

# Revisiting a Reactive Router with Ivy

Brandon Roberts



**Brandon Roberts**

@brandontroberts

Software Engineer at Nrwl.io

Google Developer Expert

NgRx Maintainer



## **Making Waves with Nx, Monorepos & Angular**

We're a software development firm that helps teams to build large applications using monorepos and proven development practices, plus the best talent and latest engineering practices.

## **Consulting Engineering Training**

[\*\*nrwl.io/services\*\*](https://nrwl.io/services)



hello@nrwl.io



@nrwl\_io

# Agenda

**AngularJS/Angular Routers**

**Ivy**

**Building a Router**

# Shoutouts

# AngularJS Routers

# ngRoute

`<ng-view></ng-view>`

# ui-router

`<ui-view></ui-view>`



# ui-router

- Promises
- Multiple outlets
- Resolvers/Guards
- Lifecycle events
- Just components

# Angular 2 (Beta)

# Angular V2 Router

- Promises





Oh no baby, what is you doing???



# Angular V2 Router

- Promises
- Tree of Routers
- Just components
- Backported to AngularJS

# @ngrx/router

- Observables
- Route Traversal
- Route Params/Query Params
- Route Link/Route Views
- Just components

# Angular Router V3





**Secret Meetings...**

**@ngrx/router goes to  
Vladivostok**

**Comes back as  
Angular Router  
(with NgModules)**

# Angular V3?

Later...



# Ivy Features

- **New Rendering Engine**
- **Locality**
- **New Compiler**

**What does Ivy do  
for routing?**



# The Dream

# All About Injectors

**Develop primarily with  
optional NgModules**

# **All about components**

# What does it mean?

- **Observable In -> Observable Out**
  - **Stream of URLs into a stream of components**
  - **Streams of information**
    - **Route Params/Query Params**
    - **Active Route**



Everything is a stream

# Guards

Route  
Params

Router State

# Resolvers

URL  
Serializers

# NgModules

Lazy  
Loading

Route Data

# SSR

# Outlets

Query  
Params

# Events

Location  
Strategies

# What do we need?

- History Web API
- Router (Global Data)
  - Stream of URL changes
  - URL Parser
  - Query Params/Hash
- Register Routes
- Match Routes
- Render components



# History API

- **URL Changes**
- **Push State Events**
- **Angular Location Service**

## Angular Location Service

```
export class Router {  
  constructor(private location: Location) {  
    this.location.onUrlChange(() => {  
  
    });  
    this.location.subscribe(() => {  
  
    });  
  
  }  
}
```

# URL Parser

- URL Web API
  - Path
  - Query Params
  - Hash

```
export class Router {  
  parseUrl(location) {  
    return new URL(location);  
  }  
}
```

**<ngrx-router>**

```
combineLatest(url$, routes$)  
  .pipe(  
    findMatch(),  
    setRoute()  
  )
```

**</ngrx-router>**

```
<ngrx-route path="/about" [component]="aboutComponent">
```

**<ngrx-router>**



**<ngrx-route path="/about" [component]="aboutComponent">**

**</ngrx-router>**

```
<ngrx-route path="/about" [component]="aboutComponent">  
  #outlet  
</ngrx-route>
```

```
<ngrx-route path="/about" [component]="aboutComponent">  
  ɵrenderComponent(AboutComponent)  
</ngrx-route>
```



```
<ngrx-route path="/blog/posts/:id" [component]="blogPostComponent">  
</ngrx-route>
```

**<ngrx-router>**

**<ngrx-route path="/blog/posts/:id" [component]="blogPostComponent">**  
**<ngrx-route path="/about" [component]="aboutComponent">**

**</ngrx-router>**

```
export class AboutComponent implements OnInit {  
  me$ = this.routeParams$.pipe(map(params => params['me']));  
  
  constructor(private routeParams$: RouteParams, router: Router) {}  
  
  ngOnInit(): void {  
    router.url$.subscribe();  
    router.queryParams$.subscribe();  
    router.hash$.subscribe();  
    routeParams$.subscribe();  
  }  
  
  goHome(): void {  
    this.router.go('/home');  
  }  
}
```

# Lazy Loading

```
<ngrx-route path="/about" [component]="aboutComponent">
```

**loadComponent**

```
<ngrx-route path="/about" [loadComponent]="loadAboutComponent">
```

```
export class AppComponent {  
  loadAboutComponent = () => import('./about/about.component')  
    .then(m => m.AboutComponent);  
}
```



**RouterModelRefRouterRoot([])**

# Guards

Router State

## Resolvers

URL  
Serializers

Route Data

# SSR

# Events

Location  
Strategies

# Recap

- **AngularJS/Angular Routers**
- **Ivy**
  - **All about components**
- **Observable In -> Observable Out**

**[github.com/brandonroberts/ngconf2020-reactive-router](https://github.com/brandonroberts/ngconf2020-reactive-router)**





NGRX CONF





NGRX CONF

2 Days

20 Speakers

200 Attendees

U.S. Space & Rocket Center

November 5th & 6th 2020





**coupon code: “ngconf”**



**Just Launched April 1st!**

**Speed up your build and test times in CI by 10 X!**

Sign up today and get a free 25-hour time-saving coupon!

**Nx.App**

**Consulting  
Engineering  
Training**

**[nrwl.io/services](https://nrwl.io/services)**



hello@nrwl.io



@nrwl\_io



**Thank You**