

# An Introduction to Static Site Generators for Drupalists

November 17, 2018

Brian Perry

Lead Front End Developer

**bounteous**

Slides: <http://bit.ly/ssg-drupal>



# BRIAN PERRY

- Lead Front End Dev at Bounteous
- Cranston Boy in the Chicago 'burbs
- Lover of all things components...  
...and Nintendo
- JAMStack curious

**d.o:** brianperry  
**twitter:** bricomedy  
**github:** backlineint

**nintendo:** wabrian  
**brianperryinteractive.com**



# bounteous



# Static Site Generators

An Overview

# THIS IS A STATIC SITE

The screenshot shows a static website with a teal background featuring a pattern of light blue stylized flowers and darker teal abstract shapes. At the top, a navigation bar contains links: Jam Posters, Screen Savers, Backboard Patterns, Browser Icons, and Postcards. The main heading is "SOUND CLIPS" in a large, red, bubbly font. Below the heading is a paragraph: "While they may not have quite the same punch as Dolby SurroundSound, these sound clips will provide you with minutes of fun!". The page displays five sound clip entries, each with a small image, a quote, and a link to download in MP3 or WAV format with a download count:

- "What do we care?" [MP3](#) or [WAV](#) (14K) - Image of a man in a white shirt.
- "You guys are nuts." [MP3](#) or [WAV](#) (13K) - Image of a man's face.
- Monstar growl [MP3](#) or [WAV](#) (32K) - Image of a person in a green costume.
- "We are taking you to our theme park in outer space..." [MP3](#) or [WAV](#) (35K) - Image of a character in a red and white striped suit.
- "You were expecting maybe the Easter Bunny?" [MP3](#) or [WAV](#) (27K) - Image of a person in a white costume.

A vertical sidebar on the left contains navigation links with corresponding icons: STELLAR SOUVENIRS, JAM CENTRAL, BEHIND THE JAM, JUNIOR JAM, THE LINEUP, LUNKER TUNES, PLANET B-BALL, and HUMP STATION. At the bottom center, there is a "NEXT" button with a pink globe icon.

# THIS IS ALSO A STATIC SITE

The image shows a screenshot of the React.js documentation website. At the top, there is a dark navigation bar with the React logo and 'React' on the left, and 'Docs', 'Tutorial', 'Community', and 'Blog' in the center. On the right side of the navigation bar, there is a search icon, 'Search docs', the version 'v16.6.1', and a 'GitHub' link. Below the navigation bar, the main content area has a large heading 'Rendering Elements'. Underneath the heading, there is a paragraph: 'Elements are the smallest building blocks of React apps.' followed by another paragraph: 'An element describes what you want to see on the screen:'. Below this is a code block with the following code: 

```
const element = <h1>Hello, world</h1>;
```

 Under the code block, there is another paragraph: 'Unlike browser DOM elements, React elements are plain objects, and are cheap to create. React DOM takes care of updating the DOM to match the React elements.' Below this is a yellow callout box with the heading 'Note:' and the text: 'One might confuse elements with a more widely known concept of "components". We will introduce components in the [next section](#). Elements are what components are "made of", and we encourage you to read this section before jumping ahead.' On the right side of the page, there is a sidebar with a table of contents. The sections are: 'INSTALLATION' (with a dropdown arrow), 'MAIN CONCEPTS' (with an up arrow), a list of 12 items (1. Hello World, 2. Introducing JSX, 3. Rendering Elements, 4. Components and Props, 5. State and Lifecycle, 6. Handling Events, 7. Conditional Rendering, 8. Lists and Keys, 9. Forms, 10. Lifting State Up, 11. Composition vs Inheritance, 12. Thinking In React), 'ADVANCED GUIDES' (with a dropdown arrow), 'API REFERENCE' (with a dropdown arrow), and 'HOOKS (PROPOSAL)' (with a dropdown arrow). The '3. Rendering Elements' item is highlighted with a blue bar on the left.

# EVEN THIS IS A (MOSTLY) STATIC SITE



# SO WHAT IS A STATIC SITE GENERATOR?

- Generates a completely static build (html, css, js) of a site based on raw data
  - Data is often markdown but could be a database, an API, CSVs and so on
  - Typically makes use of a templating engine
- End result doesn't require a monolithic stack, can be hosted anywhere
- Popular projects: Jekyll, Hugo, Hexo, Gatsby, and many, many others.

## Data

Markdown, API, Database, CMS, etc.



## Static Site Generator

Consumes data at build time and spits out html, css, and js



## Hosting

Static assets, ideally on a CDN. Github Pages, Netlify, pretty much any host will do.

# ADVANTAGES OF STATIC SITES

- Speed
  - Browsers are pretty great at rendering static html
  - CDNs
- Security
  - Data source only needs to exist at build time
- Hosting
  - Simpler stack, easier to scale
- Developer Experience (depends a bit on the tool)

# BUT... I LOVE DRUPAL

Why would you use a static site generator with Drupal?

- Avoiding caching hell
  - Think of the effort that goes into making Drupal sites close to static – varnish, memcache, Drupal caching, etc.
- Think about the next inevitable security update
- Impact on hosting costs
- Drupal's admin UI

# **The Static Site Generator Landscape**

**From a Drupal perspective**

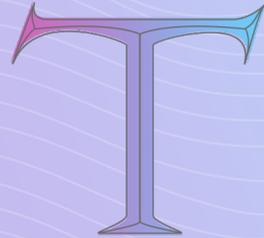
# SO... MANY... OPTIONS...

**StaticGen**  
A List of Static Site Generators for JAMstack Sites

Navigation: About | Contribute | About JAMstack | Need a Static CMS?

Filter: Any Language | Any Template | Any License | Sort: GitHub stars

Generator	Stars	Issues	PRs	Twitter
Jekyll	35970 +1781	143 +23	7891 +356	6575 +252
Next	31434 +6410	331 +80	3322 +951	N/A
Hugo	30286 +4831	305 +49	3528 +434	5186 +1544



# JEKYLL



The granddaddy of modern static site generators.

- Ruby project
- Liquid templating engine
- Markdown source
- Content in version control
- Supported natively by Github Pages
- Incremental builds
- Hugely popular

# JEKYLL - INSTALLING AND RUNNING

## Instructions

1. Install a full [Ruby development environment](#)

2. Install Jekyll and [bundler gems](#)

```
gem install jekyll bundler
```

3. Create a new Jekyll site at `./myblog`

```
jekyll new myblog
```

4. Change into your new directory

```
cd myblog
```

5. Build the site and make it available on a local server

```
bundle exec jekyll serve
```

6. Now browse to <http://localhost:4000>

# JEKYLL - DATA

```
1 ---
2 layout: post
3 title: "Welcome to Jekyll!"
4 date: 2018-11-13 17:51:09 -0600
5 categories: jekyll update
6 ---
7 You'll find this post in your `_posts` directory. Go ahead and edit it and
8 re-build the site to see your changes. You can rebuild the site in many
9 different ways, but the most common way is to run `jekyll serve`, which
10 launches a web server and auto-regenerates your site when a file is updated.
11
12 To add new posts, simply add a file in the `_posts` directory that follows the
13 convention `YYYY-MM-DD-name-of-post.ext` and includes the necessary front
14 matter. Take a look at the source for this post to get an idea about how it
15 works.
16
17 Jekyll also offers powerful support for code snippets:
18
19 ```ruby
20 {% highlight ruby %}
21 def print_hi(name)
22   puts "Hi, #{name}"
23 end
24 print_hi('Tom')
25 #=> prints 'Hi, Tom' to STDOUT.
26 {% endhighlight %}
```

Check out the [\[Jekyll docs\]\[jekyll-docs\]](#) for more info on how to get the most out of Jekyll. File all bugs/feature requests at [\[Jekyll's GitHub repo\]\[jekyll-gh\]](#). If you have questions, you can ask them on [\[Jekyll](#)

# JEKYLL - TEMPLATING

```
1 ---
2 layout: default
3 ---
4
5 <div class="home">
6   {%- if page.title -%}
7   <h1 class="page-heading">{{ page.title }}</h1>
8   {%- endif -%}
9
10  {{ content }}
11
12  {%- if site.posts.size > 0 -%}
13  <h2 class="post-list-heading">{{ page.list_title | default: "Posts" }}</h2>
14  <ul class="post-list">
15    {%- for post in site.posts -%}
16    <li>
17      {%- assign date_format = site.minima.date_format | default: "%b %-d, %Y" -%}
18      <span class="post-meta">{{ post.date | date: date_format }}</span>
19      <h3>
20        <a class="post-link" href="{{ post.url | relative_url }}">
21          {{ post.title | escape }}
22        </a>
23      </h3>
24      {%- if site.show_excerpts -%}
25      {{ post.excerpt | slice: 0, 400 }}...
26      {%- endif -%}
27    </li>
28    {%- endfor -%}
29  </ul>
```

# JEKYLL – WITH DRUPAL

Not a lot of formal options with Drupal 8

- Drupal 6 and 7 importers exist, but not Drupal 8 (that I could find)
- So then what?
  - Custom process to export data to markdown
  - Jekyll RSS importer
  - Jekyll CSV importer

```
$ ruby -r rubygems -e 'require "jekyll-import";  
  JekyllImport::Importers::CSV.run({  
    "file" => "my_posts.csv"  
  })'
```

# JEKYLL – OUTPUT

---

Umami Food Magazine About

---

## Posts

Nov 13, 2018

### [The umami guide to our favorite mushrooms](#)

We think mushrooms are one of the most enjoyable ingredients to cook with. There are plenty of edible varieties to try, each with their own distinctive shape, size, and taste. And with curious names such as chanterelle, the gypsy, horn of plenty, or hen of the woods, who wouldn't want to know more about cooking with the mighty mushroom?

One of the best things about mushrooms is their ver...

Nov 13, 2018

### [The real deal for supermarket savvy shopping](#)

This may not surprise you - but your supermarket is a hot bed of marketing mayhem, designed to improve their profit and to encourage the consumer to spend more than they intended. The tricks that all supermarkets employ are sometimes sensible ploys that any retailer should do to improve sales - but some may be more subtle and less obvious than you might think.

...

Nov 13, 2018

### [Skip the spirits with delicious mocktails](#)

Having a cocktail party? Then why not serve up something just as special for those who don't drink alcohol but do want to join in on all the fun? After all, what cocktail party is complete without those incredible looking glasses of mixed fruits, bright colors and of course, the little umbrellas? Do your bit

# JEKYLL



For you if:

- You want a simple road tested option
- You're all in on markdown
- Github Pages integration is important
- You don't hate Ruby

# JEKYLL



## Other Resources:

- [Jekyll Docs](#)
- [The New Medicurrent.com: Adventures in Decoupled Drupal](#)

## Similar projects:

- [Hexo](#)
- [Hugo](#)

# GATSBY



Build blazing fast apps and websites  
with React

- React project
- JSX Templating
- GraphQL to query data
- Thriving plugin ecosystem
- The new hotness
- Crazy fast. Seriously.
- Not just for static sites

# GATSBY - INSTALLING AND RUNNING

## Install Gatsby's command line tool

```
npm install --global gatsby-cli
```

## Using the Gatsby CLI

1. Create a new site.

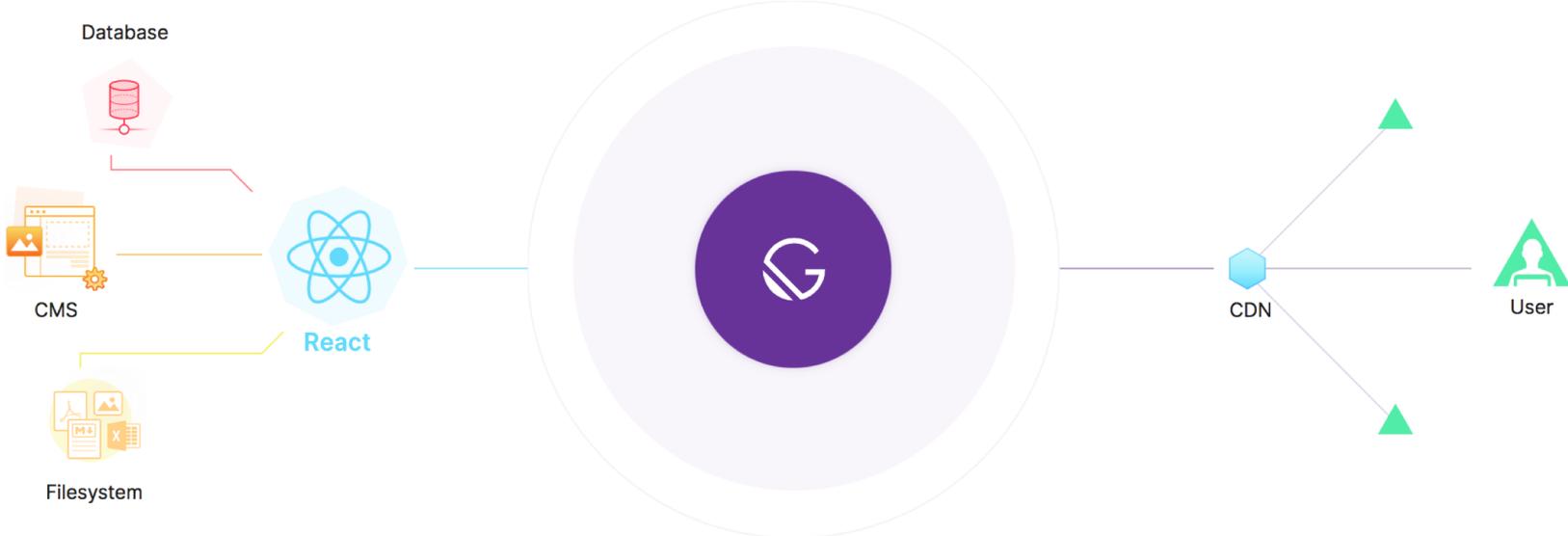
```
gatsby new gatsby-site
```

2. `cd gatsby-site`
3. `gatsby develop` — Gatsby will start a hot-reloading development environment accessible at `localhost:8000`
4. Try editing the JavaScript pages in `src/pages` . Saved changes will live reload in the browser.
5. `gatsby build` — Gatsby will perform an optimized production build for your site generating static HTML and per-route JavaScript code bundles.
6. `gatsby serve` — Gatsby starts a local HTML server for testing your built site.

Or with a starter:

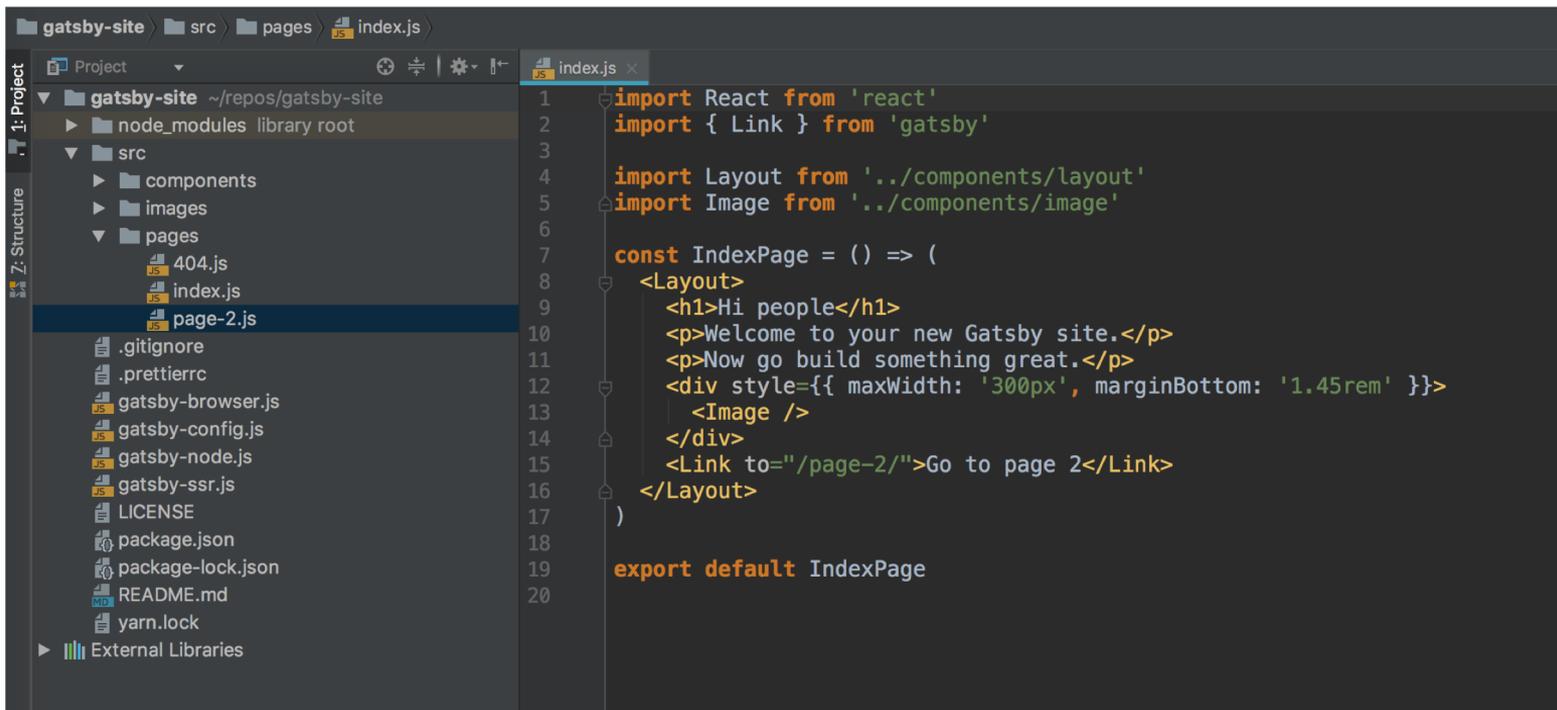
```
gatsby new gatsby-blog  
https://github.com/gatsbyjs/gatsby-starter-blog
```

# GATSBY - DATA



# GATSBY – DATA

Could be react components in src/pages



```
gatsby-site > src > pages > index.js
Project
  gatsby-site ~/repos/gatsby-site
    node_modules library root
    src
      components
      images
      pages
        404.js
        index.js
        page-2.js
      .gitignore
      .prettierrc
      gatsby-browser.js
      gatsby-config.js
      gatsby-node.js
      gatsby-ssr.js
      LICENSE
      package.json
      package-lock.json
      README.md
      yarn.lock
    External Libraries

index.js
1  import React from 'react'
2  import { Link } from 'gatsby'
3
4  import Layout from '../components/layout'
5  import Image from '../components/image'
6
7  const IndexPage = () => (
8    <Layout>
9      <h1>Hi people</h1>
10     <p>Welcome to your new Gatsby site.</p>
11     <p>Now go build something great.</p>
12     <div style={{ maxWidth: '300px', marginBottom: '1.45rem' }}>
13       <Image />
14     </div>
15     <Link to="/page-2/">Go to page 2</Link>
16   </Layout>
17 )
18
19 export default IndexPage
20
```

# GATSBY – DATA

Could be markdown from filesystem

```
gatsby-config.js x
1  module.exports = {
2    siteMetadata: {
3      title: `Pandas Eating Lots`,
4    },
5    plugins: [
6      {
7        resolve: `gatsby-source-filesystem`,
8        options: {
9          name: `src`,
10         path: `${__dirname}/src/`,
11       },
12     ],
13     `gatsby-transformer-remark`,
14     `gatsby-plugin-glamor`,
15     {
16       resolve: `gatsby-plugin-typography`,
17       options: {
18         pathToConfigModule: `src/utils/typography`,
19       },
20     },
21   ],
22 };
```

# GATSBY – CONSTRUCT QUERIES VIA GRAPHQL

GraphQL ▶ Prettify History < Docs

```
1 {
2   allMarkdownRemark {
3     edges {
4       node {
5         frontmatter {
6           title
7           date
8         }
9         html
10        excerpt
11      }
12    }
13  }
14 }
15 }
```

```
{
  "data": {
    "allMarkdownRemark": {
      "edges": [
        {
          "node": {
            "frontmatter": {
              "title": "Sweet Pandas Eating Sweets",
              "date": "2017-08-10"
            },
            "html": "<p>Pandas are really sweet.</p>\n<p>Here's a video of a panda eating sweets.</p>\n<iframe width=\"560\" height=\"315\" src=\"https://www.youtube.com/embed/4n0xNbfJLR8\" frameborder=\"0\" allowfullscreen></iframe>",
            "excerpt": "Pandas are really sweet. Here's a video of a panda eating sweets."
          }
        },
        {
          "node": {
            "frontmatter": {
              "title": "Pandas and Bananas",
              "date": "2017-08-21"
            },
            "html": "<p>Do Pandas eat bananas? Check out this short video that shows that yes! pandas do\nseem to really enjoy bananas!</p>\n<iframe width=\"560\" height=\"315\" src=\"https://www.youtube.com/embed/4SZ1r20_bY\" frameborder=\"0\" allowfullscreen></iframe>",
            "excerpt": "Do Pandas eat bananas? Check out this short video that shows that yes! pandas do\nseem to really enjoy bananas!"
          }
        }
      ]
    }
  }
}
```

QUERY VARIABLES

# GATSBY – 'TEMPLATING'

JS recipes.js x

```
1 import React from "react"
2 import { Link, graphql } from "gatsby"
3
4 import Layout from "../layouts"
5 import Container from "../components/container"
6
7 const AllRecipes = ({ data }) => (
8   <Layout>
9     <Container>
10      <h1>Recipes</h1>
11      <ul>
12        {data.allRecipes.edges.map(({ node }) => (
13          <li>
14            <Link to={node.fields.slug}>{node.title}</Link>
15          </li>
16        ))}
17      </ul>
18    </Container>
19  </Layout>
20 )
21
22 export default AllRecipes
```

```
23
24 export const query = graphql`
25   query {
26     allRecipes(limit: 1000) {
27       edges {
28         node {
29           title
30           fields {
31             slug
32           }
33         }
34       }
35     }
36   }
37 `
```

# GATSBY – WITH DRUPAL



## gatsby-source-drupal

Source plugin for pulling data (including images) into Gatsby from Drupal sites.

Pulls data from Drupal 8 sites with the [Drupal JSONAPI module](#) installed.

An example site built with the headless Drupal distro [ContentaCMS](#) is at <https://using-drupal.gatsbyjs.org/>

`apiBase` Option allows changing the API entry point depending on the version of `jsonapi` used by your Drupal instance. The default value is `jsonapi`, which has been used since `jsonapi` version `8.x-1.0-alpha4`.

## Install

```
npm install --save gatsby-source-drupal
```

[See starters that use this](#)

## How to use

```
// In your gatsby-config.js
plugins: [
  {
    resolve: `gatsby-source-drupal`,
    options: {
      baseUrl: `https://live-contentacms.pantheonsite.io/`,
      apiBase: `api`, // optional, defaults to `jsonapi`
    },
  },
]
```

## How to query

You can query nodes created from Drupal like the following:

```
{
  allArticle {
    edges {
      node {
        title
        internalId
        created(formatString: "DD-MMM-YYYY")
      }
    }
  }
}
```

# GATSBY – DEMO

🔍 Search by keyword, ingredient, dish

Login

## Umami

Food Magazine

Recipes

Our Recipe Pick  
**Frankfurter salad with mustard dressing**



Main course  
**Majorcan vegetable bake**



Main course  
**Barbecued beef - Chinese style**

# GATSBY



For you if:

- You prioritize performance
- You think in React
- You want data source flexibility
- You want to leverage the power of GraphQL
- You like the new and exciting
- You want to go beyond static

# GATSBY



## Other resources:

- [Gatsby Docs](#)
- Understanding the Limitations of Gatsby Source Drupal – [Blog post](#), [Video](#)
- Tons of talks and tutorials...

## Similar projects:

- [VuePress](#)
- [Next.js](#)
- [Static-site-generator-webpack-plugin](#)

# TOME



Drupal based static site generator and  
static content storage system

- A Drupal Project!
- Generates a static site that looks like... your Drupal site
- Exports and imports content as json

# TOME - INSTALLING AND RUNNING

- `composer create-project drupal-tome/tome-project my_site --stability dev --no-interaction --no-install`

- `drush tome:init`

Or, in an existing project:

- `composer require drupal/tome drupal-tome/tome_drush`

- `drush en tome -y`

- `drush tome:export`

- `drush tome:static --run-server`

# TOME - DATA

```
1
2   "uuid": [
3     {
4       "value": "60e94770-2e74-4f13-a2f8-9ae6fd94fe62"
5     }
6   ],
7   "langcode": [
8     {
9       "value": "en"
10    }
11  ],
12  "type": [
13    {
14      "target_id": "article",
15      "target_type": "node_type",
16      "target_uuid": "f9c64e8b-5a0f-4b07-9086-7c5d7f6508a0"
17    }
18  ],
19  "revision_timestamp": [
20    {
21      "value": "2018-06-19T14:33:20+00:00",
22      "format": "Y-m-d\\TH:i:sP"
23    }
24  ],
25  "revision_uid": [
26    {
27      "target_type": "user",
28      "target_uuid": "12d73d3a-832c-487f-bf92-d70d619238c5",
29      "url": "\/user\/1"
30    }
31  ]
}
```

# TOME - TEMPLATING

It's Drupal!



# TOME – OUTPUT

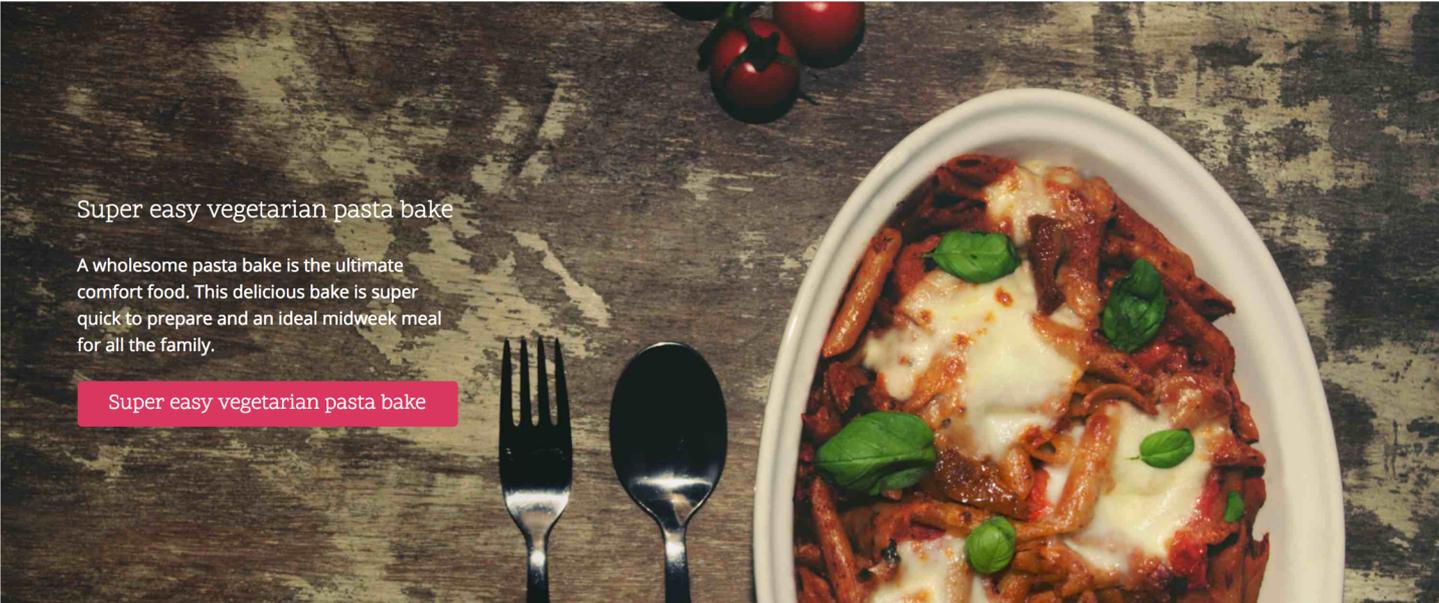
Log in



[Home](#)

[Articles](#)

[Recipes](#)



## Super easy vegetarian pasta bake

A wholesome pasta bake is the ultimate comfort food. This delicious bake is super quick to prepare and an ideal midweek meal for all the family.

Super easy vegetarian pasta bake

# TOME



For you if:

- You want to keep using the tools provide by Drupal
- You want Drupal content under version control
- You want to import flat file Drupal data into another system.

# TOME

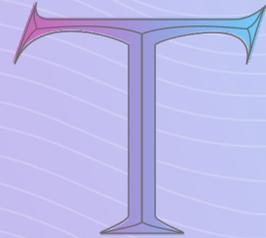


Additional resources:

- [Tome Docs](#)
- Try it out on your site

Similar projects:

- [static\\_generator](#)
- [default\\_content](#)
- [yaml\\_content](#)



# Automating Deployments

“I updated content in Drupal, now what?”

# DEPLOY A BUILD ASSET

backlineint / backlineint.github.io

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Deployed to github-pages

d279d0c was deployed by backlineint on Oct 16 Active [View deployment](#)

## Activity log

Show: All environments ▾

-  **github-pages** at d279d0c  
Deployed by backlineint on Oct 16 Active [View deployment](#)
-  **github-pages** at f668045  
Deployed by backlineint on Sep 11 Active [View deployment](#)
-  **github-pages** at ede8b47  
Deployed by backlineint on Aug 28 Active [View deployment](#)
-  **github-pages** at 39a1a42  
Deployed by backlineint on Jul 28 Active [View deployment](#)

# TRIGGER A BUILD WITH A COMMIT

General

**Build & deploy**

Continuous Deployment

Post processing

Deploy notifications

Domain management

Functions

Identity

Forms

Access control

## Continuous Deployment [?](#)

Settings for Continuous Deployment from a Git repository

### Deploy settings

Repository: <https://github.com/backlineint/camp-perry-gatsby>

Build command: **gatsby build**

Publish directory: **public/**

Production branch: **master**

Branch deploys: **Deploy only the production branch and its deploy previews**

Public deploy logs: **Logs are public**

[Learn more about common configuration directives in the docs](#) →

Edit settings

# TRIGGER A BUILD WITH A WEBHOOK

## Add webhook ☆

[Home](#) » [Administration](#) » [Configuration](#) » [Web services](#) » [Webhook](#)

### Label \*



Machine name: staging\_preview [\[Edit\]](#)

Label for the Webhook.

### Type \*

The webhook type, e.g. incoming or outgoing.

### Payload URL \*



Target URL for your payload.

### Secret



Secret that the target website gave you.

Active

Shows if the webhook is active or not.

### Content Type

The Content Type of your webhook.

ENTITY TYPE

EVENT

User

Create

User

Update

# Challenges

**“But my site can’t be static...”**

# POTENTIAL ROADBLOCKS TO A STATIC BUILD

- Content volume / build times
- Preview experience
- Alternative solutions exist for:
  - Form API
  - Authentication
- Politics

# CHALLENGE YOUR PERCEPTIONS



Why can't this site be static?

---

**You**

To yourself, after this session

**I should probably create a static site about this...**

**Thank You!**

**Brian Perry**

Lead Front End Developer

**Email:** [brian.perry@bounteous.com](mailto:brian.perry@bounteous.com)

**bounteous**