

## Web-basierte Redaktionssysteme in den Geisteswissenschaften. Architekturen und Trends

Eckhart Arnold, Bayerische Akademie der Wissenschaften (arnold@badw.de)

Oktober 2014



# Überblick

---

1. Das Material
2. Organisatorische Bedingungen
3. Softwarearchitekturen
  1. Web, Netz- *oder* Desktop - Anwendung
  2. Monolithisch *oder* Werkzeugkasten
  3. Maßgeschneidert, Spartenlösung *oder* von der Stange
4. Datenstrukturen
  1. Monokultur (XML) *oder* Vielfalt (DSLs)
  2. Das Buch als Referenz



# 1. Das Material

---

## Kleine Typologie textartigen Materials

- Editionen
  - e.g. Gesammelte Werke Immanuel Kants
  - e.g. Notizbücher Theodor Fontanes
- Wörterbücher
  - e.g. Tibetisches Wörterbuch
  - e.g. Rheinisches Wörterbuch
- Kataloge und Lexika
  - e.g. Repertorium „Geschichtsquellen des Mittelalters“

Blatt 3



## 2. Organisatorische Bedingungen den DG

---

- kleine Entwicklerteams
- oft kurze Zeithorizonte (projektfinanziert!)
- relativ häufig: dezidierte Entwickler



# Organisatorische Bedingungen

Konsequenzen:

- fragiler Entwicklungsprozess
- Zusammenarbeit im Stil der „Komplizenschaft“: kurzzeitige intensive Zusammenarbeit, keine langfristige Verantwortung
- manchmal eigenwillige Systeme

Blatt 5



# 3. Softwarearchitekturen

---

1. **Web-, Netz- oder Desktop**
2. Monolithisch *oder* Werkzeugkasten
3. Maßgeschneidert, Spartenlösung *oder* von der Stange



# Desktopanwendungen

---

- Läuft auf dem Rechner des Nutzers
- Arbeitet auf dem lokalen Dateisystem

**Vorteile:** leistungsfähig und komfortabel

**Nachteile:**

- schwächere Unterstützung für Zusammenarbeit
- Installation notwendig; u.U. plattformabhängig
- getrenntes Redaktions- und Publikationssystem

Beispiel: Classical Text Editor ([cte.oeaw.ac.at/](http://cte.oeaw.ac.at/))

# Netzbasierte Anwendungen

---

- Client/Server-Architektur
- Daten werden zentral gespeichert

## Vorteile:

- Komfort von Desktopanwendungen
- Unterstützung für gemeinschaftliches Arbeiten leichter zu realisieren.

## Nachteile:

- meist Installation erforderlich
- Plattformunabhängigkeit schwerer zu realisieren



# Webanwendungen

---

- Client/Server-Architektur
- Client läuft im Browser

## Vorteile:

- Größte Plattformunabhängigkeit
- Keine Installation erforderlich
- einheitliches Redaktions- und Publikationssystem

## Nachteile:

- Aufwändigere GUI-Programmierung
- oft unkomfortablere Anwendungen
- u.U. Umständlicher Umgang mit Dateien

Beispiel: OpenWISSKI ([wiss-ki.eu/node/163](http://wiss-ki.eu/node/163))

Blatt 9



# Trend: Vom Desktop zum Web

- Webanwendungen umso geeigneter je größer, verteilter und diffuser die Forschergruppe
- Web-Technik nicht immer geeignet, e.g. WYSIWYG-Anwendungen
- Technischer Gesamttrend zum Web  
Folge: Webanwendungen bald nicht mehr im Nachteil (?)

Blatt 10



# 3. Softwarearchitekturen

---

1. Web-, Netz- *oder* Desktop
2. **Monolithisch vs Werkzeugkasten**
3. Maßgeschneidert, Spartenlösung  
*oder* von der Stange



# Monolithische Architekturen

---

- Alle (für ein Forschungsvorhaben benötigten) Funktionen werden innerhalb eines (integrierten) Systems oder Frameworks realisiert

## Vorteile:

- Alles aus einem Guss
- Komponenten gut aufeinander abgestimmt

## Nachteile:

- „Herstellerabhängigkeit“
- Bindung an Vorgaben des Frameworks
- u.U. größerer Eigenentwicklungsaufwand

# Werkzeugkasten

---

- Benötigte Funktionen jeweils durch unterschiedliche Programme realisiert.

## Vorteile:

- Für jede Aufgabe jeweils die beste Software
- Ausfalls einzelner Komponenten gefährden nicht die Gesamtlösung

## Nachteile:

- Heterogenes System
- u.U. höherer Verwaltungs- und Pflegeaufwand

Beispiel: Ediarum ([www.bbaw.de/telota/software/ediarum](http://www.bbaw.de/telota/software/ediarum))

# Trend zum Werkzeugkasten

- Werkzeugkastenlösungen m.E. besser für fragile Entwicklungsprozesse
- Beispiele aus dem technischen Gesamttrend:
  - Open Source, insbes. Linux-Software
  - Web-Entwicklung:  
Plone->Django->Pylons

# 3. Softwarearchitekturen

---

1. Web-, Netz- *oder* Desktop
2. Monolithisch *oder* Werkzeugkasten
3. **Maßgeschneidert, Spartenlösung**  
**oder von der Stange**



# Maßgeschneidert

---

- Eigenentwicklung eines Systems für ein bestimmtes Forschungsprojekt
- Speziell auf die Bedürfnisse des Projekts zugeschnitten

## Vorteile:

- kaum Kompromisse erforderlich
- am ehesten Möglichkeit zum Experimentieren

## Nachteile:

- Zeitlicher Vorlauf
- Entwicklungskosten; eher für Langzeitprojekte
- Pflege nach Ende des Projektes fraglich

Beispiel: [ptolemaeus.badw.de](http://ptolemaeus.badw.de)

Blatt 16





# Spartenlösung

---

- Eigenentwicklung von Systemen für alle Forschungsprojekte eines Typs (z.B. Wörterbücher)

## Vorteile:

- Geringere Entwicklungskosten
- Größere Chance auf Nachhaltigkeit
- Synergieeffekte

## Nachteile:

- Komplexere Systemarchitektur

Beispiel: TAREs ([www.mhdwb.uni-trier.de/TAREs](http://www.mhdwb.uni-trier.de/TAREs))

Blatt 17



# Von der Stange

---

- Verwendung eines bestehenden Systems mit großer Entwicklergemeinschaft
- geringe Anpassungen (Templates, ggf. Plugins)

## Vorteile:

- leistungsfähiges, gut getestetes System
- Geringer Eigenentwicklungsaufwand
- Dynamische Weiterentwicklung durch die Community

## Nachteile:

- u.U. müssen Kompromisse gemacht werden
- Richtung der Weiterentwicklung nur bedingt steuerbar
- Ggf. hoher Anpassungsaufwand bei großen Versionssprüngen

Beispiel: Wien Geschichte Wiki ([www.wien.gv.at/wiki](http://www.wien.gv.at/wiki))

# Von der Stange: Fallbeispiel



www.geschichtsquellen.de/index.html

Bayerische Akademie der Wissenschaften

Repertorium  
„Geschichtsquellen des deutschen Mittelalters“

DFG

BSB Bayerische Staatsbibliothek  
Information in aller Linie

**Startseite**  
**Quellen:**  
Autoren  
Werke  
**Suche**  
**Thesauri:**  
Orte  
Schlagwörter  
Überlieferungen  
Klöster  
Heilige  
Personen  
**Über das Projekt**  
**Hilfe**  
**Impressum**

**Das Digitale Repertorium "Geschichtsquellen des deutschen Mittelalters"**

Das Repertorium "Geschichtsquellen des deutschen Mittelalters" ist ein bibliographisches und quellenkundliches Nachschlagewerk auf digitaler Grundlage zu den erzählenden Geschichtsquellen des mittelalterlichen Deutschen Reiches für die Zeit von ca. 750 bis 1500.

Es ist hervorgegangen aus dem damals lateinisch verfassten "Repertorium Fontium Historiae Medii Aevi" (11 Bände, 1962-2007). Die deutschen Geschichtsquellen dieses Werkes wurden von der Bayerischen Akademie der Wissenschaften beigesteuert und sie werden jetzt, in erweitertem Umfang, im Web digital publiziert und fortgesetzt. Dazu werden auch ständig neue Lemmata verfasst.

Die Aktualisierung der Daten erfolgt kontinuierlich und kritisch durch die Kommission für das Repertorium "Geschichtsquellen des deutschen Mittelalters" der Bayerischen Akademie der Wissenschaften. Die Bereitstellung erfolgt in Zusammenarbeit mit dem Münchener Digitalisierungszentrum/ Digitale Bibliothek und dem Zentrum für Elektronisches Publizieren (ZEP) der Bayerischen Staatsbibliothek München, die auch die digitale Umsetzung vorgenommen haben.

Berichte und Informationen über dieses Repertorium und die Kommission findet der Interessierte auf der Homepage der Kommission [www.repfont.badw.de](http://www.repfont.badw.de). Dort findet sich auch eine Link-Liste zur mittelalterlichen Quellenkunde.

**Einfache Suche im Freitext:**

  

[www.geschichtsquellen.de](http://www.geschichtsquellen.de)



# Fallbeispiel: Geschichtsquellen

Repertorium Geschichtsquellen  
des deutschen Mittelalters

- Quellen
  - Autoren
  - Werke
- Suche
- Thesauri
  - Orte
  - Schlagwörter
  - Überlieferungen
  - Klöster
  - Hellige
  - Personen
- Über das Projekt
- Hilfe
- Werkzeuge
  - Links auf diese Seite
  - Änderungen an verlinkten Seiten
  - Datel hochladen
  - Spezialseiten
  - Druckversion
  - Permanenter Link
  - Seiteninformationen
  - Attribute anzeigen

Dies ist eine Testinstallation. Sie demonstriert die Funktionalität eines *MediaWiki* in Kombination mit der Erweiterung *Semantic MediaWiki* am Beispiel des Repertoriums *Geschichtsquellen des deutschen Mittelalters*. Der Zugriff, insbesondere der Schreibende, ist noch relativ langsam, weil noch keine Maßnahmen zur Optimierung ergriffen wurden. Die Bedienung erfordert MediaWiki-Grundkenntnisse, die man sich z. B. erworben hat, wenn man sich schon einmal schreibend bei Wikipedia betätigt hat. Das Design entspricht derzeit dem Standard-Design der MediaWiki-Software, ließe sich aber beliebig anpassen, um sich vollständig von der typischen Wiki-Optik zu lösen.

### Anmerkungen

Mit der aktuellen Wiki-Installation sind alle grundlegenden Funktionalitäten des aktuellen Angebotes unter [geschichtsquellen.de](#) erreicht. Es sind, bis auf wenige Ausnahmen, alle **Quellen**, **Autoren** und **Literaturangaben** importiert. Dabei wurden auch die vorhandenen (TEI-codierten) semantischen Informationen übernommen und eine erste Kategorisierung vorgenommen (siehe weiter unten). Quellen und Autoren wurden miteinander verknüpft. Aufgrund des Umfangs des Repertoriums sind mit dem gegenwärtigen Stand *noch nicht alle semantischen Elemente aus den Ausgangsdaten erfasst* und verarbeitet. Daher fehlen noch einige Inhalte oder sind als Rohtext ohne Auszeichnungen vorhanden (z. B. die Informationen unter *Überlieferungen*). Gleiches gilt für Übersichtsseiten wie z. B. zur Stadt Rom oder zu einer Person wie Johannes von Buch, die nur in wenigen Fällen manuell angelegt wurden, aber beim endgültigen Import der Daten automatisch erzeugt werden sollen. Die Repertorienliteratur wurde vorerst in Form von einfachen Listen importiert (Zeitschriften, Reihen, Abgekürzt zitierte Werke, Nachschlagewerke, Kompendien, Quellenwerke, Sammelwerke, Periodika). Es fehlt eine direkte Verknüpfung von Literaturangaben und Quellen. Hierzu müssen noch grundlegende strukturelle Entscheidungen getroffen werden.

### Beispieleiten aus dem Repertorium

### Quellen und Autoren

- Der *Sachsenspiegel* von *Elke von Reggov*

Ein Projekt der BSB

BSB Bayerische Staatsbibliothek  
Bereitgestellt von der BSB

DFG  
Gefördert von der DFG

Neues Repertorium Geschichtsquellen vom Entwicklerteam der Bayerischen Staatsbibliothek mit *Semantic Media Wiki* realisiert



Bayerische  
Akademie der Wissenschaften

# Fallbeispiel: Geschichtsquellen

## Vorher:

- Maßgeschneiderte Werkzeugkastenlösung
- mühsam für die Redakteure
- kontinuierlicher Entwicklereinsatz notwendig

## Nachher:

- Monolithische Lösung „von der Stange“
- erheblich einfachere Bedienung
- attraktiveres Erscheinungsbild (mitgeliefert!)
- Einmaliger Entwicklereinsatz

Blatt 21



## 4. Datenstrukturen

---

1. **Monokultur (XML) oder Vielfalt (DSL) – von XML zu NoXML**
2. Das Buch als Referenz  
– Ein Abgesang



# Monokultur oder Vielfalt

---

Zwei Trends in der Softwareentwicklung:

1. **Programmierung**
  - Universalsprachen -> Sprachenvielfalt und domänenspezifische Sprachen
  - (strenge) statisch typisierte Sprachen -> agile dynamisch typisierte Sprachen
2. **Datenhaltung**
  - Standardmodell (e.g. SQL) -> Modellvielfalt (e.g. NoSQL)
  - Einheitssprache (XML) -> Vielfalt (JSON, Markdown, YAML...)

Blatt 23



# Von XML zu NoXML

## XML

```
<?xml version="1.0" encoding="UTF-8"?  
>  
<ui version="4.0">  
  <class>MainWindow</class>  
  <widget class="QMainWindow"  
name="MainWindow">  
    <property name="geometry">  
      <rect>  
        <x>0</x>  
        <y>0</y>  
        <width>400</width>  
        <height>300</height>  
      </rect>  
    </property>  
    <property name="windowTitle">  
      <string>MainWindow</string>  
    </property>  
    <widget class="QWidget"  
name="centralWidget"/>  
    <widget class="QMenuBar"  
name="menuBar">  
      <property name="geometry"> ...
```

## QML

```
ApplicationWindow {  
  visible: true  
  width: 640  
  height: 480  
  title: qsTr("Hello World")  
  
  menuBar: MenuBar {  
    Menu {  
      title: qsTr("File")  
      MenuItem {  
        text: qsTr("&Open")  
        onTriggered:  
          console.log("Open action triggered");  
      }  
      MenuItem {  
        text: qsTr("Exit")  
        onTriggered: Qt.quit();  
      }  
    }  
  }  
} ...
```





# Vor und Nachteile von XML

## Vorteile:

- Strenge Definition -> gute Fehlerkontrolle
- wichtige Standards in XML formuliert

## Nachteile:

- Kompliziert und nicht besonders leserlich
- Aufgeblähte Dateien
- Monopol bei XML-Editoren (Oxygen)

„Making humans edit xml is sadistic“

([docs.djangoproject.com/en/1.7/topics/templates/](https://docs.djangoproject.com/en/1.7/topics/templates/))



# Not only XML: Eine Prognose

- XML eher als Zwischen- oder Entstufe (Archivierung) der Textverarbeitung
- XML-Schicht künftig nur noch selten gegenüber dem Nutzer exponiert
- Entwicklung von DSLs (intuitivere Markup-Sprachen, reduziertes XML)

Blatt 26



# 4. Datenstrukturen

---

1. Monokultur (XML) oder Vielfalt (DSL)  
– von XML zu NoXML
  
2. **Das Buch als Referenz**  
– **Ein Abgesang**



# Heute: Das Buch als Referenz

---

- Gegenwärtig: Publikationsprozesse auf Druckausgaben hin orientiert.
- Druckpublikationsprozess als etablierter und wohlverstandener Prozess sehr robust
- Ergebnis (gedruckte Werke) relativ dauerhaft

Blatt 28



# In Zukunft: Buch als Option

---

- Ergebnis: Nicht Druckwerk, sondern Text als strukturierter Datenbestand
- Druckausgaben künftig nur eine mögliche Ausgabeform neben andern
- Zitation nach Absatznummern statt Seitenzahlen
- Referenzausgabe aus zwei Komponenten: 1) maschinenlesbar und 2) menschenlesbar
- Identität der Referenzausgabe durch URL

Blatt 29



# Die Zukunft der digitalen Geisteswissenschaften



„Junger Mann! Wie wollen Sie meinem Goethe-Seminar folgen, wenn Sie noch nicht die Grundvorlesung Elektrotechnik gehört haben?“

Bildquelle: Faust, R.: Peter Gorski. D. 1960. Gustav Gründgens und Uwe Friedrichsen TC: 00:49:55



Bayerische  
Akademie der Wissenschaften