

# **Die Asset Evolution**

Professionalisierung von CSS & JS

# Me

- Georg Leciejewski
- [github.com/Schorsch](https://github.com/Schorsch)
- Wirtschf. Inform, LKW Mechaniker
- PHP, Ruby, JS, CSS & Co
- [SalesKing.eu](https://SalesKing.eu)

# Wat is en' Website

- **Content** Inhalte Texte, Bilder, Videos
- **HTML** Präsentation Überschrift, Absätze, Listen

## Assets

- **CSS - Styles** schön Bunt
- **JS - Javascript** Actions, Popup's, Bewegung
- **Images** Background Images, Logo
  - Fonts

# Die Anfänge

Inline Styles:

```
<h1 style="font: bold">Überschrift</h1>
```

Inline JS

```
<input type="submit" onclick="alert('Hi')">
```

# In den Kopf

```
<html><head>  
  <styles> h1{font: bold}</styles>  
  <script type="text/javascript">  
    document.getElementById('buttonID').onclick=function(){ alert('Hi') }  
  </script>  
</head>  
<body>  
  <h1>Überschrift</h1>  
  <input type='submit' id='buttonID'>  
</body>  
</html>
```

# Die Entkoppelung

```
<html><head>  
  <link type="text/css" media="all" rel="stylesheet" href="styles.css">  
  <script type="text/javascript" src="functions.js"></script>  
</head><body>  
  <h1>Überschrift</h1>  
  <input type='submit' id='buttonID'>  
</body>  
</html>
```

styles.css:

```
h1{font: bold}
```

functions.js:

```
document.getElementById('buttonID').onclick=function(){ alert('Hi') }
```

# Die Aufteilung

- Einzelne Dateien für bestimmte Bereiche
  - CSS: buttons.css, navi.css
  - JS: form.js, startpage.js
- Verbreitung von Frameworks:
  - jQuery JS, Bootstrap CSS
- Professionelle Snippets
  - Cross Browser CSS Hacks z.b, für runde Ecken
- Ziele: Wiederverwendung, Vereinfachung

# Neue Sprachen

- CSS: SASS, LESS
  - HTML - HAML, SLIM
  - JS: CoffeeScript, DART
- 
- Weniger & leichter Tippen
  - Besserer Code
    - Compiler übernimmt Fehlerprüfung
    - automatische Code-Formatierung



# Praktische Handhabung früher

- kopieren ganzer Seiten / Template
  - Intensive HTML, CSS, JS Programmierung
  - Manuelle CSS/JS Updates mehrerer Webseiten
- 
- Neue Seite == Neues Chaos

# Praktische Handhabung heute

- Einsatz bestehender Bausteine/Frameworks
  - jQuery, Bootstrap, Foundation, FontAwesome
- Konzentration auf HTML Struktur & Inhalte
  - das meiste sieht per se schon gut aus
- Programmierung von Ausnahmen
  - Buttons Grün, Headline rot
- Automatische Updates externer Bausteine
  - Bower

# Probleme

- Ladezeiten
- Neue Endgeräte (Auflösungen, Browser)
- Testbarkeit
- Breaking changes in externen Bausteinen

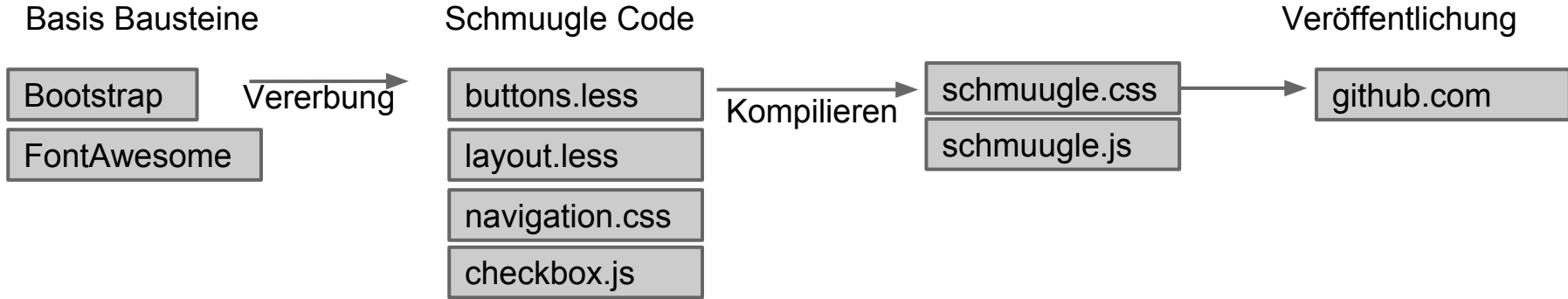
# Ziele / Ergebnisse

- Wiederverwendung
- weniger Code == weniger Fehler
- schneller Einstieg
- Zukunftssicher
- Programmierer brauchen keine Grafiker mehr sein

# Fallbeispiel - Schmuugle

- CSS Framework für Business Software
- Whats included?
  - Layout Grids
  - Menü & Buttons
  - JS Helferlein
- [github.com/Schmuugle](https://github.com/Schmuugle)

# Grundstruktur



Verwendung im eigenen Projekt

```
bower install https://github.com/Schmuugle/schmuugle.git
```

# Tools

## Bower

- managed Abhängigkeiten innerhalb von Schmuugle
- ermöglicht Dritten die einfache Nutzung

## Grunt

- Fügt CSS & JS Dateien zusammen
- Testet
- Beobachtet Dateien