

gatsby-transformer-sharp

Creates `ImageSharp` nodes from image types that are supported by the [Sharp](#) image processing library and provides fields in their GraphQL types for processing your images in a variety of ways including resizing, cropping, and creating responsive images.

[Live demo](#) ([source](#))

Install

```
npm install gatsby-transformer-sharp gatsby-plugin-sharp
```

How to use

```
// In your gatsby-config.js
module.exports = {
  plugins: ['gatsby-plugin-sharp', 'gatsby-transformer-sharp'],
}
```

Please note that you must have a source plugin (which brings in images) installed in your project. Otherwise, no `ImageSharp` nodes can be created for your files. Examples would be [gatsby-source-filesystem](#) or source plugins for (headless) CMSs like [gatsby-source-wordpress](#) .

Note: An exception to this is when using [gatsby-source-contentful](#) , as the source plugin and the assets are not [downloaded to the local filesystem](#). By default, the `gatsby-source-contentful` plugin creates a `ContentfulAsset` node for every image with links to Contentful's CDN, therefore it is not necessary to use `gatsby-transformer-sharp` together with `gatsby-source-contentful` .

Parsing algorithm

It recognizes files with the following extensions as images.

- jpeg
- jpg
- png
- webp
- tif
- tiff

Each image file is parsed into a node of type `ImageSharp` .

Configuration options

```
checkSupportedExtensions [boolean][optional]
```

Sharp only supports certain image formats (see the Parsing algorithm section above) and hence throws a warning when you e.g. use a .gif in an `ImageSharp` query. You'll need to use `publicURL` instead. With this option you can disable the warning behavior.

```
// In your gatsby-config.js
module.exports = {
  plugins: [
    `gatsby-plugin-sharp`,
    {
      resolve: `gatsby-transformer-sharp`,
      options: {
        // The option defaults to true
        checkSupportedExtensions: false,
      },
    },
  ],
}
```

Troubleshooting

Incompatible library version: sharp.node requires version X or later, but Z provides version Y

This means that there are multiple incompatible versions of the `sharp` package installed in `node_modules`. The complete error typically looks like this:

```
Something went wrong installing the "sharp" module

dlopen(/Users/misiek/dev/gatsby-starter-
blog/node_modules/sharp/build/Release/sharp.node, 1): Library not loaded:
@rpath/libglib-2.0.dylib
  Referenced from: /Users/misiek/dev/gatsby-starter-
blog/node_modules/sharp/build/Release/sharp.node
  Reason: Incompatible library version: sharp.node requires version 6001.0.0 or
later, but libglib-2.0.dylib provides version 5801.0.0
```

To fix this, you'll need to update all Gatsby plugins in the current project that depend on the `sharp` package. Here's a list of official plugins that you might need to update in case your projects uses them:

- `gatsby-plugin-sharp`
- `gatsby-plugin-manifest`
- `gatsby-remark-images-contentful`
- `gatsby-source-contentful`
- `gatsby-transformer-sharp`
- `gatsby-transformer-sqip`

To update these packages, run:

```
npm install gatsby-plugin-sharp gatsby-plugin-manifest gatsby-remark-images-
contentful gatsby-source-contentful gatsby-transformer-sharp gatsby-transformer-sqip
```

If updating these doesn't fix the issue, your project probably uses other plugins from the community that depend on a different version of `sharp`. Try running `npm list sharp` or `yarn why sharp` to see all packages in the current project that use `sharp` and try updating them as well.