Angular Dependency Injections

John Papa

@john_papa| www.johnpapa.net

Protecting Your Code with Gulp

Mangling code renames variables

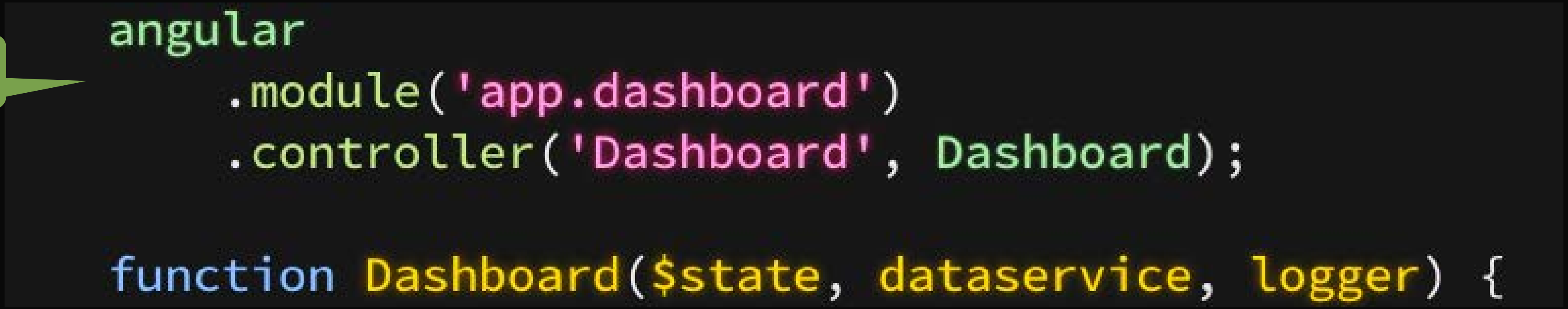
Angular uses variable names

Service location / dependency injection

Manual injections solve this

Automated injection can provide security blanket

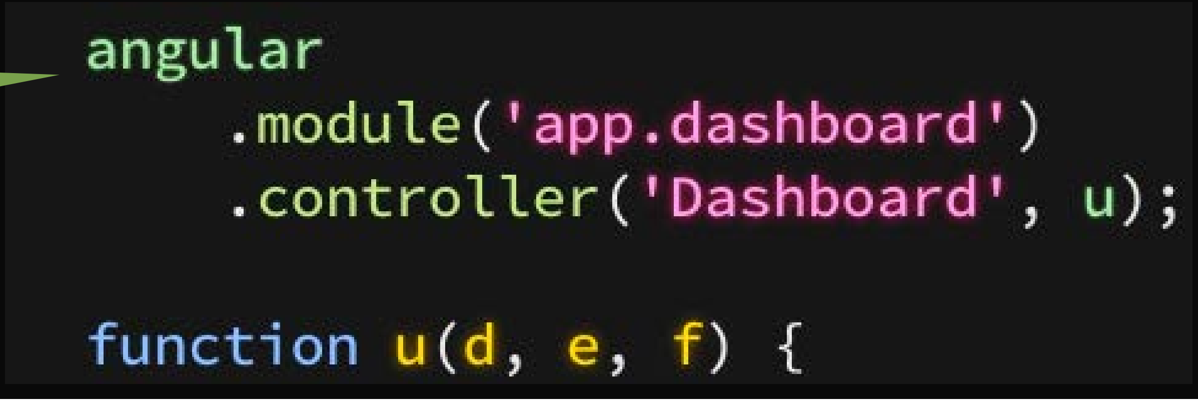
Code



```
angular
    .module('app.dashboard')
    .controller('Dashboard', Dashboard);

function Dashboard($state, dataservice, logger) {
```

Mangled



```
angular
    .module('app.dashboard')
    .controller('Dashboard', u);

function u(d, e, f) {
```

Mangling JavaScript

Variables are “mangled”

Angular can no longer locate $state, dataservice or logger

Adding Angular's Dependency Injection Annotations

```
$ npm install gulp-ng-annotate --save-dev
```

https://www.npmjs.com/package/ng-annotate

https://www.npmjs.com/package/gulp-ng-annotate

gulp-ng-annotate

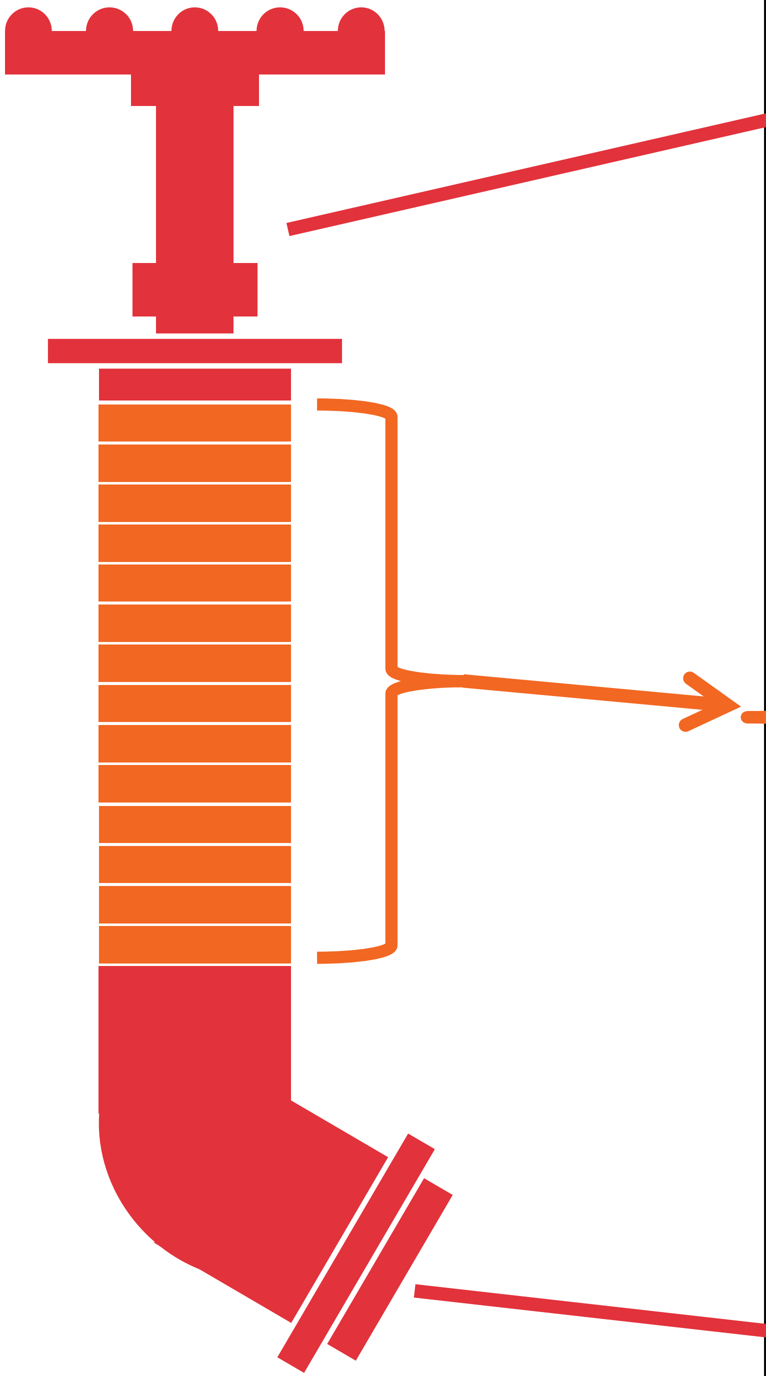
Searches for dependency injection

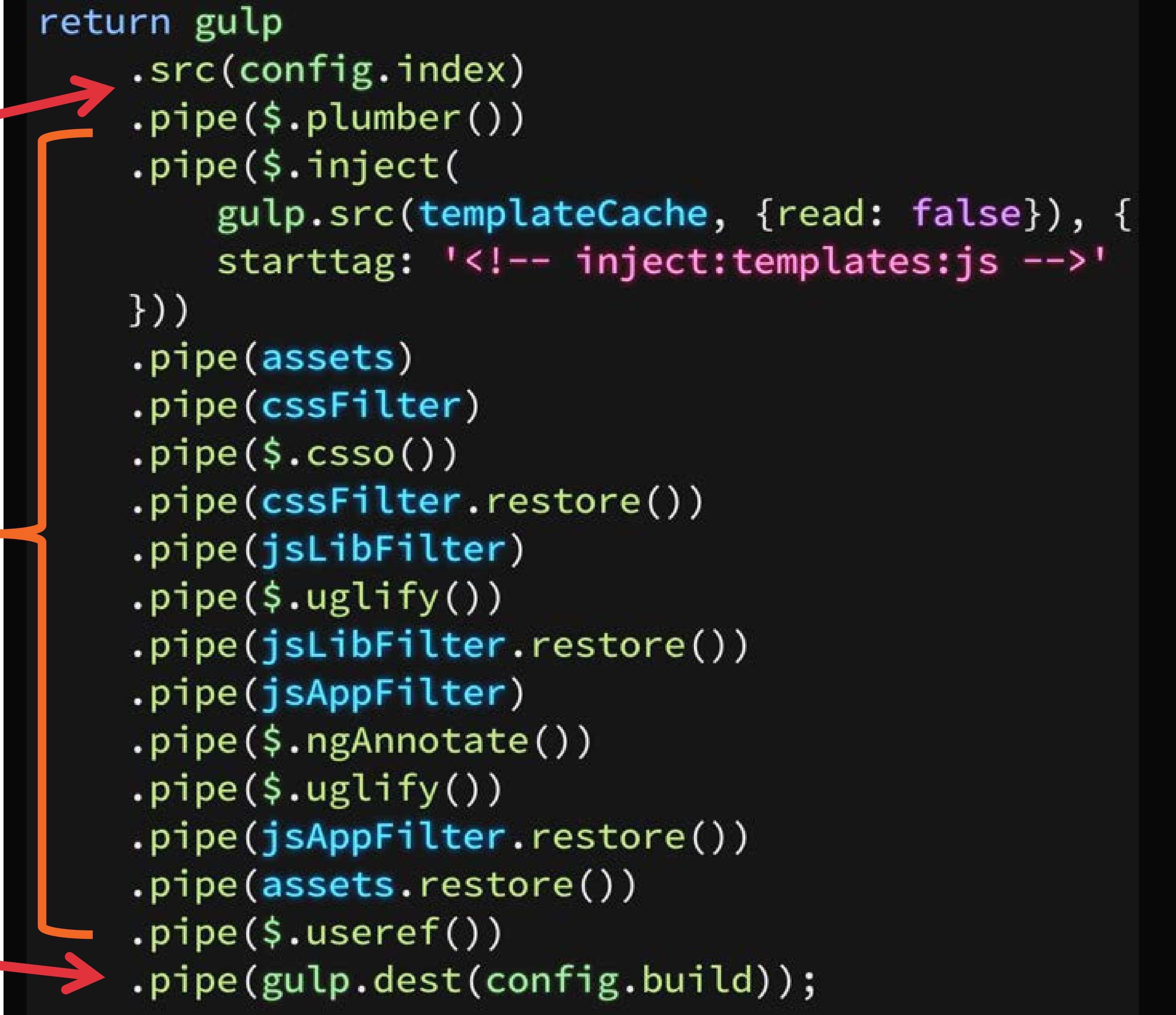
Add injection code (if not found)

Options allow some control

gulp-ng-annotate Options

- add
- remove
- single_quotes





```
return gulp
    .src(config.index)
    .pipe($.plumber())
    .pipe($.inject(
        gulp.src(templateCache, {read: false}), {
        starttag: '<!-- inject:templates:js -->'
    }))
    .pipe(assets)
    .pipe(cssFilter)
    .pipe($.csso())
    .pipe(cssFilter.restore())
    .pipe(jsLibFilter)
    .pipe($.uglify())
    .pipe(jsLibFilter.restore())
    .pipe(jsAppFilter)
    .pipe($.ngAnnotate())
    .pipe($.uglify())
    .pipe(jsAppFilter.restore())
    .pipe(assets.restore())
    .pipe($.useref())
    .pipe(gulp.dest(config.build));
```

Recap

mangling	Reduces file size
gulp-ng-annotate	Adds Angular injections
/* @ngInject */	Provide code hints