Mapping Apps w/ Angular JS

Patrick Arlt

Experience Developer - Esri Portland R&D Center

@patrickarlt - http://bit.ly/1cGPDLf

- directives?
- transclusion?


WHAT A PIECE OF JUNK

Web Components

Extend the vocabulary of HTML

- Shadow DOM
- HTML Templates
- Custom Elements
- HTML Imports

```
<!DOCTYPE html>
<html>
  <head>
    <title>Mind Blown</title>
    <link rel="import" href="esri-maps.html">
  </head>
  <body>
    <esri-map id="map"center="45.528, -122.680" zoom="15">
      <esri-feature-layer id="parks" url="http://services.arcgi
    </esri-map>

    <script type="text/javascript">
      var map = document.getElementById("map");

      map.on("click", function(e){
        console.log("map was clicked on");
      });

      var parks = document.getElementById("parks");

      parks.on("click", function(feature){
        console.log("you clicked on " + feature.name);
      });
    </script>
  </body>
```

So This Angular JS Thing...

Angular is the only production-ready way to do ANYTHING like Web Components

Angular also has a TON of features for building web apps

Angular JS Features

Data Binding

Deep Linking

Directives

Embeddable

Controller

Form Validation

Reusable Components

Injectable

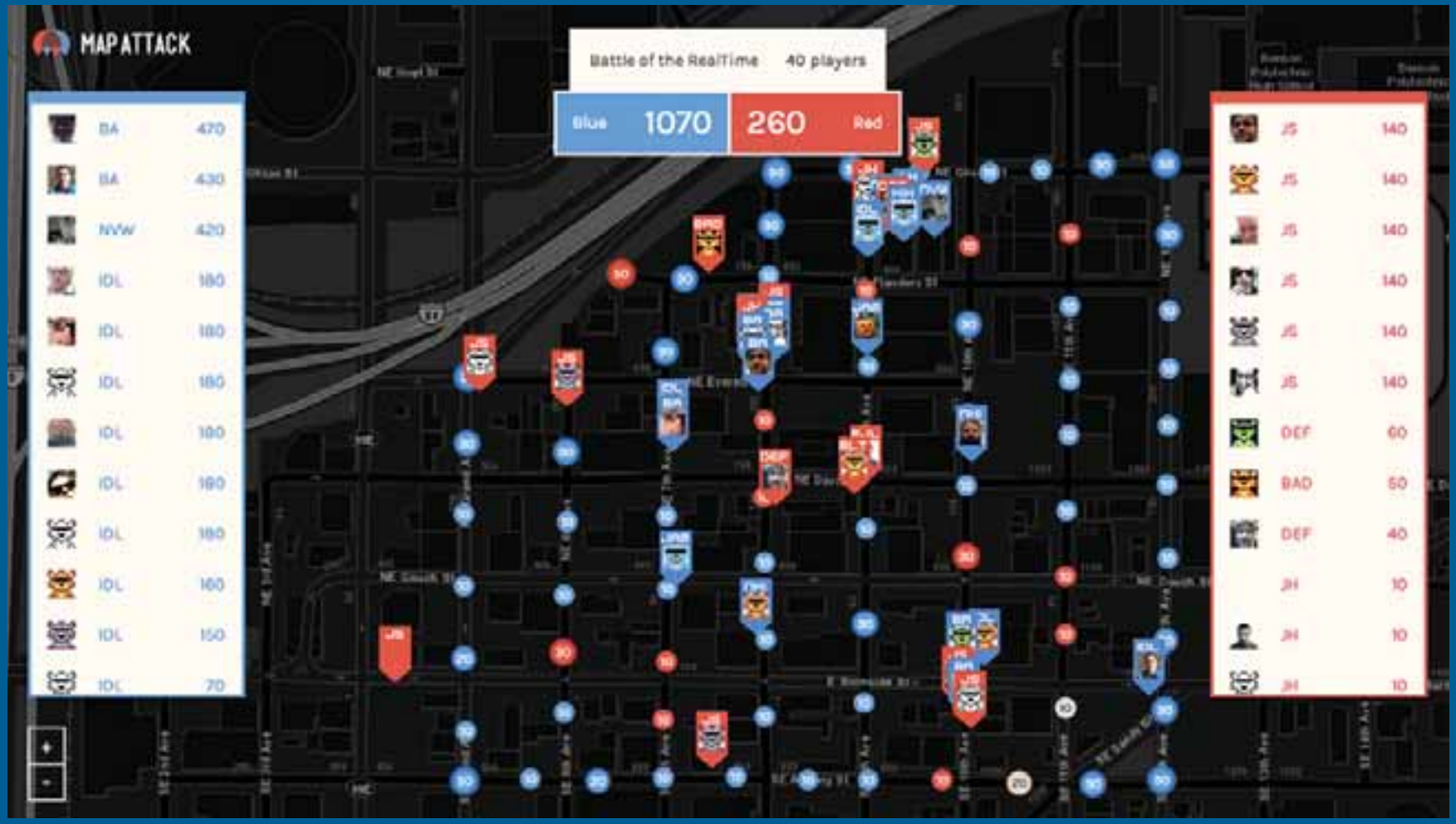
Plain JavaScript

Server Communication

Localization

Testable

My First Angular App - Map Attack



~275 lines of code - http://mapattack.org

How Angular JS Works

How Angular Works - App

Angular has its own system for scoping and namespaceing modules.

```
var app = angular.module('todo', []);
```

```
<div ng-app="todo">
  ...
</div>
```

How Angular Works - Controllers and $scope

$scope represents our "Model" and how we interact with our data

Controllers bind our $scope to chunks of DOM

```
<div ng-app>
  <div ng-controller="MainCtrl">
    {{title}}
  </div>
</div>
```

```
function MainCtrl($scope){
  $scope.title = "Hello World";
}
```

http://jsfiddle.net/patrickarlt/qv2v5/

How Angular Works - Expressions

Expressions are simple code peices that Angular can parse and evaluate

- `{{ title }}` is an expression.
- So is `{{ "Angular Says " + title }}`
- Expressions are ALWAYS evaluated against the current `$scope`

How Angular Works - Directives

Directives tell Angular HOW to bind $scope to the DOM

```
<div ng-app="">
    <div ng-controller="MainCtrl">
        <h1 ng-class="{big: isBig}">{{title}}</h1>
    </div>
</div>
```

```
function MainCtrl($scope){
    $scope.isBig = true;
    $scope.title = "Hello World"
}
```

http://jsfiddle.net/patrickarlt/5CRkr/1/

How Angular Works - Data Binding

Bind data from `$scope` to DOM and vica-versa

```
<div ng-app="">
    <div ng-controller="MainCtrl">
        <input type="text" ng-model="title">
        <h1>{{title}}</h1>
    </div>
</div>
```

```
function MainCtrl($scope){
    $scope.title = "Hello World"
}
```

http://jsfiddle.net/patrickarlt/w3ZU3/

How does Angular know the scope changed!?!?!

How Angular Works - Dirty Checking

Angular compares the new $scope the old $scope

If $scope changed, directives will handle changing the DOM

http://stackoverflow.com/questions/9682092/databinding-in-angularjs

How does Angular know when to do dirty checking?

How Angular Works - Digest Loop

First lets break Angular…

```
<div ng-app>
  <div ng-controller="MainCtrl">
    <h1>{{title}}</h1>
  </div>
</div>
```

```
function MainCtrl($scope){
  $scope.title = "Hello World"

  setTimeout(function(){
    $scope.title = "Angular is Magic!";
  }, 500);
}
```

http://jsfiddle.net/patrickarlt/u5T9v/

How Angular Works - `$scope.$apply`

Tells Angular you are changing the `$scope`.

```
<div ng-app>
  <div ng-controller="MainCtrl">
      <h1>{{title}}</h1>
  </div>
</div>
```

```
function MainCtrl($scope){
  $scope.title = "Hello World"

  setTimeout(function(){
    $scope.$apply(function(){
      $scope.title = "Angular is Magic!";
    });
  }, 500);
}
```

How Angular Works - `$scope.$apply`

Most of the time you don't have to call `$scope.$apply`. Just use Angulars helpers.

```
function MainCtrl($scope, $timeout){
    $scope.title = "Hello World"

    $timeout(function(){
        $scope.title = "Angular is Magic!";
    }, 500);
}
```

http://jsfiddle.net/patrickarlt/MzjL6/

One last $scope.$apply example

```
var socket = io.connect('http://api.mapattack.org:8000');

socket.on('data', function(payload){
  $scope.$apply(function(){
    $scope.gameState = payload;
  });
});
```

Think of $scope.$apply as APPLYING changes to the SCOPE

A Simple Angular App

http://jsfiddle.net/patrickarlt/LS2Vp/4/

I thought this was about maps...

Fine.
Lets Make
<esri-map>

But first we have to make Angular and Dojo play nice

The Good

- Angular has modules
- Dojo has modules
- They will work togather

The Bad

- Angular wants to manages module initalization via dependency injection
- Dojo wants to manage module loading via AMD

The Ugly

- No good docs on combining the module systems
- Most examples are overly complex or broken

Angular JS + JS API Setup on GitHub

Custom Directives

Warning! Writing your own directives are a HUGE topic!

```
<div ng-controller="MapCtrl">
  <esri-map id="map" center="-73.97163391113281,40.705790602241
    <esri-feature-layer url="http://services.arcgis.com/rOo16Hc
    <esri-feature-layer url="http://services.arcgis.com/rOo16Hc
  </esri-map>
</div>
```

esri-map | esri-feature-layer

Putting All Togather

Final Demo