Welcome to `gatsby@2.27.0` release (November 2020 #2).

Key highlights of this release:

- `create-gatsby` - new, interactive way to create a Gatsby site
- Experimental: Queries on Demand - improves `gatsby develop` bootup time
- Experimental: Lazy page bundling - also improves `gatsby develop` bootup time
- Experimental: SSR in develop - catch SSR errors early
- `gatsby-plugin-mdx@1.5.0` - new option for better performance
- Notable bugfixes

Sneak peek to next releases:

- Experimental: Lazy images in develop
- Documentation Reorganization

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

`create-gatsby`

There is now a new, interactive way to create a Gatsby site. As long as you have Node.js installed you're ready to go.

```
npm init gatsby
```

This command will ask a series of questions, allowing you to select your CMS, preferred styling tools and common Gatsby plugins. After you've made your selections let the installation complete and you'll have a working Gatsby site, all packages included and configured for use.

Making `gatsby develop` faster

The Gatsby develop server can get slow to start on larger sites. We're working hard to speed it up. We're addressing different scaling problems one by one and have shipped several improvements behind flags as detailed below. If you'd like to enable all these dev speedups (along with all future ones!), we've created a single flag for you. Run your site like `GATSBY_EXPERIMENTAL_FAST_DEV=true gatsby develop` and zoom zoom!

Experimental: Queries on Demand

When developing a Gatsby site there's no need to run all graphql queries before serving the site. Instead, Gatsby will run queries for pages as they're requested by a browser. Think of it like lazily loading the data your pages need, when they need it!

This avoids having to wait for slower queries (like image processing) if you're editing an unrelated part of a site. What this means for you: faster local development experience, up to 2x faster in many cases!

This feature is available under a flag.

```
GATSBY_EXPERIMENTAL_QUERY_ON_DEMAND=1 gatsby develop
```

Please try it and [let us know](#) if you have any issues (it may become a default for `develop` in the future)

[Details and discussion](#).

Experimental: SSR in Development

One of the least enjoyable bugs to encounter in Gatsby is when your build fails due to code trying to reference `window` or `document` or other browser globals that are not accessible in Node.js during SSR.

Currently the only way to debug these is to change some code and rebuild and repeat until the problem is solved. This is a very slow way to debug. Worst, these sorts of bugs are often only encountered after a long development period. It's no fun to push code you're proud of to CI only to discover it's broken.

With this coming feature, we'll SSR pages during development when do a full refresh of a page (navigating between pages will still only do a client-side render). This will help you both discover build errors earlier *and* fix them faster.

Try it out immediately by running `GATSBY_EXPERIMENTAL_DEV_SSR=true gatsby develop`. Join in the discussion in its [umbrella issue](#).

Experimental: Lazy page bundling in development

UPDATE: After a number of community members tested the change on their website, we decided it wasn't going to work out — https://github.com/gatsbyjs/gatsby/discussions/28137#discussioncomment-138998

An obstacle to Gatsby being a delightful experience for larger sites is JavaScript compilation can start to get annoyingly slow. For example, gatsbyjs.com takes over two minutes currently (with a cold cache) to compile and bundle the code for the many page components. Not acceptable!

We knew we needed to make this lazy and have shipped experimental support for this.

Now when you run `GATSBY_EXPERIMENT_DEVJS_LAZY=true gatsby develop`, webpack won't look at any of your page components until you visit them. You'll notice a slight (generally under 0.5s) delay when you first visit a page while webpack compiles it but thereafter, it'll be instantaneous.

All sites should see some speedups but it's especially noticeable for larger sites like gatsbyjs.com which now starts webpack 81% faster than before! Please test it out and tell us how fast your dev server boots up over at the [umbrella issue](#) along with any bugs you might run across.

`gatsby-plugin-mdx@1.5.0`

Adds a `lessBabel` option to the plugin which allows you to use a fast path for scanning exports during sourcing. The savings are significant, especially at scale, but as the new scanner does not use Babel, if your site depends on Babel then you can't use this. Please give the option a try and report any problems unrelated to not running Babel so we can improve the support.

See PR [#27941](#)

Notable bugfixes

- fix: endless page refresh in develop [#28034](#)
- fix: performance regression when running queries [#28032](#)

Work in progress

Experimental: Lazy images in develop

We've got some feedback that the more images your website contains, the slower your local development experience gets. 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.

Try early alpha (and [let us know](#) if you have any issues with it):

```
npm install gatsby-plugin-sharp@lazy-images
```

Lazy images are enabled by-default in this version.

[Details and discussion](#).

Documentation Reorganization

We've heard repeatedly from the community that Gatsby is a powerful tool, but it has a steep learning curve. We want to make it easier to get started with Gatsby. And that means having documentation that helps y'all find the information you need when you need it.

[Announcement and discussion](#).

Contributors

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

- [jerrydevs](#): fix(gatsby-transformer-asciidoc): fails when doc doesn't have title [PR #27865](#)
- [shoxton](#)
    - feat(gatsby-recipes): add Chakra UI recipe [PR #27721](#)
    - feat(recipes-list): add chakra-ui [PR #28013](#)
- [vhiairrassary](#): Fix #26359: Support HTTPS for the develop status server [PR #27955](#)
- [stefanjudis](#): docs(gatsby-source-contentful): clean up changelog [PR #28000](#)
- [lellky](#): chore(docs): Update add-seo-component [PR #28022](#)
- [MichaelDeBoey](#)
    - chore(gatsby-plugin-glamor): Add pluginOptionsSchema validation [PR #27602](#)
    - chore(gatsby-plugin-twitter): Add pluginOptionsSchema validation [PR #27601](#)
- [trevorblades](#): Make optional SVG favicon come after the fallback [PR #27843](#)
- [moonmeister](#): fix(plugin-manifest): Allow for all valid WebAppManifest properties [PR #27951](#)
- [nm123github](#): add query pluginOption for mongodb plugin [PR #27756](#)
- [sreeharisj23](#): chore: change .org to .com [PR #28053](#)
- [DecliningLotus](#): chore(docs): replace typefaces with fontsource [PR #27313](#)
- [yaacovCR](#): chore(gatsby-source-graphql): upgrade graphql-tools to v7 [PR #27792](#)