: Reserved;
12    106: URG [rotate = 270];
13    107: ACK [rotate = 270];
14    108: PSH [rotate = 270];
15    109: RST [rotate = 270];
16    110: SYN [rotate = 270];
17    111: FIN [rotate = 270];
18    112-127: Window;
19    128-143: Checksum;
20    144-159: Urgent Pointer;
21    160-191: (Options and Padding);
22    192-223: Data [colheight = 3];

```

(continues on next page)

(continued from previous page)

23 }

## 2.8 Tabbed Content

**Attention:** Only practicable and usable for HTML builder.

**PyPI Package** <https://pypi.org/project/sphinx-tabs/>**Documentation** <https://sphinx-tabs.readthedocs.io/>**Git Repository** <https://github.com/executablebooks/sphinx-tabs>

Create tabbed content in Sphinx documentation when building HTML.

### Features

1. Basic and nested tabs.
2. Grouped Tabs.
3. Code Tabs.

---

**Todo:** activate “Tabbed Content” extension.

---

**.. tabs::**

**.. tab::**

For more details, see [Simple Tabs](#) in the extension demonstration and the `README.md` in the extension Git repository.

### The example

```
1 .. tabs::
2
3     .. tab:: **Apples**
4
5         Apples are green, or sometimes red.
6
7     .. tab:: **Pears**
8
9         Pears are green.
10
11 .. tab:: **Oranges**
```

(continues on next page)

(continued from previous page)

```

12
13 Oranges are orange.

```

```
:which gives:
```

```
.. include:: sphinx-tabs-example.rsti
```

Nested tabs are also possible.

### The example

```

1 .. tabs::
2
3   .. tab:: Stars
4
5       .. tabs::
6
7           .. tab:: The Sun
8
9               The closest star to us.
10
11           .. tab:: Proxima Centauri
12
13               The second closest star to us.
14
15           .. tab:: Polaris
16
17               The North Star.
18
19   .. tab:: Moons
20
21       .. tabs::
22
23           .. tab:: The Moon
24
25               Orbits the Earth
26
27           .. tab:: Titan
28
29               Orbits Jupiter

```

```
:which gives:
```

```
.. include:: sphinx-tabs-nested-example.rsti
```

```
.. group-tab::
```

Also tabs can stick together in groups.

### The example

```
1  .. rubric:: operating systems
2
3  .. tabs::
4
5      .. group-tab:: Linux
6
7          **Linux** is Unix-like, but was developed without any Unix
8 ↪code.
9          The Linux kernel originated in 1991, as a project of Linus
10         Torvalds, while a university student in Finland.
11
12     .. group-tab:: Mac OS X
13
14         **Mac OS X** is a line of open core graphical operating
15 ↪systems
16         developed, marketed, and sold by Apple Inc.
17
18     .. group-tab:: Microsoft Windows
19
20         **Microsoft Windows** is a family of proprietary operating
21 ↪systems
22         designed by Microsoft Corporation and primarily targeted to
23         Intel architecture based computers.
24
25 .. rubric:: integrated development environments
26
27 .. tabs::
28
29     .. group-tab:: Linux
30
31         **There is no dedicated or default integrated development
32         environment (IDE)** on *Linux*. `Here is a list`_ of IDEs
33         which will run natively on *Linux*.
34
35     .. group-tab:: Mac OS X
36
37         **Xcode** is an integrated development environment (IDE) for
38         *Mac OS X* containing a suite of software development tools
39         developed by Apple Inc.
40
41     .. group-tab:: Microsoft Windows
42
43         **Microsoft Visual Studio** is an integrated development
44         environment (IDE) from Microsoft Corporation. It is used to
```

(continues on next page)

(continued from previous page)

```

42     develop computer programs uses Microsoft software development
43     platforms such as *Windows API*, *Windows Forms*, *Windows
44     Presentation Foundation*, *Windows Store* and *Microsoft
45     Silverlight*.
46
47     .. _`Here is a list`:
48         https://en.wikipedia.org/wiki/Category:Linux_integrated_
         ↪development_environments

```

:which gives:

```
.. include:: sphinx-tabs-group-example.rst
```

## 2.9 Paneled Content

**Attention:** Only practicable and usable for HTML builder.

**PyPI Package** <https://pypi.org/project/sphinx-panels/>

**Documentation** <https://sphinx-panels.readthedocs.io/>

**Git Repository** <https://github.com/executablebooks/sphinx-panels>

Create paneled content in Sphinx documentation when building HTML.

### Features

1. Panels in grid or cards layout.
2. Panels with click-able link-button.
3. Panels with toggle-able content by drop-downs.
4. Panels with styling: header, footer, images, icons, badges, animations

For more details, see [sphinx-panels](#) in the extension demonstration and the README .md in the extension Git repository.

**Todo:** activate “Paneled Content” extension.

```
.. panels::
```

For more details, see [Panels Usage](#).

```
.. dropdown::
```

For more details, see [Dropdown Usage](#).



**.. link-button::**

For more details, see [Link Buttons](#).

**.. div::**

For more details, see [Div Directive](#).

**:badge:**

**:link-badge:**

For more details, see [Link Badges](#).

**:opticon:**

**:fa:**

For more details, see [Inline Icons](#).

---

### Extension not applicable

This Sphinx extension is quite new and is under constant development. The current behavior disturbs the integration, so the extension is disabled for now (see `conf.py`). Currently known bugs are:

- annoying side effects with the [Tabbed Content](#) (page 42) extension by the automatically integrated and delivered Bootstrap 4.0 CSS
- no proper and practical LaTeX builder support

---

## 2.10 Email Obfuscate

**Attention:** Only practicable and usable for HTML builder.

**PyPI Package** <https://pypi.org/project/sphinxcontrib-email/>

**Documentation** <https://github.com/sphinx-contrib/email/blob/master/README.rst>

**Git Repository** <https://github.com/sphinx-contrib/email>

**Python 3 Fixes** <https://github.com/rexut/sphinxcontrib-email/tree/python3-fixes>

---

**Todo:** activate “Email Obfuscate” extension.

---

To obfuscate an email address use something like:

```
:email:`Name Surname <user@myplace.org>`  
:email:`Name Surname (user@myplace.org)`
```

That renders as Name Surname with the appropriate mailto link.

```
:email:`user@myplace.org`
```

That renders as `user@myplace.org` with the appropriate `mailto` link.

**:email:**

#### The example

```
1 * :email:`Name Surname <user@myplace.org>`  
2 * :email:`Name Surname (user@myplace.org)`  
3 * :email:`user@myplace.org`
```

:which gives:

```
.. include:: email-example.rsti
```

Section author: Stephan Linz <[linz@li-pro.net](mailto:linz@li-pro.net)>

Let's decorate the project documentation. There are a lot of themes for the Sphinx HTML builder available on the [Sphinx Themes Demo Page](#).

This documentation use the [Read the Docs Sphinx Theme](#) as demonstrate at the [Sphinx RTD Theme Demo Page](#).

Another interesting and actively developed theme is [The Sphinx Book Theme](#), the theme by The Executable Book Project.

## 3.1 Read the Docs Sphinx Theme

**PyPI Package** <https://pypi.org/project/sphinx-rtd-theme/>

**Documentation** <https://sphinx-rtd-theme.readthedocs.io/>

**Git Repository** [https://github.com/readthedocs/sphinx\\_rtd\\_theme](https://github.com/readthedocs/sphinx_rtd_theme)

This theme is primarily focused to be used on [Read the Docs](#) but can work with your own sphinx projects. You can find a working demo of the theme in the [theme documentation](#).

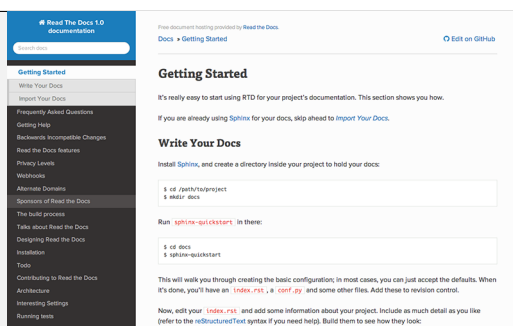


Figure 3.1: Sphinx RTD theme on Desktop

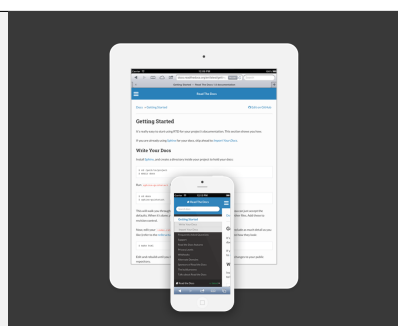


Figure 3.2: Sphinx RTD theme on Mobiles

*Section author: Stephan Linz <[linz@li-pro.net](mailto:linz@li-pro.net)>*

We have made a cheat sheet for helping you remember the syntax for *reStructuredText* & *Sphinx* programs. The basic [reStructuredText Cheat Sheet](#) could also be very helpful.

reStructuredText  
& Sphinx  
Cheatsheet

Styles

```
*Italic/emphasis text*
**Bold/strong text**
``Inline literal text/code``
:sup: super \ Script
:sub: sub \ Script
```

Bullet Lists

```
* Unordered item
* Unordered item

1. Nested ordered item
2. Nested ordered item

    a. Nested ordered item

* Unordered item
```

Definition Lists

```
First term
Definition of first term. It can
span multiple lines which alined
on same indentation.

Second term
Definition of second term.

    Indent addition paragraphs, which
    even can span multiple lines that
    aligned on same indentation.
```

Lieral Code Blocks

```
Here is a literal block::

    Its content are indented.

::

    The :: marker is omitted here
```

The `::` marker will inserted a ":" in the output from the example above. To omit the ":" in the output, precede the `::` marker with white space, or use the marker on a line of its own.

Your additional notes and comments

Section Headings

```
=====
Level 1 Heading
=====

Level 2 Heading
-----
```

- Heading structure is determined only by occurrence order.
  - Heading overline is optional.
  - Under/overlines use the following characters:
- Recommended:** = - ` : . ' " ~ ^ \* + #  
! \$ % & ( ) , / ; < > ? @ [ \ ] { | }

Targets and Links

```
Anchor target
.. _anchorbyref:
.. _Anchor by text:
.. _External link name: http://example.com/
.. [1] A footnote
.. [cit1] A global citation

External links
External link <http://example.com/>`
External link name`_ or `Example <External link name>`_

Internal links
`Anchor by text`_ or `Anchor <Anchor by text>`_
`Anchor by ref <anchorbyref>`_
:ref: anchorbyref`

Footnote
Citation
Reference a footnote [1]_
or a global citation [citel]_

Section link
Section Heading
-----
Link <Section Heading>`_`
```

Tables

Simple table	Header 2	Header 3
Column 1	Column 2	Column 3
Horizontal column span		...
...	...	...

Grid table	Header 2	Header 3
Column 1	Column 2	Vertical column
Horizontal span		span

Simple tables and grid tables can be replaced with external CSV files, using the `csv-table` directive.

Images and Figures

```
.. image:: image.png
:height: 100px
:width: 100px
:align: bottom
:target: target_

.. firgure:: image.png
:height: 100px

Figures are images with captions. They support
all image options.
```

Comments

```
.. This is a single line comment, comments can span multiple lines as well.
.. This is a comment that
   span multiple line.
```

Figure 4.1: Cheat Sheet reStructuredText & Sphinx 1/2

## Directives

reStructuredText **directives** consist of a directive **type**, **arguments** and any number of **options**. Some directives expect a block of indented **content**.

```
.. type:: arguments
   :option: option value

Directive content
```

```
.. container:: [container class]
.. csv-table:: [table title]
   :header: CSV data for headers
   :widths: number [, number]
   :file: filename
   :encoding: encoding
   :header-rows: number
   :delim: character, "tab", or "space"
   :quote: character
   :escape: character
.. rubric:: title
```

## Content Block Directives

```
.. topic:: [title]
.. sidebar:: [title]
   :subtitle: subtitle
.. admonition:: title
.. attention::
.. caution::
.. danger::
.. error::
.. hint::
.. important::
.. note::
.. tip::
.. warning::
.. seealso::
.. deprecated:: [version]
.. versionadded:: [version]
.. versionchanged:: [version]
.. math::
.. raw:: output format
```

```
.. topic:: Example

A topic block.

.. note::
    This is a note.

.. versionchanged:: 1.3
    Something changed.

.. math::
    a_1 = b_1 + c_1

.. raw:: html

<b>HTML output</b>
```

## Table of Contents

```
.. toctree::
   :maxdepth: number (of title levels)
   :glob:
   :hidden:
   :numbered:
   :caption: caption text
   :titlesonly:

[Include paths]
```

## Python Domain

**Python Domain Directives**

```
.. py:module:: module
.. py:class:: signature
.. py:function:: signature
.. py:data:: name
.. py:exception:: name
.. py:attribute:: name
.. py:method:: signature
.. py:staticmethod:: signature
.. py:classmethod:: signature
.. py:decorator:: signature
.. py:currentmodule:: module
```

**Python Domain Directive Options**

```
:param [type] name: description
:type name: type
:raises class: description
:var name: description
:vartype name: type
:returns description:
:rtype type:
```

**Python Domain References**

```
:py:mod:`module`
:py:class:`signature`
:py:func:`signature`
:py:data:`name`
:py:exc:`name`
:py:attr:`name`
:py:meth:`signature`
```

```
:py:const:`name`
```

There are a number of other Sphinx language domains for representing code constructs in reference documentation: *rst*, *c*, *cpp*, *js*, *ruby*, *php*, *dotnet*, *scala*, *go*, *lisp*, *coffee*, and others.

## Code Examples

```
.. highlight:: language
.. linenothreshold: number
.. codeblock:: [language]
   :linenos:
   :emphasize-lines: numbers [, numbers]
   :caption: caption text
   :name: block target name

[Code example, indented]
.. literalinclude:: filename
   :language: language
   :linenos:
   :emphasize-lines: numbers [, numbers]
   :encoding: encoding
   :diff: filename
   :dedent: number
```

Some of the language lexers supported by the code example **language** option: *none*, *python*, *js*, *php*, *ruby*, *perl*, *c*, *cpp*, *csharp*, *go*, *scala*, *lisp*, *coffee*, *dart*, *julia*, *lua*, *html*, *css*, *sass*, *json*, *yaml*, *diff*, *sql*, *bash*, and *shell-session*

## References

```
:role:`title <target>`      Link to target with link text title
:role:`!title`              Don't create link or reference
:role:`-module.Object`      Use last element, Object, for link text
```

**Cross-reference roles**

```
:any:`name`
:doc:`document name`
:download:`filename`
:envvar:`name`
:keyword:`python keyword`
:option:`CLI option`
```

**Cross-reference roles**

```
:abbr:`long (abbreviation)`
:command:`name`
:kbd:`key strokes`
:program:`name`
:pep:`number`
:rfc:`number`
```

```
Brought to you by **Red the Docs**.
We offer private hosting <https://readthedocs.com/>`_
and open source hosting <https://readthodocs.org/>`_
fo *Sphinx* documentation projects.
```

## Your additional notes and comments

Figure 4.2: Cheat Sheet reStructuredText &amp; Sphinx 2/2



Section author: Stephan Linz <[linz@li-pro.net](mailto:linz@li-pro.net)>

## A.1 License

Listing 1.1: License text of the Li-Pro.Net Sphinx Primer

Creative Commons Legal Code

Attribution-ShareAlike 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

(continues on next page)



(continued from previous page)

## 1. Definitions

- a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.
- b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined below) for the purposes of this License.
- c. "Creative Commons Compatible License" means a license that is listed at <https://creativecommons.org/compatiblelicenses> that has been approved by Creative Commons as being essentially equivalent to this License, including, at a minimum, because that license: (i) contains terms that have the same purpose, meaning and effect as the License Elements of this License; and, (ii) explicitly permits the relicensing of adaptations of works made available under that license under this License or a Creative Commons jurisdiction license with the same License Elements as this License.
- d. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.
- e. "License Elements" means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, ShareAlike.
- f. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.
- g. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers,

(continues on next page)

(continued from previous page)

musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

- h. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.
- i. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
- j. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.
- k. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce,

(continues on next page)

(continued from previous page)

limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

- a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
- b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified.";
- c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
- d. to Distribute and Publicly Perform Adaptations.
- e. For the avoidance of doubt:
  - i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
  - ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,
  - iii. Voluntary License Schemes. The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made

(continues on next page)

(continued from previous page)

subject to and limited by the following restrictions:

- a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(c), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(c), as requested.
- b. You may Distribute or Publicly Perform an Adaptation only under the terms of: (i) this License; (ii) a later version of this License with the same License Elements as this License; (iii) a Creative Commons jurisdiction license (either this or a later license version) that contains the same License Elements as this License (e.g., Attribution-ShareAlike 3.0 US)); (iv) a Creative Commons Compatible License. If you license the Adaptation under one of the licenses mentioned in (iv), you must comply with the terms of that license. If you license the Adaptation under the terms of any of the licenses mentioned in (i), (ii) or (iii) (the "Applicable License"), you must comply with the terms of the Applicable License generally and the following provisions: (I) You must include a copy of, or the URI for, the Applicable License with every copy of each Adaptation You Distribute or Publicly Perform; (II) You may not offer or impose any terms on the Adaptation that restrict the terms of the Applicable License or the ability of the recipient of the Adaptation to exercise the rights granted to that recipient under the terms of the Applicable License; (III) You must keep intact all notices that refer to the Applicable License and to the disclaimer of warranties with every copy of the Work as included in the Adaptation You Distribute or Publicly Perform; (IV) when You Distribute or Publicly Perform the Adaptation, You may not impose any effective technological measures on the

(continues on next page)

(continued from previous page)

Adaptation that restrict the ability of a recipient of the Adaptation from You to exercise the rights granted to that recipient under the terms of the Applicable License. This Section 4(b) applies to the Adaptation as incorporated in a Collection, but this does not require the Collection apart from the Adaptation itself to be made subject to the terms of the Applicable License.

- c. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv) , consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(c) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.
- d. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation,

(continues on next page)

(continued from previous page)

modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

#### 5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

#### 7. Termination

- a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
- b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

#### 8. Miscellaneous

- a. Each time You Distribute or Publicly Perform the Work or a Collection,

(continues on next page)

(continued from previous page)

- the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
- b. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
  - c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
  - d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
  - e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
  - f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

#### Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it

(continues on next page)

(continued from previous page)

shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of the License.

Creative Commons may be contacted at <https://creativecommons.org/>.

## A.2 Credits

Listing 1.2: Authors cited when creating the Li-Pro.Net Sphinx Primer

---

AUTHORS (in alphabetical order)

---

Armin Ronacher <armin.ronacher@active-4.com>	Sphinx Code and Documentation
Bareos GmbH & Co. KG <info@bareos.org>	Bareos Documentation
David Goodger <goodger@python.org>	Docutils and reStructuredText
Eric Holscher <eric@ericholscher.com>	Sphinx Tutorial
Georg Brandl <georg@python.org>	Sphinx Code and Documentation
Richard Jones <rjones@ekit-inc.com>	ReStructuredText Primer
Stephan Linz <linz@li-pro.net>	Li-Pro.Net Sphinx Primer

---



Section author: Stephan Linz <[linz@li-pro.net](mailto:linz@li-pro.net)>

## B.1 Terms

### B.1.1 Commons

**Docutils** [Docutils](#) is an open-source text processing system for processing plaintext documentation into useful formats, such as HTML, LaTeX, man-pages, open-document or XML. It includes *reStructuredText*, the easy to read, easy to use, what-you-see-is-what-you-get plaintext markup language.

**See also:**

- [English Wikipedia: reStructuredText](#)

**LaTeX** [LaTeX](#) is a document preparation system for high-quality typesetting. It is most often used for medium-to-large technical or scientific documents but it can be used for almost any form of publishing. LaTeX uses the *TeX* typesetting program for formatting its output, and is itself written in the *TeX* macro language.

**See also:**

- [English Wikipedia: LaTeX](#)

**PyEnchant** [PyEnchant](#) is a Python binding for *Enchant*.

**reStructuredText** [reStructuredText](#) (**RST**, **ReST**, or **reST**) is a file format for textual data used primarily in the Python programming language community for technical documentation. It is part of the *Docutils* project of the Python Doc-SIG (Documentation Special Interest Group)

**See also:**

- [English Wikipedia: reStructuredText](#)

**Sphinx** [Sphinx](#) is a documentation generator written and used by the Python community. It is written in Python, and also used in other environments. Sphinx converts reStructuredText files into HTML websites and other formats including PDF, EPub, Texinfo and man.

*reStructuredText* is extensible, and Sphinx exploits its extensible nature through a number of extensions—for autogenerating documentation from source code, writing mathematical notation or highlighting source code, etc.

**See also:**

- [English Wikipedia: Sphinx \(documentation generator\)](#)

### B.1.2 Technologies

**BibTeX** [BibTeX](#) is a widely used bibliography management tool in *LaTeX*, with BibTeX the bibliography entries are kept in a separate file and then imported into the main document.

**See also:**

- [English Wikipedia: BibTeX](#)

**Enchant** [Enchant](#) is a free software project developed as part of the AbiWord word processor with the aim of unifying access to the various existing spell-checker software.

**See also:**

- [English Wikipedia: Enchant \(software\)](#)

**PGF**

**TikZ**

**PGF/TikZ** [PGF/TikZ](#) is a pair of languages for producing vector graphics (for example: technical illustrations and drawings) from a geometric/algebraic description, with standard features including the drawing of points, lines, arrows, paths, circles, ellipses and polygons. PGF, the *Portable Graphic Format*, is a lower-level language, while TikZ, which is written in *TeX*, is a set of higher-level macros that use PGF.

- [English Wikipedia: PGF/TikZ](#)

**TeX** [TeX](#) is a computer language designed for use in typesetting system; in particular, for typesetting math and other technical material. It has been noted as one of the most sophisticated digital typographical systems and is also used for many other typesetting tasks, especially in the form of *LaTeX*, ConTeXt, and other macro packages.

- [English Wikipedia: TeX](#)



## Listings

2.1	TikZ example file (ctrloop.tikz) . . . . .	26
2.2	Block Diagram example file (block.diag) . . . . .	30
2.3	Sequence Diagram example file (seq.diag) . . . . .	33
2.4	Activity Diagram example file (act.diag) . . . . .	36
2.5	Network Diagram example file (nw.diag) . . . . .	39
2.6	Rack Diagram example file (rack.diag) . . . . .	40
2.7	Packet Diagram example file (packet.diag) . . . . .	41
1.1	License text of the Li-Pro.Net Sphinx Primer . . . . .	53
1.2	Authors cited when creating the Li-Pro.Net Sphinx Primer . . . . .	61



## List of Tables

1	Li-Pro.Net Sphinx Primer Document Revisions . . . . .	5
---	---	---



## List of Figures

3.1	Sphinx RTD theme on Desktop . . . . .	48
3.2	Sphinx RTD theme on Mobiles . . . . .	48
4.1	<i>Cheat Sheet</i> <b>reStructuredText &amp; Sphinx 1/2</b> . . . . .	50
4.2	<i>Cheat Sheet</i> <b>reStructuredText &amp; Sphinx 2/2</b> . . . . .	51





## List of Equations



## List of Downloads

---

**Note:** *List of Downloads* is not fully supported for LaTeX. All entries in the list are not linked and can not be provided together with the document.

---

### Legal Notice of Li-Pro.Net Sphinx Primer

All artifacts were selected and download by using this reference URLs:

- LICENSE
- CREDITS



## List of Issues (To-Do)

---

**Todo:** activate “BibTeX Citations” extension.

---

(The [original entry](#) (page 10) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/bibtex.rst, line 28.)

---

**Todo:** activate “Block Diagram Family” extensions.

---

(The [original entry](#) (page 27) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/blockdiag.rst, line 9.)

---

**Todo:** activate “Activity Diagram” extension.

---

(The [original entry](#) (page 34) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/blockdiag/actdiag.rst, line 25.)

---

**Todo:** activate “Block Diagram” extension.

---

(The [original entry](#) (page 28) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/blockdiag/blockdiag.rst, line 25.)

---

**Todo:** activate “Network Diagram” extension.

---

(The [original entry](#) (page 36) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/blockdiag/nwdiag.rst, line 25.)

---

---

**Todo:** activate “Sequence Diagram” extension.

---

(The [original entry](#) (page 31) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/blockdiag/seqdiag.rst, line 25.)

---

**Todo:** activate “Email Obfuscate” extension.

---

(The [original entry](#) (page 46) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/email.rst, line 18.)

---

**Todo:** activate “LinuxDoc” extension.

---

(The [original entry](#) (page 12) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/linuxdoc.rst, line 22.)

---

**Todo:** activate “Mathematical Plots” extension.

---

(The [original entry](#) (page 19) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/matplotlib.rst, line 30.)

---

**Todo:** activate “Program Output” extension.

---

(The [original entry](#) (page 15) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/programoutput.rst, line 18.)

---

**Todo:** activate “Paneled Content” extension.

---

(The [original entry](#) (page 45) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/sphinx-panels.rst, line 29.)

---

**Todo:** activate “Tabbed Content” extension.

---

(The [original entry](#) (page 42) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/sphinx-tabs.rst, line 25.)

---

**Todo:** activate “PGF/TikZ LaTeX Pictures” extension.

---

(The [original entry](#) (page 25) is located in /home/docs/checkouts/readthedocs.org/user\_builds/lpn-doc-sphinx-primer/checkouts/0.0.3/source/extensions/tikz.rst, line 39.)



## Bibliography

[juh2019swdocwsp] Jan Ulrich Hasecke. *Software-Dokumentation mit Sphinx*. CreateSpace (was part of Amazon.com Inc.), today Kindle Direct Publishing (KDP), Seattle, United States of America, 2. edition, 2019. ISBN 1793008779. ISBN-10: 1-79300-877-9, ISBN-13: 978-1793008770, OCLC: 889425279, URL: <https://www.amazon.com/dp/1793008779> (March 2020).

## Symbols

`:fill-cells:` (*directive option*)  
     `flat-table` (*directive*), 13  
`:header-rows:` (*directive option*)  
     `flat-table` (*directive*), 13  
`:stub-columns:` (*directive option*)  
     `flat-table` (*directive*), 13  
`:widths:` (*directive option*)  
     `flat-table` (*directive*), 13

## A

`actdiag` (*directive*), 34  
 Activity Diagram  
     Sphinx Extension, 33  
 Appendix  
     Li-Pro.Net Sphinx Primer, 53

## B

`badge` (*role*), 46  
 Bibliography  
     Li-Pro.Net Sphinx Primer, 77  
 BibTeX, 63  
 BibTeX Citations  
     Sphinx Extension, 10  
 Block Diagram  
     Sphinx Extension, 28  
 Block Diagram Family  
     Sphinx Extension, 27  
`blockdiag` (*directive*), 28

## C

Cheat Sheet  
     Li-Pro.Net Sphinx Primer, 48  
`command-output` (*directive*), 16  
 Concepts  
     Li-Pro.Net Sphinx Primer, 7  
 configuration value  
     `spelling_word_list_filename`, 10

## Credits

    Li-Pro.Net Sphinx Primer, 61  
`cspan` (*role*), 13

## D

`div` (*directive*), 46  
 Docutils, 62  
 Downloads  
     Li-Pro.Net Sphinx Primer, 71  
`dropdown` (*directive*), 45

## E

`email` (*role*), 47  
 Email Obfuscate  
     Sphinx Extension, 46  
 Enchant, 63  
 Equations  
     Li-Pro.Net Sphinx Primer, 69  
 Extension  
     Activity Diagram, Sphinx, 33  
     BibTeX Citations, Sphinx, 10  
     Block Diagram Family, Sphinx, 27  
     Block Diagram, Sphinx, 28  
     Email Obfuscate, Sphinx, 46  
     LinuxDoc, Sphinx, 12  
     Mathematical Plots, Sphinx, 19  
     Network Diagram, Sphinx, 36  
     Paneled Content, Sphinx, 45  
     PGF/TikZ LaTeX Pictures, Sphinx, 24  
     Program Output, Sphinx, 14  
     Sequence Diagram, Sphinx, 31  
     Spelling Checker, Sphinx, 9  
     Tabbed Content, Sphinx, 42  
 Extensions  
     Li-Pro.Net Sphinx Primer, 8

## F

`fa` (*role*), 46

Figures

Li-Pro.Net Sphinx Primer, 67

flat-table (directive), 12

:fill-cells: (directive option), 13

:header-rows: (directive option), 13

:stub-columns: (directive option), 13

:widths: (directive option), 13

## G

Glossary

Li-Pro.Net Sphinx Primer, 61

group-tab (directive), 43

## I

Issues

Li-Pro.Net Sphinx Primer, 73

## L

LaTeX, 62

Li-Pro.Net Sphinx Primer

Appendix, 53

Bibliography, 77

Cheat Sheet, 48

Concepts, 7

Credits, 61

Downloads, 71

Equations, 69

Extensions, 8

Figures, 67

Glossary, 61

Issues, 73

License, 53

List of Downloads, 71

List of Equations, 69

List of Figures, 67

List of Issues, 73

List of Listings, 63

List of Tables, 65

Listings, 63

Tables, 65

Themes, 47

License

Li-Pro.Net Sphinx Primer, 53

link-badge (role), 46

link-button (directive), 45

LinuxDoc

Sphinx Extension, 12

List of Downloads

Li-Pro.Net Sphinx Primer, 71

List of Equations

Li-Pro.Net Sphinx Primer, 69

List of Figures

Li-Pro.Net Sphinx Primer, 67

List of Issues

Li-Pro.Net Sphinx Primer, 73

List of Listings

Li-Pro.Net Sphinx Primer, 63

List of Tables

Li-Pro.Net Sphinx Primer, 65

Listings

Li-Pro.Net Sphinx Primer, 63

## M

Mathematical Plots

Sphinx Extension, 19

mathmpl (directive), 19

## N

Network Diagram

Sphinx Extension, 36

nwdiag (directive), 36

## O

opticon (role), 46

## P

packet (directive), 41

Paneled Content

Sphinx Extension, 45

panels (directive), 45

PGF, 63

PGF/TikZ, 63

PGF/TikZ LaTeX Pictures

Sphinx Extension, 24

plot (directive), 20

Program Output

Sphinx Extension, 14

program-output (directive), 15

PyEnchant, 62

## R

rack (directive), 39

Read the Docs

Sphinx Themes, 48

reStructuredText, 62

rspan (role), 13

## S

seqdiag (directive), 31

Sequence Diagram

Sphinx Extension, 31

spelling (directive), 9

Spelling Checker

Sphinx Extension, 9

spelling\_word\_list\_filename

configuration value, 10

Sphinx, 63

Sphinx

Extension Activity Diagram, 33

- Extension BibTeX Citations, [10](#)
- Extension Block Diagram, [28](#)
- Extension Block Diagram Family, [27](#)
- Extension Email Obfuscate, [46](#)
- Extension LinuxDoc, [12](#)
- Extension Mathematical Plots, [19](#)
- Extension Network Diagram, [36](#)
- Extension Paneled Content, [45](#)
- Extension PGF/TikZ LaTeX Pictures, [24](#)
- Extension Program Output, [14](#)
- Extension Sequence Diagram, [31](#)
- Extension Spelling Checker, [9](#)
- Extension Tabbed Content, [42](#)
- Themes Read the Docs, [48](#)

## T

- `tab` (*directive*), [42](#)
- Tabbed Content
  - Sphinx Extension, [42](#)
- Tables
  - Li-Pro.Net Sphinx Primer, [65](#)
- `tabs` (*directive*), [42](#)
- TeX, [63](#)
- Themes
  - Li-Pro.Net Sphinx Primer, [47](#)
  - Read the Docs, Sphinx, [48](#)
- TikZ, [63](#)
- `tikz` (*directive*), [25](#)
- `tikz` (*role*), [25](#)