

Documenting Python Code

Christian Heitzmann | SimplexaCode AG



SimplexaCode AG | Grimselweg 11/501 | 6005 Lucerne | Switzerland
+41 43 810 06 03 | info@simplexacode.ch | www.simplexacode.ch



Swiss Python Summit | Eastern Switzerland University of Applied Sciences
Rapperswil SG | September 21, 2023

Shortlink and Target Audience

- shortlink to these slides
 - ***link.simplexacode.ch/jwh6***
- target audience
 - everyone working on professional and/or non-small Python projects
 - not relevant for personal hobby Python development, e.g., on Raspberry Pi

Documentation Debts

- 1 Every day you do not document, you increase the software's debts.
- 2 The debts show up more and more every day.
If not with you, then with your employees, especially the new ones.
- 3 Until you reach the point where reverse engineering takes up most of the time.
- 4 Documentation debts are technical debts¹ – with all their consequences.

¹C. Lilienthal, *Langlebige Software-Architekturen*, 3. Auflage, dpunkt.verlag, 2020

Typical Work Environment

- ① **appliance phase:** promise the stars
- ② **first day:** setup the machine
 - best case: disk image
 - worst case: ask coworkers for help
 - typical case: outdated and incomplete wiki pages
- ③ **first month:** try to gain initial understanding of the “architecture”
 - no API or source code comments
 - outdated or missing documentation
 - *“We don’t do that here.”* or *“&\$!#%”*
- ④ **following months and years:** reverse engineering, day in and day out

Typical Excuses (and Undeniable Reasons)









- **“We are agile!”**
 - \Rightarrow Agile does *not* mean no documentation.
- **“No time! No money!”**
 - \Rightarrow Let’s do the math.
- **“We do Clean Code / TDD / XP / [you name it], so the source code is the documentation!”**
 - \Rightarrow Dream on!

\Rightarrow many undeniable reasons for software documentation: ² ³

²C. Heitzmann, *Javadoc mit Stil*, in: JavaSPEKTRUM, 6/2022, link.simplexacode.ch/ne6c

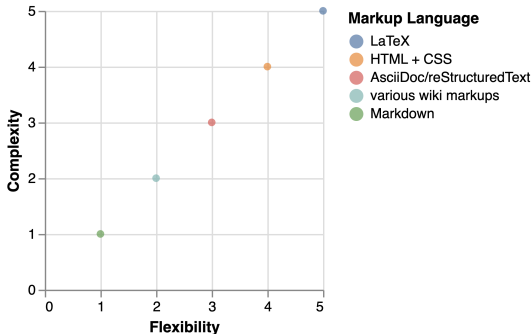
³C. Heitzmann, *Auszeichnungssprachen in der Softwaredokumentation*, in: IT Spektrum, 4/2023

Audiences and Altitudes

- **Who** do you want to address (**audience**)?
 -  external developers, end users, ...
 -  product owners, managers, ...
 -  internal developers
 -  ...
- **What level of detail** do you want to document (**“altitude”**)?
 -  overall software architecture → external documents and diagrams
 -  classes, interfaces, APIs → docstrings, Javadoc, JSDoc, ...
 -  functions internals → source code comments
 -  ...

(Lightweight) Markup Languages

- **markup language:**
machine- and human-readable language for structuring and formatting text and other data
 - e.g., *HTML + CSS*, *LaTeX*, ...
- **lightweight markup language:**
simpler syntax without disturbing the writing and reading flow
 - e.g., *Markdown*, *reStructuredText*, *AsciiDoc*, various wiki markups, ...



Docs as Code

- **Documentation is treated like source code.**
- same IDE
- same version control
- same process integration
- same CI/CD pipeline integration

⇒ documentation in text format (~~binary format~~)

Near Source Code Documentation

- 📖 **Javadoc** using HTML and defined Javadoc tags
- 🔗 **docstrings** usually using reStructuredText syntax
- 📄 **JSDoc** using HTML and defined tags for JavaScript
- 👍 close **proximity between source code and documentation**

Docstrings

- Python uses **docstrings** for documentation.
 - They appear at the **beginning of a** class or function **body**.
 - Cf. **Javadocs** which appear **above** a class or method **definition**.
- Docstrings are data that can be **accessed at runtime**.
- Docstrings can and should contain **reStructuredText syntax**.^{4 5}

```
1 class MyClass:
2     """
3     Example for docstrings.
4
5     This class shows the use of docstrings.
6     """
7     pass
8
9
10 def my_function():
11     """
12     This function does nothing.
13
14     Except that it returns 42.
15
16     :return: 42
17     """
18     return 42
19
20
21 def main():
22     print(my_function.__doc__)
23
24
25 if __name__ == '__main__':
26     main()
```

⁴www.sphinx-doc.org/en/master/usage/restructuredtext/basics.html

⁵docutils.sourceforge.io/rst.html

reStructuredText

docutils.sourceforge.io/docs/user/rst/quickref.html

Generating Python Documentation

- `help(...)`
- `pydoc`
- `Sphinx`⁶
 - `sphinx-apidoc`⁷ using the `autodoc`⁸ extension
 - *PEP 287 – reStructuredText Docstring Format*⁹
or
 - NumPy docstrings format¹⁰

⁶www.sphinx-doc.org

⁷www.sphinx-doc.org/en/master/man/sphinx-apidoc.html

⁸www.sphinx-doc.org/en/master/usage/extensions/autodoc.html

⁹peps.python.org/pep-0287/

¹⁰numpydoc.readthedocs.io/en/latest/format.html

Sphinx Documentation Examples

- Python¹¹
- NumPy¹²
- pandas¹³

- Projects using Sphinx¹⁴
- Learn the documentation style from other modules' source code.

¹¹docs.python.org/3.11/library/csv.html

¹²numpy.org/doc/stable/reference/arrays.ndarray.html

¹³pandas.pydata.org/docs/reference/api/pandas.read_pickle.html

¹⁴www.sphinx-doc.org/en/master/examples.html

AsciiDoc as Lightweight Markup Language

- 21 lightweight markup languages¹⁵
- exclude
 - product or application focus
 - outdated or shadowy existence
- top 4
 - 1 **AsciiDoc**
 - 2 **reStructuredText**
 - 3 **MediaWiki**
 - 4 **Textile**

Comparison of language features [\[edit \]](#)

Comparing language features								
Language	HTML export tool	HTML import tool	Tables	Link titles	class attribute	id attribute	Release date	
AsciiDoc	Yes	Yes	Yes	Yes	Yes	Yes	2002-11-25 ^[1]	
BBCode	No	No	Yes	No	No	No	1998	
Creole	No	No	Yes	No	No	No	2007-07-04 ^[2]	
Gemtext	Yes	?	No	Yes	No	No	2020	
GitHub Flavored Markdown	Yes	No	Yes	Yes	No	No	2011-04-28+	
Jira Formatting Notation	Yes	No	Yes	Yes	No	No	2002 ^[3]	
Markdown	Yes	Yes	No	Yes	Yes/No	Yes/No	2004-03-19 ^{[4][5]}	
Markdown Extra	Yes	Yes	Yes ^[6]	Yes	Yes	Yes	2013-04-11 ^[7]	
MediaWiki	Yes	Yes	Yes	Yes	Yes	Yes	2002 ^[8]	
MultiMarkdown	Yes	No	Yes	Yes	No	No	2009-07-13	
Org-mode	Yes	Yes ^[9]	Yes	Yes	Yes	Yes	2003 ^[10]	
PmWiki	Yes ^[11]	Yes	Yes	Yes	Yes	Yes	2002-01	
POD	Yes	?	No	Yes	?	?	1994	
reStructuredText	Yes	Yes ^[9]	Yes	Yes	Yes	auto	2002-04-02 ^[12]	
setext	Yes	Yes	No	Yes	No	No	1992 ^[13]	
Slack	No	No	No	Yes	No	No	2013+ ^{[14][15]}	
TiddlyWiki	Yes	No	Yes	Yes	Yes	No	2004-09 ^[16]	
Textile	Yes	No	Yes	Yes	Yes	Yes	2002-12-26 ^[17]	
Texty	Yes	Yes	Yes	Yes	Yes	Yes	2004 ^[18]	
txt2tags	Yes	Yes ^[19]	Yes ^[20]	Yes	Yes/No	Yes/No	2001-07-26 ^[21]	
WhatsApp	No	No	No	No	No	No	2016-03-16 ^[22]	

¹⁵en.wikipedia.org/wiki/Lightweight_markup_language

Markdown

- 👍 most popular lightweight markup language
- 👎 no standard
- 👎 many flavors
- 👎 plug-ins or HTML code necessary
- 👎 no tables, cross references, footnotes, YouTube videos, ...

⇒ **Don't use it.**¹⁶

¹⁶E. Holscher, *Why You Shouldn't Use "Markdown" for Documentation*,
www.ericholscher.com/blog/2016/mar/15/dont-use-markdown-for-technical-docs/

AsciiDoc for Python Docstrings?

- I'm not aware of any well-established (Sphinx) converter/extension/plugin.
- Within Python, **stick to Python standards** (here: reStructuredText).
- Compare to **Javadoc for Java** or **JSDoc for JavaScript**.
- **Keep docstrings simple.**

reStructuredText for *All* Software Documentation?

- technologically fine
- **AsciiDoc**
 - 👍 more generic, esp. for non-techies
 - 👎 non-Pythonic in Python
- **reStructuredText**
 - 👍 covers Python docstrings (but not Javadoc or JSDoc)
 - 👎 more nerdy, clumsy, and aimed at tech-savvy people
- compromise: **tagged source code includes in AsciiDoc?**
 - 👍 works similar across all programming languages
 - 👎 increases distance between source code and documentation

⇒ *Carefully* discuss with all developers and stakeholders.

Three Ingredients for Successful Software Documentation

① docs as code infrastructure

- reStructuredText/Sphinx
- AsciiDoc/Asciidoctor
- Javadoc/HTML
- Markdown

② few, but responsible people (“doctator”, “technical writer”)

③ documentation guide¹⁷

¹⁷specimen: GitLab, *Documentation Style Guide*,
docs.gitlab.com/ee/development/documentation/styleguide/

Questions?

SimplexaCode

