

Headless CMS - brauche ich das?

Peter Köllner

München, 24.03.2020

Agenda

- Einführung
- Grundlagen zu Headless CMS
- Headless Varianten
- Vor- und Nachteile vom Headless Ansatz
- Empfehlung für CMS-Architekturen
- Lessons Learned in Headless Projekten

Jahre Partner
von Sitecore

Mitarbeiter an
5 Standorten



13

Sitecore
Experience Awards

Gemeinsam erfolgreich!

Peter Köllner, 43

APPLICATION ARCHITECT

#agile #münchen #microservices #boardgames

”



PURPOSE VON UNIC

**Wir machen Digitales
menschlich.**

Unic: digitale Lösungen seit 1996

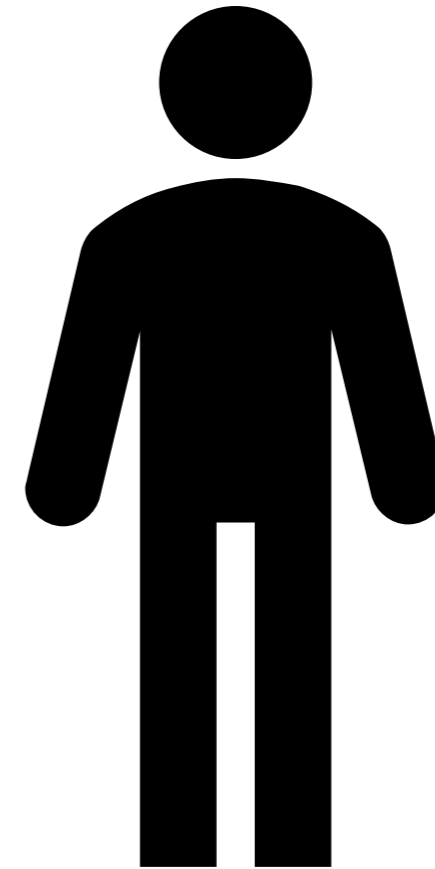
- Inhabergeführte Digitalagentur mit 250+ Mitarbeitenden
- 5 Standorte in Deutschland, Schweiz und Polen
- Digitale Lösungen für Aufgaben aus Marketing, Verkauf & After Sales Service – z.B. mit Global Websites, E-Commerce-Anwendungen oder Kundenportale
- Einsatz von kommerzieller Software & OpenSource-Frameworks

Grundlagen zu Headless CMS

Coupled CMS

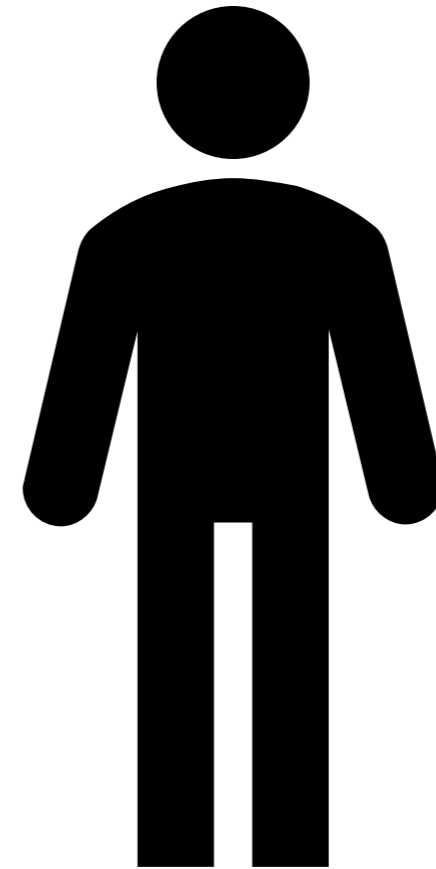
1 System ist zuständig für Erstellung und Auslieferung:

- Redakteursystem
(Inhaltserstellung / -speicherung)
- Deliverysystem
(Inhaltsauslieferung / -darstellung)

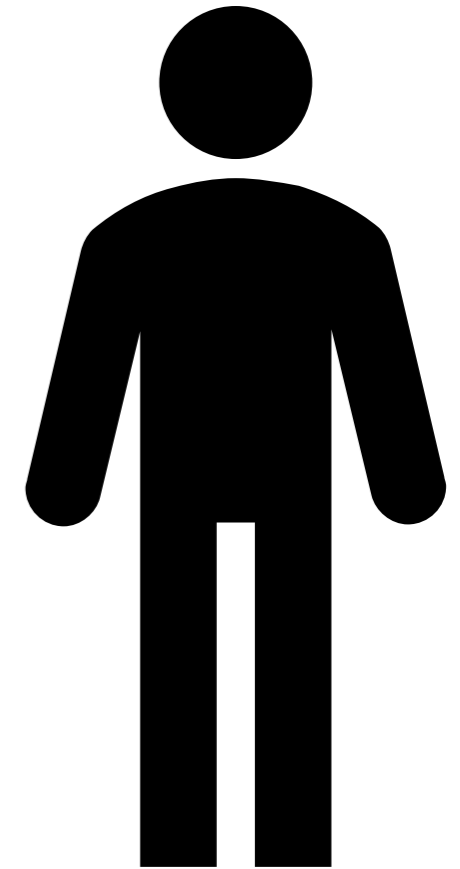


Decoupled CMS

- Wie Coupled CMS, nur wird der Content zur Auslieferung auf ein Delivery System repliziert
- Vorteile: Sicherheit und Performance



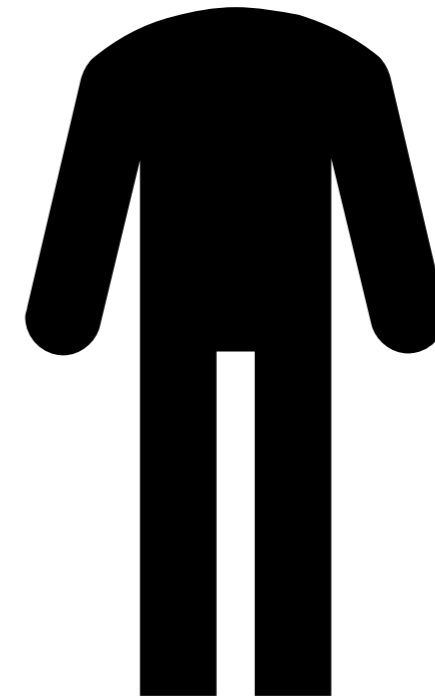
Redaktion



Auslieferung

Headless CMS

- System zur Speicherung und Verwaltung von Inhalten
- Zur Auslieferung steht eine API zur Verfügung
- Die Darstellungsmöglichkeiten (entspricht dem Kopf) fehlen

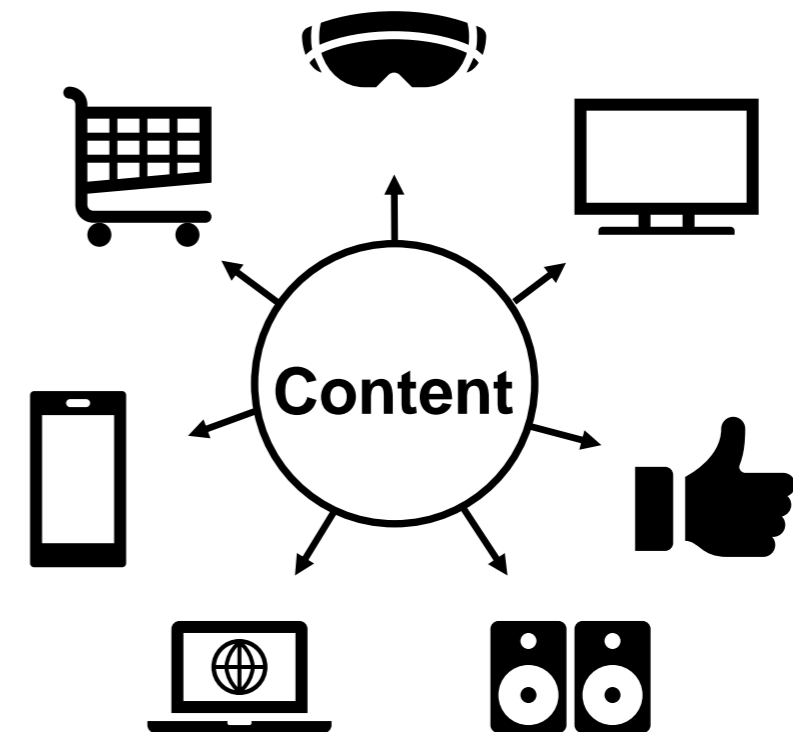


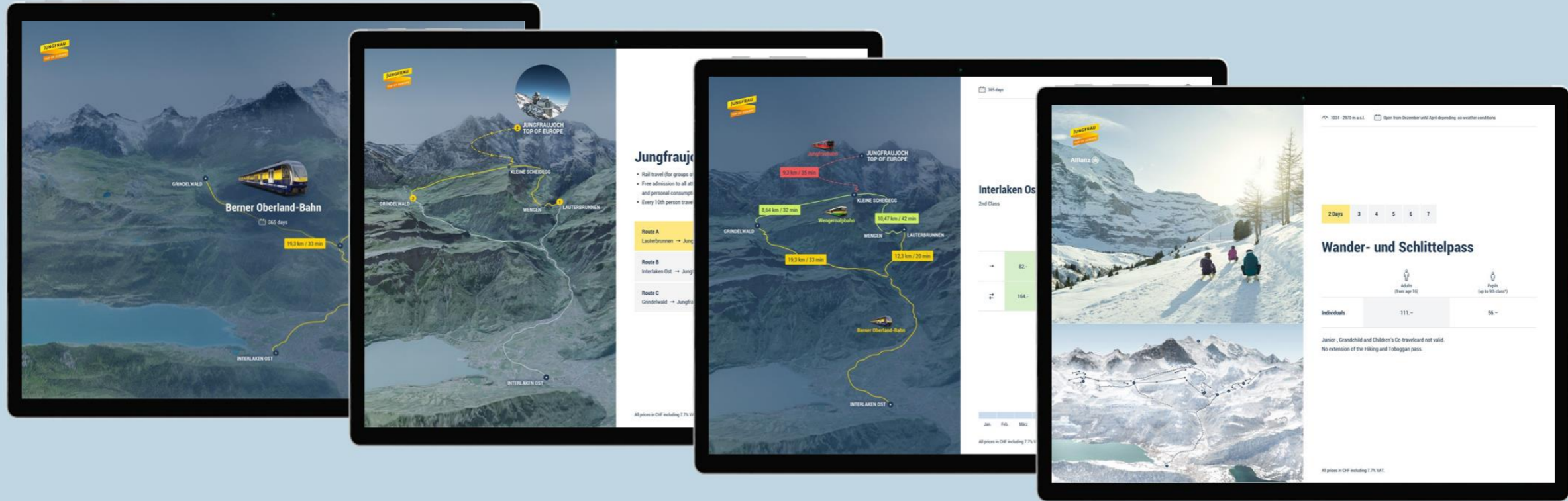
Content Model



Wieso Headless CMS

- Traditionell: (W)CMS wird für Web verwendet
- Neu: weitere Kanäle benötigen andere Formate wie Audio, Video, proprietäre Protokolle
- Beispiele:
Point of Sales Displays, Alexa, Virtual Reality, Smartwatches, ...





Headless Varianten

Pure Headless Systeme

- Größtenteils SaaS Lösungen
- Wesentlich niedrigere Lizenzkosten
verglichen mit klassischen Enterprise
CMS Lösungen
- Keine Layoutfunktionalität und
Personalisierungsmöglichkeiten
- Etabliertere Produkte:
 - Contentful
 - Contentstack
 - CosmicJS
 - GraphCMS
 - Prismic
 - Zesty

Hybrid Headless Systeme

- Ausspielung headless oder klassisch
 - Preview, Layout Features und Navigation sind vorhanden
 - Bieten als Digital Experience Plattform (DXP) meist weitere Features wie Analytics, Personalisierung, Marketing Automation
 - Reichen von On Premise Lösungen bis in die Cloud
- Etabliertere Produkte:
 - Adobe AEM
 - CoreMedia
 - Drupal
 - FirstSpirit
 - Magnolia
 - Sitecore

Vor- und Nachteile vom Headless Ansatz

Vorteile

- **Eine Content-Basis für alle Kanäle**
- **Parallelisierung von Content- und Applikationsentwicklung**
- **Höhere Verfügbarkeit von Entwicklern**
- **Microservice Architektur**

Nachteile

- **Präsentationsschicht für Webauftritte**
 - Redakteure haben meist keinen Einfluss auf die Navigation, die Anordnung des Inhalts
 - Wenig Flexibilität für Landing Pages / Microsites
- **Höhere Anforderungen an Redakteure**
- **Höherer Anteil an Individual-Entwicklung**
- **Personalisierung wird in der Regel nicht unterstützt**

Empfehlungen für CMS-Architekturen

Klassisches WCMS / DXP

- Fokus Webkanäle / E-Mail
- Niedrige/Keine Abhängigkeit von Entwicklern für Standard Webanwendungsfälle (z.B. neue Microsites)
- Beeinflussung der Darstellung durch Redakteure ist wichtig
- Personalisierung ist wichtig

Pro Headless Ansatz

- Multichannel Strategie
- Stark strukturierte Darstellung des Contents (Zeitungsartikel, Blogs, Produktkataloge)

Welcher Headless Ansatz

- Multichannel Strategie
- Stark strukturierte Darstellung des Contents (Zeitungsartikel, Blogs, Produktkataloge)
- **Hybrid Headless CMS**
 - Personalisierung ist wichtig
 - keine Microservice Strategie
- **Pure Headless CMS**
 - Personalisierung ist nicht wichtig
 - Microservice Strategie

Viel inspiriert

Ideen für kleine und grosse Erlebnisse in der ganzen Schweiz.

▾ **JETZT ENTDECKEN**

Home » [Inspirationen](#)

Alle **19**

 ÖV **4**

 Ausflüge **7**

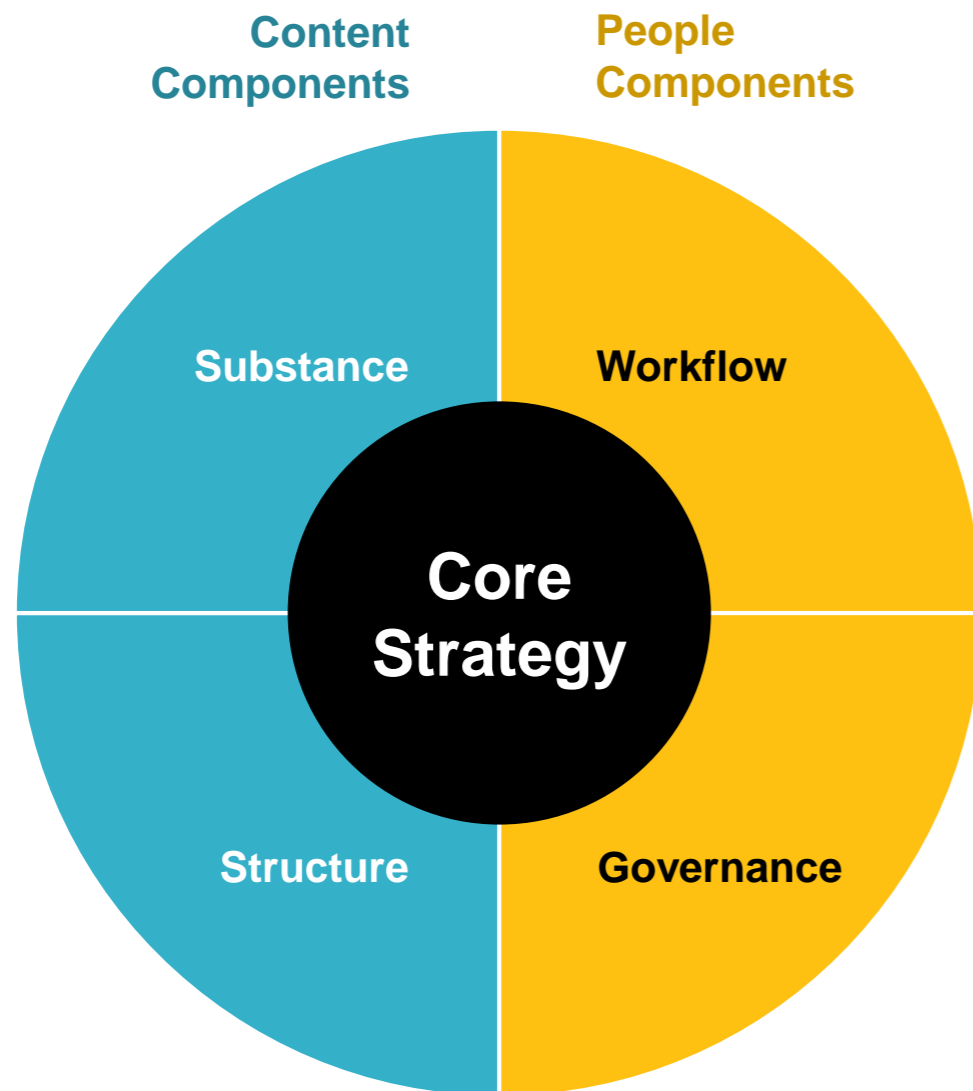
 Essen & Trinken **2**

 Lifestyle **6**



Lessons Learned in Headless Projekten

Content Strategie als MUST für Headless

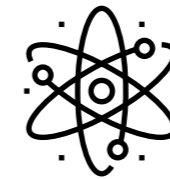


- **Core Strategy:** Zielerreichung, User-Bedürfnisse
- **Substance:** Was für Inhalte? Welche Informationen?
- **Structure:** Priorisierung, Organisation, Formatierung, Anzeige
- **Workflow:** Prozesse, Instrumente, Ressourcen
- **Governance:** Wer trifft Entscheidungen?

Erfolgsfaktoren



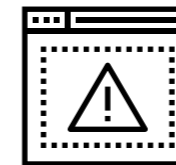
Content früh erfassen



Inhaltsmodell einfach halten



Inhaltsmodell mit allen
Stakeholdern abstimmen
(Konsistenz!)



Kanalspezifische Inhalte planen
(SEO, Metadaten)



Peter Köllner

SENIOR APPLICATION ARCHITECT

peter.koellner@unic.com

@ COPYRIGHT

**Alle Ideen sind urheberrechtlich
geschützt und geistiges Eigentum
der Unic-Gruppe.**
