

# Bootstrap

Web-Engineering 2

Prof. Dr.-Ing. Thorsten Riemke-Gurzki

# Historie von Bootstrap

- Ursprünglich internes Projekt von Twitter, um die Entwicklung unternehmensinterner Verwaltungswerkzeuge zu vereinfachen.
- Ziel: Speed up development.
- Bootstrap umfasst
  - Grid-System
  - Gestaltungselemente (Icons, Buttons...)
  - Typografie
  - JavaScript-Erweiterungen.
- Erschienen im Jahr 2011.
- Lizenz: [Apache License 2.0](#)

# Bootstrap: Bestandteile

Layout/Grid

Components

Content

Forms

Helpers

Utilities

# Exkurs: Sass

- Bootstrap verwendet Sass, eine auf CSS basierende Stylesheet-Sprache.
- Sass umfasst auch Programmkonstrukte wie Fallunterscheidungen, Variablen und Konstanten.
- Mixins: Mehrere CSS-Regeln können benannt und unter diesem Namen an beliebigen Stellen eingefügt werden.
- Sie wird auch als CSS Präprozessor bezeichnet, da sie zu CSS übersetzt wird.
- Bootstrap kann ohne eigene Arbeiten in Sass verwendet werden, komplexe eigene Designs erfordern jedoch Sass.
- Bootstrap sorgt über Sass für das richtige Vendor-Prefixing.

# Exkurs: Sass/SCSS

```
@basecolor: #FF0000;
```

```
h1 {  
  color: @basecolor;  
}
```

```
div {  
  background-color: @basecolor;  
}
```

Sass Preprocessor

```
h1 {  
  color: #FF0000;  
}
```

```
div {  
  background-color: #FF0000;  
}
```

# Bootstrap nutzen

Website von Bootstrap:

<http://getbootstrap.com/>

Zuerst muss Bootstrap eingebunden werden (siehe nächste Folie).

Danach kann ausschließlich mit den CSS-Klassen von Bootstrap gearbeitet werden.

# Bootstrap einbinden

Bootstrap kann auf dem eigenen Server bereitgestellt werden.

Alternativ kann Bootstrap auch von einem CDN geladen werden.

Sowohl Download als auch der Code zum Einbinden aus dem CDN finden sich hier:

<https://getbootstrap.com/docs/5.1/getting-started/download/>

Bootstrap besteht aus einer CSS- und einer Javascript-Datei, die beide in das HTML-Dokument eingebunden werden müssen.

# Ein schnelles Beispiel: Buttons

Bootstrap enthält vorgestylte Komponenten, wie zum Beispiel Buttons.

```
<button type="button" class="btn btn-primary">Primary</button>  
<button type="button" class="btn btn-success">Primary</button>
```



# Typografie

Bootstrap bringt seinen eigenen Font mit.

Es sind entsprechende Schriftgrößen für **h1** bis **h6** hinterlegt.

Interessant: Bootstrap resettet nicht die Fontgröße, sondern basiert auf der Browser-Einstellung. (warum?)

# Styling Inline-Elemente

Inline-Elemente sind standardmäßig von Bootstrap mit gestylt, also auch insbesondere:

`<mark>`, `<del>`, `<em>`

# Beispiel Bootstrap-Komponenten: Cards



## Melolontha

Der praktische Helfer in Haus und Garten

[Mehr Info](#)

# Cards: Code

```
<div class="card" style="width: 18rem;">
  
  <div class="card-body">
    <h5 class="card-title">Melolontha</h5>
    <p class="card-text">
      Der praktische Helfer in Haus und Garten
    </p>
    <a href="#" class="btn btn-primary">Mehr Info</a>
  </div>
</div>
```

# Grid System

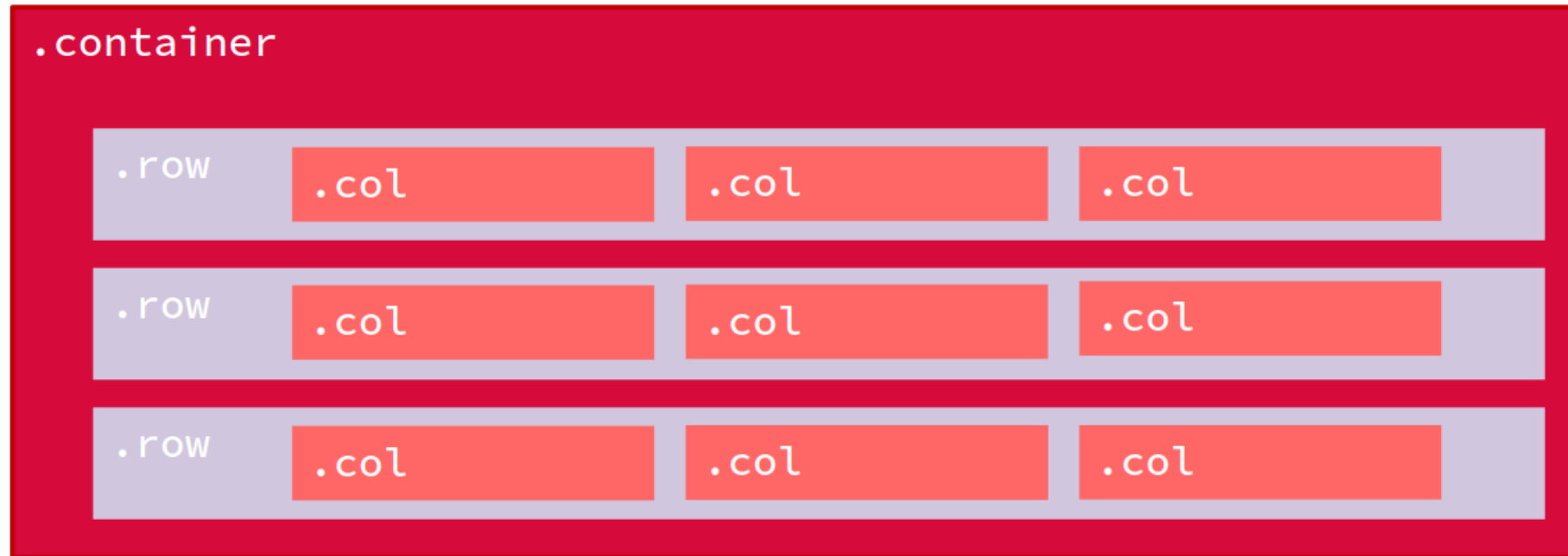
Bootstrap beinhaltet ein 940 Pixel breites Grid mit 12 Spalten.

# Bootstrap Breakpoints

Breakpoint	Class infix	Dimensions
X-Small	<i>None</i>	<576px
Small	sm	≥576px
Medium	md	≥768px
Large	lg	≥992px
Extra large	xl	≥1200px
Extra extra large	xxl	≥1400px

Quelle: Bootstrap, [www.getbootstrap.com](http://www.getbootstrap.com)

# Container, Rows und Columns



# Container

- **.container**: Erstellt einen responsiven Container mit fixer Breite (max-width zu Breakpoint).
- **.container-fluid**: Erstellt einen responsiven Container mit fixer Breite (width 100% zu Breakpoint).

```
<div class="container">  
  <h1>Bootstrap Container</h1>  
  <p>Dies wurde mit Bootstrap 4 container realisiert.</p>  
</div>
```

```
<div class="container-fluid">  
  <h1>Bootstrap Container</h1>  
  <p>Dies wurde mit Bootstrap 4 container-fluid realisiert.</p>  
</div>
```



# Container Abmessungen

	<b>Extra small</b> <576px	<b>Small</b> ≥576px	<b>Medium</b> ≥768px	<b>Large</b> ≥992px	<b>X-Large</b> ≥1200px	<b>XX-Large</b> ≥1400px
<code>.container</code>	100%	540px	720px	960px	1140px	1320px
<code>.container-sm</code>	100%	540px	720px	960px	1140px	1320px
<code>.container-md</code>	100%	100%	720px	960px	1140px	1320px
<code>.container-lg</code>	100%	100%	100%	960px	1140px	1320px
<code>.container-xl</code>	100%	100%	100%	100%	1140px	1320px
<code>.container-xxl</code>	100%	100%	100%	100%	100%	1320px
<code>.container-fluid</code>	100%	100%	100%	100%	100%	100%

Quelle: Bootstrap, [www.getbootstrap.com](http://www.getbootstrap.com)

# Zeilen mit row und col

```
<div class="container-fluid">
  <h1>Bootstrap Container, Row, col</h1>
  <p>mit einer row, die drei Spalten enthält. Responsiv!</p>
  <div class="row">
    <div class="col">.col</div>
    <div class="col">.col</div>
    <div class="col">.col</div>
  </div>
</div>
```

Zum Ausprobieren weist man am Besten den **col** jeweils einen farbigen Hintergrund zu.

# Zeilen mit row und col im Grid

```
<div class="container-fluid">
  <h1>Bootstrap Container, Row, col</h1>
  <p>mit einer row, die drei Spalten enthält. Responsiv!</p>
  <div class="row">
    <div class="col-sm-3">.col</div>
    <div class="col-sm-3">.col</div>
    <div class="col-sm-6">.col</div>
  </div>
</div>
```

Jetzt werden die Container in den Grid-Spalten angeordnet. Die Breiten sind 3-3-6 (=12). Breakpoint ist sm (small=576px).

# Zeilen mit row und col - mehr Breakpoints

```
<div class="container-fluid">
  <h1>Bootstrap Container, Row, col</h1>
  <p>mit einer row, die drei Spalten enthält. Responsiv!</p>
  <div class="row">
    <div class="col-sm-3 col-md-4">.col</div>
    <div class="col-sm-3 col-md-4">.col</div>
    <div class="col-sm-6 col-md-4">.col</div>
  </div>
</div>
```

sm-Aufteilung: 3-3-6

md-Aufteilung: 4-4-4

# Bootstrap: Menu

Riemke's Home Funktionen Preise Über uns

Riemke's



Home

Funktionen

Preise

Über uns

Der ausgewählte Menüpunkt wird durch die Klasse active anders dargestellt.

# Bootstrap: Tabellen

```
<table class="table table-striped">
```

```
</table>
```

Tabellen sind ebenfalls im Bootstrap-Stil gestylt. Mit der Klasse **table-striped** sind gestreifte Tabellen oder mit **table-dark** dunkle Tabellen einfach realisierbar.

Mehr zu Tabellen: <https://getbootstrap.com/docs/5.2/content/tables/>

# Form Validation

Für die Form Validierung ist neben Bootstrap CSS Klassen auch Javascript erforderlich.

<https://getbootstrap.com/docs/5.2/forms/validation/>



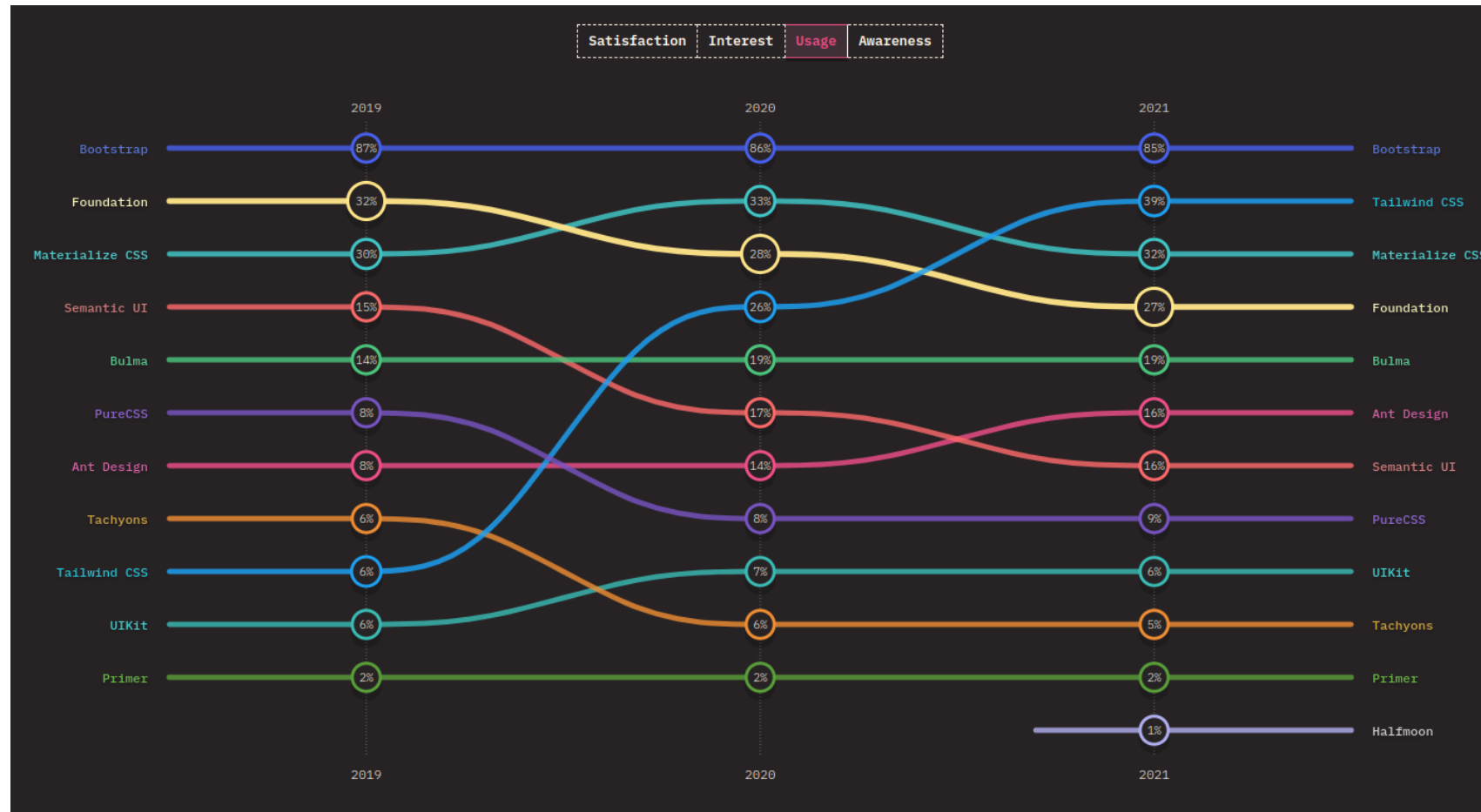
# Mehr Bootstrap!

Dann am Besten in die Doku schauen!

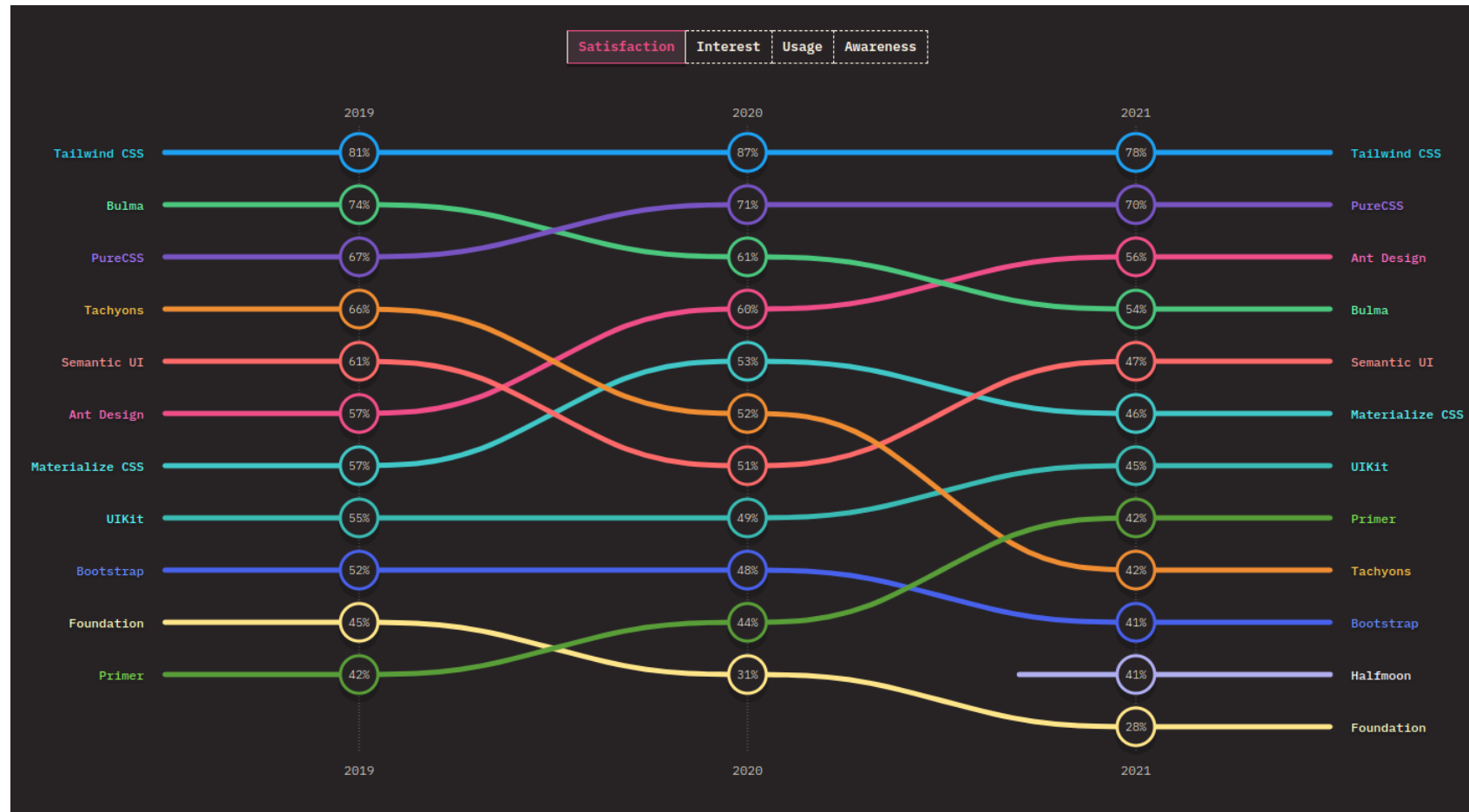
<https://getbootstrap.com/docs/5.2/>

Und das machen wir jetzt!

# State of CSS: Nutzung



# State of CSS: Zufriedenheit



© 2024 Prof. Dr.-Ing. Thorsten Riemke-Gurzki. Verwendung nur für Lernzwecke im Rahmen der Lehrveranstaltung. Eine Nutzung und/oder analoge bzw. digitale Weiterverbreitung außerhalb der Veranstaltung ist nicht gestattet. Stand der Folien: 18.03.2024

automatisch generiert mit lecturemark markdown presentations © 2024 Thorsten Riemke- Gurzki, Version: 0.9

Prof. Dr.-Ing. Thorsten Riemke-Gurzki