

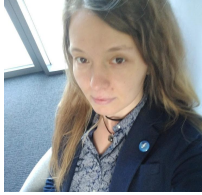
The fundamentals of technical writing

Tooling

Alexandra Nikandrova
Technical Writer

Petr Kovář
Documentation Program Manager

About the authors:



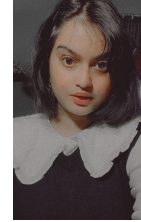
Alexandra Nikandrova

Technical Writer.
Former Devops



Ioanna Gkioka

Technical Writing Manager
Support technical writers
Former university teacher



Kalyani Desai

I am a Technical Writer, Open source contributor,
Fun fact: I tried some strawberry shampoo.
It doesn't taste as good as it smells :P



Arati Ajit Belgaonkar

I am a Technical Writer by profession and
personally, I am an athlete. My hobbies are
gardening and swimming.



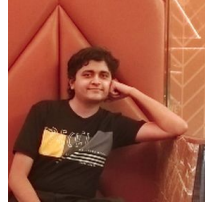
Dominika Vagnerova

Technical Writer
Former journalist

About the authors:



Vendula Ferschmannova
Global Manager



Mayur Patil
Technical Writer 2
RHEL Networking and RDMA



Petr Kovář
Documentation Program Manager
Open source documentation writer and translator.

What we'll discuss today

- Homework follow-up
- Introduction
- Editors
- Languages
 - XML, DocBook
 - Dita toolkit
 - reStructuredText (RTS), Sphinx

Why

- Single source → multiple outputs
- Reusing text
- Unified structure and templating
 - Usability: Structure of each article in documentation must look similar. Must have the same font, highlighting, levelling, and use blank space

How

- Languages and structure created specifically for technical writers
- Templates
- Simplified English



Editors

Editors

- Multiple editors: vi/vim, VS Code, Notepad, Pages, Emacs, etc.
- Choose whatever is comfortable

VS Code preview



Vim editor

There are modes: **normal** (default), insert and command line.

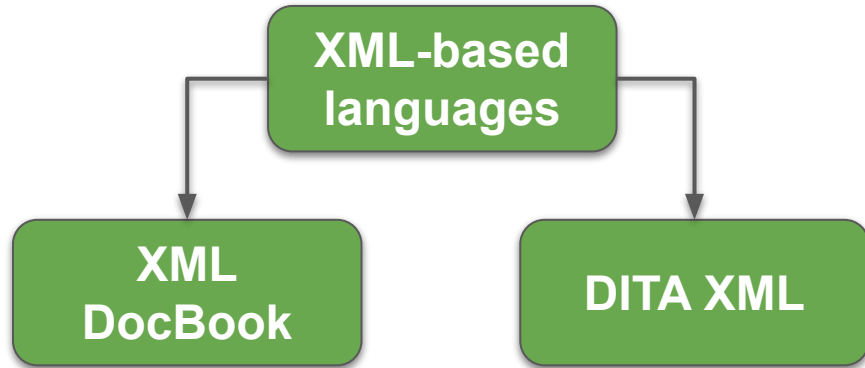
Command	Description
vim FILE_NAME	Create or modify the FILE_NAME in vim.
:q or :ZQ	Quit the file without saving. Perform in command line mode.
:x or :qw!	Save and quit file. Perform in command line mode.
dd	Delete the highlighted text or the current line. Perform in normal mode.
v	Highlight the text. Use <i>left</i> and <i>right</i> arrows to expand the text area. Perform in normal mode.
y	Copy the highlighted text or the current line. Perform in normal mode.
p	Paste the highlighted text or the current line. Perform in normal mode.

Vim tutorial: [vimtutor](https://www.vim.org/vimtutor/)

Useful links: [vim.org](https://www.vim.org/)

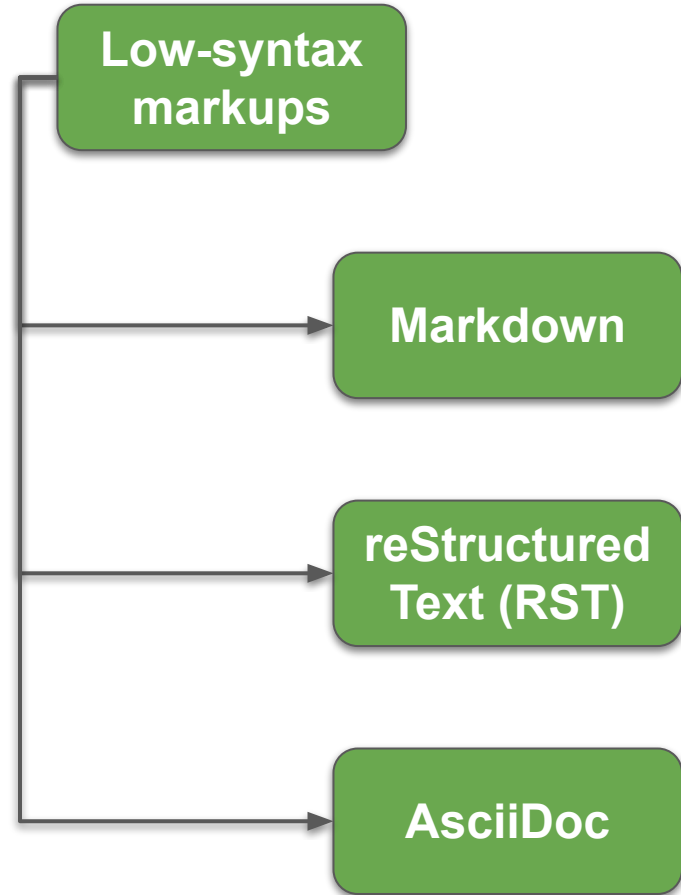


Languages



- Robust Content Management Systems
- Multi-language support
- Single source
- Modular content management
- Commercial/corporate world

- Easy adoption by community members
- Single source
- Modular content management
- Natural connection with GitHub
- Popular in open source world



XML

Example

```
<section>

  <title>Extensible Markup
  Language</title>

  <para>Extensible Markup Language
  (XML) is a markup language that marks
  up data content with tags. XML
  uses
  tags to define content
  structure.</para>

</section>
```

- Extensible Markup Language (XML) is a markup language is a [markup language](#) and [file format](#) for storing, transmitting, and reconstructing arbitrary data (Wikipedia)
- Define data with pair of tags
 - Human-readable
 - Machine-readable
- Each XML document needs to contain XML header (XML declaration):
`<?xml version="1.0" encoding="UTF-8"?>`
- Nesting

DocBook

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<book xml:id="simple_book" xmlns="http://docbook.org/ns/docbook" version="5.0">
  <title>Very simple book</title>
  <chapter xml:id="chapter_1">
    <title>Chapter 1</title>
    <para>Hello world!</para>
    <para>I hope that your day is proceeding <emphasis>splendidly</emphasis>!</para>
  </chapter>
  <chapter xml:id="chapter_2">
    <title>Chapter 2</title>
    <para>Hello again, world!</para>
  </chapter>
</book>
```

- Semantic markup language created just for the purpose of technical writing
- XML uses tags to define content structure
- It consists of:
 - Elements – the basic building blocks
 - Attributes – information related to a specific element
 - Entities – representing an item of data instead of using the data itself
- Advantages: self-describing, widely used, extensibility
- Disadvantages: more lengthy, less readable compared to other markup languages, strict syntax rules
- Learn more:

<https://tdg.docbook.org/tdg/4.5/docbook.html>

DocBook

Example

- Single source, multiple outputs (HTML, PDF, CHM)
- Templates
- Tooling: Oxygen, xMetaL, etc.

```
<?xml version="1.0" encoding="UTF-8"?>
<book xml:id="simple_book" xmlns="http://docbook.org/ns/docbook" version="5.0">
  <title>Very simple book</title>
  <chapter xml:id="chapter_1">
    <title>Chapter 1</title>
    <para>Hello world!</para>
    <para>I hope that your day is proceeding <emphasis>splendidly</emphasis>!</para>
  </chapter>
  <chapter xml:id="chapter_2">
    <title>Chapter 2</title>
    <para>Hello again, world!</para>
  </chapter>
</book>
```

Darwin Information Typing Infrastructure (DITA)

- DITA is an XML-based architecture for creating modular content that allows you to reuse content and interchange content from various sources: topic-based authoring
- Advantages:
 - CMS support
 - WYSIWYG
 - Compounding guides based on versions
- DITA Open Toolkit – the open-source publishing engine: <https://github.com/dita-ot/dita-ot/>

Example

```
<topic xml:lang="en"
id="sample">
  <title>Sample
  title</title>
  <body>
    <p audience="teacher">
      This text is for the
      teacher.</p>
    <p audience="student">
      This text is for the
      student.</p>
  </body>
</topic>
```

DocBook vs. DITA

Language	DocBook	DITA	Markdown	RST	AsciiDoc
Single Source	x	x	x	x	x
Modular content		x		x	x
Reusing content	x	x		x	x
CMS	x	x		x	x
Proprietary / open source	x	x	x	x	x

Markdown

Example

Heading

=====

Alternative heading

Sub-heading

Block of text with *_italic_*,
****bold**** and ``monospace``
formatting. This is a
[link](http://example.com).

1. numbered list
 - * bulleted list
 - * another bulleted list
2. another list item

![Image](some-picture.png
"picture")

- *"Markdown is a text-to-HTML conversion tool for web writers. Markdown allows you to write using an easy-to-read, easy-to-write plain text format, then convert it to structurally valid XHTML (or HTML)."*
John Gruber
- Simplicity: plain text with Markdown syntax but very basic options (e.g. support for tables, modularity)
- Created in 2004, it became the first popular lightweight markup language, especially for blogging, online forums and hosting platforms like GitHub
- Many markup flavors, limited success in standardization
- Learn more:
<https://daringfireball.net/projects/markdown/>

reStructuredText (RST) + Sphinx

Example

```
Chapter 1 Title
=====

.. toctree::
   :maxdepth: 2

   other_documents/included_document

This is a paragraph.

Section 1.1 Title
-----

    This is an indented block of text.

::

    This is a block of preformatted text.

* a bullet point
  - a sub-list item

* another bullet point

.. image:: image_folder/included_image.png
```

- reStructuredText (RST) is a lightweight markup language popular in Python-based communities
- Sphinx generates documentation from RST sources using the Docutils Python tools
- Sphinx supports cross-references and including documents in a hierarchy with automated linking, or generating indices
- Supports multiple output formats including HTML, PDF, man pages, etc.
- Can build whole documentation sites, including translations and custom themes
- Supported on GitHub and used by documentation hosting platforms such as <https://readthedocs.org/>

Prerequisites for the next class (14th March)

- Install AsciiDoctor on your computer
(https://rh-writers.github.io/technical-writing-course-brno/#_introduction_to_asciidocctor)
- Sign up for a GitHub account.
- Set up Git and SSH keys on your computer if you work from the command line
(https://rh-writers.github.io/technical-writing-course-brno/#_git_installation)
- .
- Alternatively, use the GitHub web interface.