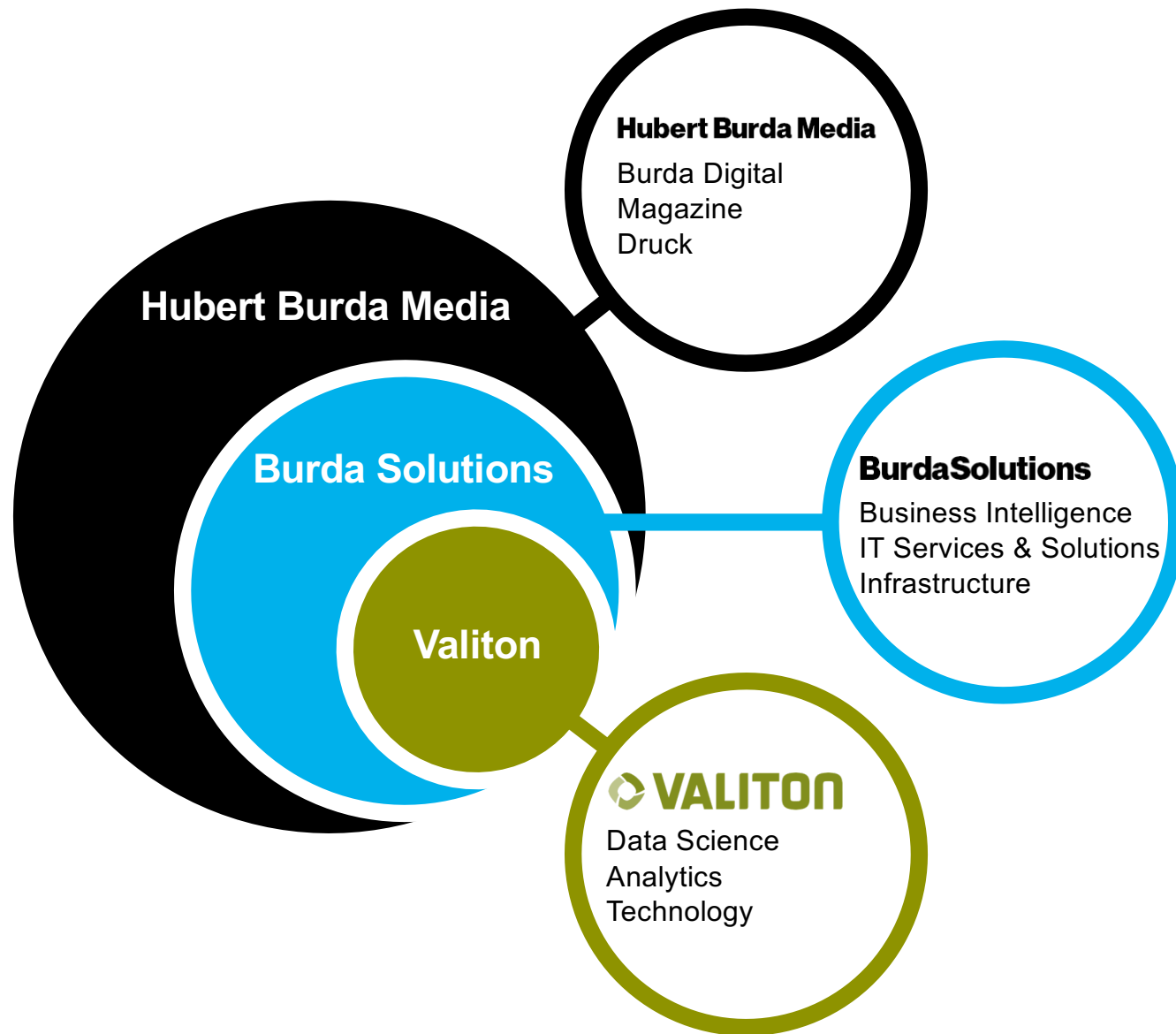


# Websites – Es muss nicht immer Wordpress sein ... (oder doch?)

Uwe Jäger  
Valiton GmbH



# VALITON – EIN BURDA SOLUTIONS UNTERNEHMEN



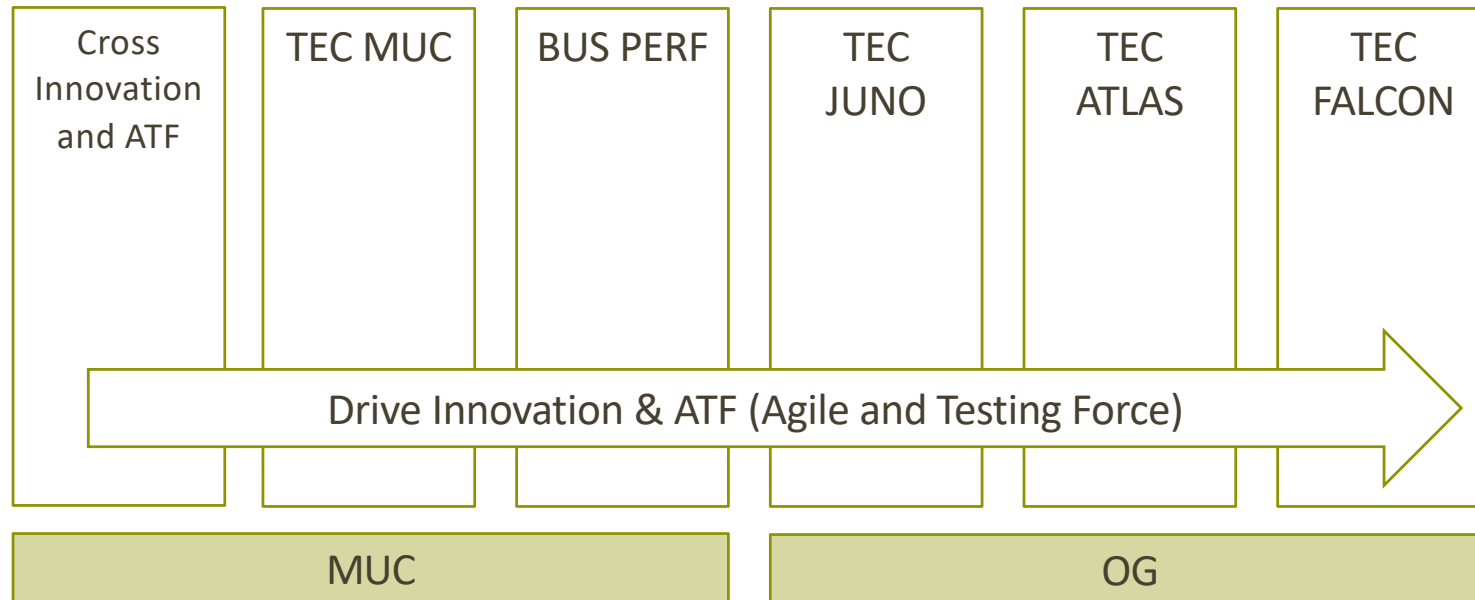
# VALITON - ECKDATEN

---

## Key Facts

- Team
  - MUC 20 Mitarbeiter / OG 20 Mitarbeiter
  - Aktuell 5 Werkstudenten/Praktikanten/Thesis in München und Offenburg > davon wird 1 in Festanstellung wechseln am 1.1.19
  
- Kunden
  - Betreute e-Commerce Kunden: 10 mit über 1.000 Mio. € Umsatz über die Plattformen
  - Betreute Analyticskunden: 5 mit mehr als 3 Mio OptIns / Kundendaten
  - Portfolio HBM Kunden und Nicht-HBM Kunden
  
- KPIs
  - Kununu Score : 4,06
  - Kundenzufriedenheits Score : 1,79 (1,87 in 2016)
  - Durchschnittsalter : 36 Jahre

# VALITON - ORGANISATION

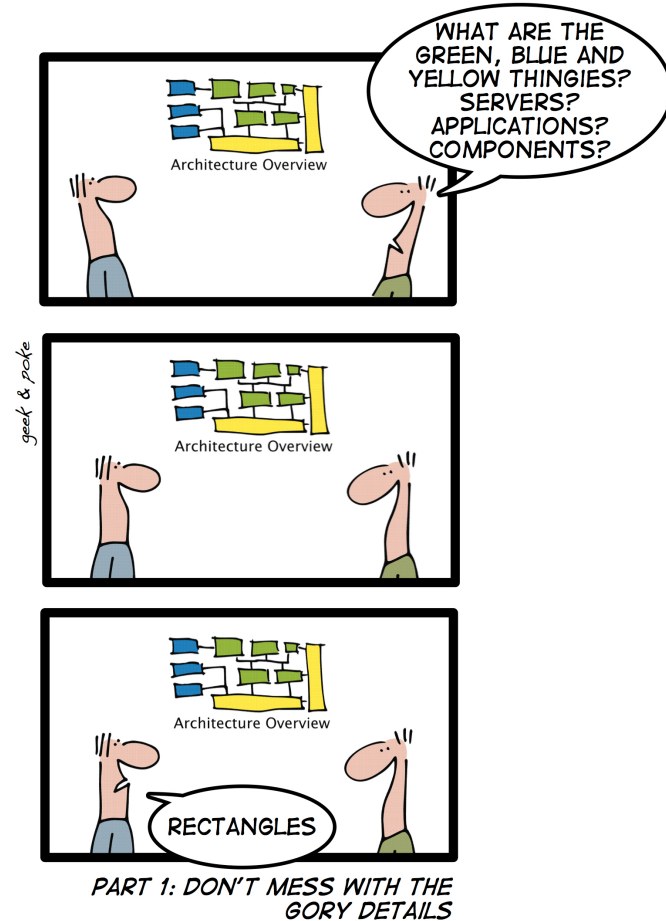


# SOFTWARE ARCHITEKTUR

## Was macht ein Software Architekt?

- Analysiert und bewertet Technologien
- Wählt Technologien aus
- Leitet Projekte
- Stellt technische Eigenschaften der Software sicher
- Trifft technische Entscheidungen in einem Software-Projekt

### ENTREPRISE ARCHITECTURE MADE EASY



# AUSGANGSSITUATION

---

## Valiton soll eine neue Website bekommen

- Die vorhandene Website auf Basis Typo3 war in die Jahre gekommen
- Ein Refresh von Design und Content war notwendig
  - Neue Informationsarchitektur
  - Neues Design
    - Unterschiedliche Seiten/Templates
    - Content eher statisch
    - Einfach pflegbarer Blog
- Wir nehmen Wordpress – oder nicht?

# WORDPRESS

---

## Wordpress ist ein freies Content Management System

"**WordPress (WordPress.org)** ist ein [freies Content-Management-System](#), welches ursprünglich zum Aufbau und zur Pflege eines [Weblogs](#) entwickelt wurde, da es jeden Beitrag einer oder mehreren frei erstellbaren Kategorien zuweisen kann und dazu automatisch die entsprechenden Navigationselemente erzeugt. Sie ist Teil der Hosting-Plattform [WordPress.com](#) (**WordPress**).“

<https://de.wikipedia.org/wiki/WordPress>

- Pros:
  - Einfaches und kostenloses CMS
  - Viele Plugins für Erweiterungen und Themes für Look & Feel
- Cons:
  - Nur Blog-Beiträge als Standardinhaltstyp
  - Ziel von Hackern und automatisierten Skripten
  - Performance
  - Qualität der Erweiterungen
  - Art der Programmierung von Erweiterungen

# WORDPRESS ERWEITERUNGEN

---

Programmierung erfolgt durch „Hooks“

- Beispiel einfache Erweiterung für neue Felder



# ALTERNATIVEN

---

## Gibt es Alternativen zu Wordpress?

- Andere CMS-Systeme
  - Typo 3, Drupal, Liferay, Magnolia, ...
    - Erweiterungen
    - Performance
    - Security
- Eigenentwicklung
  - Z.B. mit Symfony CMF
    - Aufwändige Implementierung
    - Server-Betrieb
- Statische Website
  - HTML, CSS
    - Anpassungen und Inhaltspflege nur durch Entwickler
- Generator für statische Website
  - Automatische Generierung der kompletten Website
    - Einfacher Serverbetrieb
    - Schnelle Website
    - Inhaltspflege muss geklärt werden

# WEBSITE GENERATOREN

---

Aus Templates und Inhalten werden HTML-Seiten erzeugt.

- Jekyll
  - <https://jekyllrb.com/>
  - Ruby & Liquid
- GatsbyJS
  - <https://www.gatsbyjs.org/>
  - Javascript/React
- Hugo
  - <https://gohugo.io/>
  - Go
- ... und viele Weitere (siehe z.B. <https://www.staticgen.com/>)

# WEBSITE GENERATOREN - TEMPLATES

---

Je nach Programmiersprache/Generator gibt es Unterschiede

- Templatesprachen wie Liquid bei Jekyll oder Go Templates bei Hugo
- React-Komponenten bei GatsbyJS

# WEBSITE GENERATOREN - INHALTE

---

Üblicherweise werden die Inhalte in (Markdown-) Dateien gepflegt

- Markdown: vereinfachte Auszeichnungssprache, die auch ohne Konvertierung gut lesbar ist:

```
Überschrift  
=====
```

```
Dies ist ein Absatz
```

```
2. Überschrift  
-----
```

```
# ungeordnete Liste  
# mit zwei Einträgen
```

- Aus dem Markdown wird HTML generiert

# GATSBYJS UND REACT

---

## GatsbyJS generiert eine React-Anwendung

- Statt Templates verwendet GatsbyJS React-Komponenten
- React
  - komponenten-basiertes Javascript UI Framework
  - Virtual DOM
  - Properties & State
  - render Methode
- Sehr guter und einfacher Einstieg in React
- Für das Deployment wird serverseitig HTML generiert (wichtig unter SEO-Aspekten)

# GATSBYJS BEISPIEL

---

Eine Seite ist eine React-Komponente

- [GatsbyJS Starter-Seite](#)

# GATSBYJS UND DATEN

---

GatsbyJS kann über Plugins Inhalte aus beliebigen Quellen verwenden

- Daten werden über Source Plugins eingelesen und im Data-Layer gespeichert
- Transformer Plugins können die Daten im Data-Layer verändern/anreichern
- Über GraphQL-Abfragen werden React-Komponenten mit Daten versorgt oder Seiten generiert
- Während der Entwicklung kann man einfach die [Datenbank abfragen](#)

# GATSBYJS SOURCE PLUGIN BEISPIEL

---

Aktuelle Stellenangebote werden per API ausgelesen

- [Beispiel Source-Plugin](#)



# GATSBYJS TRANSFORMER

---

Daten im Data-Layer werden angereichert

- Markdown parsen
- Bilder in unterschiedlichen Größen berechnen

# GATSBYJS GRAPHQL

---

Mit GraphQL werden Daten aus dem Data-Layer abgefragt

- Inhalte auslesen
- Bildgrößen bereitstellen

# GATSBYJS INTEGRATION WORDPRESS

---

Viele Plugins sind vorhanden, z.B. eine Wordpress Integration

- Redakteure erfassen Inhalte im Admin von Wordpress
- Über die JSON-API werden die Posts ausgelesen und im Data-Layer von GatsbyJS gespeichert
- Über GraphQL-Queries werden die Daten selektiert und den Komponenten als Properties übergeben

# GATSBYJS DYNAMISCHE FUNKTIONEN

---

Eine statische Website muss nicht auf Funktionen verzichten

- Dynamik mit Javascript/React ist schon eingebaut
- Formulare
  - Serverseitige Lösung für Formularhandling notwendig, z.B. <https://formsfree.io/>
- Kommentare
  - Wie Formulare, oder <https://staticman.net/> Kommentare landen als Pull Requests auf github.com
- Suche
  - Geht auch schon in Javascript: <http://elasticlunr.com/>

# GATSBYJS GENERIERUNG UND DEPLOYMENT

---

## Automatisches Generieren und Deployen mit CI Pipelines

- Mit dem Kommando `gatsby build` wird die Website generiert
- Inhalt des Verzeichnisses „public“ wird einfach auf den Webserver kopiert
- Dieser Prozess lässt sich einfach automatisieren und z.B. über eine CI-Pipeline ausführen (z.B. Jenkins oder in unserem Fall Gitlab-CI)
- Der Prozess kann manuell getriggert werden, oder
- automatisch bei Änderungen im Code und
- automatisch ein Mal täglich

# FAZIT

---

## GatsbyJS

- + Schnelle und sichere Website
  - + Einfache Pflege der Inhalte mit Wordpress
  - + Schnelles Setup zum Lernen von React
  - + Gute Abstraktion der Daten im Data-Layer
  - + Dynamische Funktionen realisierbar
  - + Einfaches Hosting und Deployment
- 
- Hohe Lernkurve
  - Nur Teile der Inhalte einfach pflegbar

## LINKS

---

- Übersicht statischer Website Generatoren
  - <https://www.staticgen.com/>
  - <https://snipcart.com/blog/choose-best-static-site-generator>
  - <https://www.netlify.com/blog/2017/05/25/top-ten-static-site-generators-of-2017/>
- GatsbyJS
  - <https://www.gatsbyjs.org/>
- Beispielseite
  - <https://www.valiton.com/>
- Beispiel-Code
  - <https://github.com/uwej711/gatsby-demo>

## KONTAKT

---



**Uwe Jäger**

Senior Software Architect

Hubert-Burda-Platz 1  
D-77652 Offenburg

Telefon: 0781 – 84 3572

- <mailto:uwe.jaeger@valiton.com>
- <https://www.valiton.com>
- <https://github.com/uwej711>
- <https://github.com/valiton>
- <https://stackoverflow.com/jobs/companies/valiton-gmbh>