

# TypeScript is ‘not worth it’ for developing libraries, says Svelte author, as team switches to JavaScript and JSDoc

By **Tim Anderson** - May 11, 2023



Svelte creator Rich Harris has made the case for switching from TypeScript to JavaScript and JSDoc, countering a trend that has seen strong growth in TypeScript usage.

A [pull request](#) for TS to JSDoc conversion attracted some puzzled comments. “Why is this change being made? I was looking around for an issue or discussion around this but I couldn’t find anything,” said one, before the issue was locked on GitHub as “too heated.”

There is further background in this [Svelte Society interview](#) last month where Harris said: “One of the things that we’ve decided to do is migrate away from TypeScript in the Svelte core codebase, and use JavaScript instead. There’s some nuance there that I failed to adequately communicate.”

“My position is, types are fantastic, TypeScript is a bit of a pain ... as soon as you use a .ts file, then you have to have the tooling to support that ... there’s all of these points of friction when you use a non-standard language like TypeScript that I have come to believe makes it not worth it. So instead, we have put all our types in JSDoc annotations, and we get all of the type safety, but none of the drawbacks, because it is just JavaScript, everything is in comments, you can just run the code. This is what we do in the Sveltekit codebase and it has worked out fantastically so

for Svelte 4.0, we're going to do the same for Svelte because it's going to enable us to move much more quickly."

Svelte/SvelteKit is not the most popular JavaScript framework but it is one of the most admired.

Developers have embraced TypeScript because they find that strong typing reduces errors, as well as improving the developer experience in code editors with features like code completion and pop-help. However, [JSDoc](#), which is primarily an API documentation tool, can also be used for type checking. Support for this is built into Visual Studio Code as documented [here](#). Developers simply add:

```
// @ts-check
```

to the top of a JavaScript file. "When types cannot be inferred, they can be specified using JSDoc comments," the documentation explains. This feature is actually powered by TypeScript, meaning that in reality TypeScript and JSDoc are working side by side.

One of the nuances that can easily be missed, though, is that Harris is mainly concerned about TypeScript in the context of library development. The switch to JSDoc is "less beneficial if you're building an app because if you're building an app, you're going to have a build step anyway. You want to optimize your code, you want to minify it, you want to bundle everything up. If you're building a library that is when I really strongly urge to use JSDoc instead."

Harris [added on Hacker News](#) that "as a user of Svelte, this won't affect your ability to use TypeScript with Svelte at all – functions exported from Svelte will still have all the same benefits of TypeScript that you're used to (typechecking, intellisense, inline documentation etc). Our commitment to TypeScript is stronger than ever."

Fluent UI proposal for Visual Studio raises fears about another botched redesign

Return of the applet? CheerpJ 3.0 runs a full JVM in HTML5 and WebAssembly

Despite 'extremely disruptive' Jakarta EE change, Red Hat releases the Quarkus 3.0 container-optimiz...

Reduce costs by 90% by moving from microservices to monolith: Amazon internal case study raises eyeb...

90 minutes from code to deployment: VS Code team describes internal development process

Interview: Next.js company Vercel adds hosted services for storage and security - but adds cost as w...

Amazon now the most popular Java Development Kit vendor for production, says observability survey

Brussels greenlights AI Act, chews over Cyber Resilience legislation

Updates for Git and Git for Windows after severe bug rash

RSA boss warns of 'scary truth': AI 'co-pilots' swooping on dev and sec jobs

KubeCon panel: Platform engineering isn't killing DevOps. It's just quietly unloading it

Linux Foundation Europe chief warns EU bill could fragment open source – and load risk onto devs