

# **Creating specifications with makespec**

Jakob Voß

2013-07-13 16:16:25 +0200

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Usage</b>	<b>4</b>
2.1	Setup . . . . .	4
2.2	Workflow . . . . .	4
2.3	Configuration . . . . .	5
2.4	Document variables . . . . .	6
2.5	Plugins . . . . .	6
2.5.1	Include plugin . . . . .	6
2.6	Requirements . . . . .	6
<b>3</b>	<b>Examples</b>	<b>7</b>
<b>4</b>	<b>Revision history</b>	<b>8</b>

# 1 Introduction

{DOCUMENT\_ABSTRACT}

## 2 Usage

### 2.1 Setup

Best practice using makedoc is to host your specification in a git repository, optionally published at [GitHub](#). Your repository should contain at least:

- The specification as Pandoc Markdown source file (e.g. `myspec.md`)
- A Makefile with basic settings and **configuration**, such as:

```
NAME=myspec
GITHUB=https://github.com/myaccount/myspec/
```

```
include makespec/Makefile
```

- A subdirectory `makespec` containing `makespec`. Best practice is inclusion as a git submodule. To do so in a new git repository:

```
git submodule add https://github.com/jakobib/makespec.git
```

To clone an existing specification that uses makespace this way:

```
git clone --recursive http://github.com/myaccount/myspec.git
```

If you publish the specification as GitHub pages, you should not forget to also clone the `gh-pages` branch:

```
git checkout -b gh-pages origin/gh-pages && git checkout master
```

### 2.2 Workflow

Make sure you have the most recent version

```
git pull origin master gh-pages
```

Edit the specification source file in Markdown syntax, for instance with vim editor

```
vim myspec.md
```

Create a HTML version

```
make html
```

Commit to your repository

```
git add Makefile myspec.md
git commit -m "improved my fancy specification"
```

Create a gh-pages branch with the HTML version

```
make website
```

Publish

```
git push origin master gh-pages
```

## 2.3 Configuration

**NAME** Required short name. Should not contain spaces and similar nasty characters.

**SOURCE** Source file in Pandoc Markdown syntax, set to NAME.md by default.

**GITHUB** Github repository to link to in revision history.

**REVISIONS** Number of revisions to show in the revision history (GIT-CHANGES).

**FORMATS** If your specification contains a RDF ontology written in Turtle syntax, set FORMATS=ttml owl. More formats may be supported in the future.

**TITLE** Title, unless already specified in the source file.

**AUTHOR** List of authors, unless already specified in the source file.

**DATE** Fixed date of publication. Set to a timestamp of the latest commit by default.

**ABSTRACT** A short abstract (in Markdown syntax) which is used as template variable DOCUMENT\_ABSTRACT.

**ABSTRACT\_FROM** A file to read abstract from if no ABSTRACT was defined. Set to abstract.md by default.

## 2.4 Document variables

The following variables, if enclosed in curly brackets (such as `{THIS_VARIABLE}`) are automatically replaced before conversion from Markdown syntax.

**GIT\_REVISION\_DATE** timestamp of the latest commit.

**GIT\_REVISION\_HASH** Short revision hash of the latest commit.

**GIT\_CHANGES** Revision history. Length can be set with configuration variable `REVISIONS`.

**GIT\_ATOM\_FEED** URL of an Atom feed with revisions at GitHub.

**DOCUMENT\_ABSTRACT** Abstract, if defined with configuration variable `ABSTRACT` or `ABSTRACT_FROM`.

## 2.5 Plugins

Makespec makes use of several plugins of Pandoc Markdown syntax.

### 2.5.1 Include plugin

One can include markdown files with the following syntax:

```
`filename.md`{.include}
```

The inclusion statement should be placed on the begin of a single line. By now, inclusion is implemented on syntax level. A later version of this plugin will act on the abstract syntax tree.

## 2.6 Requirements

- GNU Make
- Perl `>= 5.14.1`
- [Pandoc](#) version `>= 1.9`
- [Rapper](#) from Raptor RDF library (only if writing an RDF ontology)

## 3 Examples

The following specifications make use of makespec:

- [Simple Service Status Ontology \(SSSO\)](#)
- [Document Service Ontology \(DSO\)](#)
- [Document Availability Information API \(DAIA\)](#)
- [Patrons Account Information API \(PAIA\)](#)
- ...

Last but not least, the documentation of makespec is also created with makespec. [This document](#) was last modified at 2013-07-13 16:16:25 +0200 with hash [817fd18](#).

## 4 Revision history

Also available as [Atom feed](#).

- [2013-07-13 16:16:25 +0200](#): include plugin closes issue #11
- [2013-07-11 11:08:27 +0200](#): refactoring
- [2013-07-11 10:26:26 +0200](#): major refactoring
- [2013-05-21 18:24:08 +0200](#): changed document variable syntax to {CURLY\_BRACKETS}
- [2013-05-21 18:17:34 +0200](#): added ABSTRACT (issue #9)