wagtail-personalisation Documentation

Release 0.12.1

Lab Digital BV

Sep 26, 2018

Contents

Wagxperience is a fully-featured personalisation module for Wagtail. It enables editors to create customised pages - or parts of pages - based on segments whose rules are configured directly in the admin interface.

- **Get up and running**
  - *Getting Started*
- **For developers**
  - *Usage Guide*
- **For editors & marketeers**
  - *Editor Guide*

CHAPTER 1

Index

1.1 Getting Started

1.1.1 Installing Wagxperience

Wagtail Personalisation requires Wagtail 2.0 or 2.1 and Django 1.11 or 2.0.

To install the package with pip:

```
pip install wagtail-personalisation
```

Next, include the `wagtail_personalisation`, `wagtail.contrib.modeladmin` and `wagtailfontawesome` apps in your project's `INSTALLED_APPS`:

```
INSTALLED_APPS = [
    # ...
    'wagtail.contrib.modeladmin',
    'wagtail_personalisation',
    'wagtailfontawesome',
    # ...
]
```

Make sure that `django.contrib.sessions.middleware.SessionMiddleware` has been added in first, this is a prerequisite for this project.

```
MIDDLEWARE = [
    'django.contrib.sessions.middleware.SessionMiddleware',
    # ...
]
```

1.1.2 Using the sandbox

To experiment with the package you can use the sandbox provided in the repository. It includes a couple of segments with rules, a personalisable page with a variant and a personalisable StreamField block.

To install this you will need to create and activate a virtualenv and then run `make sandbox`. This will start a fresh Wagtail install, with the personalisation module enabled, on http://localhost:8000 and http://localhost:8000/cms/. The superuser credentials are `superuser@example.com` with the password `testing`.

1.2 Usage Guide

1.2.1 Implementation

Extending a page to be personalisable

Wagxperience offers a `PersonalisablePage` base class to extend from. This is a standard `Page` class with personalisation options added.

Creating a new personalisable page

Import and extend the `personalisation.models.PersonalisablePage` class to create a personalisable page.

A very simple example for a personalisable homepage:

```
from wagtail.wagtailcore.models import Page
from wagtail_personalisation.models import PersonalisablePageMixin

class HomePage(PersonalisablePageMixin, Page):
    pass
```

All you need is the `PersonalisablePageMixin` mixin and a Wagtail `Page` class of your liking.

Adding personalisable StreamField blocks

Taking things a step further, you can also add personalisable StreamField blocks to your page models. Below is the full Homepage model used in the sandbox.

```
from __future__ import absolute_import, unicode_literals

from wagtail.admin.edit_handlers import RichTextFieldPanel, StreamFieldPanel
from wagtail.core import blocks
from wagtail.core.fields import RichTextField, StreamField
from wagtail.core.models import Page

from wagtail_personalisation.models import PersonalisablePageMixin
from wagtail_personalisation.blocks import PersonalisedStructBlock


class HomePage(PersonalisablePageMixin, Page):
    intro = RichTextField()
    body = StreamField([
```

(continues on next page)

(continued from previous page)

```
        ('personalisable_paragraph', PersonalisedStructBlock([
            ('paragraph', blocks.RichTextBlock()),
        ], icon='pilcrow'))
    ])

    content_panels = Page.content_panels + [
        RichTextFieldPanel('intro'),
        StreamFieldPanel('body'),
    ]
```

Using template blocks for personalisation

Please note that using the personalisable template tag is not the recommended method for adding personalisation to your content, as it is largely decoupled from the administration interface. Use responsibly.

You can add a template block that only shows its contents to users of a specific segment. This is done using the "segment" block.

When editing templates make sure to load the `wagtail_personalisation_tags` tags library in the template:

```
{% load wagtail_personalisation_tags %}
```

After that you can add a template block with the name of the segment you want the content to show up for:

```
{% segment name="My Segment" %}
    <p>Only users within "My Segment" see this!</p>
{% endsegment %}
```

The template block currently only supports one segment at a time. If you want to target multiple segments you will have to make multiple blocks with the same content.

1.2.2 Creating custom rules

Rules consist of two important elements, the model fields and the `test_user` function. They should inherit the `AbstractBaseRule` class from `wagtail_personalisation.rules`.

A simple example of a rule could look something like this:

```
class UserIsLoggedInRule(AbstractBaseRule):
    """User is logged in rule to segment users based on their authentication
    status.

    Matches when the user is authenticated.

    """
    icon = 'fa-user'

    is_logged_in = models.BooleanField(default=False)

    panels = [
        FieldPanel('is_logged_in'),
    ]

    class Meta:
```

(continues on next page)

(continued from previous page)

```
        verbose_name = _('Logged in Rule')

    def test_user(self, request=None):
        return request.user.is_authenticated() == self.is_logged_in

    def description(self):
        return {
            'title': _('These visitors are'),
            'value': _('Logged in') if self.is_logged_in else _('Not logged in'),
        }
```

As you can see, the only real requirement is the `test_user` function that will either return `True` or `False` based on the model fields and optionally the request object.

That's it!

1.3 Editor Guide

The editor guide is meant for content editors and marketers using Wagxperience to offer a personalised experience to their visitors.

1.3.1 Introduction

Wagxperience is an open source module developed by Lab Digital for the Wagtail content management system. It allows editors and marketeers to create personalised experiences by harnessing the power of segmentation and rules.

In this guide, we'll take you step by step through the process of offering your visitors a tailor made online experience. The subjects covered are:

- Using the segments dashboard
- Defining a new segment
- Setting up rules used to match visitors to a segment
- Personalize a page by creating a variant
- Using the StreamField to personalize content blocks
- And even more helpful stuff...

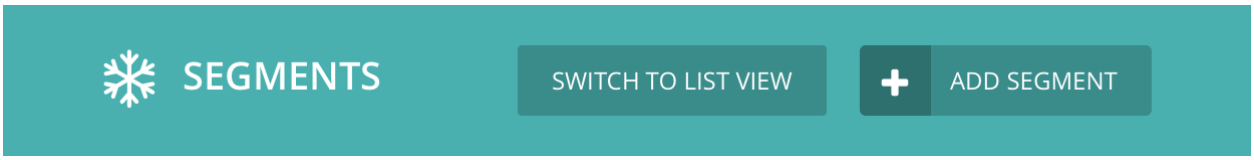
So without further ado, let's get started!

1.3.2 The segments dashboard

Wagxperience comes with two different views for it's segment dashboard. A "list view" and a "dashboard view". Where the dashboard view attempts to show all relevant information and statistics in a visually pleasing manner, the list view is more fitted for sites using large amounts of segments, as it may be considered more clear in these cases.

Switching between views

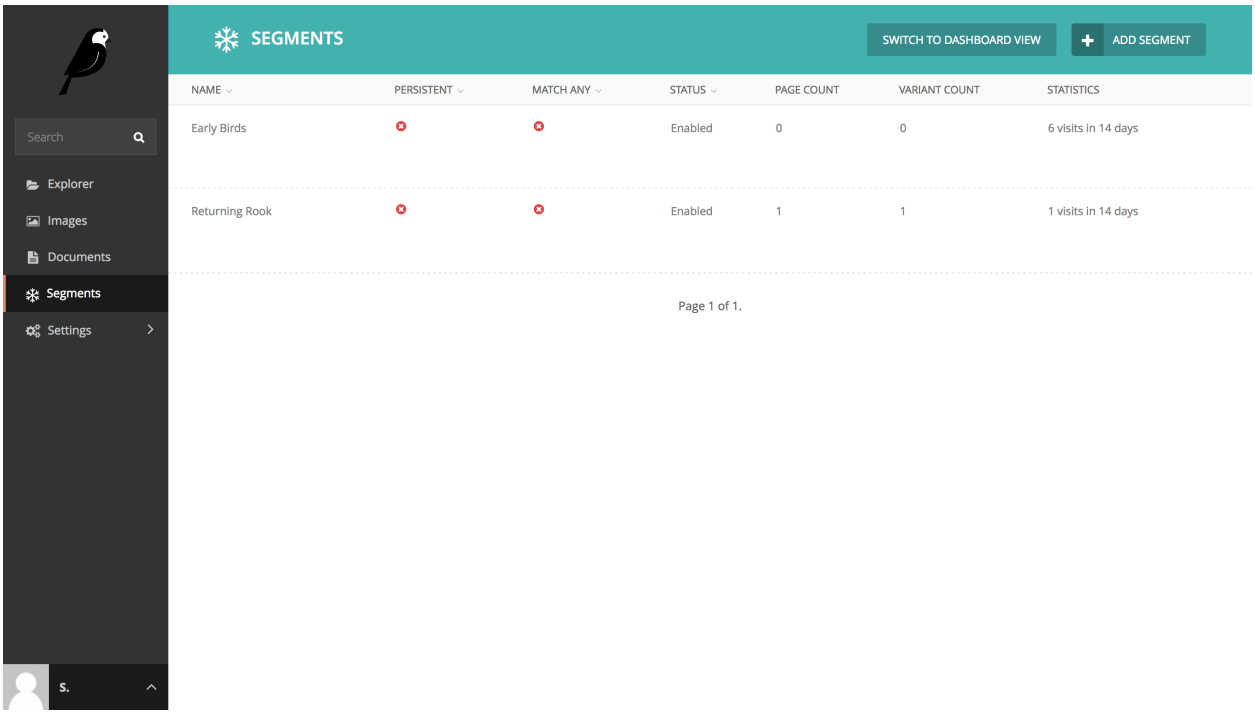
By default, Wagxperience's "dashboard view" is active on the segment dashboard. If you would like to switch between the dashboard view and list view, open the segment dashboard and click the "Switch view" button in the green header at the top of the page.



Using the list view

Advantages of using the list view:

- Uses the familiar table view that is used on many other parts of the Wagtail administration interface.
- Offers a better overview for large amounts of segments.
- Allows for reordering based on fields, such as name or status.



Definitions

Name The name of your segment.

Persistent If this is disabled (default), whenever a visitor requests a page, the rules of this segment are reevaluated. This means that when the rules no longer match, the visitor is no longer a part of this segment. However, if persistence is enabled, this segment will "stick" with the visitor, even when the rules no longer apply.

Match any If this is disabled (default) all rules of this segment must match a visitor before the visitor is appointed to this segment. If this is enabled, only 1 rule has to match before the visitor is appointed.

Status Indicates whether this segment is active (default) or inactive. If it has been set to 'inactive', visitors will not be appointed to this segment and no personalised content for this segment will be shown to visitors.

Page count The amount of pages that have variants using this segment.

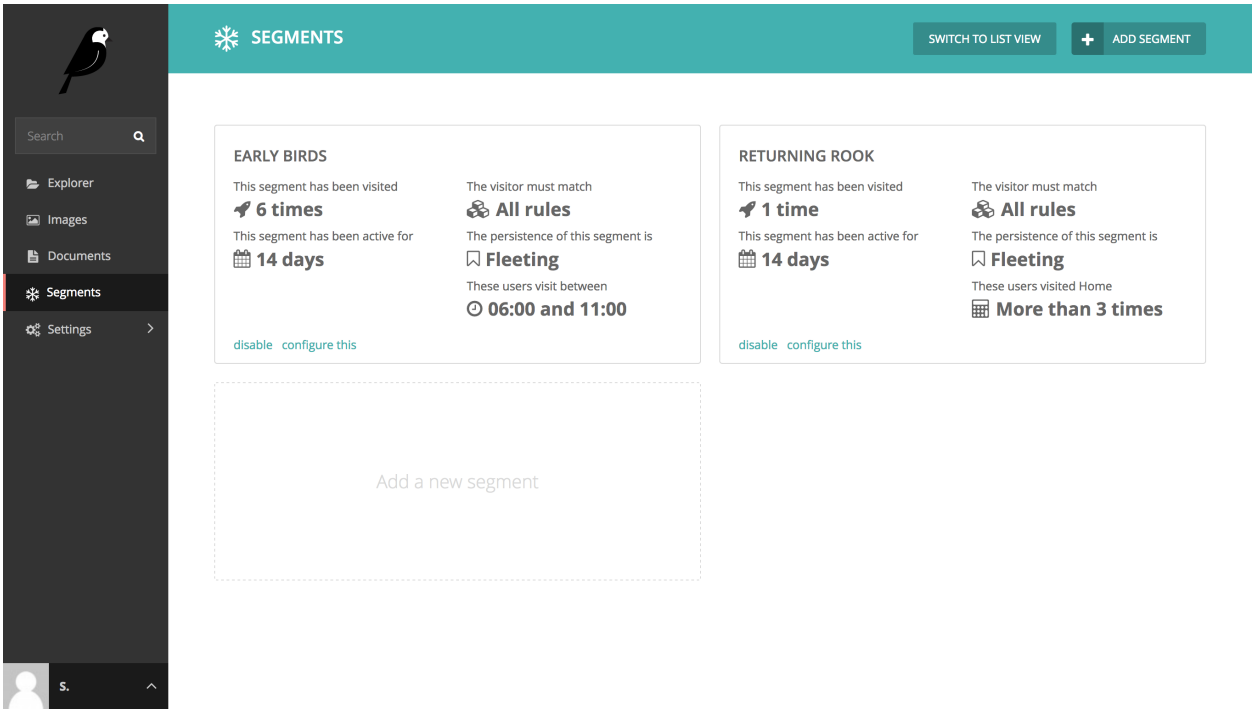
Variant count The total amount of variants for this segment. Does not yet apply, as this will always match the amount of pages in the "Page count".

Statistics Shows the amount of visits of this segment and the days it has been enabled. If the segment is disabled and then re-enabled, these statistics will reset.

Using the dashboard view

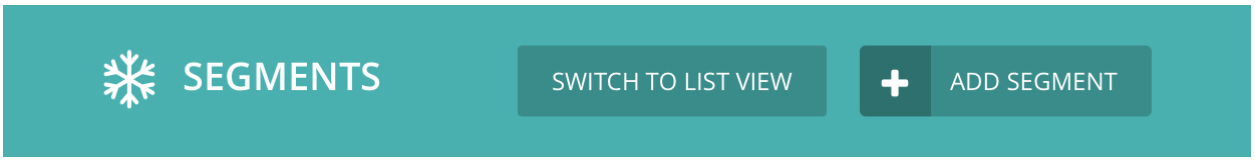
Advantages of using the dashboard view:

- Offers a more pleasing visual representation of segments.
- Focused on giving insights about your segments at a glance.
- Shows the actual rules of a segment.
- Gives more wordy explanation about the information shown.



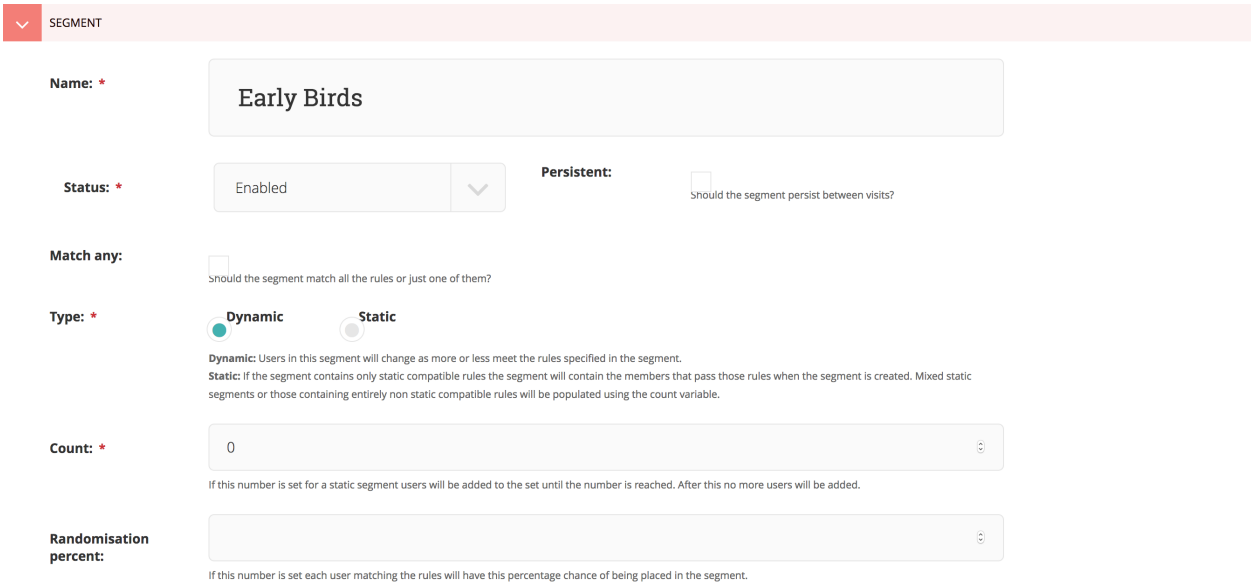
1.3.3 Creating a segment

To create a segment, go to the "Segments dashboard" and click "Add segment". You can find the segments dashboard in the administration menu on the left of the page.



On this page you will be presented with two forms. One with specific information about your segment, the other allowing you to choose and configure your rules.

Set segment specific options



1. Enter a name for your segment

   Choose something meaningful like "Newsletter campaign visitors". This will ensure you'll have a general idea which visitors are in this segment in other parts of the administration interface.

2. Select the status of the segment *Optional*

   You will generally keep this one **enabled**. If for some reason you want to disable the segment, you can change this to **disabled**.

3. Set the segment persistence. *Optional*

   When persistence is **enabled**, your segment will stick to the visitor once applied, even if the rules no longer match the next visit.

4. Select whether to match any or all defined rules. *Optional*

   Match any will result in a segment that is applied as soon as one of your rules matches the visitor. When **match all** is selected, all rules must match before the segment is applied.

5. The segment type *Required*

   Dynamic: Users in this segment will change as more or less meet the rules specified in the segment.

   Static: If the segment contains only static compatible rules the segment will contain the members that pass those rules when the segment is created. Mixed static segments or those containing entirely non static compatible rules will be populated using the count variable.

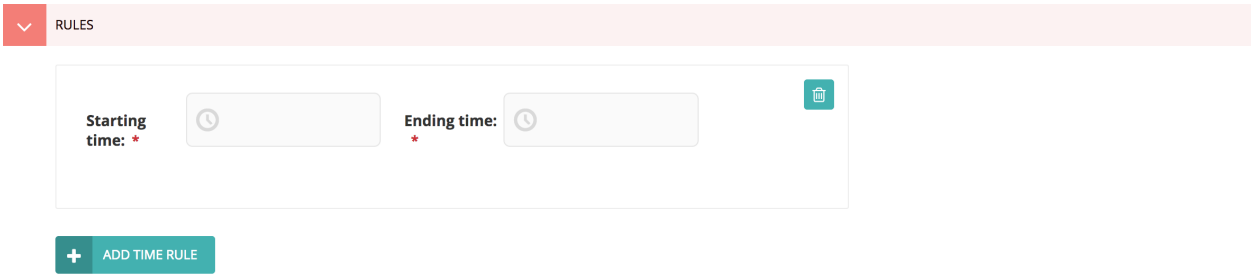
6. The segment count *Optional*

   If this number is set for a static segment users will be added to the set until the number is reached. After this no more users will be added.

7. Randomisation percentage *Optional*

   If this number is set each user matching the rules will have this percentage chance of being placed in the segment.

Defining rules



5. Choose the rules you want to use.

   Wagxperience comes with a basic set of *Included rules* that allow you to get started quickly. The rules you define will be evaluated once a visitor makes a request to your application.

   The rules that come with Wagxperience are as follows:

Included rules

Wagxperience comes with a base set of rules that allow you to start segmenting your visitors quickly.

Time rule

The time rule allows you to segment visitors based on the time of their visit. Define a time frame in which visitors are matched to this segment.

Option	Description
Start time	The start time of your time frame.
End time	The end time of your time frame.

`wagtail_personalisation.rules.TimeRule`

Day rule

The day rule allows you to segment visitors based on the day of their visit. Select one or multiple days on which you would like your segment to be applied.

Option	Description
Monday	Matches when the visitors visits on a monday.
Tuesday	Matches when the visitors visits on a tuesday.
Wednesday	Matches when the visitors visits on a wednesday.
Thursday	Matches when the visitors visits on a thursday.
Friday	Matches when the visitors visits on a friday.
Saturday	Matches when the visitors visits on a saturday.
Sunday	Matches when the visitors visits on a sunday.

`wagtail_personalisation.rules.DayRule`

Referral rule

The referral rule allows you to match visitors based on the website they were referred from. For example:

```
example\.com|secondexample\.com|.*subdomain\.com
```

Option	Description
Regex string	The regex string to match the referral header to.

`wagtail_personalisation.rules.ReferralRule`

Visit count rule

The visit count rule allows you to segment a visitor based on the amount of visits per page. Use the operator to to set a maximum, minimum or equal amount of visits.

Option	Description
Page	The page on which visits will be counted.
Count	The amount of visits to match.
Operator	Whether to match for more than, less than or equal to the specified visit count.

`wagtail_personalisation.rules.VisitCountRule`

Query rule

The query rule allows you to match a visitor based on the query included in the url. It let's you define both the parameter and the value. It will look something like this:

```
example.com/?campaign=ourbestoffer
```

Option	Description
Parameter	The first part of the query ('campaign').
Value	The second part of the query ('ourbestoffer').

`wagtail_personalisation.rules.QueryRule`

Device rule

The device rule allows you to match visitors by the type of device they are using. You can select any combination you want.

Option	Description
Mobile phone	Matches when the visitor uses a mobile phone.
Tablet	Matches when the visitor uses a tablet.
Desktop	Matches when the visitor uses a desktop.

`wagtail_personalisation.rules.DeviceRule`

User is logged in rule

The user is logged in rule allows you to match visitors that are authenticated and logged in to your app.

Option	Description
Is logged in	Whether the user is logged in or logged out.

`wagtail_personalisation.rules.UserIsLoggedInRule`

Origin country rule

The origin country rule allows you to match visitors based on the origin country of their request. This rule requires to have set up a way to detect countries beforehand.

Option	Description
Country	What country user's request comes from.

You must have one of the following configurations set up in order to make it work.

- Cloudflare IP Geolocation - `cf-ipcountry` HTTP header set with a value of the alpha-2 country format.
- CloudFront Geo-Targeting - `cloudfront-viewer-country` header set with a value of the alpha-2 country format.
- The last fallback is to use GeoIP2 module that is included with Django. This requires setting up an IP database beforehand, see the Django's GeoIP2 instructions for more information. It will use IP of the request, using HTTP header the `x-forwarded-for` HTTP header and `REMOTE_ADDR` server value as a fallback. If you want to use a custom logic when obtaining IP address, please set the `WAGTAIL_PERSONALISATION_IP_FUNCTION` setting to the function that takes a request as an argument, e.g.

```
# settings.py

WAGTAIL_PERSONALISATION_IP_FUNCTION = 'yourproject.utils.get_client_ip
↪'


# yourproject/utils.py

def get_client_ip(request):
    return request['HTTP_CF_CONNECTING_IP']
```

`wagtail_personalisation.rules.OriginCountryRule`

Click "save" to store your segment. It will be enabled by default, unless otherwise defined.

1.3.4 Creating personalised content

Once you've created a segment you can start serving personalised content to your visitors. To do this, you can choose one of three methods.

1. Create a page variant for a segment.

2. Use StreamField blocks visible for a segment only.
3. Use a template block visible for a segment only.

Method 1: Create a page variant

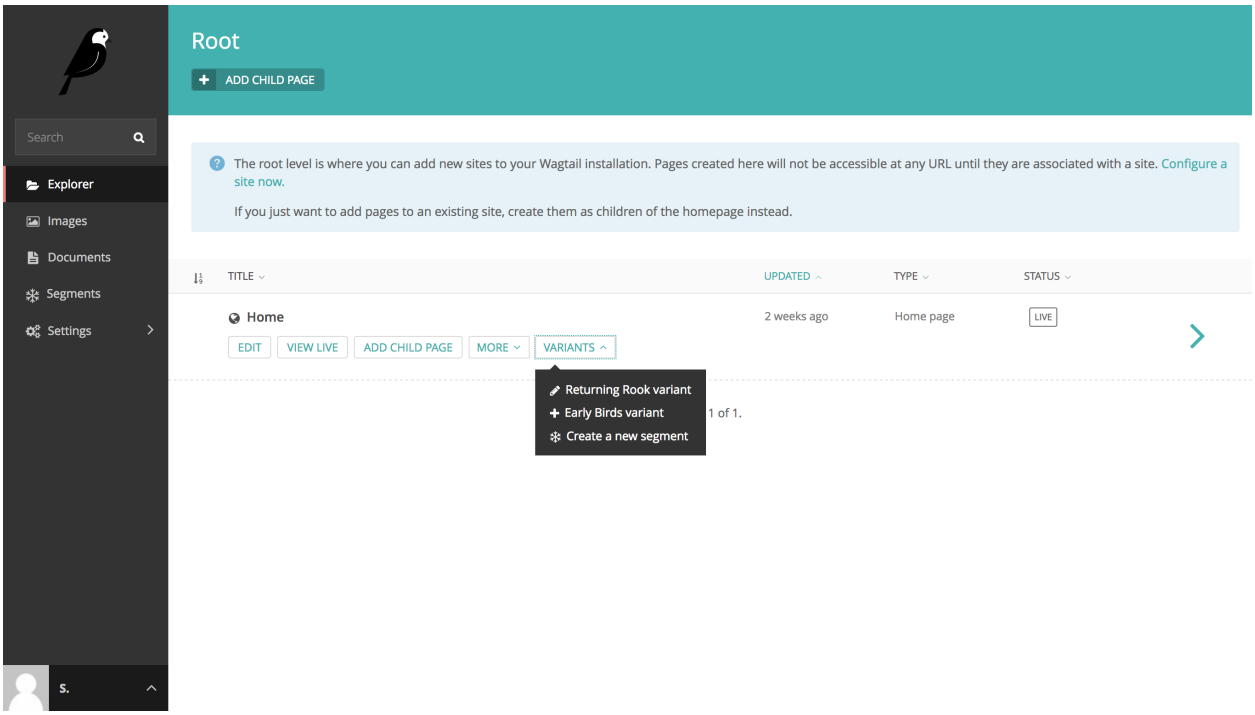
Why you would want to use this method

- It has absolutely no restrictions, you can change anything you want.
- That's pretty much it.

Why you would want to use a different method

- You are editing a page that changes often. You would probably rather not change the variation(s) every time the original page changes.

To create a variant of a page for a specific Segment (which you can change to your liking after creating it), simply go to the Explorer section and find the page you would like to personalize.



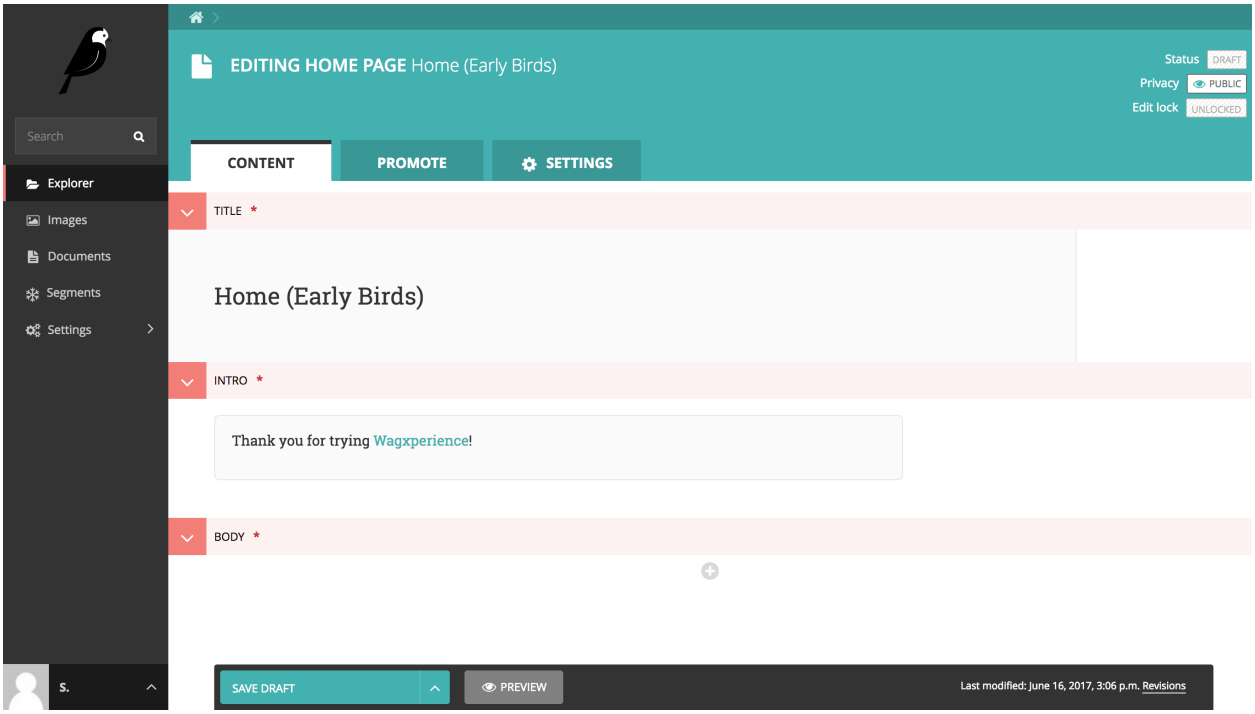
When you hover over a page, you'll notice a "Variants" dropdown button appears. Click the button and select the segment you would like to create personalised content for.

Once you've selected the segment, a copy of the original page will be created with a title that includes the segment. Don't worry, your visitors won't be able to see this title. It's only there for your reference.

You can change everything on this page you would like. Visitors that are appointed to your segment will automatically see the new variant you've created for them when attempting to visit the original page.

Method 2: Use a StreamField block

Preparing a page and it's StreamField blocks for this method is described in the Usage guide for developers. Please refer to *Adding personalisable StreamField blocks* for more information.



Why you would want to use this method

- Allows you to create personalised content in the original page (without creating a variant).
- Create multiple StreamField blocks for different segments inline.

Why you would want to use a different method

- You need someone tech savvy to change the back-end implementation.

To create personalised StreamField blocks, first select the page you wan't to create the content for. Note that the personalisable StreamField blocks must be activated on the page by your developer.

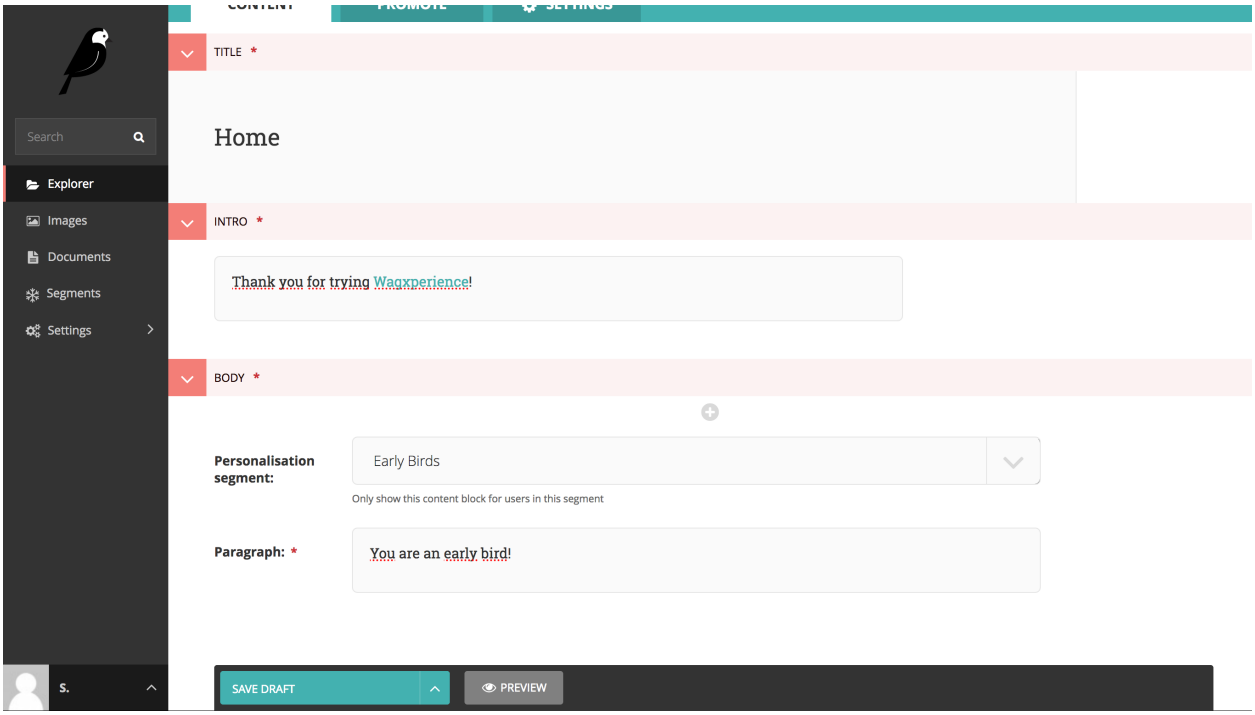
Scroll down to the block containing the StreamField and add a personalisable block. The first input field in the block is a dropdown allowing you to select the segment this StreamField block is ment for.

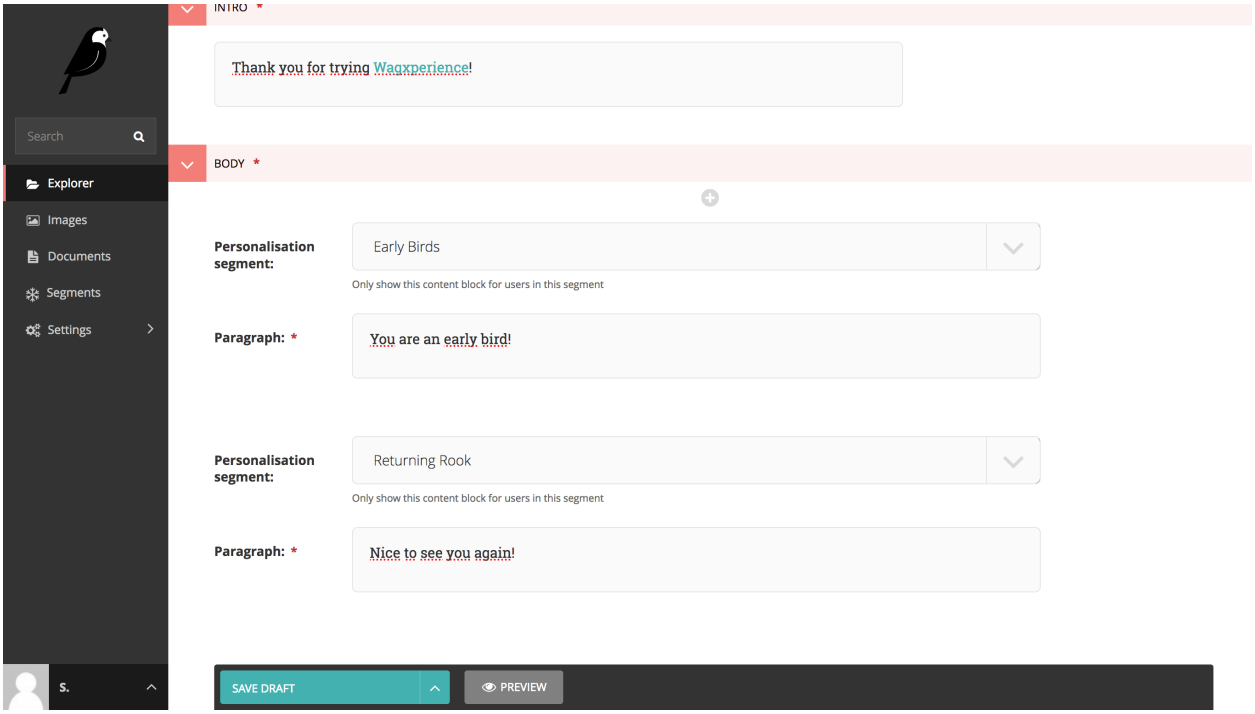
If you want, you can even add multiple blocks and change the segment to show different content between segments!

Once saved, the page will selectively show StreamField blocks based on the visitor's segment.

Method 3: Use a template block

Setting up content in this manner is described in the Usage guide for developers. Please refer to *Using template blocks for personalisation* for more information.


TITLE *
Home
INTRO *
Thank you for trying Wagxperience!
BODY *
Personalisation segment:
Early Birds
Only show this content block for users in this segment
Paragraph: *
You are an early bird!
Search
Explorer
Images
Documents
Segments
Settings
S.
SAVE DRAFT
PREVIEW


INTRO *
Thank you for trying Wagxperience!
BODY *
Personalisation segment:
Early Birds
Only show this content block for users in this segment
Paragraph: *
You are an early bird!
Personalisation segment:
Returning Rook
Only show this content block for users in this segment
Paragraph: *
Nice to see you again!
Search
Explorer
Images
Documents
Segments
Settings
S.
SAVE DRAFT
PREVIEW