

# ToiletPaper #160

## Ein Server ist auch nur ein Stream

Autoren: Robert Gruner / Software Engineer / Standort Leipzig  
Daniel Straub / Software Architect / Business Division Media

### ✘ Problem

Der nächste Node.js Server muss her – sei es wegen eines neuen Projekts oder weil etwas in einem bestehenden System umgebaut werden soll. Es stellt sich also wieder einmal die Frage nach der Technologie-Auswahl. Ihr wollt:

- am liebsten so wenig Code wie möglich schreiben?
- TypeScript benutzen?
- auf funktionale Programmierparadigmen setzen?

### ✓ Lösung

Dann ist das Framework namens **Marble** etwas für euch! Dieses basiert auf RxJS und verfolgt somit die Idee, dass alles(!) ein Stream ist. Marble ist eine gute Alternative zu Express, Koa, Hapi und co., wenn man RxJS oder Reactive Programming bereits aus anderen Programmiersprachen kennt. Die bei RxJS im Mittelpunkt stehende **Observable**-Monade ermöglicht es, dem Railway Oriented Programming Ansatz zu folgen. Wir behandeln eine Anfrage an einen Server als Input-Stream, welcher in einen Output-Stream umgewandelt wird. Das passiert anhand von **Effects** und sieht ungefähr so aus:

### ➔ Beispiel

```
const validateRequest = requestValidator$({
  params: t.type({
    adjective: t.string,
  }),
});

const getCoffee$ = r.pipe(
  r.matchPath('/coffee/:adjective'), // make the coffee effect available at /coffee
  r.matchType('GET'), // effect for GET requests only
  r.useEffect(req$ => req$.pipe(
    validateRequest,
    // tap allows side-effect execution without changes to the stream
    tap(({ url, method, params }) => console.log({ method, url, params })),
    // process request and create response
    map((req) => ({ body: `Kaffee ist ${req.params.adjective}!` })),
  )),
);

// use via await server()
const server = createServer({ port: 1337,
  listener: httpListener({ effects: [
    combineRoutes('/', { effects: [
      getCoffee$,
      // add more endpoints here
    ] })),
  ] })),
});
```

### + Weiterführende Aspekte

- Marble.js – Getting started: <https://docs.marblejs.com/getting-started/quick-setup>
- RxJS – Intro: <https://rxjs.dev/guide/overview>
- Railway Oriented Programming – Intro: <https://medium.com/geekculture/better-architecture-with-railway-oriented-programming-ad4288a273ce>