



ManfredSteyer



# Maintainable Client Architecture w/ Micro-Apps and Angular Elements

Manfred Steyer

**SOFTWARE**architekt.at

# About me...

- Manfred Steyer
- **SOFTWARE**architekt.at
- Angular Trainings and Consultancy
- Google Developer Expert (GDE)



*Manfred Steyer*



Public: Munich, November 2018  
Early-Bird Tickets available now!  
Discount Code: ngMunich (15%)

In-House: everywhere

<http://www.softwarearchitekt.at/workshops>



✓ Commodore 64

✓ 1982

✓ ~ 1 MHz

✓ 64 KB RAM

✓ Disks with 170 KB



Version with two cores ;-)

Challenge: Resources!





Challenges  
today:  
Complexity





There are milk  
products lasting  
longer than  
some JavaScript  
frameworks

---





# Contents

1)  
Micro Apps

2)  
Web Components

3)  
Implementation  
w/ Angular

4)  
DEMO



## 1) Micro Apps



# Idea



System

# Idea

μService

μService

μService



# Idea

μApp

μApp

μApp

Pro

Contra

Less Complexity

Less Coordination

Different Technologies

Separate Deployment

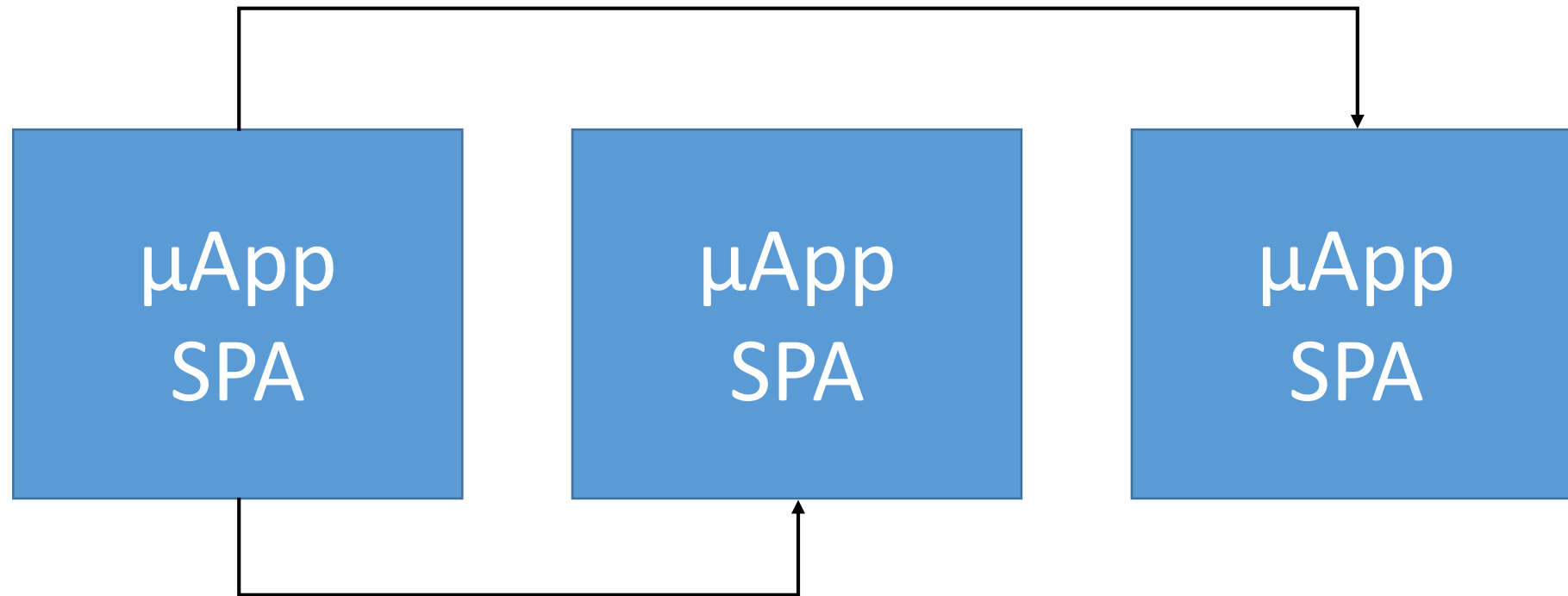
UI Composition

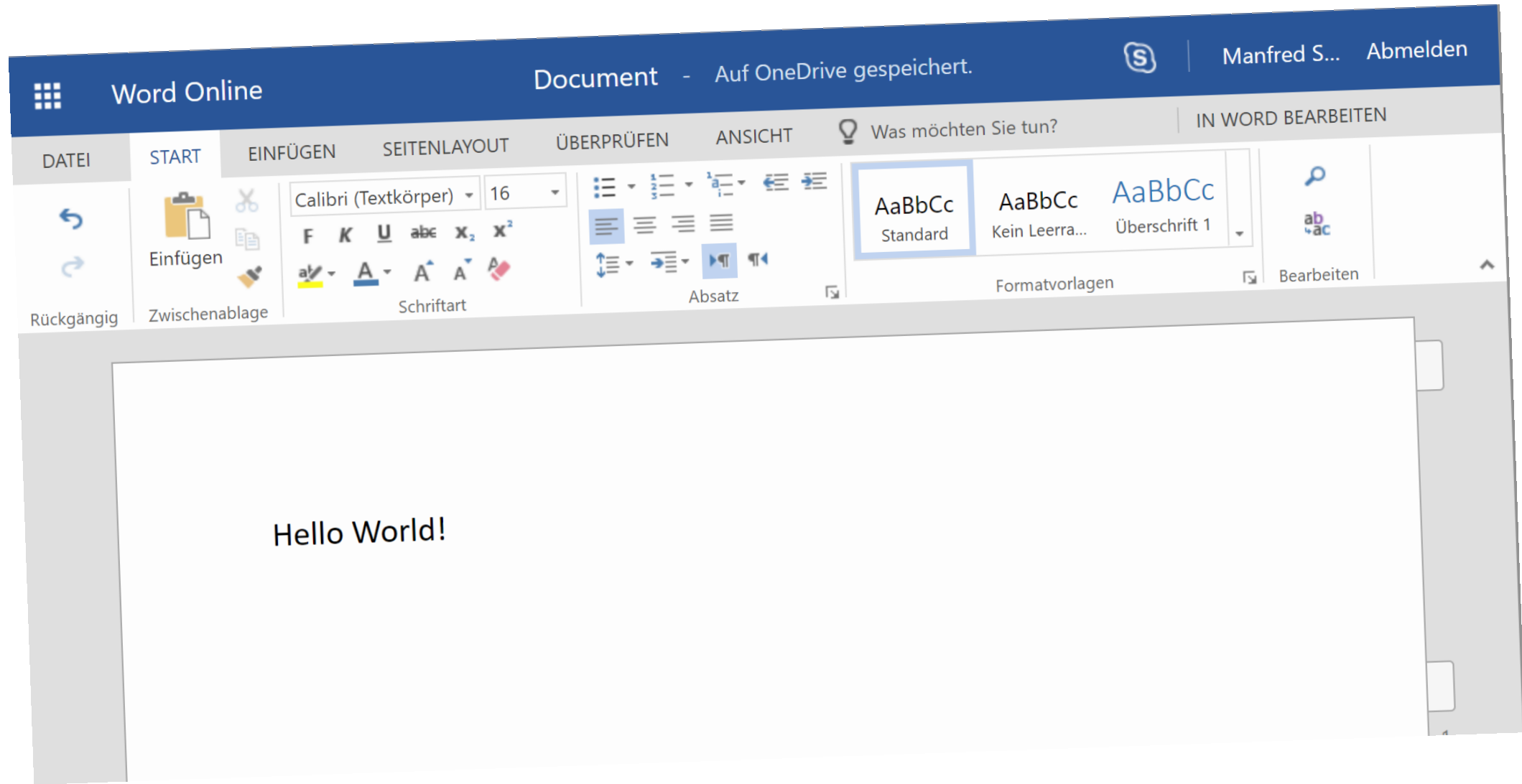
Distributed Data

Distributed System

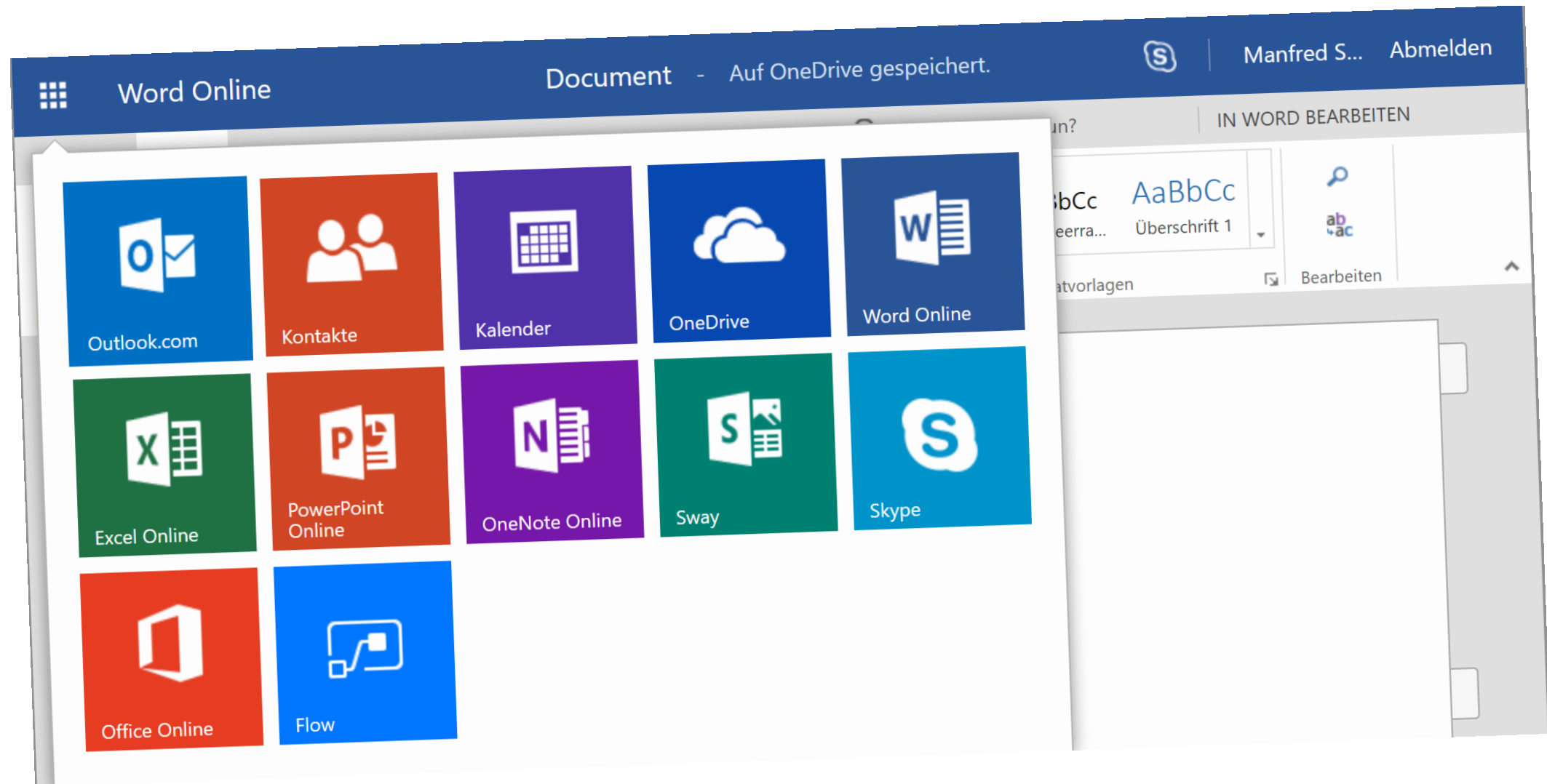


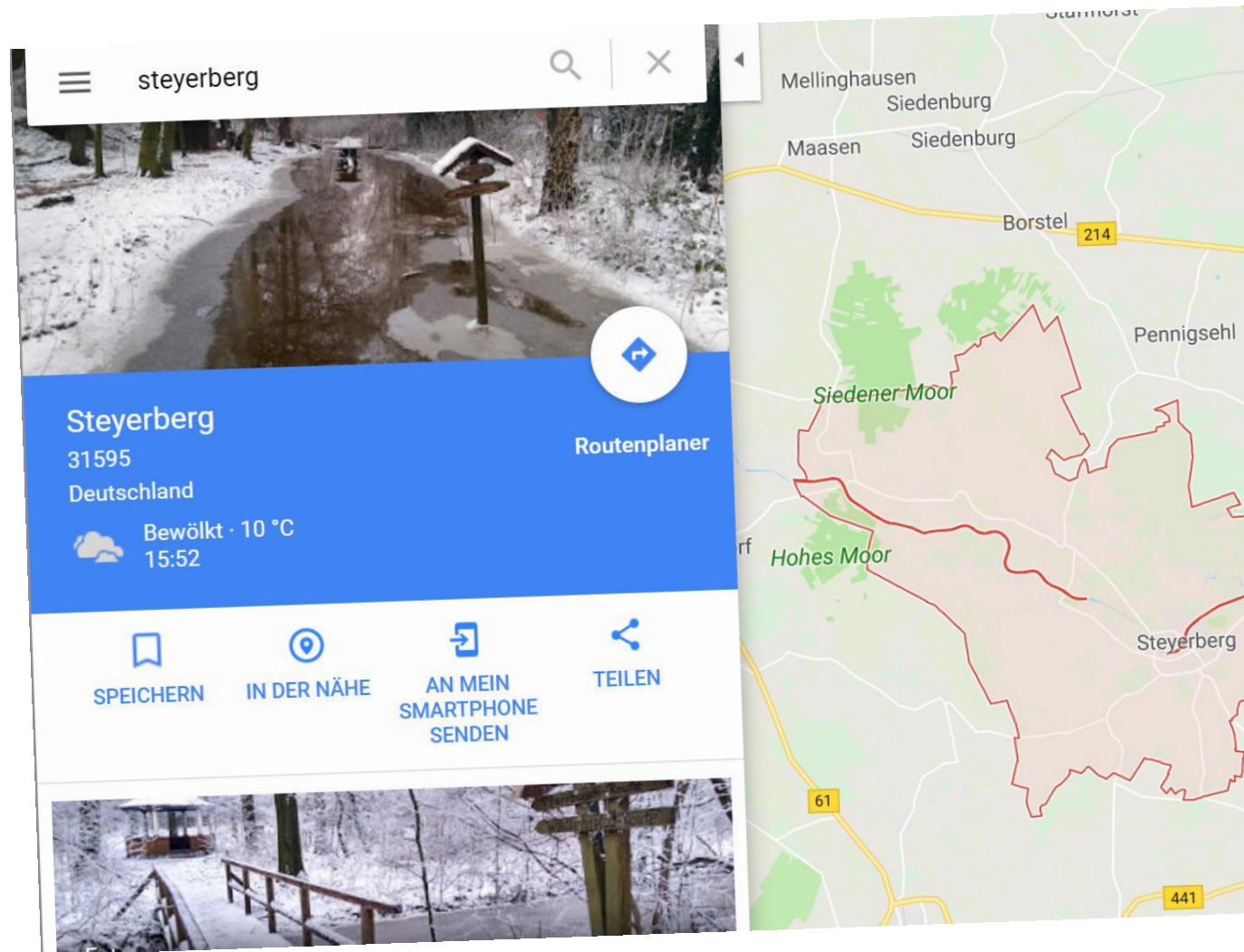
# UI Composition w/ Hyperlinks



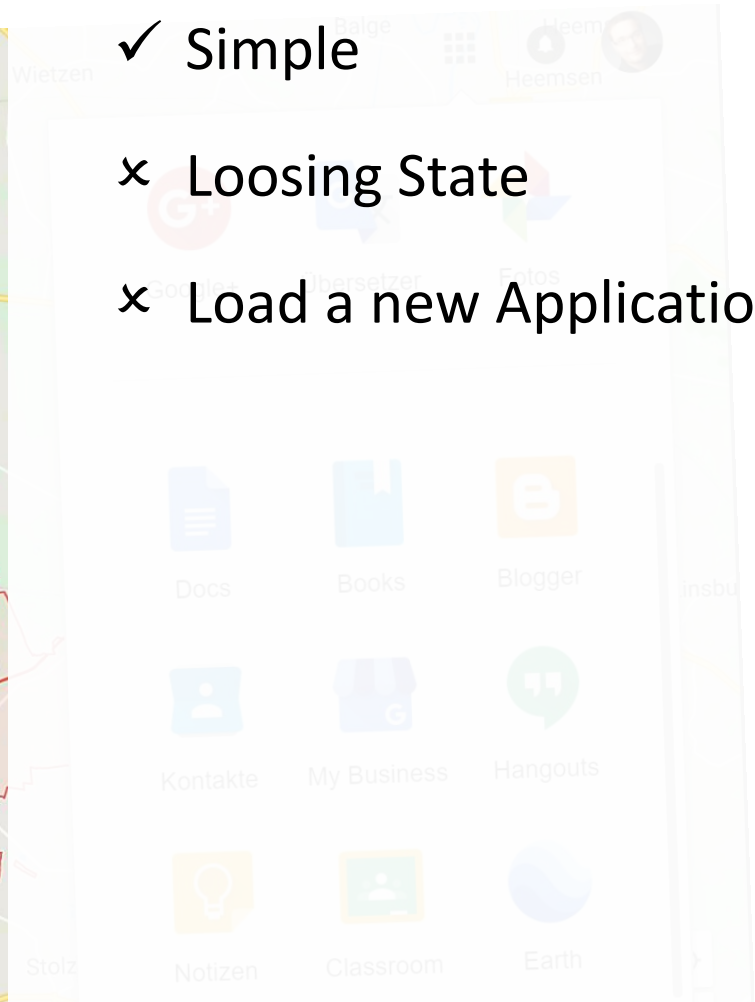








- ✓ Simple
- ✗ Loosing State
- ✗ Load a new Application





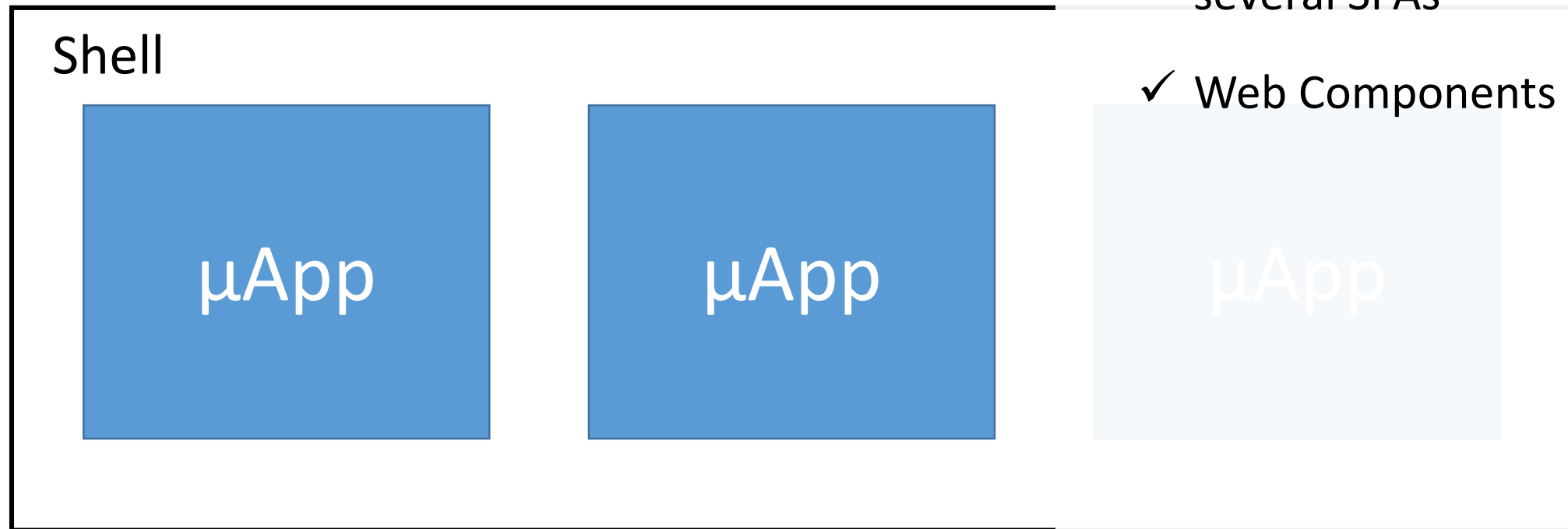
# Providing a (SPA based) Shell

✓ iframes

✓ Bootstrapping

several SPAs

✓ Web Components







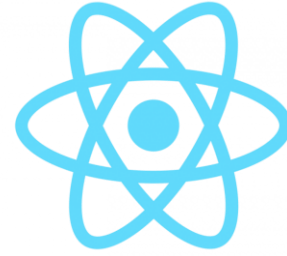
## 2) Web Components

---



The JavaScript logo, consisting of the letters 'JS' in a bold, black, sans-serif font, centered within a bright yellow square. This square is positioned inside a white circle with a light gray drop shadow.

JS



Framework independent  
Components



@ManfredSteyer

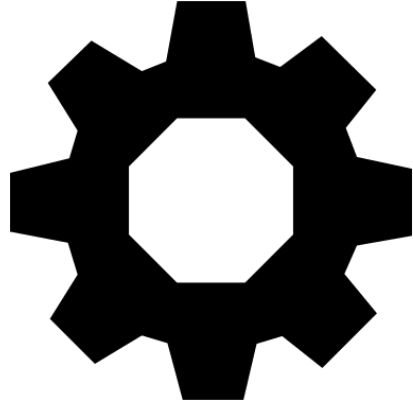
Can be dynamically  
loaded into the  
browser!





# Umbrella Term for several Standards

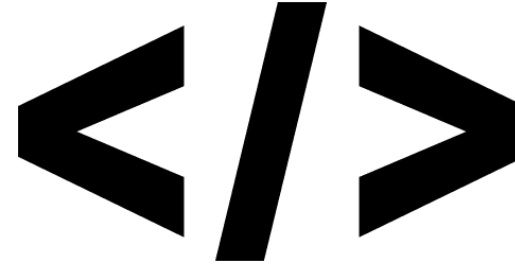




Templates



HTML Imports



Custom Elements



Shadow DOM

Standards

# Custom Elements

---

Custom HTML-Element, e. g. `<date-picker>`

---

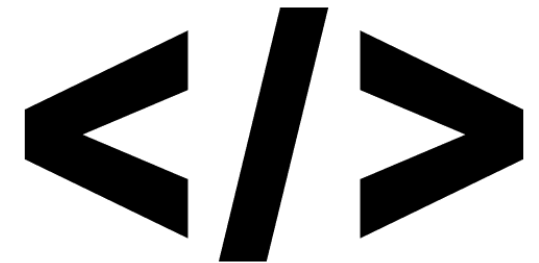
Behaves like build-in ones

---

Write once; consume with any framework

---

Can be polyfilled for all current browser

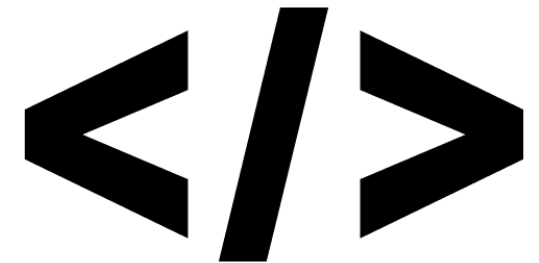


# Examples

`<slider min="0" max="100"> ...`

`<customer-selector selected="selected()"> ...`

`<my-micro-app state="sharedState" message="handle(...)">...`





# Shadow DOM

---

Isolate CSS

---

Parent cannot influence Component's layout

---

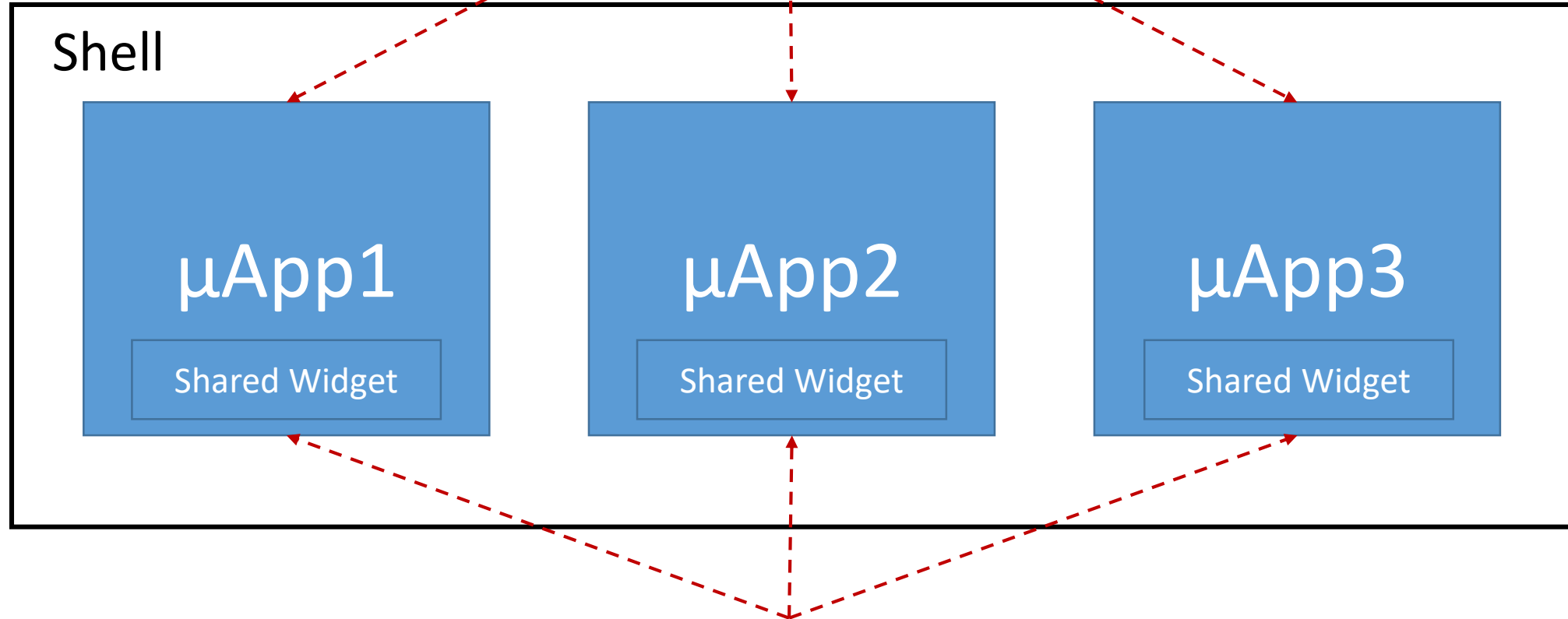
Component cannot influence Parent or Siblings

---

Can be emulated for all current browsers (w/ limitations)



## Web Components for **Macro**-Architecture



## Web Components for **Micro**-Architecture



3) Implementation with Angular



# Shadow DOM

- Baked-in
- Emulation
  - `@Component({ encapsulation: ViewEncapsulation.Emulated })`
  - Default
- Shadow DOM
  - `@Component({ encapsulation: ViewEncapsulation.Native })`
- Can be turned off:
  - `@Component({ encapsulation: ViewEncapsulation.None })`



# Custom Elements

---

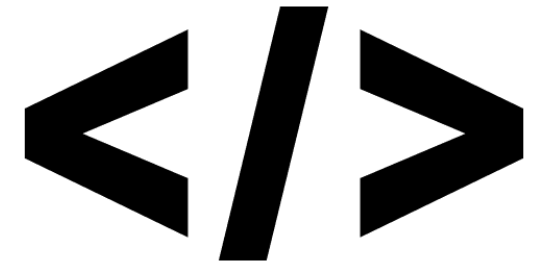
Angular Elements

---

Wraps Angular Component

---

Expose Custom Element



# MicroApp as Angular Elements

```
@NgModule({  
  imports: [BrowserModule],  
  declarations: [MicroAppComponent],  
  bootstrap: [],  
  entryComponents: [MicroAppComponent]  
})  
export class AppModule { }
```



# MicroApp as Angular Elements

```
@NgModule({  
  imports: [BrowserModule],  
  declarations: [MicroAppComponent],  
  bootstrap: [],  
  entryComponents: [MicroAppComponent]  
})  
export class AppModule { }
```

# MicroApp as Angular Elements

```
@NgModule({
  imports: [BrowserModule],
  declarations: [MicroAppComponent],
  bootstrap: [],
  entryComponents: [MicroAppComponent]
})
export class AppModule {

  constructor(private injector: Injector) { }

  ngDoBootstrap() {
  }
}
```

# MicroApp as Angular Elements

```
@NgModule({
  imports: [BrowserModule],
  declarations: [MicroAppComponent],
  bootstrap: [],
  entryComponents: [MicroAppComponent]
})
export class AppModule {

  constructor(private injector: Injector) { }

  ngDoBootstrap() {
    const MicroAppElement = createCustomElement(
      MicroAppComponent, { injector: this.injector });

    customElements.define('micro-app', MicroAppElement);
  }
}
```



# Loading MicroApps

```
<script src="micro-app.bundle.js"></script>
```

```
<micro-app></micro-app>
```

# 4) DEMO

# Conclusion

Separate  
Deployment

Less  
Coordination

Mixing  
Technologies

Shadow DOM  
& Angular  
Elements

Macro vs.  
Macro Level



## Blog > SOFTWAREarchitekt.at

- A Software Architect's Approach Towards Using Angular (And SPAs In General) For Microservices Aka Microfrontends
- Micro Apps With **Web Components** Using Angular Elements
- Angular Elements: A dynamic Dashboard ...
- Angular, React, Vue.Js And Co. **Peacefully United** Thanks To Micro Apps And Web Components

# Contact and Downloads

[mail] manfred.steyer@SOFTWAREarchitekt.at

[web] SOFTWAREarchitekt.at

[twitter] ManfredSteyer

*Slides & Examples*



Munich, November 2018 or In-House  
Early-Bird Tickets available now!  
Discount Code: ngMunich (15%)

<http://www.softwarearchitekt.at/workshops>