


# WebGIS Workshop

(Web-) Entwicklung | Julian Kissling



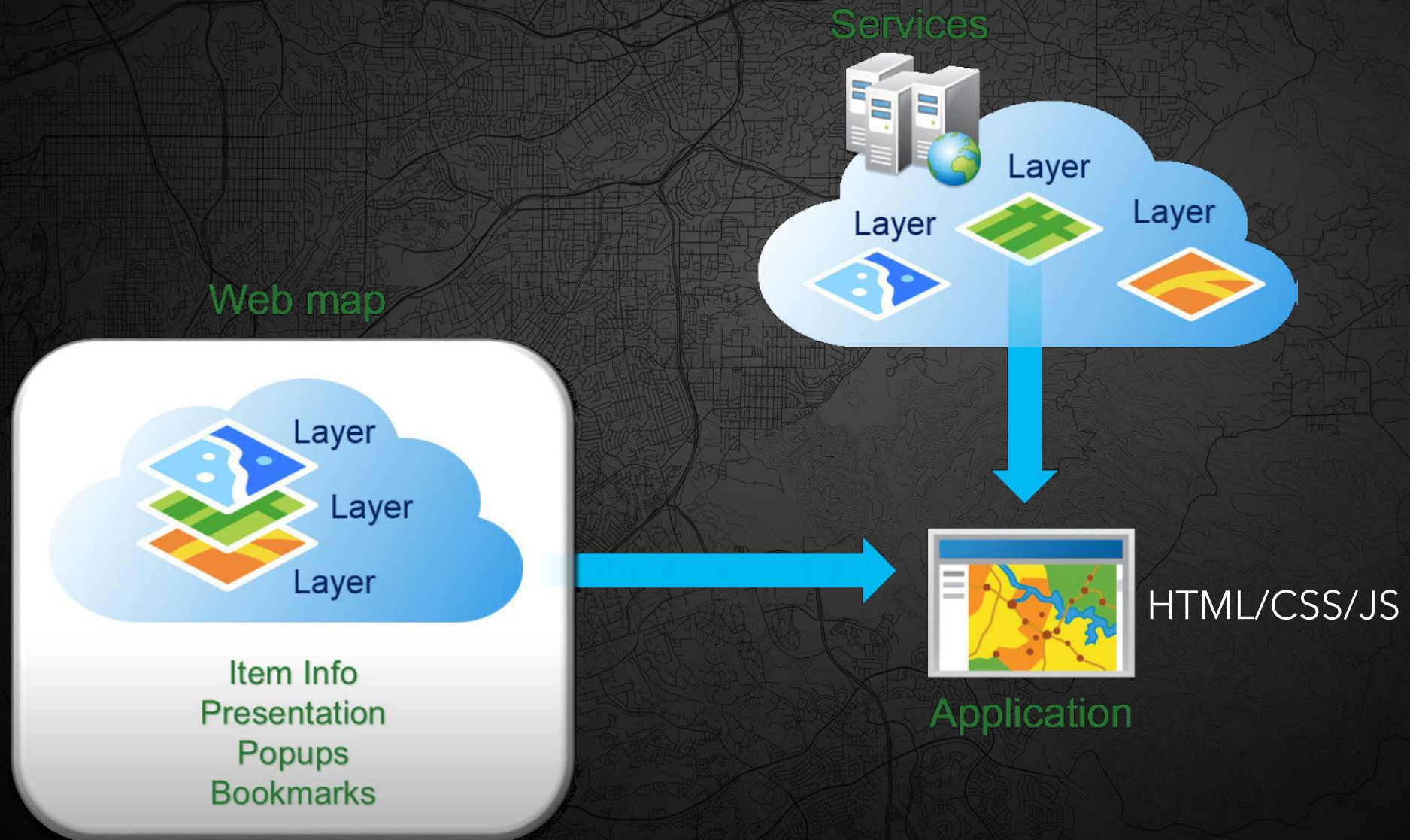


Präsentation

<https://github.com/jkissling/presentations>



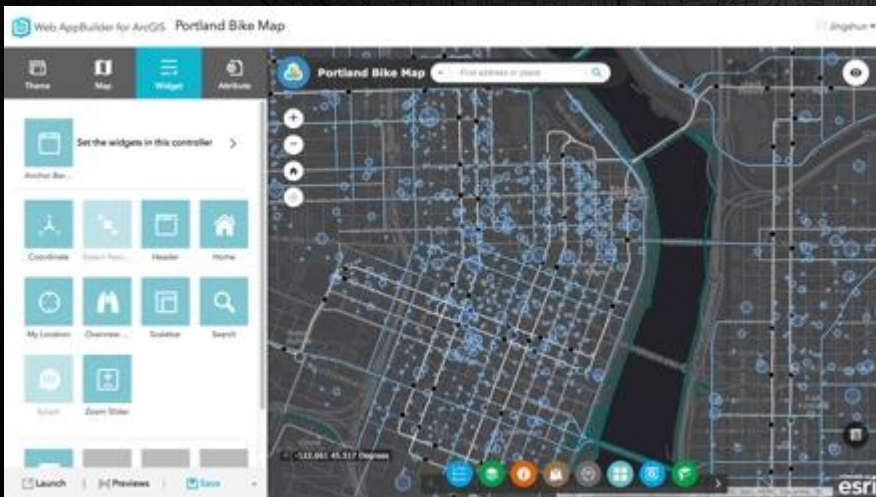
# Aufbau einer Web App





# Web Apps

Konfigurierbare  
Apps



Eigenentwicklung

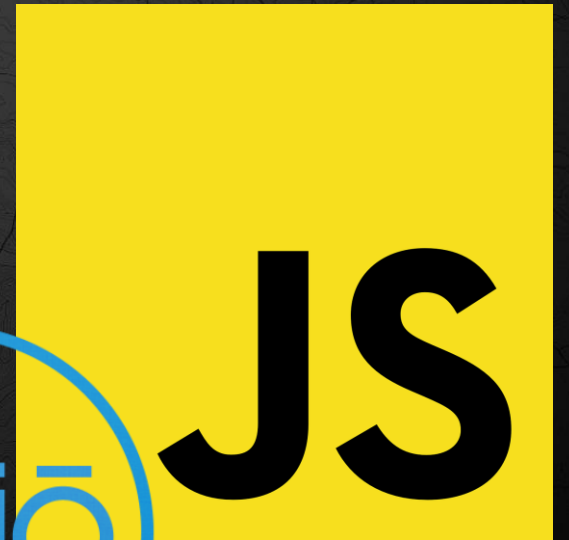


```
require(function(M) {  
  var myMap = new Map("divMap",  
    basemap : "streets"  
  });
```



# ArcGIS API for JavaScript

- Zwei Versionen 3x und 4x
- 4 Releases pro Jahr (April, Juli, September, Dezember)
  - Zusammen mit AGOL und WAB
- Basierend auf Dojo
  - AMD Module
  - Wir laden was wir brauchen



# ArcGIS API for JavaScript

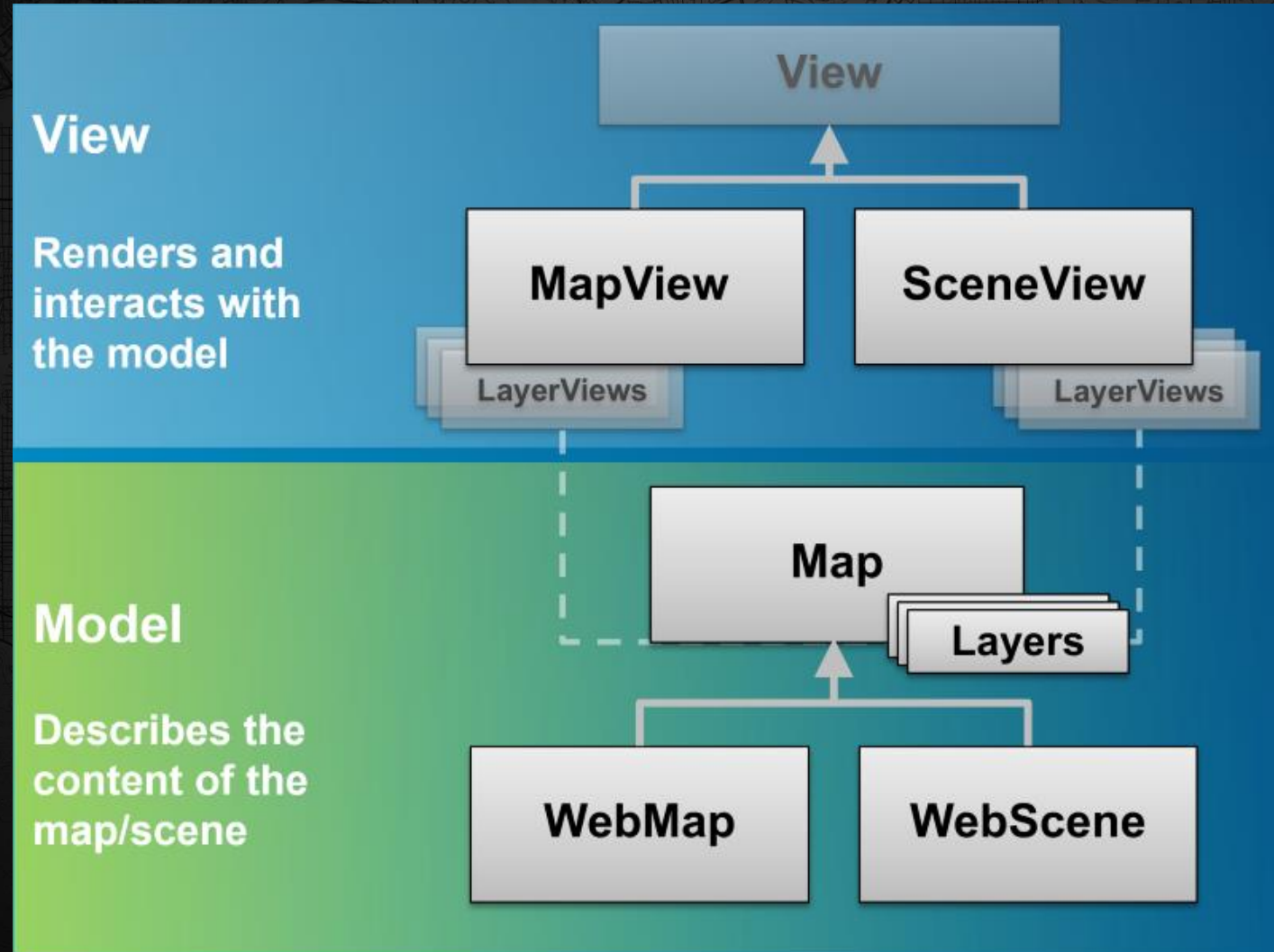
3x	4x
<ul style="list-style-type: none"><li>• Time Support</li><li>• Full Editing and Sketching</li></ul>	<ul style="list-style-type: none"><li>• Observable Properties (<i>.watch()</i>)</li><li>• ES Promises (<i>async/await</i>)</li><li>• WebGL</li><li>• 3D Support</li><li>• Map Rotation</li><li>• Autocasting</li></ul>

→ 4x aktive Weiterentwicklung

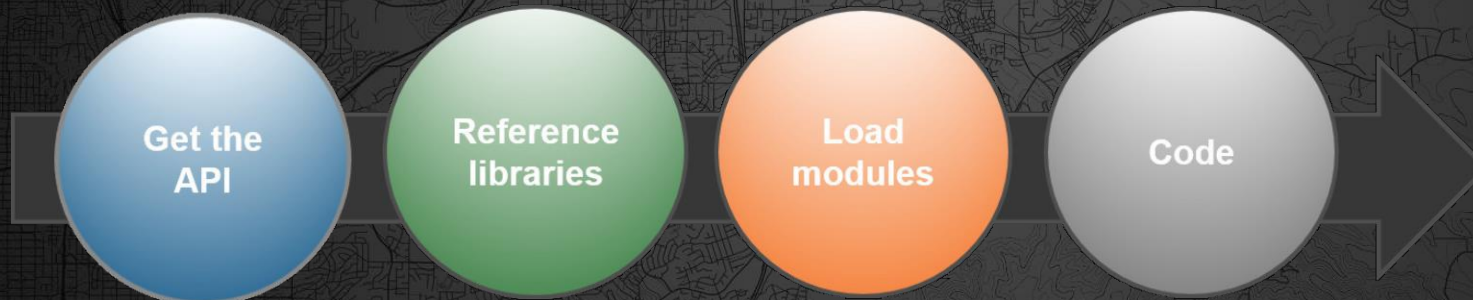
→ 3x nur noch Bugfixes und Key-Features



# Konzept 4x



# Workflow





# Loading The API

- Index.html

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="https://js.arcgis.com/4.10/esri/css/main.css">
</head>

<body>
  <script src="https://js.arcgis.com/4.10/"></script>
</body>

</html>
```



# Loading Modules

- app.js

```
require(["esri/map", "esri/geometry/Point"], function(Map, Point) {  
  
});
```





# Code

- index.html

```
<div id="mapDiv"></div>
```

Element

- app.css

```
#mapDiv {  
  height: 100vh;  
  width: 100vw;  
}
```

Style

- app.js

```
require(["esri/Map", "esri/views/MapView", ...], function(Map, MapView){  
  var myMap = new Map({  
    basemap : "streets"  
  } ...  
});
```

Action



# Let's get started

## Tools

- VS Code/ Brackets/ Notepad++/ Code Editor deiner Wahl
- Chrome/ Firefox / (IE)

## Nützliches

- ArcGIS for Developers - <https://developers.arcgis.com/javascript/>
- Geonet - <https://community.esri.com/>
- Stackoverflow - <https://stackoverflow.com/>



# Aufgabe 1

? **Eine Karte integrieren:** Stelle eine Karte in deiner Web App dar.

- Class: Map
  - <https://developers.arcgis.com/javascript/latest/api-reference/esri-Map.html>
- Class: MapView
  - <https://developers.arcgis.com/javascript/latest/api-reference/esri-views-MapView.html>



# Aufgabe 2

? **MapView Konfiguration:** Konfiguriere die Web Map nach deinem Geschmack.

- Class: Map
  - <https://developers.arcgis.com/javascript/latest/api-reference/esri-Map.html>
- Class: MapView
  - <https://developers.arcgis.com/javascript/latest/api-reference/esri-views-MapView.html>



# Aufgabe 3

? **Daten laden:** Integriere einen Feature Service und stelle diesen auf der Karte dar.

- Class: Feature Layer
  - <https://developers.arcgis.com/javascript/latest/api-reference/esri-layers-FeatureLayer.html>
- Service(s):
  - Erdbeben - [http://services.arcgis.com/V6ZHFr6zdgNZuVG0/arcgis/rest/services/Earthquakes\\_Since\\_1970/FeatureServer/0](http://services.arcgis.com/V6ZHFr6zdgNZuVG0/arcgis/rest/services/Earthquakes_Since_1970/FeatureServer/0)
  - Andere - <http://services.arcgis.com/V6ZHFr6zdgNZuVG0/ArcGIS/rest/services>



# Aufgabe 4

- **PopUp**: Füge dem Feature Layer Pop-Ups hinzu.
- Class: PopupTemplate
  - <https://developers.arcgis.com/javascript/latest/api-reference/esri-PopupTemplate.html>



# Aufgabe 5

? **Renderer:** Benutze einen Class-Breaks-Renderer um den Feature Service zu stylen.

- Class: ClassBreaksRenderer
  - <https://developers.arcgis.com/javascript/latest/api-reference/esri-renderers-ClassBreaksRenderer.html>



# Aufgabe 6

? **Legende:** Füge der Karte nun eine Legende hinzu.

- Class: Legend
  - <https://developers.arcgis.com/javascript/latest/api-reference/esri-widgets-Legend.html>



# Aufgabe 7

- **Layer Liste:** Füge der Web App eine Layer Liste hinzu.
- Class: LayerList
  - <https://developers.arcgis.com/javascript/latest/api-reference/esri-widgets-LayerList.html>



# Aufgabe 7b

- **Widgets:** Füge der Web App weitere Widgets hinzu.
- Siehe Hier:
  - <https://developers.arcgis.com/javascript/latest/api-reference>



# Aufgabe 8

? **Events/Observables:** Überwache eine Property und reagiere darauf.

- z.B.: Basemap ändern, wenn der FeatureLayer ausgeblendet wird.
- Watch:
  - <https://developers.arcgis.com/javascript/latest/api-reference/esri-core-Accessor.html#watch>



# Aufgabe 9

- **2D -> 3D:** Stelle nun alles in 3D dar.
- Tipp: 3 kleine Dinge ;-)



# Aufgabe 10

? **2D und 3D synchronisieren:** Stelle eine 2D Karte und eine 3D Szene dar und synchronisiere diese.



