
django-modeltranslation-wagtail Documentation

Release 0.4.1

Benjamin Balder Bach

Dec 25, 2021

Contents

1	Features	3
2	How to use	5
3	When to use this	7
4	Background	9
5	Using django-modeltranslation or wagtail-modeltranslation?	11
6	Contents	13
6.1	Installation	13
6.2	Usage	13
6.3	modeltranslation_wagtail	14
6.4	Contributing	15
6.5	History	16
7	Feedback	19
	Python Module Index	21
	Index	23

Keep-it-simple glue between django-modeltranslation and Wagtail.

CHAPTER 1

Features

- Field-based translation for Wagtail using `django-modeltranslation`
- **TODO** Makes translated fields easily accessible to the Wagtail admin.
- Sensible to the currently activated language.
- The `Page` model isn't modified (no migrations added to Wagtail), translations of `title`, `seo_title` and `search_description` are local to the translated models.

CHAPTER 2

How to use

You create Wagtail models and activate them with a `<yourapp>.translation` module, just like you would with any other modeltranslation-based model.

In Wagtail, it's the active language that decides what language you are editing in your translated model.

For instance, navigating to `/en/wagtail/pages/add/<yourapp>/<yourmodel>/<ptr_id>/` will create a new page in English.

Drawback: There is currently no UI for the translation workflow. This means that users have to *know* that they need to switch `/en/` with `/fr/` in the URL path after creating an English entry to edit it in French.

When to use this

Actually, you might not want to use this! Consider carefully to use [wagtail-modeltranslation](#) as it has substantially improved by not hard-copying `django-modeltranslation` anymore.

If you need to have 1:1 translation for Pages or Snippets in Wagtail, you could use `django-modeltranslation`. However, `wagtail-modeltranslation` is a complex project that patches a lot of external code and therefore often breaks with the latest versions of Wagtail.

If you only need freeform translation (when pages in the page-tree aren't translated 1:1 but translations exist independently from one another), look no further than either Wagtail's built-in [simple_translation](#) or [wagtail-localize](#).

CHAPTER 4

Background

Originally, I was using [wagtail-modeltranslation](#). The project has been sparsely maintained, but more seriously, it was monkey-patching Wagtail and using a hard-copy of the modeltranslation codebase instead of a dependency reference to `django-modeltranslation`. The latter has now been restored, which as lowered the motivation for this project.

A few years later, I returned to use `wagtail-modeltranslation` but once again found myself spending way too much time understanding non-merged PRs etc. To be fair, the project has been maintained in the meantime, but it was lacking behind and I couldn't figure out what to do to use it with the latest version of Wagtail.

This project introduces a bit of naming hell. But remember it like this: *django-modeltranslation-wagtail* has `wagtail` at the end because it depends on *django-modeltranslation*, which depends on *django*.

Using django-modeltranslation or wagtail-modeltranslation?

Switching is easy!

Since this project is directly based on django-modeltranslation, the creation of fields in the database and django migrations remains the same. In case you are switching, just revisit your `translation.py` files and change the imports to point to `modeltranslation_wagtail`.

6.1 Installation

At the command line either via `easy_install` or `pip`:

```
$ easy_install modeltranslation_wagtail
$ pip install modeltranslation_wagtail
```

Or, if you have `virtualenvwrapper` installed:

```
$ mkvirtualenv modeltranslation_wagtail
$ pip install modeltranslation_wagtail
```

6.2 Usage

In your Django settings, you need to add the following

```
INSTALLED_APPS = [
    # NOTE THE ORDER - IT MATTERS!
    "modeltranslation_wagtail",
    "modeltranslation",
]
```

To use `django-modeltranslation-wagtail` in `myapp.translation`

```
from modeltranslation.translator import register
from modeltranslation_wagtail.translator import TranslationOptions

from myapp.models import MyModel

@register(MyModel)
```

(continues on next page)

(continued from previous page)

```
class MyModelTranslation(TranslationOptions):
    fields = (
        'content',
    )
```

6.3 modeltranslation_wagtail

6.3.1 modeltranslation_wagtail package

Subpackages

modeltranslation_wagtail.migrations package

Submodules

modeltranslation_wagtail.migrations.0001_initial module

```
class modeltranslation_wagtail.migrations.0001_initial.Migration(name,
                                                                    app_label)
    Bases: django.db.migrations.migration.Migration
    dependencies = []
    initial = True
    operations = []
```

Module contents

Submodules

modeltranslation_wagtail.apps module

```
class modeltranslation_wagtail.apps.TransAppConfig(app_name, app_module)
    Bases: django.apps.config.AppConfig
    name = 'modeltranslation_wagtail'
    ready()
        Lifted from the great minds in the infoportugal/wagtail-modeltranslation community.
    verbose_name = 'Translated Wagtail Models'
```

modeltranslation_wagtail.translator module

```
class modeltranslation_wagtail.translator.TranslationOptions(*args, **kwargs)
    Bases: modeltranslation.translator.TranslationOptions
    fields = ()
```

Module contents

6.4 Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

You can contribute in many ways:

6.4.1 Types of Contributions

Report Bugs

Report bugs at https://github.com/benjaoming/modeltranslation_wagtail/issues.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

Write Documentation

django-modeltranslation-wagtail could always use more documentation, whether as part of the official django-modeltranslation-wagtail docs, in docstrings, or even on the web in blog posts, articles, and such.

Submit Feedback

The best way to send feedback is to file an issue at https://github.com/benjaoming/modeltranslation_wagtail/issues.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

6.4.2 Get Started!

Ready to contribute? Here's how to set up *modeltranslation_wagtail* for local development.

1. Fork the *modeltranslation_wagtail* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/modeltranslation_wagtail.git
```

3. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

4. When you're done making changes, check that your changes pass style and unit tests, including testing other Python versions with tox:

```
$ tox
```

To get tox, just pip install it.

5. Commit your changes and push your branch to GitHub:

```
$ git add .  
$ git commit -m "Your detailed description of your changes."  
$ git push origin name-of-your-bugfix-or-feature
```

6. Submit a pull request through the GitHub website.

6.4.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.6, 2.7, and 3.3, and for PyPy. Check https://travis-ci.org/benjaoming/modeltranslation_wagtail under pull requests for active pull requests or run the `tox` command and make sure that the tests pass for all supported Python versions.

6.4.4 Tips

To run a subset of tests:

```
$ py.test test/test_modeltranslation_wagtail.py
```

6.5 History

6.5.1 0.4.1 (2021-12-24)

- Support for Wagtail StreamField

- Very sparse documentation (please do help to expand)

6.5.2 0.4 (2021-12-24)

- New tests
- Updates for Django 3 and Wagtail 2+

6.5.3 0.2.1 (2018-02-20)

- Changes to release meta data

6.5.4 0.2.0 (2017-12-17)

- Do not translate URL slugs for now, it breaks and needs a more patching in order to work.

6.5.5 0.1.0 (2017-01-20)

- First release on PyPI.

CHAPTER 7

Feedback

If you have any suggestions or questions about **django-modeltranslation-wagtail** feel free to email me at benjamin@overtag.dk.

If you encounter any errors or problems with **django-modeltranslation-wagtail**, please let me know! Open an Issue at the GitHub http://github.com/benjaoming/modeltranslation_wagtail main repository.

m

- `modeltranslation_wagtail`, [15](#)
- `modeltranslation_wagtail.apps`, [14](#)
- `modeltranslation_wagtail.migrations`, [14](#)
- `modeltranslation_wagtail.migrations.0001_initial`,
[14](#)
- `modeltranslation_wagtail.translator`, [14](#)

D

dependencies (modeltranslation_wagtail.migrations.0001_initial.Migration attribute), 14

F

fields (modeltranslation_wagtail.translator.TranslationOptions attribute), 14

I

initial (modeltranslation_wagtail.migrations.0001_initial.Migration attribute), 14

M

Migration (class in modeltranslation_wagtail.migrations.0001_initial), 14

modeltranslation_wagtail (module), 15

modeltranslation_wagtail.apps (module), 14

modeltranslation_wagtail.migrations (module), 14

modeltranslation_wagtail.migrations.0001_initial (module), 14

modeltranslation_wagtail.translator (module), 14

N

name (modeltranslation_wagtail.apps.TransAppConfig attribute), 14

O

operations (modeltranslation_wagtail.migrations.0001_initial.Migration attribute), 14

R

ready() (modeltranslation_wagtail.apps.TransAppConfig method), 14

T

TransAppConfig (class in modeltranslation_wagtail.apps), 14

TranslationOptions (class in modeltranslation_wagtail.translator), 14

V

verbose_name (modeltranslation_wagtail.apps.TransAppConfig attribute), 14