AngularJS & RequireJS

twitter: @ImSeanThompson

github: seanmthompson

ROOT Metrics®

What are we talking about here?

- Maintainable, organized app structure
- Lazy loading controllers and directives per route at runtime
- Small components via reusable directives
- Separation of concerns and single responsibility

Where I came from...

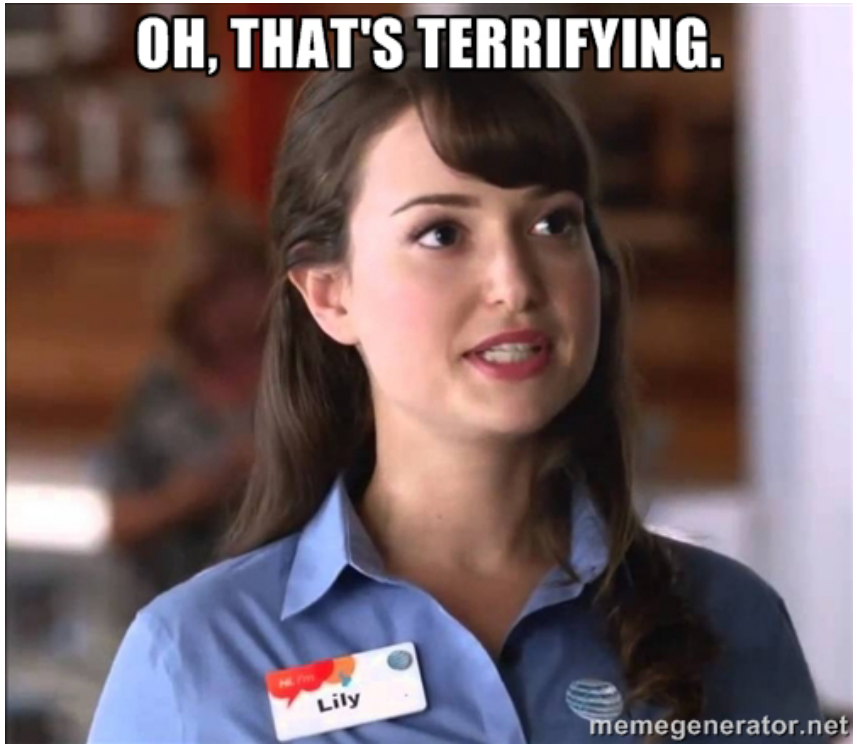
Anyone been here?

```
    <!-- build:js scripts/modules.js -->
    <script src="build/js/rsr.libs.min.js"></script>
    <script src="resources/icon_config.js"></script>
    <script src="services/helpers.js"></script>
    <script src="app.js"></script>
    <script src="controllers/modal.js"></script>
    <script src="controllers/mapModal.js"></script>
    <script src="controllers/list.js"></script>
    <script src="controllers/report.js"></script>
    <script src="controllers/map.js"></script>
    <script src="services/services.js"></script>
    <script src="filters/filters.js"></script>
</body>
</html>
```

Or here?

```
app.directive('navigation', ['$translate', '$timeout', function($translate, $timeout, $state) {
    return {
        restrict: 'E',
        templateUrl: '/_themes/root/angular/views/nav.html',
        link: function(scope) {
            scope.language = $translate.proposedLanguage() || $translate.use();

            scope.setLanguage = function(lang) {
                $translate.use(lang);
                scope.language = $translate.proposedLanguage() || $translate.use();

            };

            $timeout( function() {
                $(".global-nav-toggle").click(function() {
                    $(this).toggleClass("active");
                    $(".global-nav").toggleClass("show");
                });

                $(".country-switcher").click(function(e) {
                    e.preventDefault();
                    var val = $(this).attr("id");
                    window.location.href = window.location.href.replace(/(us|uk|en-ca|fr-ca)/, val).split("?")[0];
                });
            });
        }
    }
}]);


app.directive('airportDetails', [ function() {
    return {
        restrict: 'E',
        templateUrl: '/_themes/root/angular/views/airport.html'
    }
}]);

app.directive('venueDetails', [ function() {
    return {
        restrict: 'E',
        templateUrl: '/_themes/root/angular/views/venue.html'
    }
}]);

app.directive('defaultReport', [ function() {
    return {
        restrict: 'E',
        templateUrl: '/_themes/root/angular/views/default-report.html'
    }
}]);

app.directive('reportDetails', [ function() {
    return {
        restrict: 'E',
        templateUrl: '/_themes/root/angular/views/report-details.html'
```

What we had..



- One file for directives, one file for controllers
- Business rules, presentation rules all crammed in huge unreadable files
- A million load scripts in our index.html file

To the rescue!

Your new best friends...

- $controllerProvider.register
- $compileProvider.directive
- $filterProvider.register
- $provide.factory
- $provide.service

How does it work?

Basic outline:

- Main app file defines app-level dependencies, and loads them at initial run time.

- App bootstrapped using angular.bootstrap

- Routes/states defined and loaded when route/state activated.

- Route/state level dependencies use Angular providers to register controllers and directives

- Can still call resolves and services at route activation.

CODE DEMO!

App Config:

```
var app = angular.module('App', dependencies, function () { });

app.config(function ($locationProvider, $stateProvider, $urlRouterProvider, $controllerProvider, $compileProvider, $filterProvider, $h

    // save references to the providers
    app.lazy = {
        controller: $controllerProvider.register,
        directive: $compileProvider.directive,
        filter: $filterProvider.register,
        factory: $provide.factory,
        service: $provide.service
    };
```

Controller syntax:

```
define([], function () {
    var controllerName = 'HomeCtlr';
    var fn = function ($rootScope, $scope) {

    };

    angular.module('App').lazy.controller(controllerName, fn);
});
```

Directive syntax:

```
define([], function () {
    var directiveName = 'sampleDir';
    var fn = function ($window) {
        return {
            restrict: 'C',
            templateUrl: 'js/directives/Sample.html',
            link: function ($scope, element, attrs) {

            }
        }
    };

    angular.module('App').lazy.directive(directiveName, fn);
});
```

Github:

https://github.com/seanmthompson/Angular-RequireJS-Starter