

Make it your own!

Integrating Twig templates with
your WordPress block theme

This Is Me

```
{  
  "name": "Joni Halabi",  
  "role": "Senior Web Developer @ Georgetown",  
  "socials": "@jonihalabi",  
  "subtheme": "music"  
}
```

Today's playlist

1. Come With Me (*on a story about our themes @ GU*)
2. May I Suggest (*some definitions*)
3. Somewhere Different Now (*a very different block theme*)
4. Goodbye Pluto (*and other post-project thoughts*)

Enough chit chat...



Time to get down to business.

1

Come With Me

(on a story about our themes @ GU)

Websites @ GU

- Support almost 400 sites
- Almost all are on WordPress
- 1 parent theme, 1 child theme
- Using Gutenberg since it was in alpha

Our themes

- Our classic theme accidentally turned into a **hybrid** theme when we intro'ed `theme.json`
- We ❤️ block styling customizations
- Our "hmmm..." moment: A post-editor bug with block template selection

Experiment time!

We decided to convert our classic theme into a block theme.

2

May I Suggest

(some definitions of terms we will use today)

Theme

- A folder containing front-end markup, styles, and functionality
- Includes **templates**, a file with markup for a particular kind of page

Template

- Define how the entire front-end of a page is marked up, from header to footer.
 - Kinds of pages include posts, evergreen pages, archives, and the home page.
-

The main difference between a *classic theme* and a *block theme* is how the templates in each are coded.

Classic theme

- Templates are PHP files that live in the root of the theme's folder
- Templates are automatically recognized by WP
- Back-end functionality is also in PHP

Block theme

- Declared in WP by the existence of `theme.json`
- Templates are HTML files that live in a `templates/` folder
- Templates are automatically recognized by WP

Template part

- Container block used to display other blocks, e.g. in a site header
- Available in the block inserter for use by content editors

Pattern library

- GU style guide, written in KSS Node
- Loosely based on atomic design... kinda
- Patterns written in Twig and SCSS

GU pattern library

- Follow GU visual identity guidelines
- Adhere to accessibility requirements
- Remain consistent across sites



reedcodes Update style guide with cross-references for pattern relationships (#323) 

881fedb · last week  History

Name	Last commit message	Last commit date
 ..		
 _index.scss	Update style guide with cross-references for pattern relationships (#323)	last week
 eg-heading.twig	Update style guide with cross-references for pattern relationships (#323)	last week
 heading.twig	refactor heading to add Site Title	2 years ago

```
{#- Default to <h2> if no heading level is set. -#}
{%- set heading_level = heading_level | default( 2 ) -%}

<h{{ heading_level }} gu-heading
  {{- "" }} {{ heading_class ? heading_class }}
>

{%- block heading_content -%}
  {{- heading_text -}}
{%- endblock heading_content -%}

</h{{ heading_level }}>
```

```
{% include "@text/heading/heading.twig" with {
    heading_text: "May I Suggest"
} %}
```

P.S. more info @...

- Reed Piernock's talk about building a living style guide and pattern library with KSS-Node from WPCampus 2021

3

Somewhere Different Now

(or, how we built a very different block theme)

Requirements

- Use the functionality of `theme.json`
- Lock down the core design of the theme
- Use markup/patterns from our Pattern Library

> 	assets
> 	blocks
	functions.php PHP
	style.css
> 	templates
	404.html HTML
	archive.html HTML
	front-page.html HTML
	index.html HTML
	page-chaptered.html HTML
	page-landing.html HTML
	page-parent-child.html HTML
	page.html HTML
	single.html HTML
	theme.json

Our block theme

- Uses templates
- Does **not** use template parts, because:
 - Template parts contain other blocks - does not help us hook into our patterns
 - We also do not want them editable content editors

Solution to template parts

Register theme-only blocks that call the pattern library patterns
using Timber

Template files

- All HTML and references to WP blocks
- Block references: `<!-- wp:BLOCK_NAME /-->`

```
<!-- wp:gu/site-header /-->

<main class="gu-page page-default">
  <!-- wp:gu/page-breadcrumbs /-->

  <!-- wp:gu/page-title {
      "heading-class": "page-title" } /-->

  <!-- wp:post-content /-->
</main>

<!-- wp:gu/site-footer /-->
```



blocks



Name

> featured-image

> menu-parent-child

> page-breadcrumbs

< page-title

block.json

README.md
MD

register.php
PHP

> pagination

README.md
MD

> single-post-header

Theme blocks

- Dynamic blocks that call external patterns
- Only registered server-side
- No JS – we do not want an editor UI

```
{  
  "$schema": "https://schemas.wp.org/trunk/block.json",  
  "apiVersion": 2,  
  
  "name": "gu/page-title",  
  "title": "Title",  
  "description": "Display the page title from the pattern  
}
```

```
use Timber\Timber;

class PageTitle {
    public function __construct() {
        add_action( 'init', [ $this, 'register' ] );
    }

    public function register(): void {
        register_block_type( __DIR__, [
            'render_callback' => [ $this, 'render' ]
        ]);
    }

    public function render( array $attrs ): string { ... }
}

new PageTitle;
```

```
public function render( array $attrs ): string {
    global $post;

    // If heading classes are passed in, use them.
    $class = $attrs['heading_class'] ?? '';

    // If heading text is passed in, use that.
    // Otherwise, use the post title.
    $text = $attrs['heading_text'] ?? $post->post_title;

    return Timber::fetch( '@text/heading/heading.twig', [
        'heading_level'  => 1,
        'heading_class'  => $class,
        'heading_text'   => $text,
    ] );
}
```

```
{#- Default to <h2> if no heading level is set. -#}
{%- set heading_level = heading_level | default( 2 ) -%}

<h{{ heading_level }} gu-heading
  {{- "" }} {{ heading_class ? heading_class }}
>

{%- block heading_content -%}
  {{- heading_text -}}
{%- endblock heading_content -%}

</h{{ heading_level }}>
```




+ ⌚ ← → ≡

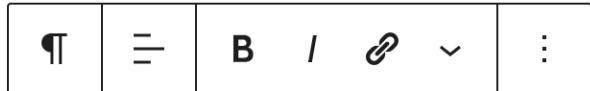
Switch to draft Preview

Update



Options

Somewhere Different Now



I took a long drive by the church and the high dive
Past the riverbank hillside, where we looked at the clouds



[Home](#) / Somewhere Different Now

Somewhere Different Now

I took a long drive by the church and the high dive
Past the riverbank hillside, where we looked at the clouds



Extensions

- Theme-only blocks are just blocks
- Can be used as children of other theme-only blocks
- Combine them to create more complex structures; e.g. site header
 - We have a pattern for that too!

4

Goodbye Pluto

(and other issues and lessons learned from this experiment)

Issue #1: The blocks

- Theme blocks are visible to the post editor
- We don't want them to be seen, edited, or moved by a content editor
- Also caused a block error in the customizer

Solution

Register the block in the JS & hide the block from the block inserter.

```
function hide( settings, name ) {
  if ( ARRAY_OF_BLOCKS.includes( name ) ) {
    return lodash.assign( {}, settings, {
      supports: { inserter: false },
    } );
  }
  return settings;
};

wp.hooks.addFilter(
  'blocks.registerBlockType', 'gu', hide );
```

Issue #2: Child themes

- Core bug: child themes cannot find the parent theme directory if both themes are block themes.
- #57141 (WP_Theme cannot locate a parent block theme)
 - (opened November 2022)

Issue #3: Too complicated!

- Dependencies with other repos
- Large amount of functionality with settings, etc.

Conclusions

- Converting a complex classic theme to a block theme is... complex
- None of this is 100% flushed out for production
- Great experiment in stretching what we can do with WP
- Best for a brand-new theme



The Last Call

- Find these slides @ talks.jhalabi.com/blocktheme
- Find me @ jhalabi.com
 - WPCampus Slack
 - @jonihalabi
 - @thatdevgirl on GH

Playlist

0. *This Is Me*, Eddie From Ohio
1. *Come With Me*, Brother Sun
2. *May I Suggest*, Susan Werner
3. *Somewhere Different Now*, Girlyman
4. *Goodbye Pluto*, Vance Gilbert
5. *The Last Call*, Red Molly

Feedback, please!



Hooray!!

Thank you everyone! ❤

Tech references

- Block theme overview, via the WordPress Editor Handbook
- Block theme template and template parts
- All about theme.json
- Core blocks reference guide
- KSS Node, to write a style guide

