

Welcome to `gatsby@2.29.0` release (December 2020 #2)

Key highlights of this release:

- [Query on Demand](#) - improves `gatsby develop` bootstrap time
- [Lazy Images](#) - do not wait for all images to be processed to start the development
- [Improvements to our CLI](#) - improved `create-gatsby` & new `plugin` command
- [Experimental: Parallel data sourcing](#) - run source plugins in parallel to speedup sourcing on sites with multiple source plugins

Other notable changes:

- [Performance improvements](#)
- [Slugify option for File System Route API](#)
- [gatsby-image codemod](#)
- [Notable bugfixes](#)

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](#)

Query on Demand

Starting with v2.29, 10% of our users are automatically opt-in to this feature. We've first shipped this feature behind a flag in [v2.27](#) and feel confident now that more people can try it out. Opt-in users will receive a notice in their terminal about that opt-in behavior and a hint on how to turn it off (in case it disturbs your workflow). As a recap of what Query on Demand will improve:

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!

By slowly rolling out this feature to more and more users we'll be getting closer to a GA (general availability) release that will benefit all users of Gatsby. Please give us your feedback in the [umbrella discussion](#).

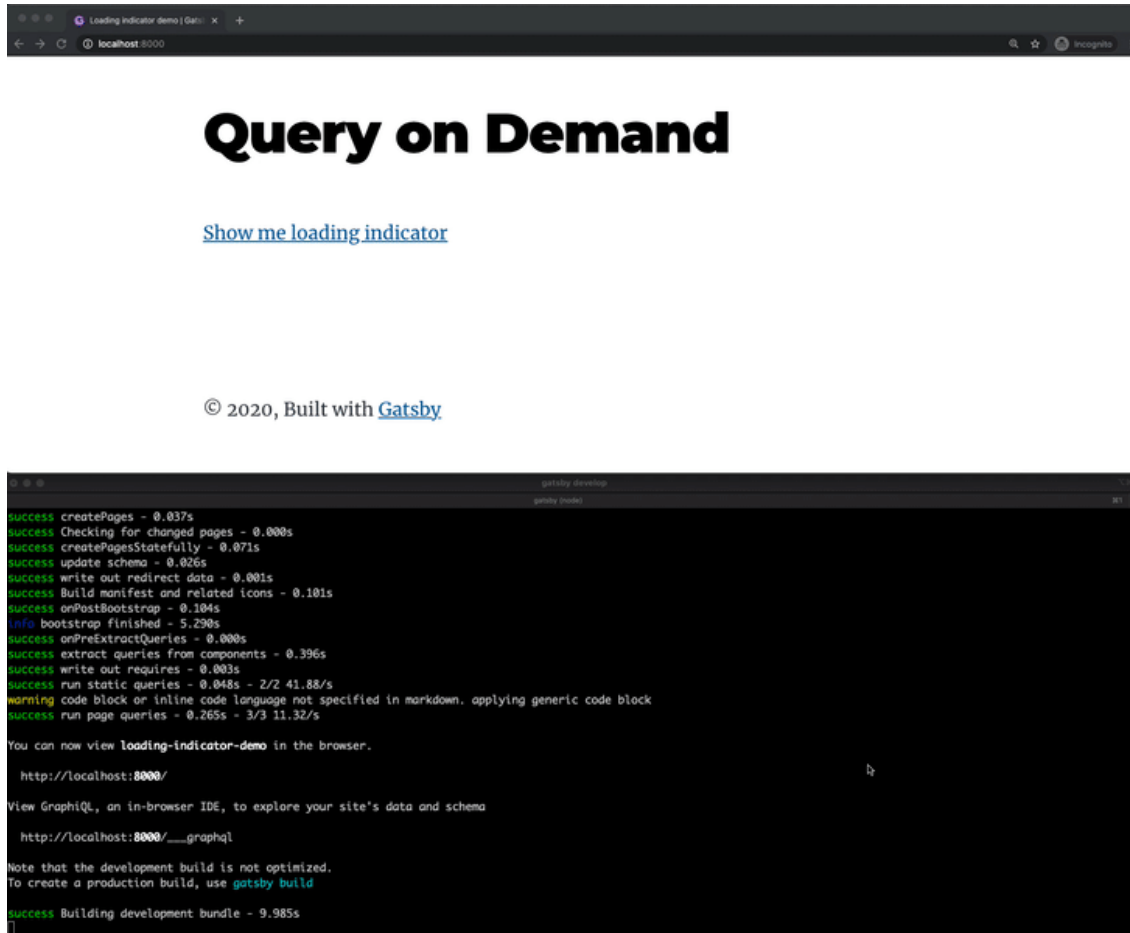
If you want to turn it on/off, you can set the corresponding flag inside `gatsby-config.js`:

```
// In your gatsby-config.js
module.exports = {
  flags: {
    QUERY_ON_DEMAND: false,
  },
  plugins: [], // your plugins stay the same
}
```

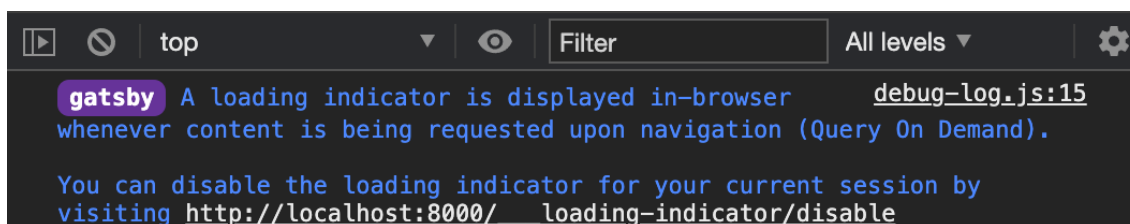
In v2.29 we improved the UX around it by adding a loading indicator and message to the browser console (only in `gatsby develop`). If you want or need to de-activate this indicator, you can! For more details please see the [umbrella discussion](#). Here's a preview of them in action:

Loading indicator

The loading indicator respects the user's settings for `prefers-reduced-motion` and `prefers-color-scheme`. It also announces itself to screen readers.



Browser console



Lazy Images

Similarly as with Query on Demand also Lazy Images will be automatically delivered to 10% of our users with this v2.29 release. We've first shipped this feature behind a flag in [v2.28](#). Opt-in users will receive a notice in their terminal about that opt-in behavior and a hint on how to turn it off (in case it disturbs your workflow). As a recap of what Lazy Images will improve:

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.

By slowly rolling out this feature to more and more users we'll be getting closer to a GA release that will benefit all users of Gatsby. Please give us your feedback in the [umbrella discussion](#).

If you want to turn it on/off, you can set the corresponding flag inside `gatsby-config.js`:

```
// In your gatsby-config.js
module.exports = {
  flags: {
    LAZY_IMAGES: false,
  },
  plugins: [], // your plugins stay the same
}
```

Improvements to our CLI

In [v2.27](#) we introduced `create-gatsby`, a new and interactive way to create a Gatsby site. You can run it in your terminal with `npm init gatsby`.

A couple of papercuts were fixed but we also added new features:

- Prompt for your site name instead of a folder name
- Automatically add this name to your `siteMetadata` and `package.json`
- Add `-y` flag. This flag bypasses all prompts other than naming your site

The regular `gatsby-cli` received a new command to list out all plugins in your site by running `gatsby plugin ls`.

Experimental: Parallel data sourcing

In [v2.28](#) we gave a sneak peek at a new feature that enables parallel data sourcing. As a recap:

Plugin APIs in Gatsby run serially. Generally this is 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.

You're now able to activate this experiment in the stable release of Gatsby by adding a flag to your `gatsby-config.js`:

```
// In your gatsby-config.js
module.exports = {
  flags: {
    PARALLEL_SOURCING: true,
  },
  plugins: [], // your plugins stay the same
}
```

Please give us your feedback in the [umbrella discussion](#).

Performance improvements

We were able to ship a bunch of performance improvements both to Gatsby itself and its plugins:

`gatsby`

- Improve query running performance for sites with large amount of data (up to 10% in our tests). See <https://github.com/gatsbyjs/gatsby/pull/28525> for details.

`gatsby-source-contentful`

- Improve performance of data sourcing for large Contentful spaces. See <https://github.com/gatsbyjs/gatsby/pull/28375> for details.
- Prevent concurrent requests to Contentful's image processing API. See <https://github.com/gatsbyjs/gatsby/pull/28438> for details.

Slugify option for File System Route API

The File System Route API uses [slugify](#) to create slugs for the generated routes. You're now able to pass custom options to that instance, e.g. when you want to change the separator. The full details are listed in the [README](#) of `gatsby-plugin-page-creator`.

`gatsby-image` **codemod**

We introduced some API changes for working with images when we published the new `gatsby-plugin-image` in [v2.26](#). In order to make it easier to migrate your code to work with the new plugin, we've created a codemod. Follow the migration instructions in the [README](#) in order to run the codemod against your project.

Notable bugfixes

- Scroll restoration issue in the browser API was fixed in [#27384](#) that affected e.g. page transitions
- Do not fail in develop when eslint loader is removed in [#28494](#)
- Respect hash as source of truth for scroll position in [#28555](#)
- Wait for jobs to complete in `onPostBuild` in [#28534](#)
- Truncated long terminal messages fixed in [#26190](#)

Contributors

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

- [bappr](#): Update kintohub documentation with kintohub V1 deployment [PR #27016](#)
- [axe312ger](#)
 - fix: improve error handling, always show API response status and message [PR #27730](#)
 - fix(documentationjs): fix example caption string [PR #27365](#)
- [olisteadman](#): Tell user to supply BOTH keys in .env.development. [PR #28405](#)
- [galihmelon](#): [docs][guides] improvements to Why Gatsby Uses GraphQL #15235 [PR #27640](#)
- [sreeharisj23](#): Remove Broken Link. [PR #28412](#)
- [vrabe](#): fix(gatsby): scroll restoration issue in browser API [PR #27384](#)
- [WillMayger](#): fix(gatsby): re-render route when location state changes [PR #28346](#)
- [kasipavankumar](#): chore(gatsby-plugin-manifest): Update pluginOptionsSchema link [PR #28344](#)

- [rburgst](#): fix: avoid joi validation error when using reporter.panic [PR #28291](#)
- [davidmauskop](#): chore(docs): Update Deploying to Render guide [PR #28272](#)
- [saintmalik](#): chore(docs): fix broken url [PR #28197](#)
- [muescha](#): fix(docs): document pluginOptionsSchema - add file extension, code block language, quotation marks [PR #27844](#)
- [herecydev](#) & [hoobdeebla](#): chore: update ink to v3 [PR #26190](#)