

Welcome to `gatsby@2.28.0` release (December 2020 #1).

Key highlights of this release:

- [New `gatsby new`](#) - new, interactive way to create a Gatsby site
- [Feature flags in `gatsby-config.js`](#) - set your feature toggles without environment variables
- [Improved Fast Refresh integration](#) - better hot reloading
- [Experimental: Lazy images in develop](#) - run image transformations only when they are needed by browser

Other notable changes:

- [Image plugin helpers](#) - make it easier for plugin authors to support the new gatsby image plugin
- [Experimental: New cache clearing behaviors](#) - we're experimenting with ways to be smarter about clearing caches
- [gatsby-plugin-emotion v5.0](#) - now uses emotion v11
- [Removed experimental lazy page bundling](#)
- [Notable bugfixes](#)

Sneak peek to next releases:

- [Experimental: Parallel data sourcing](#) - run source plugins in parallel to speedup sourcing on sites with multiple source plugins

**Bleeding Edge:** Want to try new features as soon as possible? Install `gatsby@next` and let us know if you have any [issues](#).

[Previous release notes](#)

[Full changelog](#)

## New `gatsby new`

In previous release we added [interactive way of scaffolding new gatsby project](#) (with `npm init gatsby`). As of this release, `gatsby new` (without any addition arguments) will use the same flow.

## Feature flags in `gatsby-config.js`

Gatsby traditionally used environment variables to use various modes or enable experimental features. This worked, but it was far from pleasant and had many issues, notably:

- Setting environment variables has idiosyncrasies across platforms (it's different for Windows and different for Linux or MacOS)
- Forced using NPM scripts (and correlating environment variables in continuous deployment solutions like Gatsby Cloud)

No longer! Now you can use flags in `gatsby-config.js`, which will work the same regardless of which environment you're on, where you're deploying, and for every member of your team! Woo hoo!

To use those, add `flags` to your `gatsby-config.js`:

```
// In your gatsby-config.js
module.exports = {
  // your existing config
  flags: {
```

```
    FAST_DEV: true,
  },
}
```

Currently supported flags:

- `DEV_SSR` - Server-side render (SSR) pages on full reloads during develop. Helps you detect SSR bugs and fix them without needing to do full builds.
- `QUERY_ON_DEMAND` - Only run queries when needed instead of running all queries upfront. Speeds starting the develop server.
- `LAZY_IMAGES` - Don't process images during development until they're requested from the browser. Speeds starting the develop server (requires `gatsby-plugin-sharp@^2.10.0`). See [Experimental: Lazy images in develop](#) section for additional details.
- `FAST_DEV` - Umbrella flag that enables `DEV_SSR`, `QUERY_ON_DEMAND` and `LAZY_IMAGES` features.
- `FAST_REFRESH` - Use React Fast Refresh instead of the legacy `react-hot-loader` for instantaneous feedback in your development server. Recommended for versions of React  $\geq 17.0$ .

## Improved Fast Refresh integration

Gatsby had some support for `react-refresh` (or Fast Refresh as it can be referred to) already. The PR for a [preliminary Fast Refresh integration](#) added a custom overlay, better error recovery and in general a better communication between Gatsby's dev server and React Fast Refresh. This will give you quicker feedback and an overall improved development experience.

To use it, add a flag to your `gatsby-config.js`:

```
// In your gatsby-config.js
module.exports = {
  // your existing config
  flags: {
    FAST_REFRESH: true,
  },
}
```

Visit the [umbrella issue about Fast Refresh](#) to see preview images and give feedback.

## Making `gatsby develop` faster

We continue working on speeding up your local development server. We need *your* help to test out these experiments and provide feedback, so please check 'em out! 🙌

### Experimental: Lazy images in develop

As more and more images are added to a Gatsby site, the slower the local development experience oftentimes becomes. You spend time waiting for images to process, instead of you know, developing! No longer! This experimental version of `gatsby-plugin-sharp` only does image processing when the page gets requested.

To use it, make sure you have `gatsby-plugin-sharp@^2.10.0` and add `LAZY_IMAGES` to flags in `gatsby-config.js`:

```
// In your gatsby-config.js
module.exports = {
  // your existing config
  flags: {
    LAZY_IMAGES: true,
  },
}
```

[Details and discussion.](#)

## Image plugin helpers

This release adds new utility functions to help plugin authors add support for the [new Gatsby image component](#). If you maintain a source plugin, or if you want to help with one, take a look at [the RFC](#).

## Experimental: New cache clearing behaviors

Gatsby aggressively clears its cache, sometimes too aggressively. Here's a few examples:

- You `npm install` a plugin, or update an existing
- You change your `gatsby-node.js` and add a few log statements
- You change your `siteMetadata` in `gatsby-config.js` to update your site's title

In all of these cases, your cache is entirely cleared, which means that the next time you run `gatsby develop` the experience is slower than it needs to be. We'll be working on this to ensure that your first run, and every run thereafter, is as quick and speedy as you expect!

We added two new flags for the webpack and downloaded files caches that when enabled, will preserve these caches across changes. We'll be evaluating their impact and safety to assess whether these can soon be enabled, by default. Please test and give feedback!

To enable, modify your `gatsby-config.js` as follows:

```
module.exports = {
  // your existing config
  flags: {
    PRESERVE_WEBPACK_CACHE: true,
    PRESERVE_FILE_DOWNLOAD_CACHE: true,
  },
}
```

### **gatsby-plugin-emotion@5.0.0**

The plugin is updated to the new major version of emotion: v11. Check out [this post](#) in emotion docs for updates.

## Removed experimental lazy page bundling

In `gatsby@2.27.0` we added [Experimental: Lazy page bundling](#) mode for `gatsby develop` that would delay compiling page templates until it was needed. While preliminary tests were very promising, we discovered few

showstoppers that degraded development experience. [We decided to end the experiment](#) for now and shift our efforts to [\(experimental\) new cache clearing behaviors](#).

## Notable bugfixes

- fix: hot reloading hangs on multiple fast saves in develop [#28237](#)
- fix error: `The result of this StaticQuery could not be fetched` when static query is added [#28349](#)

## Work in progress

### Experimental: Parallel data sourcing

Plugin APIs in Gatsby run serially. Generally this what we want as most API calls are CPU/IO bound so things are fastest letting each plugin have the full undivided attention of your computer. But source plugins are often *network* bound as they're hitting remote APIs and waiting for responses. We tried [changing the invocation of `sourceNodes` to parallel](#) on a few sites with 4+ source plugins and saw a big speedup on sourcing (40%+) as they were no longer waiting on each other to start their API calls.

This is a very "Your mileage may vary" situation — not all sites will notice any difference and also not all source plugins are network bound ( `gatsby-source-filesystem` reads from the local machine). We're looking at finding better heuristics so that all sites are as fast as possible at data sourcing but in the meantime, if you have sites with multiple source plugins, this could be a big help.

You can try it today using `gatsby@next` version and adding a flag to your `gatsby-config.js` :

```
// In your gatsby-config.js
module.exports = {
  // your existing config
  flags: {
    PARALLEL_SOURCING: true,
  },
}
```

## Contributors

A big **Thank You** to [our community who contributed](#) to this release 💜

- [styxlab](#): fix(gatsby-source-filesystem): fix race condition when using `publicURL` field [PR #28176](#)
- [visualfanatic](#): link to issue fixed inside release notes [PR #28193](#)
- [oorestisime](#): feat(gatsby-plugin-sitemap): allow `serialize` plugin option to be async function [PR #28207](#)
- [theskillwithin](#): breaking(gatsby-plugin-emotion): update to emotion@11 [PR #27981](#)
- [hassankhan](#): fix(gatsby-transformer-remark): ensure `getNodesByType()` is passed through [PR #28218](#)
- [nina-py](#): chore(gatsby-cli): Bump up update-notifier version to 5.0.1 [PR #28273](#)
- [leerob](#): Add Vercel Analytics to documentation. [PR #27841](#)
- [KKVANONYMOUS](#): chore(docs): Fix link to inc builds [PR #28306](#)
- [jmiazga](#): fix(gatsby-recipes): updated chakra ui recipe after v1 release [PR #28270](#)
- [blainekasten](#): feat(gatsby): Add preliminary fast-refresh integration [PR #26664](#)