

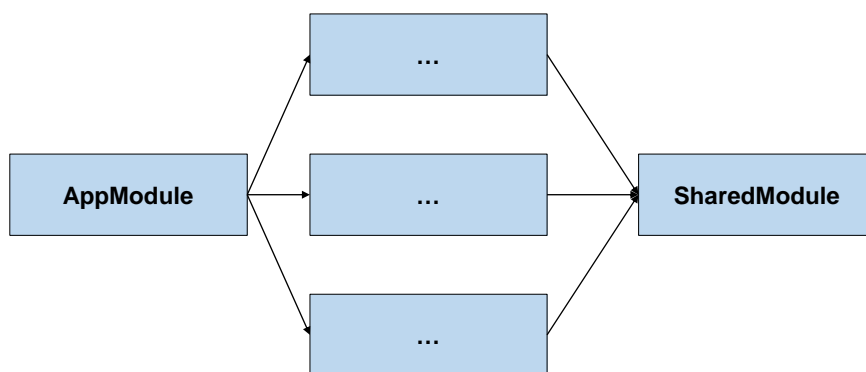


 ManfredSteyer

Architectures for huge
Enterprise Applications with Angular

Manfred Steyer
SOFTWAREarchitekt.at

Typical Module Structure



Root Module

Feature Modules

Shared Module

Contents

- (npm-)Packages
- Monorepos

 @ManfredSteyer



Structure of an npm Package

- /node_modules
- your-stuff
- package.json

 @ManfredSteyer

<https://goo.gl/hjt7G3>



Angular Package Format

 @ManfredSteyer




Angular Libraries with ng-packagr

From Components to Angular Package Format.

David Herges

 @ManfredSteyer



Angular CLI 6+
supports
Packages
using
ng-packagr

Create Library with CLI ≥ 6

```
npm install -g @angular/cli
```

```
ng new lib-project
```

```
cd lib-project
```

```
ng generate library logger-lib
```

```
ng generate application playground-app
```

```
ng serve --project playground-app
```

```
ng build --project logger-lib
```

 @ManfredSteyer

Folder Structure

```

└─ node_modules
└─ projects
  └─ logger-lib
  └─ playground-app
  └─ playground-app-e2e
  └─ src
    ├── angular.json
    ├── package-lock.json
    ├── package.json
    ├── tsconfig.json
    └── tslint.json
  
```




Publishing to npm Registry

- Increment version in package.json
 - npm version [patch | minor | major | version]
- ng build --project logger-lib
- npm publish --registry <http://localhost:4873>
- npm install --registry <http://localhost:4873>

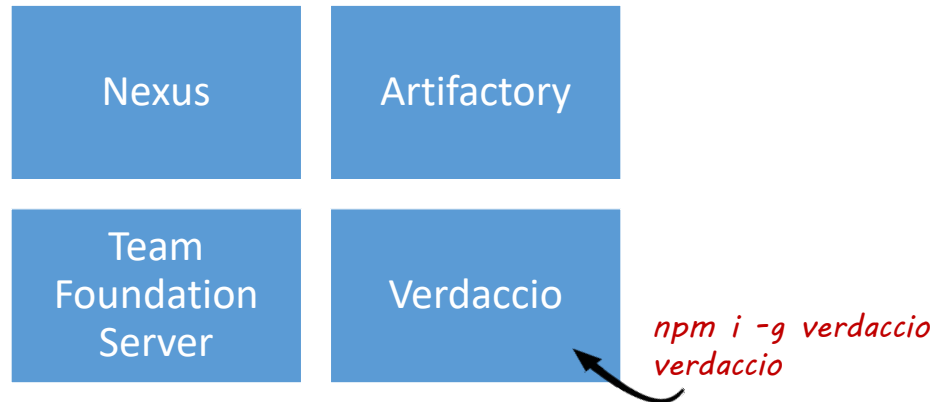
 @ManfredSteyer

Alternatives for setting the Registry

- Global: npm set registry <http://localhost:4873>
 - Default: registry.npmjs.org
 - npm get registry
- Project: .npmrc in project root 

 @ManfredSteyer

npm Registries



 @ManfredSteyer

DEMO

 @ManfredSteyer


Advantages

- Distribution
- Versioning
- Decoupling between lib authors and app authors

 @ManfredSteyer

Disadvantages

- Distribution
 - Versioning
 - Decoupling between lib authors and app authors
- ;-)

 @ManfredSteyer

Disadvantages

Distribution


- Annoying within project
- Prevents getting further libs

Versioning

- Old versions
- Conflicts
- How to force devs to use latest version?

Decoupling

- What if lib authors == app authors?

 @ManfredSteyer



Monorepo Structure

```
└─ node_modules
└─ projects
    └─ flight-admin
    └─ flight-api
    └─ flight-app
    └─ validation
    .gitignore
    angular.json
    package-lock.json
    package.json
```

Advantages

Everyone uses the latest versions

No version conflicts

No burden with distributing libs

Creating new libs: Adding folder

Experience: Successfully used at Google, Facebook, ...

Two Flavors

Project Monorepo

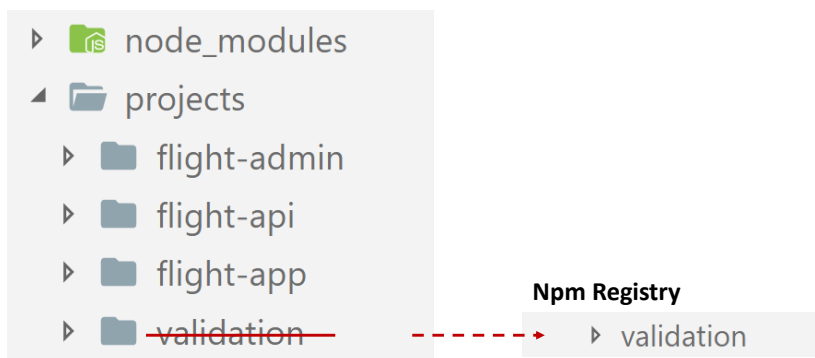
- Like Workspaces/Solutions in different IDEs


Company-wide Monorepo

- E. g. used at Google or Facebook

 @ManfredSteyer

Moving back and forth



 @ManfredSteyer

Moving back and forth



 @ManfredSteyer

Tooling & Generator

<https://nrwl.io/nx>

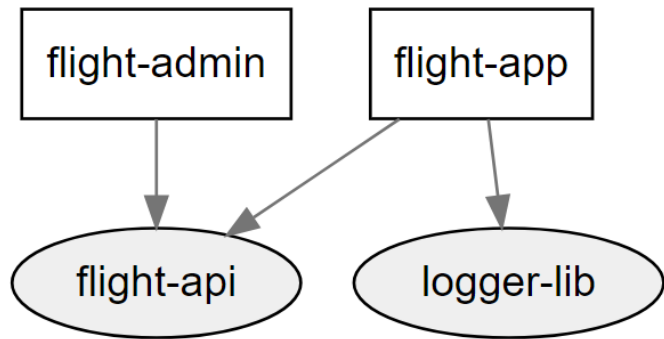


Nrwl Extensions for Angular

An open source toolkit for enterprise Angular applications.

 @ManfredSteyer

Visualize
Module
Structure



 @ManfredSteyer

DEMO



 @ManfredSteyer