

Werkzeuge zur Textpublikation

Alexander Czmieł

czmieł@bbaw.de

@alexczmieł

Lifecycle einer Digitalen Edition

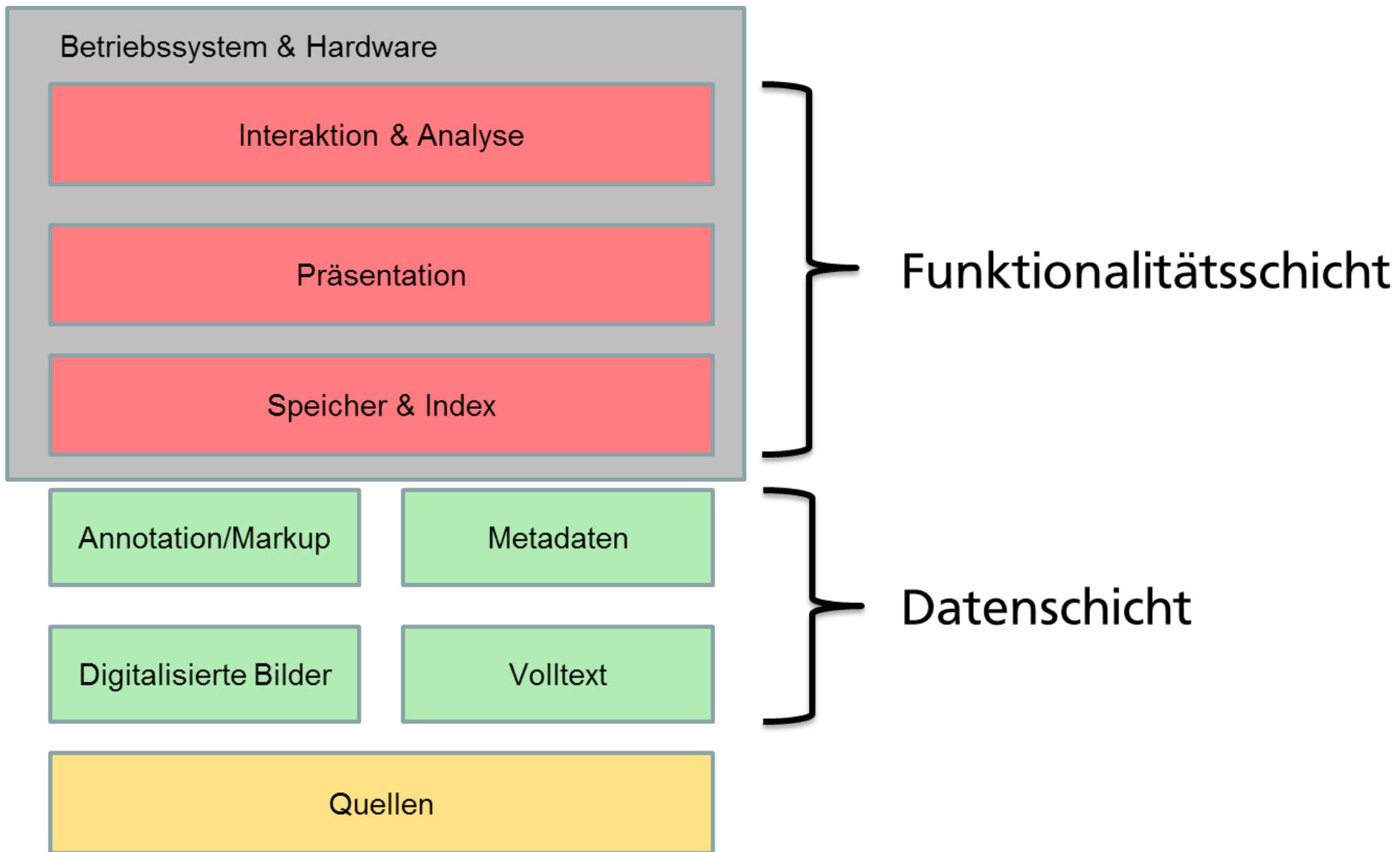
1. Analyse der Quellen
2. Anforderungsanalyse der digitalen Ressource, *Requirement Engineering*
3. Entwurf des Daten- / Dokumentenmodells, Auswahl von Standards
4. Auswahl, Anpassung bzw. Entwicklung von Tools
5. Digitalisierung, Reproduktion, Transkription
6. Aufsetzen und Betreuen der Server
7. Konzept, Design und Umsetzung der Web-Publikation
8. Vorbereitung für Langzeitverfügbarkeit / -archivierung
9. Betreuung und Wartung nach Projektende („*data curation*“)

Lifecycle einer Digitalen Edition

- 1. Analyse der Quellen**
- 2. Anforderungsanalyse der digitalen Ressource, *Requirement Engineering***
- 3. Entwurf des Daten- / Dokumentenmodells, Auswahl von Standards**
- 4. Auswahl, Anpassung bzw. Entwicklung von Tools**
- 5. Digitalisierung, Reproduktion, Transkription**
6. Aufsetzen und Betreuen der Server
7. Konzept, Design und Umsetzung der Web-Publikation
8. Vorbereitung für Langzeitverfügbarkeit / -archivierung
9. Betreuung und Wartung nach Projektende („*data curation*“)

Lifecycle einer Digitalen Edition

1. **Analyse der Quellen**
2. **Anforderungsanalyse der digitalen Ressource, *Requirement Engineering***
3. **Entwurf des Daten- / Dokumentenmodells, Auswahl von Standards**
4. **Auswahl, Anpassung bzw. Entwicklung von Tools**
5. **Digitalisierung, Reproduktion, Transkription**
6. Aufsetzen und Betreuen der Server
7. **Konzept, Design und Umsetzung der Web-Publikation**
8. Vorbereitung für Langzeitverfügbarkeit / -archivierung
9. Betreuung und Wartung nach Projektende („*data curation*“)



Grundsätzliches zur Publikation von XML

- Single Source Prinzip
 - eine Quelle, verschiedene Ausgabeformate
- Einfache vs. Komplexe Lösungen
- Selbst bauen vs. Vorlagen verwenden
- Statisch vs. Dynamisch
- Lokal vs. Serverlösung
- Publikation persistent referenzierbar machen und lizenzieren!

Creative Commons



Namensnennung (BY):

Der Name des/der Urhebers/in muss in der von ihm/ihr festgelegten Weise genannt werden. Dieses Kürzel muss immer angegeben werden, außer jemand möchte auf alle Recht verzichten.



Keine Bearbeitung (ND):

ND steht für No Derivates, also keine Bearbeitung. Das Werk darf nicht verändert werden.



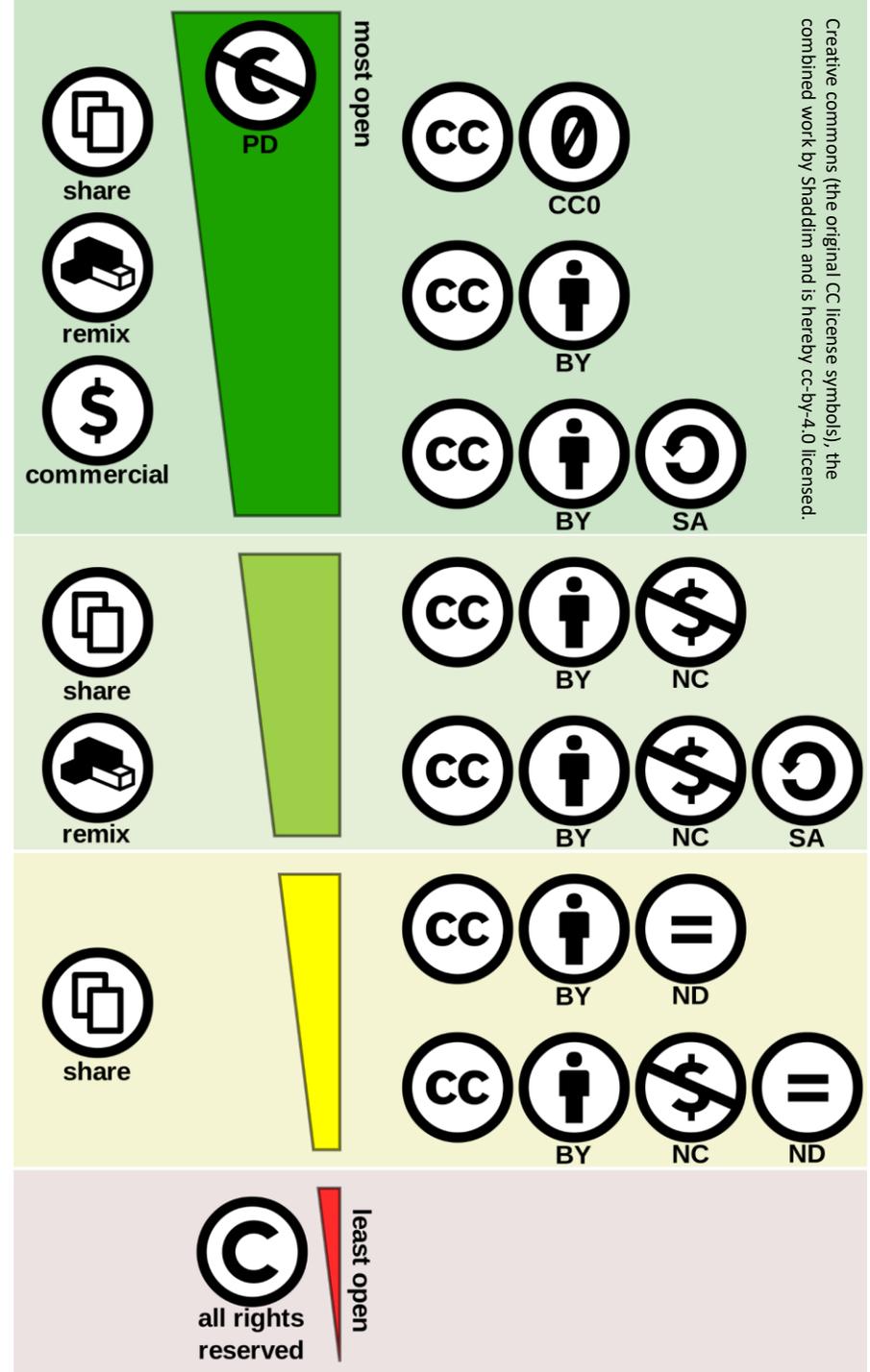
Nicht kommerziell (NC):

NC steht für Non-Commercial. Es ist keine kommerzielle Nutzung erlaubt.



Weitergabe unter gleichen Bedingungen (SA):

Der Inhalt darf unter gleichen Bedingungen (Share Alike) weitergegeben werden. Das heißt, wenn Werke mit diesem Code verwendet werden, müssen die daraufhin neu entstandenen Werke mit vergleichbaren oder identischen Lizenzbedingungen weitergegeben werden.



<https://creativecommons.org/>

Beispiel Weber-Gesamtausgabe

```
<publicationStmt>
  <publisher>
    <name>Carl-Maria-von-Weber-Gesamtausgabe</name>
    <address>
      <street>Hornsche Str. 39</street>
      <postCode>32756</postCode>
      <placeName>
        <country>D</country>
        <settlement>Detmold</settlement>
      </placeName>
    </address>
  </publisher>
  <idno type="WeGA" n="293"/>
  <availability>
    <licence target="http://creativecommons.org/licenses/by/4.0/">Creative
      Commons Attribution 4.0 International (CC BY 4.0)</licence>
  </availability>
</publicationStmt>
```

TEI-Dokumente publizieren

- XML
- XML + CSS
- XML + XSLT
- XML + XQuery
- XML + XML-Datenbanken
- XML + Publikationsframeworks

XML

- Einfachste Lösung
- Kein definiertes Zielformat
- Gute und richtige Art der Publikation
 - Transparenz der Datengrundlage
 - Weiterverarbeitung und Integration in andere Projekte
 - Langzeitarchivierung
- Immer XML publizieren!

XML & CSS

Was ist CSS?

“Cascading Style Sheets (CSS) is a simple mechanism for adding style (e.g., fonts, colors, spacing) to Web documents.”

→ Elemente anordnen und verschönern

- Spezifikation: <https://www.w3.org/TR/CSS/>
- Tutorial: <http://www.w3schools.com/css>
- Beispiele: <http://www.csszengarden.com/>
- Cheat Sheet:
<https://www.smashingmagazine.com/wp-content/uploads/images/css3-cheat-sheet/css3-cheat-sheet.pdf>

Selector



h1 {

font-family: Arial



}

Property

Value



Declaration-block

CSS-Beispiel

```
html {  
  background-color: #c0b7a8; /*yellow*/  
}  
body {  
  background-color: #F8FBEB;  
  font-family: Georgia, Times, serif;  
  color: #333333;  
}  
h4 {  
  font-size: 1.3em;  
  margin-top: 0em;  
  color: #32649c  
}
```

XML & CSS Beispiel (Übung)

```
<?xml-stylesheet type="text/css" href="mein_style.css"?>
```

```
<TEI xmlns="http://www.tei-c.org/ns/1.0">
```

```
  <teiHeader>
```

```
    <fileDesc>
```

```
      <titleStmt>
```

```
        .
```

```
        .
```

```
        .
```

```
</TEI>
```

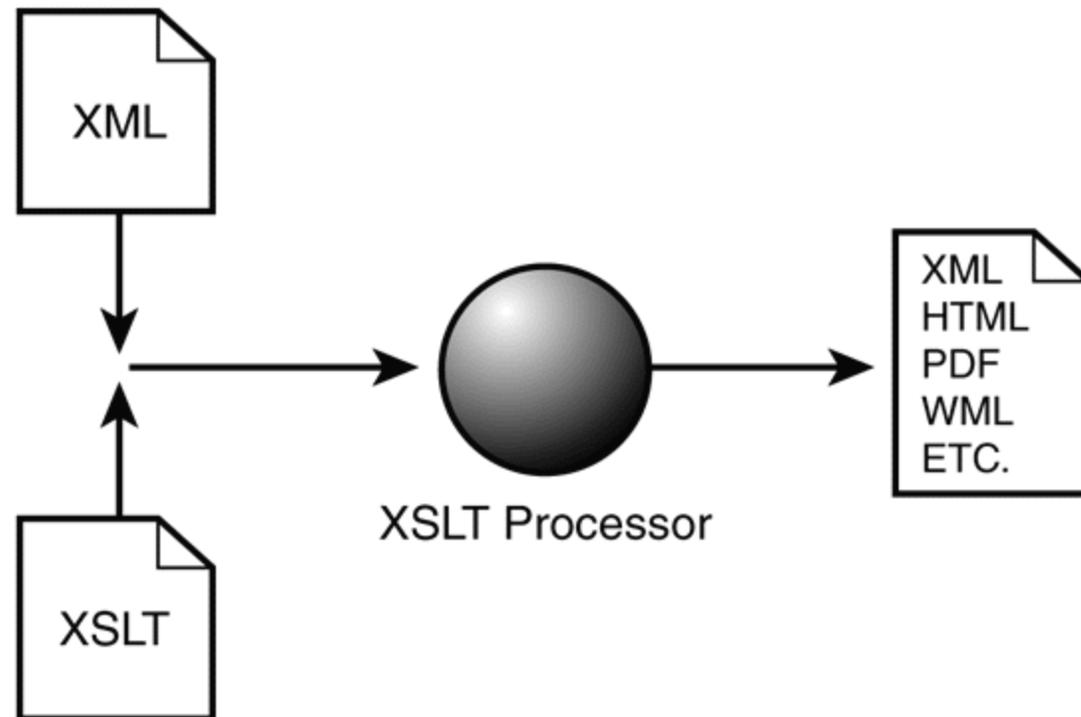
CSS Vor- und Nachteile

- Sehr einfache Lösung: funktioniert wie HTML & CSS
- Vorteile:
 - schnell
 - Semantische Struktur des Dokuments bleibt erhalten
 - Läuft in jedem modernen Webbrowser
- Nachteile:
 - statisch
 - Nur ein Zielformat
 - Keine Inline und keine eingebetteten CSS-Anweisungen möglich
 - Keine Links möglich

XML & XSLT

Was ist XSLT

- eXtensible Stylesheet Language Transformation
- W3C Standard seit 1999
- Sprache zur Transformation von XML-Dokumenten
- XSL ist auch XML
- Transformationsszenario:



XSLT-Beispiel

```
<xsl:stylesheet version="2.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">  
  <xsl:output method="text"/>  
  <xsl:template match="/">  
    Article - <xsl:value-of select="/Article/Title"/>  
    Authors: <xsl:apply-templates select="/Article/Authors/Author"/>  
  </xsl:template>  
  <xsl:template match="Author">  
    - <xsl:value-of select="." />  
  </xsl:template>  
</xsl:stylesheet>
```

Vor- und Nachteile

- Benötigt XSLT Prozessor
 - Saxon, Xalan, MSXML usw.
- Vorteile:
 - Mächtiger als CSS
 - Umstrukturierung
 - Unterschiedliche Ausgabeformate:
 - HTML, PDF (XSL-FO), ePub, RDF u.a. textbasierte Formate
- Nachteile:
 - Statisch
 - Man muss das Zielformat beherrschen
- Mehr dazu morgen...

XSLT-Frameworks

- TEI-Stylesheets von Sebastian Rahtz
 - <https://github.com/TEIC/Stylesheets>
 - Dokumentation: <http://www.tei-c.org/release/doc/tei-xsl/>
 - Integriert in Oxygen
 - Kann man selbst anpassen, wenn man möchte
 - <http://www.tei-c.org/Tools/Stylesheets/customize.xml>
- Andere Stylesheets
 - J. Hellingman → <https://github.com/jhellingman/tei2html>
 - TEI Lite nach HTML & ePub als Zielformat

Anwendung TEI-Stylesheets (Übung)

- Oxygen Transformationsszenario
- OxGarage - <http://www.tei-c.org/oxgarage/>
 - „...is a web, and RESTful, service to manage the transformation of documents between a variety of formats.”

TEI-Boilerplate

- <http://teiboilerplate.org/>
- ...is a lightweight solution for publishing styled TEI P5 content directly in modern browsers.
- Integriert einfache XSLT, CSS und JavaScript-Anweisungen
- Konvertiert wenige TEI-Elemente (z.B. <ptr>, <ref>, <figure> oder @rend) in naheliegende HTML-Elemente

XML & XQuery (& XSLT)

Was ist XQuery?

- XML Query Language bezeichnet eine vom W3C spezifizierte Abfragesprache für XML-Datenbanken (<http://www.w3.org/TR/xquery/>)
- “erweitertes XPath“
- Funktionale Programmiersprache (Turing complete)
- Benötigt einen XQuery Prozessor, meist eine XML-Datenbank
- Keine XML-Syntax
- Wird benutzt zur Selektion, Extraktion, Konstruktion, Berechnungen usw.
- ...und macht sehr viel Spaß, damit zu arbeiten! :)

XQuery-Beispiel

```
xquery version "1.0";
```

```
declare namespace xslfo="http://exist-db.org/xquery/xslfo";
```

```
let $col := collection("/db/mgh/data")
```

```
let $docID := request:get-parameter("docID", "")
```

```
let $qString := request:get-parameter("qString", "")
```

```
let $text := $col//text[.//idno = $docID]
```

```
let $pdf1 := transform:transform($text, doc('/db/mgh/scripts/pdf.xsl'), $para)
```

```
let $pdf := xslfo:render($pdf1, 'application/pdf', (), doc('/db/mgh/scripts/fop-cfg.xml'))
```

```
return response:stream-binary($pdf, 'application/pdf', concat($docID, '.pdf'))
```

XML-Datenbanken

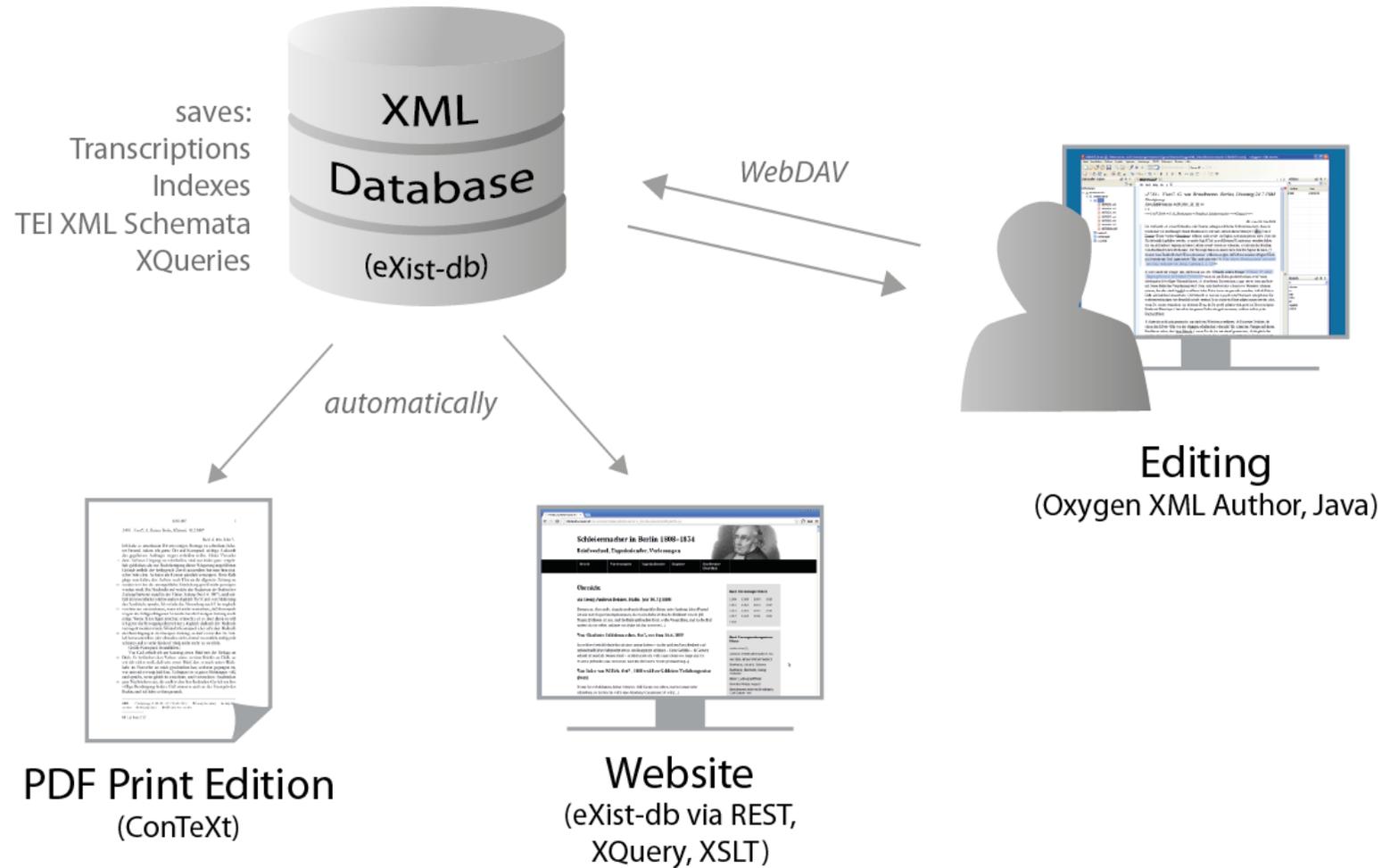
- Dient zur Speicherung und effizienten Verarbeitung von XML-Dokumenten mit X-Technologien (aber auch anderen Dateitypen)
- Analyse von XML-Dokumenten – eins, viele oder Fragmente
- Behandelt semi-strukturierte Informationen in natürlicher Art und Weise (keine Tabellen, Struktur bleibt erhalten)
- NoSQL database
- Unterstützt Validierung
- Extensive list of native XML databases:
<http://www.rpbouret.com/xml/ProdsNative.htm>

eXistdb - <http://exist-db.org>

- Ein Beispiel für einen XQuery-Prozessor
- ...und einen XPATH, XSLT, XUpdate, XPROC-Prozessor
- ...und eine native XML-Datenbank
- ...und ein Web Server
- ...und eine Volltext-Suchmaschine
- ...und eine XML/XSLT/XQuery-Entwicklungsumgebung
- ...und eine Heimat für viele Digitale Editionen



ediarum Workflow



Publikations-Frameworks & andere Tools

- Vorgefertigte Lösungen
- Unterschiedliche Technologien
- Unterschiedliche Restriktionen
 - Mögliche Elemente
 - Mögliche Ausgabeformate
 - Anwendungsfälle

Institutionelle „Werkzeugkästen“

- Graz – GAMS - <http://gams.uni-graz.at/>
- Berlin – ediarum - <http://www.bbaw.de/telota/software/ediarum>
- London – KILN - <https://kcl-ddh.github.io/kiln/>
- Den Haag – elaborate - <https://www.elaborate.huygens.knaw.nl>
- Rom – DENQ

Publiktaionsframeworks

- Versioning Machine - <http://v-machine.org/>
- SynopsX - <https://synopsx.github.io/>
- eXtensible Text Framework (XTF) - <http://xtf.cdlib.org/>
- Edition Visualization Technology (EVT) - <http://evt.labcd.unipi.it/>
- TEICHI - <http://www.teichi.org/>

TAPAS 2.0

- Publikationsrepositorium für TEI-Projekte
 - Vor allem auch kleinere, ohne institutionelle Anbindung.
- Benutzt TEI-Stylesheets und TEI-Boilerplate
- Registrierungspflichtig
- Integriert momentan ca. 20 Projekte
- <http://beta.tapasproject.org/>

TEI Processing Model Toolbox

- Basiert auf TEI Simple - <https://github.com/TEIC/TEI-Simple>
- Und eXistdb
 - Über das Packagerepository von eXistdb installierbar
- Bietet mehrere Ausgabeformate
- Anpassungen der Ausgabe über TEI-ODD (mehr dazu am Freitag)
- <http://showcases.exist-db.org/exist/apps/tei-simple/>

TEI-Wiki: Publishing and delivery tools

- [http://wiki.tei-c.org/index.php/Category:Publishing and delivery tools](http://wiki.tei-c.org/index.php/Category:Publishing_and_delivery_tools)