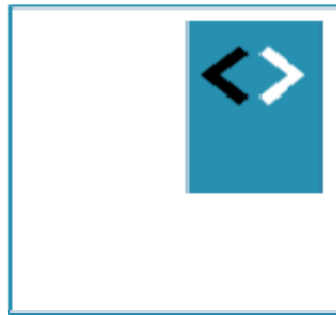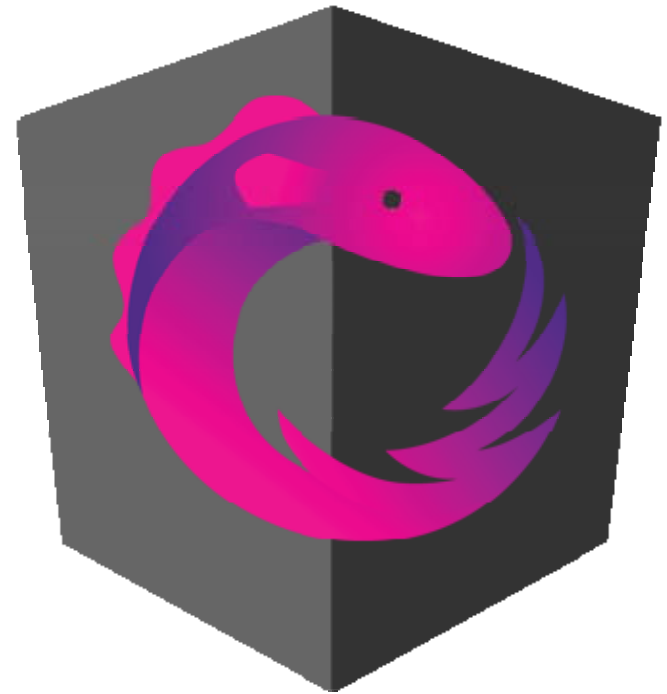# Angular Advanced @ngrx/store introspection with @ngrx/store-devtools
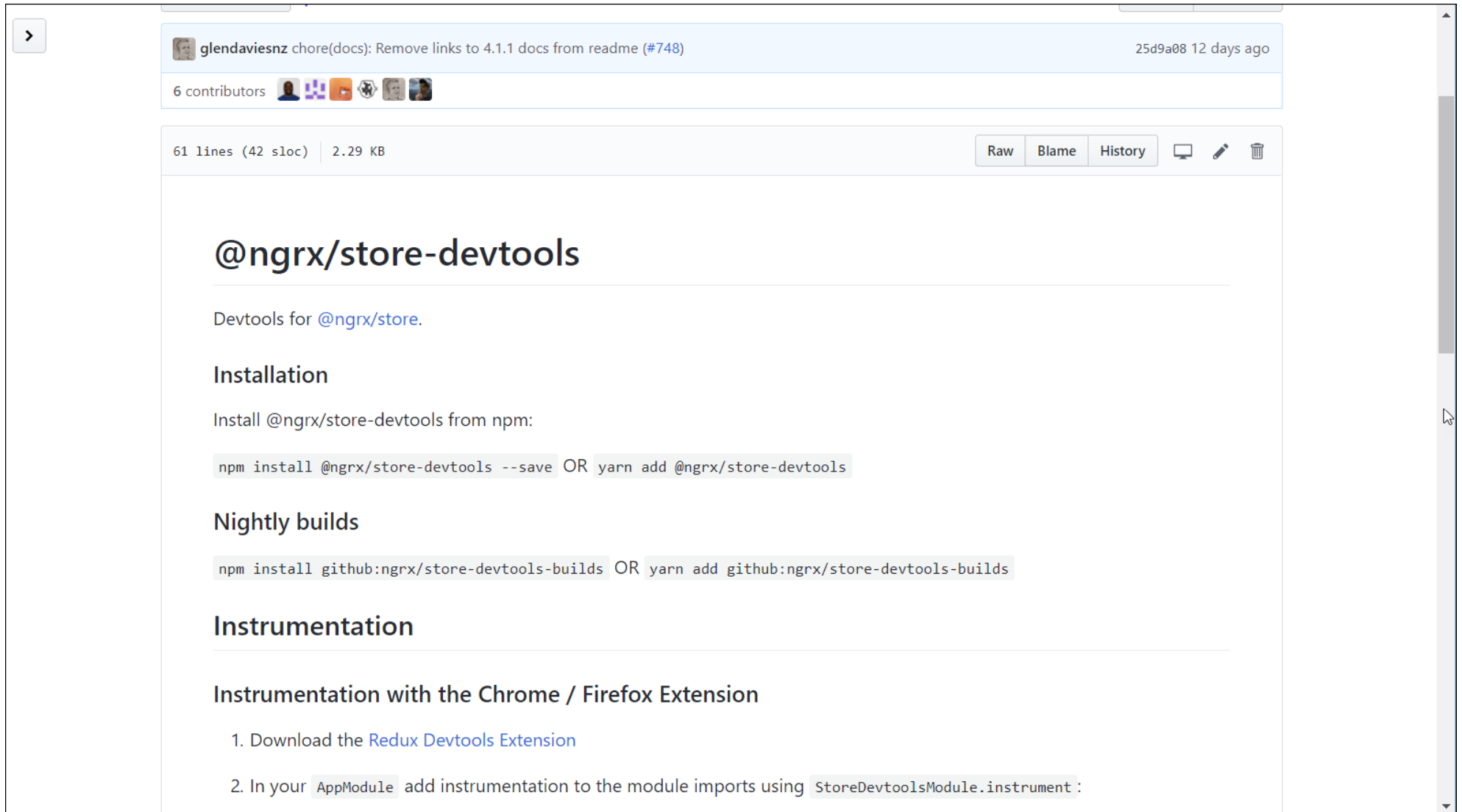
Peter Kassenaar –
info@kassenaar.com

What is @ngrx/store-devtools?

- Analyzing, inspection and debugging your State Management Store

- Time Travelling options

https://github.com/ngrx/platform/tree/master/docs/store-devtools

glendaviesnz chore(docs): Remove links to 4.1.1 docs from readme (#748)   25d9a08 12 days ago

6 contributors

61 lines (42 sloc) | 2.29 KB   Raw Blame History

@ngrx/store-devtools

Devtools for @ngrx/store.

Installation

Install @ngrx/store-devtools from npm:

`npm install @ngrx/store-devtools --save` OR `yarn add @ngrx/store-devtools`

Nightly builds

`npm install github:ngrx/store-devtools-builds` OR `yarn add github:ngrx/store-devtools-builds`

Instrumentation
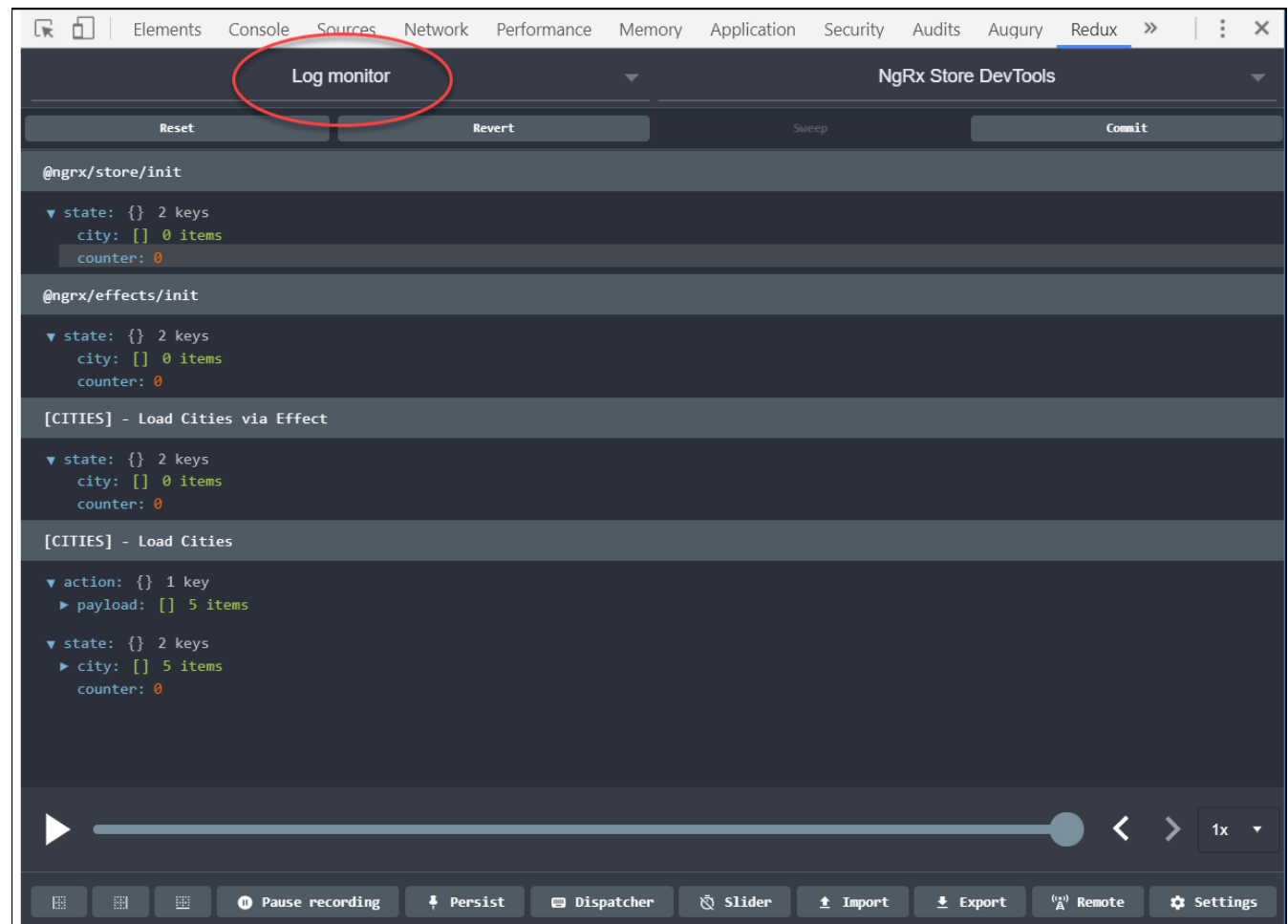
Instrumentation with the Chrome / Firefox Extension

1. Download the Redux Devtools Extension

2. In your `AppModule` add instrumentation to the module imports using `StoreDevtoolsModule.instrument` :
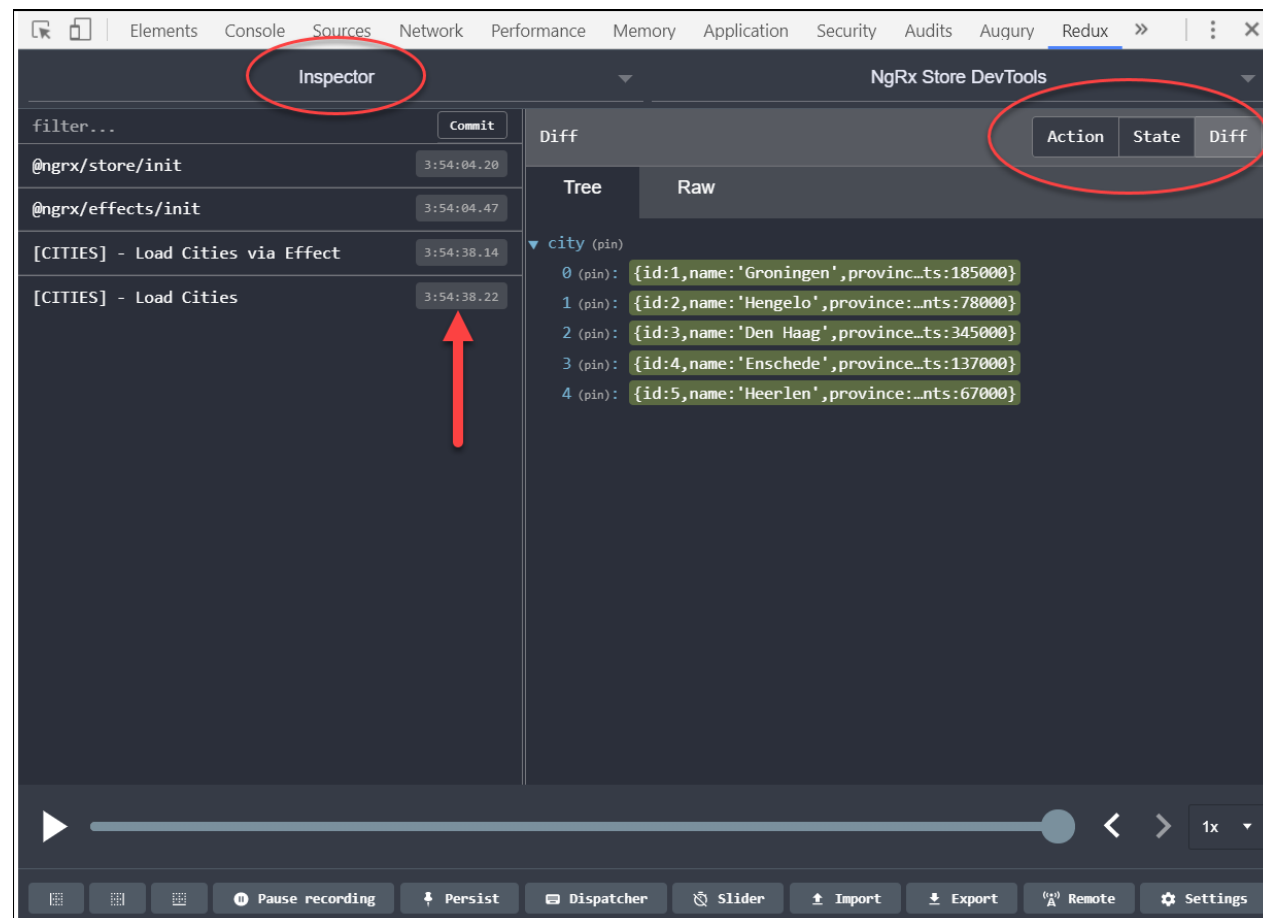
Chrome Extension
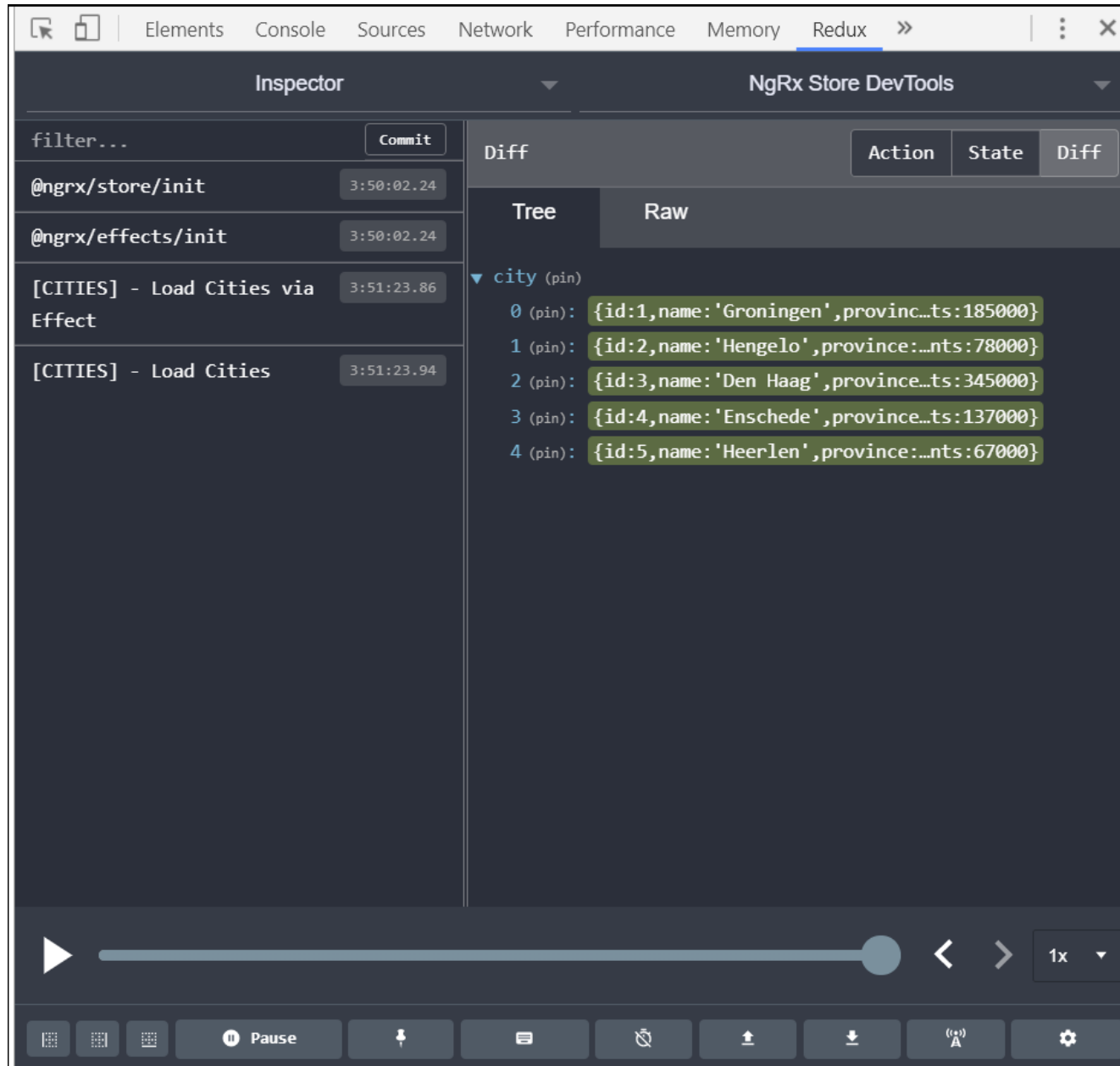
Options

- ## Log Monitor

  - Log actions from/to the store and see changes in Real Time
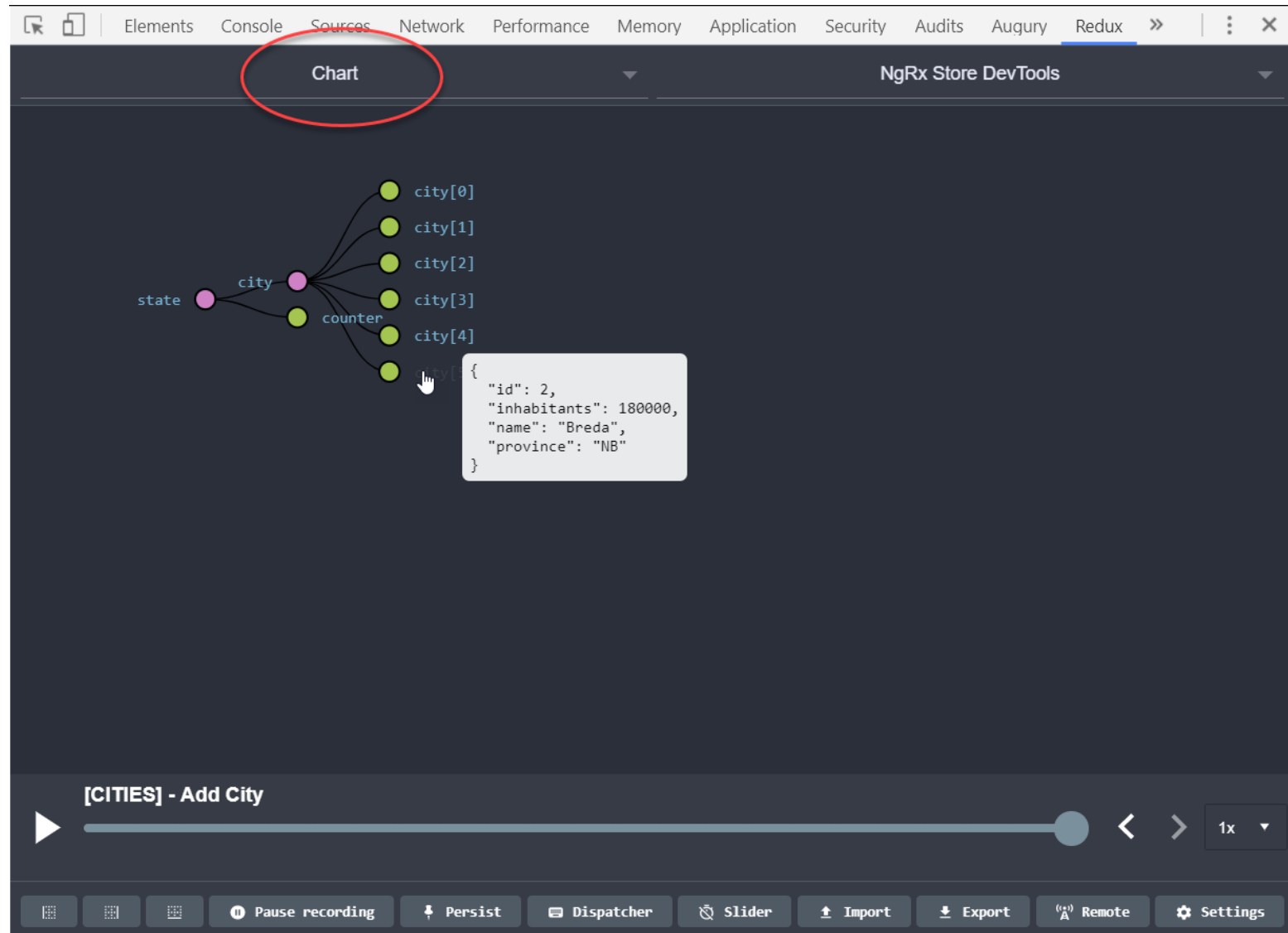
  - Reset/Revert actions

Inspector

- Filter Actions

- Jump/Skip actions

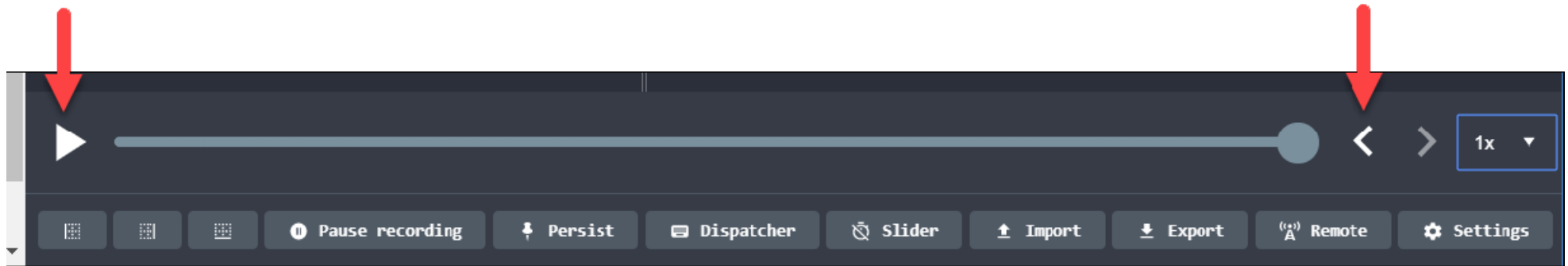- Inspect `Action`, `State` or `Diff` compared to previous State

Chart

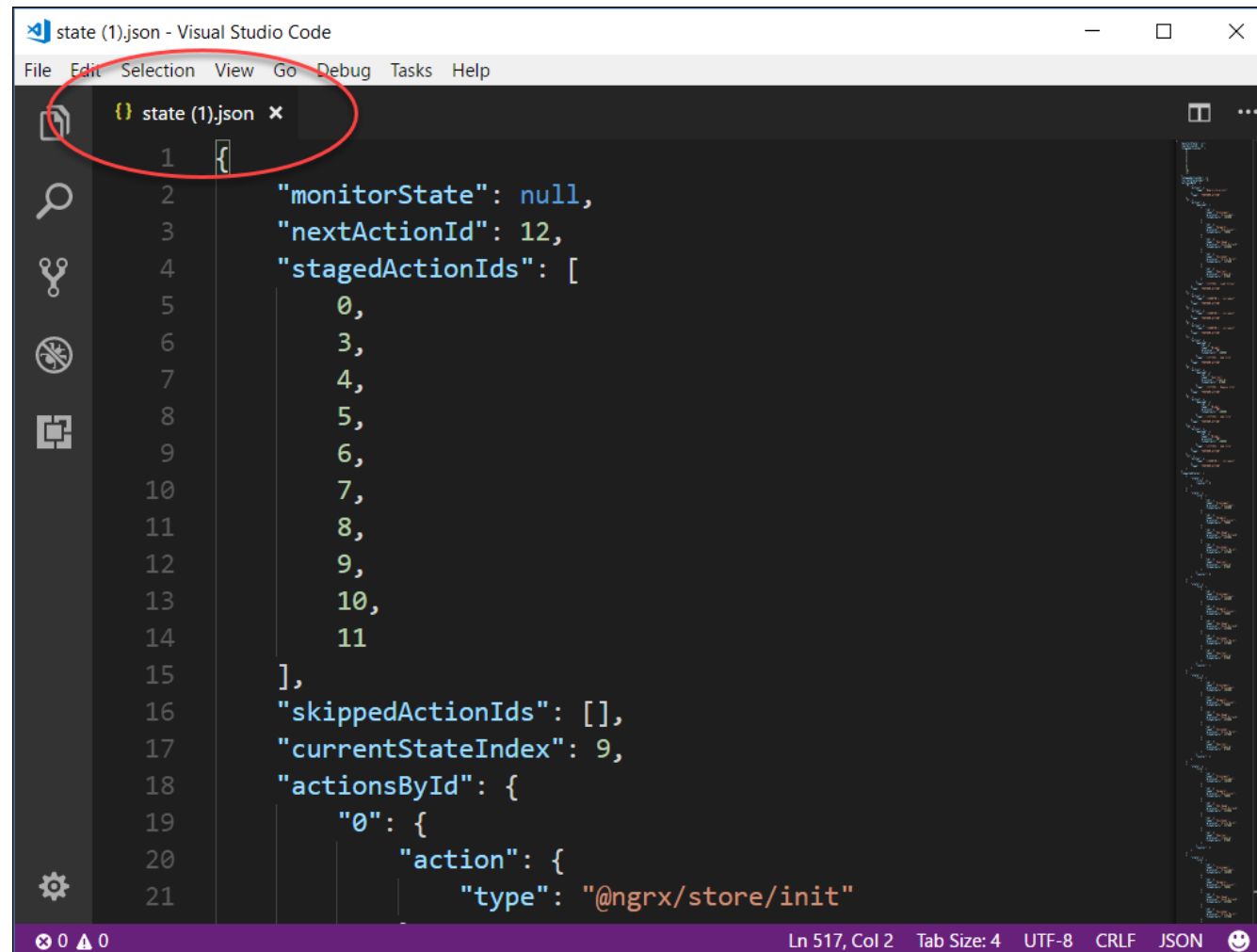- Inspect flowchart of Actions, flow of data through the Store

Slider

- (Re)play various actions and see changes in app and in state

- Jump to begin/end of timeline

Import/export State

- Save State changes to .json-file and import them to replay steps
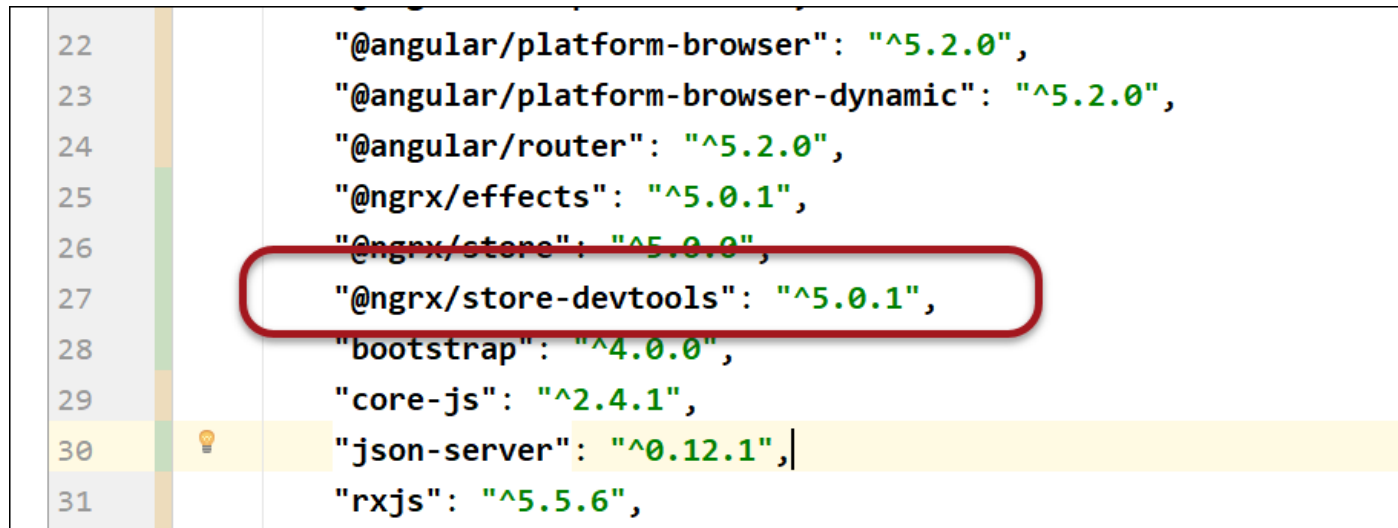
Installation

Install in project and in Google Chrome

Installation in Project

```
npm install @ngrx/store-devtools --save
```

OR:

```
yarn add @ngrx/store-devtools
```



```
22    "@angular/platform-browser": "^5.2.0",
23    "@angular/platform-browser-dynamic": "^5.2.0",
24    "@angular/router": "^5.2.0",
25    "@ngrx/effects": "^5.0.1",
26    "@ngrx/store": "^5.0.0",
27    "@ngrx/store-devtools": "^5.0.1",
28    "bootstrap": "^4.0.0",
29    "core-js": "^2.4.1",
30    "json-server": "^0.12.1",
31    "rxjs": "^5.5.6",
```

Installation in `app.module.ts`

```typescript
// app.module.ts
import {StoreDevtoolsModule} from '@ngrx/store-devtools';

@NgModule({
    …
    imports     : [
        …
        StoreDevtoolsModule.instrument({
            maxAge: 10}) // Redux Devtools debugging in Chrome
    ],
})
export class AppModule {
}
```

Install Redux DevTools

- Add from `chrome://extensions`

More info

- http://extension.remotedev.io/

- NOT specific to Angular – to be used with all Redux-like stores!
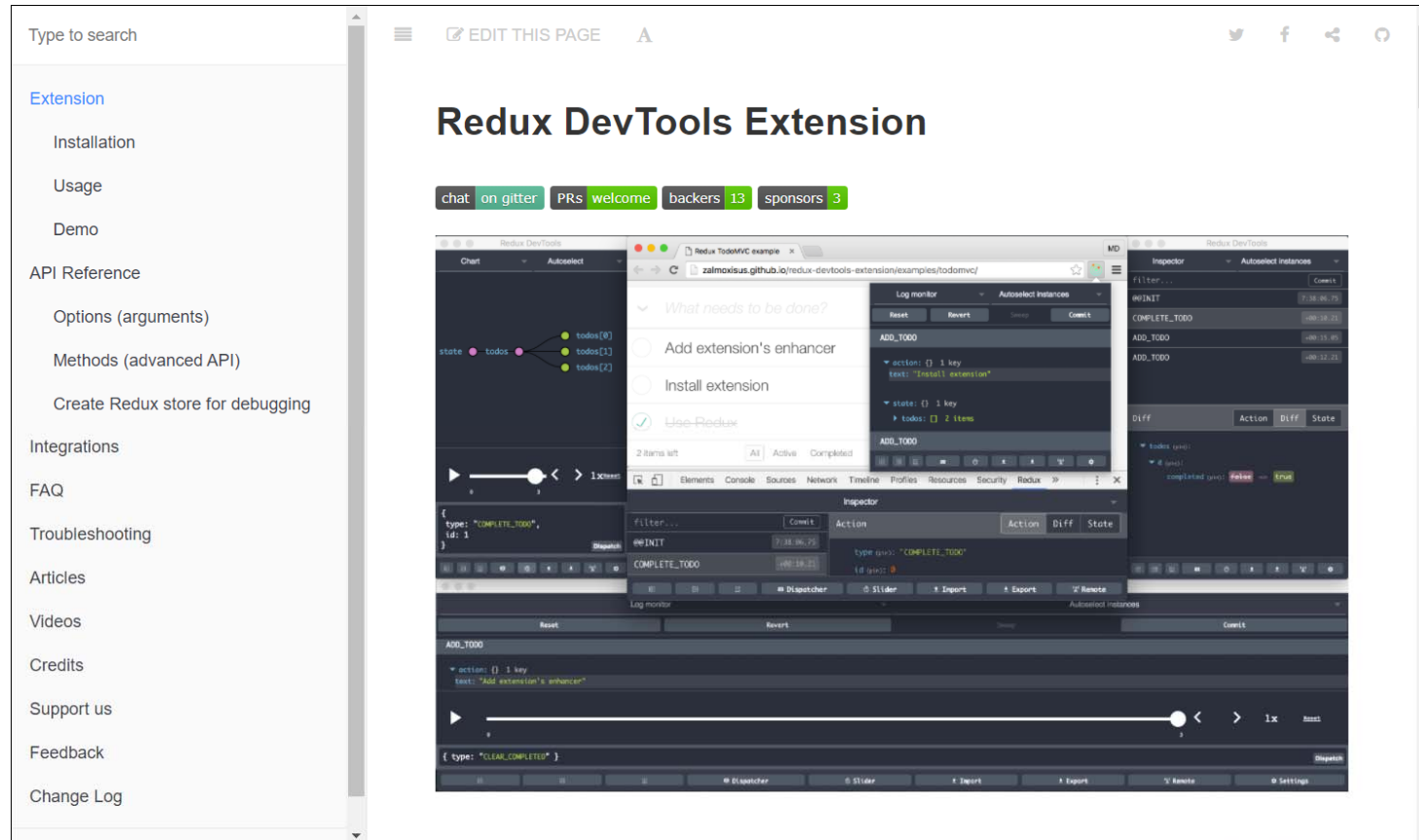
Workshop

- Use your own project, or start from one of the Store projects, for example `/200-ngrx-simple-store`

- Add Redux DevTools extension to Chrome (one-off action)

- Add `@ngrx/store-devtools` to current project via `npm` or `yarn`

- Activate devtools in `app.module.ts`

- Experiment with different views

    - Log Monitor

    - Inspector

    - Chart

- Export a State Tree to `.json` file, reload the project and import it again. See how this works