

Welcome to `gatsby@3.10.0` release (July 2021 #2)

Key highlights of this release:

- [Experimental: Parallel Query Running](#) - Improves time it takes to run queries during gatsby build
- [Experimental: webpack persistent caching for `gatsby develop`](#) - significantly speed up start of webpack server

Also check out [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](#)

Experimental: Parallel Query Running

Gatsby's build process is composed of multiple steps (see [our documentation](#) for more details) and one step that will increase in time with more and more pages/nodes is query running. You're seeing this step as `run static queries` and `run page queries` in your build log.

This step currently only runs in a singular process and the goal of Parallel Query Running is to spread out the work to multiple processes to better utilize available cores & memory. We're using [gatsby-worker](#) (highly inspired by [jest-worker](#)) and [lmdb-store](#) to accomplish this. In [Gatsby v3.7](#) we've added support for LMDB in Gatsby and are now leveraging this new data storage option to enable communication between the main process and the workers.

Depending on the type of queries you use you will see dramatic improvements in performance. You can try out different types of queries (and expected results) with the [query-filters-sort benchmark](#). Toggle the feature flag for a before/after comparison.

Here are two examples:

- Fast `eq-uniq` filter with `GATSBY_CPU_COUNT=5 NUM_NODES=100000 NUM_PAGES=10000 FILTER=eq-uniq TEXT=1`.
 - **Before:** `run page queries - 3.787s - 10001/10001 2641.07/s`
 - **After:** `run queries in workers - 3.445s - 10001/10001 2903.34/s`
 - For the already fast `eq` filters you will see smaller improvements compared to the slower filters like...
- Slow `gt` filter with `GATSBY_CPU_COUNT=5 NUM_NODES=10000 NUM_PAGES=10000 FILTER=gt TEXT=1`:
 - **Before:** `run page queries - 41.832s - 10001/10001 239.07/s`
 - **After:** `run queries in workers - 15.072s - 10001/10001 663.57/s`
 - Huge improvements for more complex queries or filters that are not "Fast Filters"

To try it out in your own site, please make sure that you're using Node v14.10 or later. Install `lmdb-store` as a dependency:

```
npm install lmdb-store
```

Then enable the config flag in your `gatsby-config.js` :

```
module.exports = {
  flags: {
    PARALLEL_QUERY_RUNNING: true,
  },
}
```

Please share your results, findings, and feedback in the [PQR GitHub Discussion](#). You'll also be able to read about known/common pitfalls there and possible solutions.

Experimental: webpack persistent caching for `gatsby develop`

After rolling out webpack 5 persistent caching for production builds in [Gatsby v3.8](#) we're now beginning the gradual rollout of it for `gatsby develop` . It greatly improves the startup time of the development server.

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

```
module.exports = {
  flags: {
    DEV_WEBPACK_CACHE: true,
  },
}
```

If you're already using the `FAST_DEV` flag you'll be using it automatically once you update to Gatsby v3.10. Please share your feedback in the [GitHub Discussion](#).

Notable bugfixes & improvements

- `gatsby` : Update `postcss` to 8.3.5 to remove deprecation warning on Node v16.
- `gatsby` : Switched `createRoot` to `hydrateRoot` . Please note, this only applies if you use [React 18 in Gatsby](#).
- `gatsby-source-wordpress` : Check preview URL earlier and give better feedback, via [PR #32251](#).
- `gatsby` : Pass `search` and `hash` to `window.location` to final URL of redirect and after the service worker updated, via [PR #32334](#) and [PR #32323](#).
- `gatsby` : Avoid the `UNHANDLED REJECTION write EPIPE` error when using `Ctrl + C` , via [PR #32311](#) and [PR #32356](#).
- `gatsby` : When a `gatsby build` fails on e.g. missing data it now prints the `page-data.json` file for this page to give more context on what's missing, via [PR #32301](#).
- `gatsby-source-contentful` : Support image corner radius from Image API, via [PR #32333](#).
- `gatsby-source-contentful` : Support `metadata.tags` property, via [PR #31746](#).

Contributors

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

- [RapTho](#): Wrong post ids in example author name filter [PR #32191](#)
- [anselm94](#): chore(docs): Update Storybook guide to v6 [PR #31653](#)
- [SarthakC](#): fix: added missing parentheses in creating a source plugin tutorial [PR #32259](#)

- [emmanuelgautier](#): fix(gatsby-plugin-gtag): replace google analytics domain with google tag manager [PR #31036](#)
- [nellaparedes](#): fix(gatsby): Pass search/hash to location after swUpdated [PR #32323](#)
- [karlhorky](#): Avoid UNHANDLED REJECTION error on ctrl-C [PR #32311](#)
- [weronikadominia](#): docs(contributing): update docs with info about translations being on hold (#31883) [PR #32328](#)
- [cabutler10](#): chore(docs): Fix typo in apollo/client npm package name [PR #32345](#)
- [ezeYaniv](#): chore(docs): Update building-a-theme to latest Theme UI [PR #32357](#)
- [SonnyBrooks](#): chore(docs): Correct JavaScript spelling [PR #32368](#)
- [axe312ger](#)
 - feat(contentful): add support image corner radius [PR #32333](#)
 - feat(contentful): add support for tags [PR #31746](#)