Lots to cover this month with releases from across the Svelte ecosystem. Most importantly, Svelte Summit Spring 2021 has an Open Call for Speakers. **The deadline is March 14** so if you have an idea for a talk, submit it now!

Let's dive into the news 🦕

What's new in `sveltejs/svelte`

- SSR store handling has been reworked to subscribe and unsubscribe as in DOM mode. SSR stores should work much more consistently now (**3.31.2**, see custom stores and Server-side component API)
- Multiple instances of the same action are now allowed on an element (**3.32.0**, example)
- The new `foreign` namespace should make it easier for alternative compile targets (like Svelte Native and SvelteGUI) by disabling certain HTML5-specific behaviour and checks (**3.32.0**, more info)
- Support for inline comment sourcemaps in code from preprocessors (**3.32.0**)
- Destructured defaults are now allowed to refer to other variables (**3.33.0**, example)
- Custom elements will now call `onMount` functions when connecting and clean up when disconnecting (**3.33.0**, checkout this PR for an interesting conversation on how folks are using Svelte with Web Components)
- A `cssHash` option has been added to the compiler options to control the classname used for CSS scoping (**3.34.0**, docs)
- Continued improvement to Typescript definitions

For a complete list of changes, including bug fixes and links to PRs, check out the CHANGELOG

New from `sveltejs/language-tools`

- For language server clients that don't support `didChangeWatchedFiles`, a fallback file watcher will be used instead
- New highlighting rules for store accessors and element directives (like `bind:` and `class:`)
- HTML tags can now be renamed together
- Mustache tags parsing is now more robust and will provide better intellisense in more situations

Haven't tried the language-tools yet? Check out Svelte Extension for VSCode or find a plugin for your favorite IDE!

Other changes from `sveltejs/*`

- eslint-plugin-svelte3 now supports TypeScript as of 3.1.0
- prettier-plugin-svelte released a number of minor versions to address whitespace and comment trimming issues.
- svelte-preprocess bug fixes this month include fixes to postcss transformations and support for both v2 and v3 of `postcss-load-config`
- sapper's 0.29.1 release fixes some bad imports in type definitions, updates typings to be compatible with express/polka, and restores hashing of all CSS file names.

Community Showcase

Apps & Sites

- Tracking the Coronavirus from NYTimes is an example of SvelteKit in production
- Budibase is an open-source low-code platform, helping developers and IT professionals build, automate, and ship internal tools 50x faster on their own infrastructure.
- Track the Parcel is a one-stop tool for tracking parcel status with all major package shippers.

- [Memo](#) is a replacement for email that uses Svelte for modern messaging
- [Userscripts Safari](#) is an open-source userscript editor for Safari… a native Svelte app for Mac OS!
- [SVGX](#) is "the desktop SVG asset manager designers and developers wished they had."
- [Armoria](#) is a procedural heraldry generator and editor
- [FictionBoard](#) is a virtual table top (VTT) platform that just released fillable and responsive character sheets
- [Castles & Crusades Treasure Generator](#) is a treasure generator for the table top RPG: Castles and Crusades.
- [NESBit Studio](#) is a toolkit to help the development of homebrew NES games
- [ElectroBlocks](#) is an online Arduino IDE with a built-in simulator (Chrome Only)
- [Goblin.life](#) is a 3D world builder whose UI is built with Svelte
- [farmbox](#) is a UAE-based grocery delivery services
- [heroeswearmasks.fun](#) is a client-side machine learning tool that determines whether or not you're wearing a mask.
- [weatherify](#) is a very pretty (and [open source](#)) weather app
- [DSN Live](#) lets you monitor connections between NASA/JPL and interplanetary spacecraft missions in real time.

Demos, Libraries, Tools & Components

- [spc](#) is a special characters picker component for the web
- [svelte-injector](#) lets you inject Svelte components in React, Angular, Vue, jQuery, Vanilla JS.
- [Felte](#) is a form library for Svelte with simple validation reporting.
- [svelte-use-form](#) is form library that "is easy to use and has 0 boilerplate."
- [Formula](#) provides "Zero Configuration Reactive Forms for Svelte."
- [Houdini](#) is "the disappearing GraphQL client built for Sapper and Sveltekit."
- [svelte-split-pane](#) is a draggable split pane component
- [svelte-virtualized-auto-sizer](#) is a high-order component that automatically adjusts the width and height of a single child.
- [svelte-window](#) are components for efficiently rendering large, scrollable lists and tabular data.
- [Svelte Persistent store](#) is a Svelte store that keep its value through pages and reloads
- [Svelte Dark](#) is a VSCode theme inspired by the svelte.dev REPL
- [Import Cost](#) has been updated to support Svelte libraries and help developers keep their bundle size under control.
- [Tree-sitter-svelte](#) provides tree-sitter grammar for svelte
- [Svelte Ripple](#) is a Material Design ripple effect that doesn't depend on `@material/ripple` (made by @karakara in the Svelte Discord)
- [Analog SVG Clock](#) is a great example of easing functions (made by @tonmcg in the Svelte Discord)
- [Console Log Styler](#) lets you generate a styled console log using pseudo HTML and CSS (made by @EmNudge in the Svelte Discord)
- [svelte-heroicons](#) is a handy wrapper for the Heroicons icon library
- [supabase-ui-svelte](#) are UI components for Supabase authentication

Have your own Svelte Component to share? Check out the [Components](#) page on the Svelte Society site. You can contribute by making [a PR to this file](#).

Learning Resources & Starters

- [The **unofficial** SvelteKit docs](#) were built using SvelteKit and are [open for contributions](#)
- 📦 [Svelte Store](#) course by lihautan covers the basics of Svelte Stores and best practices.
- [Svelte Events](#) by WebJeda explains how directives like `on:` can be used to listen to DOM events.
- [How to Set Up Protected Routes in Your Svelte Application](#) describes how to authenticate your users to access your routes

- [Using Fauna's streaming feature to build a chat with Svelte](#) demonstrates how to setup and configure Fauna to build a real-time chat interface with Svelte
- [Using TakeShape with Sapper](#) demonstrates how to connect the TakeShape CMS with Sapper
- [YastPack](#) is Yet Another Snowpack-Svelte-TailwindCss-Routify Template Pack
- [S2T2](#) is a Snowpack + Svelte + TailwindCSS + Typescript template
- [tonyketcham/sapper-tailwind2-template](#) is a Sapper Template w/ Tailwind 2.0, TypeScript, ESLint, and Prettier

See you next month!

Got something to add? Join us on [Svelte Society](#), [Reddit](#) and [Discord](#)!