

[← Back to Blog](#)

Thursday, December 18th 2025

Next.js 16.1

Posted by



Luke Sandberg
@lukeisandberg



Tim Neutkens
@timneutkens

Next.js 16.1 focuses on faster development workflows and improved stability, with major updates to Turbopack and tooling.

- **Turbopack File System Caching for `next dev` (stable)**: Improved compile times for `next dev` by default.
- **Next.js Bundle Analyzer (experimental)**: Optimize your code with our new interactive tool.
- **Easier debugging**: Debug your Next.js app with `next dev --inspect`.
- **Transitive external dependencies**: Turbopack can automatically handle transitive external dependencies with no warnings.

Upgrade Today

```
>_ terminal
```



```
# Use the automated upgrade CLI
npx @next/codemod@canary upgrade latest

# ...or upgrade manually
npm install next@latest react@latest react-dom@latest

# ...or start a new project
npx create-next-app@latest
```

Turbopack File System Caching for `next dev`

Turbopack file system caching for `next dev` is now stable and on by default. Compiler artifacts are stored on disk, leading to significantly faster compile times when restarting your development server, especially in large projects.

First route compile time

react.dev		~10× faster
Cold	3.7s	
Cached	380ms	
nextjs.org		~5× faster
Cold	3.5s	
Cached	700ms	
Large internal Vercel app		~14× faster
Cold	15s	
Cached	1.1s	

Internal applications at Vercel have been dogfooding this for the past year. To learn more about how we built file system caching for Turbopack, watch [Luke Sandberg's talk at Next.js Conf](#) ↗.

Following this release, we'll be stabilizing file system caching for `next build`. See our [documentation](#) for more information, and share your [feedback](#) on the dedicated GitHub discussion.

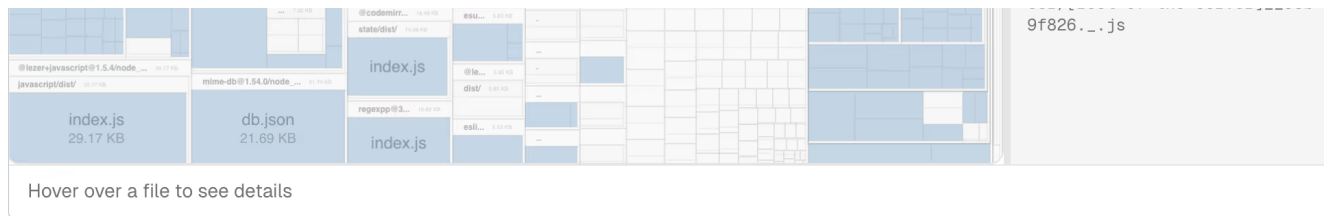
Next.js Bundle Analyzer (experimental)

Next.js 16.1 includes a new experimental [Bundle Analyzer](#) that works with Turbopack. It makes it easier to optimize bundle sizes for both server and client code—helping improve Core Web Vitals, reduce lambda cold start times, and identify bloated dependencies.

```
>_ terminal 📄  
  
next experimental-analyze
```

Running the command launches an interactive UI to inspect production bundles, identify large modules, and see why they're included.

The screenshot displays the Next.js Bundle Analyzer interface. The main area is a treemap showing the composition of the production bundle. The largest module is `react-dom` at 53.62 KB. Other significant modules include `HomeContent.js` (59.90 KB), `index.js` (52.01 KB), and `acorn.mjs` (28.66 KB). The interface includes a search bar at the top, filters for 'Compressed', 'Client', and '3 file types', and a search field for files. The right sidebar shows the selected file, `TopNav.tsx`, with its size (34.62 KB uncompressed, 12.02 KB compressed) and an import chain showing its dependencies: `TopNav.tsx` depends on `Page.tsx`, which depends on `index.tsx`, which depends on `TopNav.tsx`. The output chunks section at the bottom shows the file's location in the production bundle.



The Next.js Bundle Analyzer showing a treemap with `TopNav.tsx` selected, revealing its import chain.

Try it yourself: [Open the interactive Bundle Analyzer demo](#) to explore the module graph.

The Bundle Analyzer is deeply integrated into Next.js, allowing you to:

- Filter bundles by route
- View the full import chain showing why a module is included
- Trace imports across server-to-client component boundaries and dynamic imports
- View CSS and other imported asset sizes
- Switch between client and server views

The Bundle Analyzer is in early development and will be improved further in future releases. Share your feedback on the dedicated [GitHub discussion](#).

Easier Debugging with `next dev --inspect`

You can now enable the [Node.js debugger](#) by passing `--inspect` to `next dev`. Previously this required passing `NODE_OPTIONS=--inspect` and would attach the inspector to all processes spawned by Next.js instead of only to the process running your code.

Improved Handling of `serverExternalPackages`

Next.js allows you to keep dependencies unbundled using `serverExternalPackages`. Previously, this only worked reliably for direct dependencies. If you used a library that

internally depends on something like `sqlite`, and needed to externalize `sqlite`, you'd have to add it to your own `package.json`—even though it's not your direct dependency. This workaround leaked internal implementation details, created maintenance burden, and could lead to impossible version conflicts when multiple packages required different versions of the same dependency.

Next.js 16.1 fixes this for Turbopack, which now correctly resolves and externalizes transitive dependencies in `serverExternalPackages` without additional configuration.

Other Updates

- **20MB smaller installs:** Next.js installs are about 20MB smaller thanks to simplifications in the Turbopack file system caching layer.
- **New `next upgrade` command:** A new `next upgrade` command makes upgrading easier. Going forward, you can just run this to upgrade Next.js versions.
- **MCP `get_routes` tool:** The [Next.js DevTools MCP server](#) now has a `get_routes` tool to get the full list of routes in your application.
- **`generateStaticParams` timing:** Time spent on `generateStaticParams` is now logged as part of the timings shown for requests in development.
- **Build worker logging:** `next build` "Collecting page data" and "Generating static pages" now log the number of worker threads used.
- **Improved async import bundling:** Turbopack has improved bundling of async imports in dev to reduce the number of chunks produced, avoiding certain pathological but real-world cases.
- **Relative source map paths:** Turbopack now produces source maps with relative file paths for server-side code, improving compatibility with Node.js and other ecosystem tools.

Feedback and Community

Share your feedback and help shape the future of Next.js:

- [GitHub Discussions](#) ↗
 - [GitHub Issues](#) ↗
 - [Discord Community](#) ↗
-

Contributors

Next.js is the result of the combined work of over 3,700 individual developers. This release was brought to you by:

- The **Next.js** team: [Andrew](#) ↗, [Hendrik](#) ↗, [Janka](#) ↗, [Jiachi](#) ↗, [Jimmy](#) ↗, [Jiwon](#) ↗, [JJ](#) ↗, [Josh](#) ↗, [Jude](#) ↗, [Sam](#) ↗, [Sebastian](#) ↗, [Sebbie](#) ↗, [Wyatt](#) ↗, and [Zack](#) ↗.
- The **Turbopack** team: [Benjamin](#) ↗, [Luke](#) ↗, [Niklas](#) ↗, [Tim](#) ↗, [Tobias](#) ↗, and [Will](#) ↗.
- The **Next.js Docs** team: [Delba](#) ↗, [Rich](#) ↗, [Ismael](#) ↗, and [Joseph](#) ↗.

Huge thanks to [@kdy1](#), [@eps1lon](#), [@SyMind](#), [@bgw](#), [@swarnava](#), [@devjiwonchoi](#), [@ztanner](#), [@ijjk](#), [@huozhi](#), [@icyJoseph](#), [@acdlite](#), [@unstubbable](#), [@gnoff](#), [@gusfune](#), [@lukesandberg](#), [@sokra](#), [@hayes](#), [@shuding](#), [@wyattjoh](#), [@marjan-ahmed](#), [@timneutkens](#), [@ajstrongdev](#), [@zigang93](#), [@mischnic](#), [@Nayeem-XTREME](#), [@hamirmahal](#), [@eliOshin](#), [@tessamero](#), [@gaojude](#), [@jamesdaniels](#), [@georgesfarah](#), [@timeyoutakeit](#), [@sequencerr](#), [@Strernd](#), [@lucasadrianof](#), [@wbinssmith](#), [@hamidreza-nategi](#), [@jokokoloko](#), [@dijonmusters](#), [@H01001000](#), [@xusd320](#), [@lubieowoce](#), [@KaziMahbuburRahman](#), [@zhiyanzhaijie](#), [@feedthejim](#), [@that-one-arab](#), [@JamBalaya56562](#), [@shrink](#), [@florianliebig](#), [@allenzhou101](#), [@benmerckx](#), [@ymc9](#), [@Marukome0743](#), [@pyrytakala](#), [@danpeleg4](#), [@gaearon](#), [@styfle](#), [@jhuleatt](#), [@muhammadsyaddad](#), [@roelvan](#), and [@SukkaW](#) for helping!



Resources

[Docs](#)

[Support Policy](#)

[Learn](#)

[Showcase](#)

[Blog](#)

[Team](#)

[Analytics](#)

[Next.js Conf](#)

[Previews](#)

[Evals](#)

More

[Next.js Commerce](#)

[Contact Sales](#)

[Community](#)

[GitHub](#)

[Releases](#)

[Telemetry](#)

[Governance](#)

About Vercel

[Next.js + Vercel](#)

[Open Source Software](#)

[GitHub](#)

[Bluesky](#)

[X](#)

Legal

[Privacy Policy](#)

Subscribe to our newsletter

Stay updated on new releases and features, guides, and case studies.

© 2026 Vercel, Inc.

