Summer School “Digitale Edition”

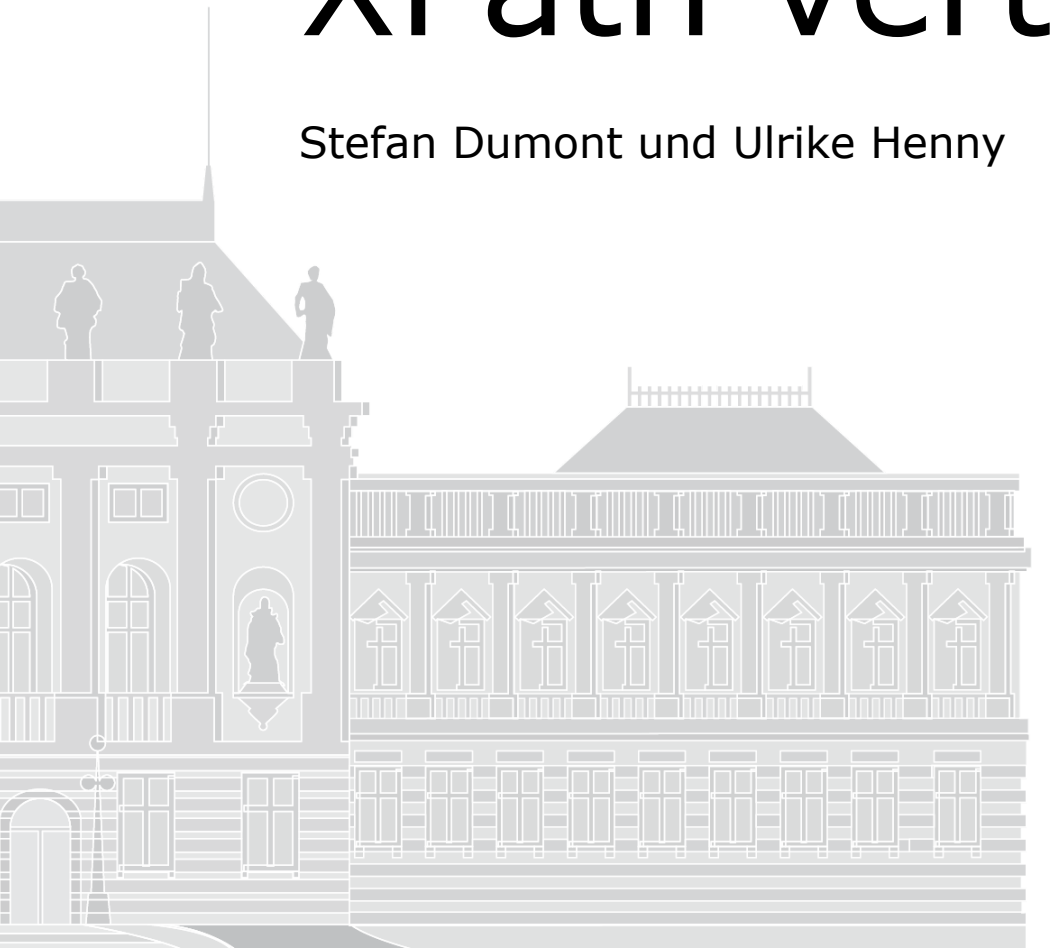
Erschließung geisteswissenschaftlicher Quellen mit digitalen Methoden

5.-9. September 2016

Zentrum für Informationsmodellierung
Austrian Centre for Digital Humanities
Elisabethstraße 59/III, SR 81.31

XPath vertieft

Stefan Dumont und Ulrike Henny



Programm:

Wiederholung XPath
Weitere XPath-Funktionen
Reguläre Ausdrücke
Namensräume

Wiederholung
XPath

Übungsdatei: briefcorpus.xml

Bewegung im Baum

Lokalisierungsschritte

„gehe von hier nach da“

Knotentests

„bist Du der, den ich suche?“

Lokalisierungspfade

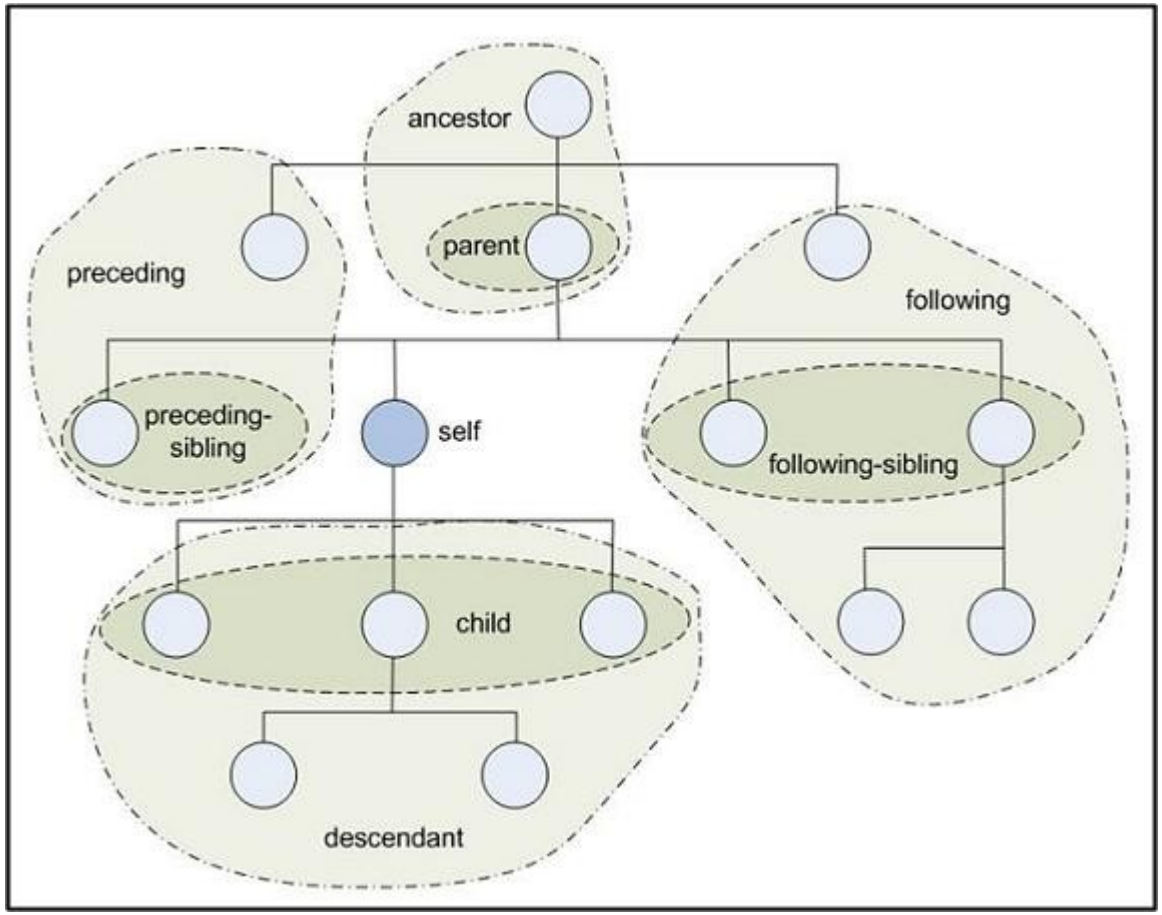
[Schritt]/[Schritt]/[Schritt]

/teiCorpus/teiHeader/titleStmt

„Kontext“

absolute Pfade (vom Dokument ausgehend) vs.
relative Pfade (vom aktuellen Kontext ausgehend)

Bewegung im Baum: Achsen



Achsen: abgekürzte Syntax

self::	.
child::	/Elementname
parent::	/..
descendant-or-self::	//Elementname
attribute::	/@Attributname
Knotentest, beliebiger Elementname	*

Knotentests

Bezeichnung	Bedeutung
node()	**Jeder Knotentyp**
text()	**Nur Textknoten**
comment()	Nur XML-Kommentare, d.h. <!- - Das ist ein Kommentar - ->
*	Jeder Element-Knoten
@*	Jeder Attribut-Knoten
@abc	**Attributknoten mit dem Namen „abc“**
abc	**Elementknoten mit dem Namen „abc“**
processing-instruction()	Verarbeitungsanweisungen / Processing instructions (PI) <?name wert ?>

XPath – Operatoren

Operator	Bedeutung
=	gleich (Vergleich zweier Werte)
!=	ungleich (Vergleich zweier Werte)
< bzw. lt	kleiner als (Vergleich zweier Werte). Maskierung innerhalb von XSLT erforderlich.
> bzw. gt	größer als (Vergleich zweier Werte). Maskierung innerhalb von XSLT zu empfehlen.
<= bzw. le	kleiner als oder gleich (Vergleich zweier Werte). Maskierung innerhalb von XSLT erforderlich.
>= bzw. ge	größer als oder gleich (Vergleich zweier Werte). Maskierung innerhalb von XSLT zu empfehlen.
and	logische Und-Verknüpfung (beide Ausdrücke)
or	logische Oder-Verknüpfung (einer von beiden Ausdrücken)

Wiederholungsübungen

1) Aufgabe: Alle Brieftitel auslesen

2) Aufgabe: Alle verschiedenen Absendeorte

3) Alle Briefe nach 1. Januar 1818

4) Alle Personennamen, die mit „B“ anfangen

Weitere XPath-Funktionen

Funktionen

Syntax einer Funktion:

```
funktionsname($argument1, $argument2, $argument3?)
```

- Funktionen führen hinter ihrem Namen immer Klammern!
- In den Klammern wird eine festgelegte Anzahl von Argumenten erwartet
- Die Argumente können unterschiedliche Datentypen erwarten, z.B. Zeichenkette (string) oder Knoten (node)
- Bestimmte Argumente können optional sein (oben: „?“)
- Online kann man in Funktions*referenzen* nachschlagen, welchen Zweck bestimmte Funktionen haben und welche Argumente sie erwarten
- Gute Referenz bei data2type (http://goo.gl/8HeGSj)

Funktionen

concat(string, string, …)

Verknüpfung mehrerer Zeichenketten

→ Aufgabe: Briefnummern auslesen und jeweils ein „Nr.“ voranstellen

string-length(string)

Länge der Zeichenkette

→ Aufgabe: Wie lang ist die längste (max()-Funktion) Grußformel (<salute>), die vorkommt?

Funktionen

substring-before(string, string)

gibt den Teil des ersten Strings zurück, der vor dem Vorkommen des zweiten steht

→ Aufgabe: Alle Personennamen, die ein Komma enthalten auswählen und nur den Familiennamen vor dem Komma anzeigen:

substring-after(string, string)

gibt den Teil des ersten Strings zurück, der nach dem Vorkommen des zweiten steht

Funktionen

substring(string, number, number)

Teilzeichenkette ab einer bestimmten Position, bis zu einer bestimmten Position

→ Aufgabe: Initialen der Brief-Empfänger ausgeben

normalize-space(string)

Entfernt „überflüssigen“ Leerraum in einer Zeichenkette

→ Aufgabe: Brieftitel auslesen und Leerraum normalisieren:

Funktionen

matches(string, regex)

bestimmt, ob ein String einem bestimmten Muster entspricht

→ Aufgabe: Alle Attribute @key, die ein Leerzeichen enthalten

Funktionen

tokenize(string, regex)

teilt eine Zeichenkette in eine Reihe von Teilzeichenketten auf, der reguläre Ausdruck bestimmt einen Teilstring als Teiler

→ Aufgabe: Ergebnis von vorhergehender Aufgabe in einzelne Werte aufteilen

Funktionen

replace(string, regex, string)

ersetzt alle Teile eines Strings, die einem bestimmten Muster entsprechen, mit einem Ersetzungsstring

→ Aufgabe: Alle „ä“ durch „ae“ ersetzen

Reguläre Ausdrücke

Reguläre Ausdrücke

- Manche XPath-Funktionen verwenden reguläre Ausdrücke
- Mächtig: Verwendung von Mustern für Zeichenketten
- Beispiele:
  - Graz → findet Graz
  - Gr[a-z]z → findet z.B. Graz, Griz, Gruz usw.
  - B.*k → findet z.B. Buschenschank, oder Bank
  - Graz(er)? → findet Graz, aber auch Grazer
  - 01\.01\.20[0-9]{2}
  - → findet z.B. 01.01.2010, 01.01.2013, 01.01.2020

Reguläre Ausdrücke

- **Reservierte Zeichen** (mit \ maskieren, wenn wörtlich gemeint)

.	Beliebiges Zeichen
^	Anfang einer Zeichenkette
$	Ende einer Zeichenkette

- **Zeichenauswahl:** [abc], [a-z], [0-9]

Reguläre Ausdrücke

- **Quantoren**

  Ohne kommt genau einmal vor

  ? kein- oder einmal

  + einmal oder mehrmals

  * beliebig oft

  {n} n mal

- **Gruppierung** mit (…)

Regex-Übungen

- Aufgabe: Regex-Ausdruck, um in einem Dokument sowohl „May“ als auch „Mai“ zu finden

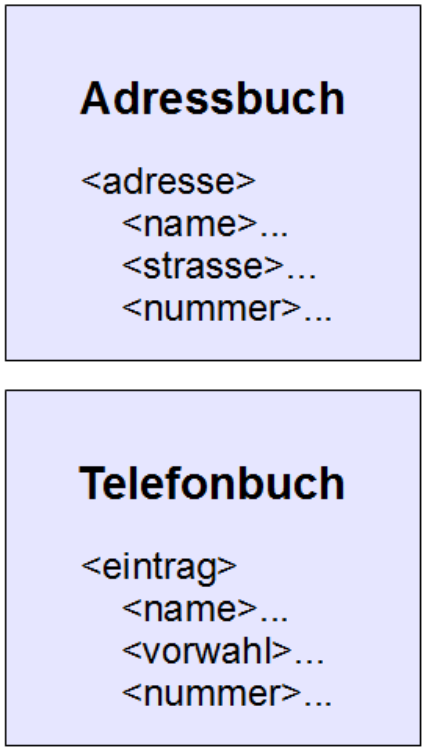
- Aufgabe: Alle maschinenlesbaren Datumsangaben, die in einen Mai fallen (unabhängig vom Jahr)

- Aufgabe: Alle Personennamen in den Brieftexten, die mit „L“ anfangen und mit Vokal enden

Namensräume

Namensräume

Wo liegt das Problem?



Namensräume

- Element- und Attributnamen sind Teil eines Namensraumes
- Namensraum:
  - Sammlung von Namen zu einem Gegenstandsbereich
  - Identifikation über URI
- Qualifzierter Elementname: <präfix:name>, <dc:title>
- Deklaration:
  - <element xmlns=''URI''>…
  - <präfix:element xmlns:präfix=''URI''>…
- Häufige Fehlerquelle!

Namensraum der TEI

TEI-Namensraum: http://www.tei-c.org/ns/1.0

TEI-XML Datei:

```
<TEI xmlns="http://www.tei-c.org/ns/1.0" xml:id="b00012">
    <teiHeader>
        <fileDesc>
            <titleStmt>
                <title>
                    <idno>B 05/02/26-1</idno>An Heinrich Wilhelm von Harnier, 26. Feb
            </titleStmt>
            <publicationStmt>
                <p>Akademie der Wissenschaften und der Literatur | Mainz</p>
            </publicationStmt>
            <sourceDesc>
                <msDesc rend="manuscript">
                    <msIdentifier>
                        <institution>Universitätsbibliothek Amsterdam</institution>
                        <collection>Handschriften, Sammlung Diederichs</collection>
                        <idno>
                            <idno type="shelfmark">91 Dv2</idno>
                        </idno>
```

Namensräume - Beispiel

```
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:dc="http://purl.org/dc/elements/1.1/">

   <rdf:Description
         rdf:about="
   http://de.wikipedia.org/w/index.php?title=Reise&amp;oldid=12
   9613470
   ">

        <dc:title>Reise</dc:title>
        <dc:type>document</dc:type>
        <dc:contributor>Wikipedia, Die freie
   Enzyklopädie</dc:contributor>
        <dc:creator>Wikipedia-Autoren</dc:creator>
        <dc:date>18. April 2014</dc:date>

   </rdf:Description>

</rdf:RDF>
```

Vielen Dank!