# Textnavigation mit XPath

Ulrike Henny
&
Patrick Sahle

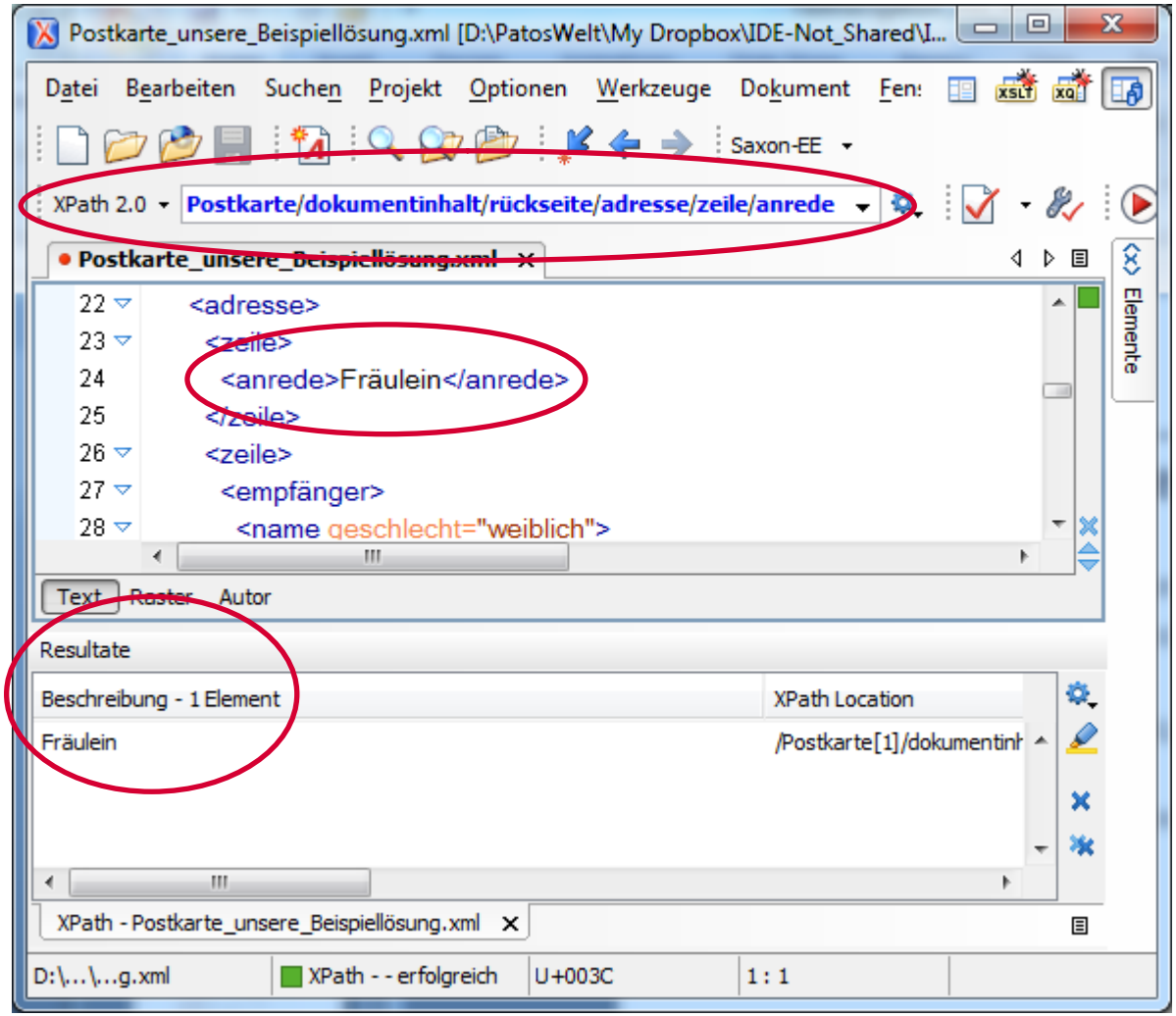
# Fahrplan

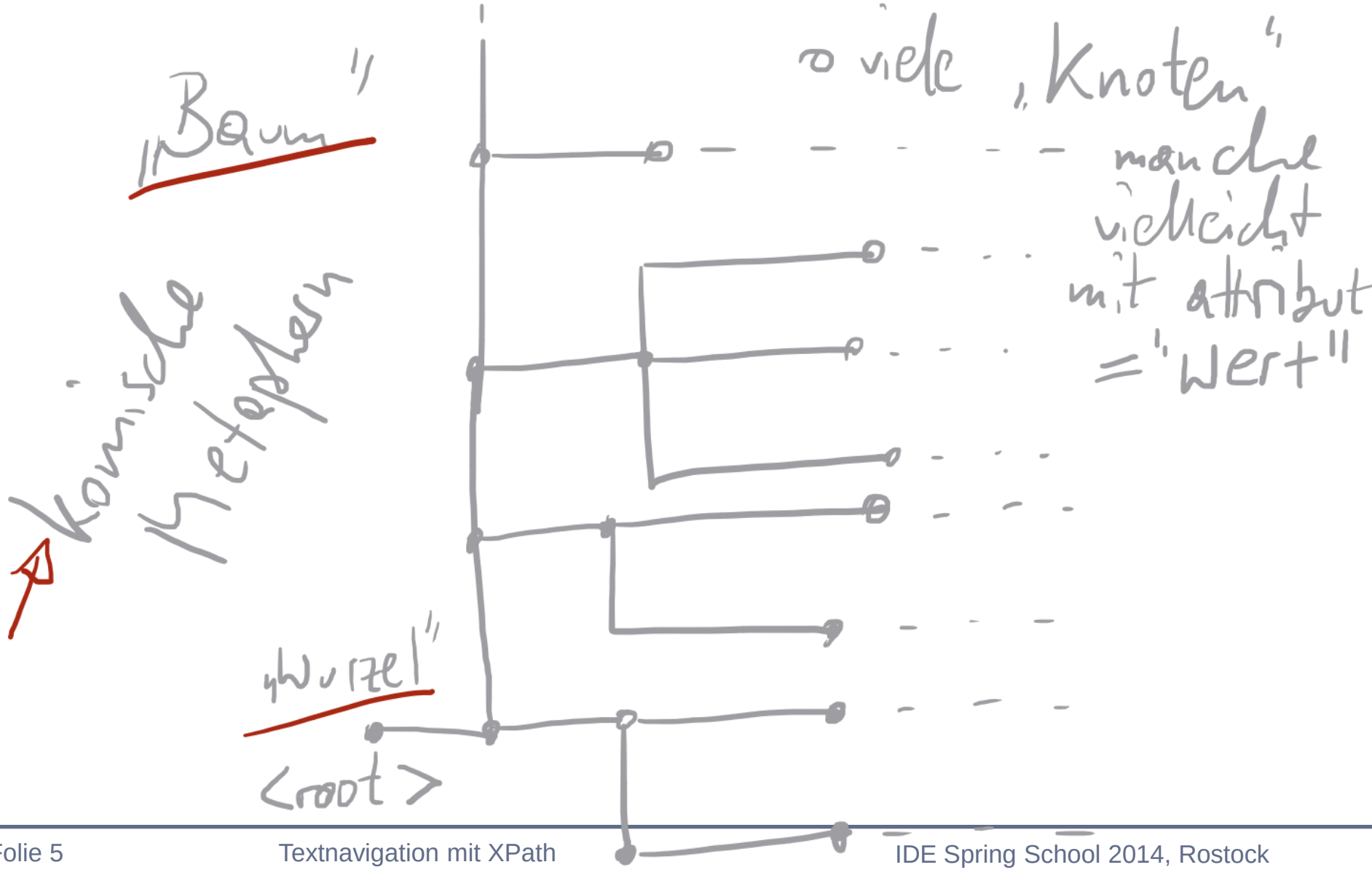
- XPath: wieso – weshalb – warum?
- XPath im Editor
- XML als Baum
- XPath: Grundkonzepte
- XPath: der erste Baukasten
- XPath: gemeinsame Übungen
- XPath: der weitere Baukasten
- XPath: Einzelübungen

# Xpath: wieso – weshalb – warum?

- Daten und Datenstrukturen verstehen und analysieren
- Daten mit anderen Technologien auswählen und verarbeiten

- Auswählen von Knoten, Knoten-Mengen, Attributwerten, Textinhalten, Textteilen
- Vergleichen, Zählen, Berechnen, Umwandeln etc.


Postkarte_unsere_Beispiellösung.xml [D:\PatosWelt\My Dropbox\IDE-Not_Shared\I...
Datei Bearbeiten Suchen Projekt Optionen Werkzeuge Dokument Fen:
Saxon-EE
XPath 2.0
Postkarte/dokumentinhalt/rückseite/adresse/zeile/anrede
Postkarte_unsere_Beispiellösung.xml
22 <adresse>
23 <zeile>
24 <anrede>Fräulein</anrede>
25 </zeile>
26 <zeile>
27 <empfänger>
28 <name geschlecht="weiblich">
Elemente
Text Raster Autor
Resultate
Beschreibung - 1 Element
XPath Location
Fräulein
/Postkarte[1]/dokumentinh
XPath - Postkarte_unsere_Beispiellösung.xml
D:\...\...g.xml
XPath - - erfolgreich
U+003C
1 : 1


„Baum"
„viele „Knoten"
manche vielleicht mit attribut ="wert"
Ikonische Metaphern
„Wurzel"
<root>

```
<Postkarte>
   <bibliographischeAngaben> ... </bibliographischeAngaben>
   <dokumentinhalt>
      <vorderseite>
         <titel><ort>Dortmund</ort>. Hauptbahnhof.</titel>
      </vorderseite>
      <rückseite>
         <adresse>
            <zeile><anrede>Fräulein</anrede></zeile>
            <zeile><empfänger>
                  <name geschlecht="weiblich">
                     <vorname>Margarete</vorname>
                     <nachname>Grogorenz</nachname></name></empfänger>
            </zeile>
            <zeile><ort>Berlin</ort> S.O. 36</zeile>
            <zeile>Wienerstr. 27.</zeile>
         </adresse>
         <haupttext>
            <zeile><anrede>Mein liebes <adressat geschlecht="weiblich"><name><vorname
                typ="koseform"  norm="Margarete">Gretchen</vorname></name></adressat>
                !</anrede> </zeile>
            <zeile>Soeben bin ich "<ironisch>glücklich</ironisch>" in <ort>Dortmund</ort></zeile>
            <zeile>angekommen und denke gleich an Dich!!</zeile>
```

# ... als Baum?

# XPath Grundkonzepte

- XPath erlaubt die Konstruktion von Pfadausdrücken
- Pfade bestehen aus Ketten von Pfadschritten
- Es gibt „Achsen“, über die man sich im Baum bewegt
- Der letzte bzw. äußerste Schritt bestimmt, was ein XPath-Ausdruck zurückgibt
  - Pfade werden von vorne nach hinten abgearbeitet
  - Klammern von innner nach außen aufgelöst

# XPath Grundkonzepte, Pfadschritte

- Schritt/Schritt/Schritt
- Postkarte/dokumentinhalt/rückseite/adresse/zeile/anrede

# XPath Grundkonzepte, Achsen

- horizontale und vertikale Achsen
- Achse::Knotentest
- * für beliebige Elemente

parent | ancestor | ancestor-or-self

child | descendant | descendant-or-self

preceding | preceding-or-self | preceding-sibling

following | following-or-self | following-sibling

attribute

self

```
child::Postkarte/child::dokumentinhalt/child::rückseite/
    child::adresse/child::zeile/child::anrede
```

# XPath Grundkonzepte, Achsen

abgekürzte Syntax

- child::Elementname = /Elementname
- parent::* = /..
- descendant-or-self::Elementname = //Elementname
- attribute::Attributname = /@Attributname
- self = .

# XPath Grundkonzepte, Rückgabe

ein XPath-Ausdruck kann verschiedene Dinge zurückgeben ...

- Knoten
- Knoten-Set
- Sequenz
- Wahrheitswert
- Text, String, Name (z.B. von Elementen)
- Zahl

# XPath: der erste Baukasten

- Verkettung von Schritten
- Achsen
- Zusammengefasste Ausdrücke: ( ... )
- Bedingungen
  - Knotentest[Bedingung]
    - //zeile[anrede]
    - //zeile[not(@typ='vordruck')]
- Operatoren
  - and or |
  - = != < >
  - + - * div %
- Funktionen

# XPath: Funktionen

- Funktionen können mit ihrem Namen aufgerufen werden.
- Sie bestehen aus Ihrem Namen und runden Klammern: *funktion*()
- Manche Funktionen erwarten in der Klammer die Übergabe von "etwas"
  - das kann ein Knoten sein, ein Knotensatz, ein String, eine Zahl …
- Die Übergaben sind durch Kommata getrennt
  - funktion(parameter,parameter)
- Funktionen geben dann etwas zurück
  - das kann ein Wahrheitswert sein, eine Zahl, ein String, eine Sequenz …

# XPath: eine Funktion

contains(string,string)

- auf Deutsch: Enthält der erste String den zweiten? Ja oder nein?
- contains('Schnecke','ecke') → Rückgabe: true
- contains(//stempel[2]/datum,'1920') → Rückgabe: false
- //stempel[2]/datum/contains(.,'1920') → Rückgabe: false
- //stempel[contains(.,'20')] → das Element "stempel" wird zurückgegeben, das den String '20' enthält
- //*[contains(stempel,'2009')] → das Element wird zurückgegeben, das ein Element stempel enthält, das '20' enthält
- //haupttext/zeile[contains(anrede,'lieb')]/adressat
  - Beachten: Strings müssen in Hochkomma stehen, Elemente nicht!

# XPath: der erste Baukasten, Funktionen

- count()
  - zählt etwas, erwartet eine Sequenz, z.B. Ein Knoten-Set
  - count(XPath)
- position()
  - gibt die Position eines Knotens an
- contains()
  - prüft, ob ein Element oder String einen anderen String enthält
  - contains(string,string)
- string-length()
  - zählt die Länge eines Strings – string-length(string)
- starts-with()
  - prüft, ob ein String mit einem anderen String beginnt
  - starts-with(string,string)
- not()
  - dreht einen Wahrheitswert um

# XPath: ~~erste~~ zweite Beispiele

- count(//ort)
  - Zähle die Menge der Elemente *Ort* in beliebiger Tiefe des Baumes
- count(//adresse/zeile)
  - Wieviele Zeilen hat die Adresse?
- //adresse/zeile[anrede]/position()
  - Die wievielte Zeile der Adresse enthält die Anrede?
- //haupttext[contains(zeile,‘herzlich‘)]
  - Gib mir den Haupttext, der eine Zeile enthält, die ‚herzlich‘ enthält.
  - Rückgabe = Knoten! Vgl. dagegen contains(//haupttextzeile,‘herzlich‘) = Boolean
- string-length(//empfänger/name/nachname)
  - Wie lang ist der Nachname des Empfängers?

  //ort[starts-with(.,‘D‘)]
  - Gib mir alle Orte, die mit D anfangen
- //name[not(@geschlecht)]
  - Bei welchen Namen fehlt das Attribut *geschlecht*?

# XPath: gemeinsame Übungen (Postkarte)

- Gib mir alle Vornamen
  - Lösung: //vorname
- Wieviele Zeilen hat der Haupttext
  - Lösung: count(//haupttext/zeile)
- Was steht in der dritten Adresszeile?
  - Lösung: //adresse/zeile[position()=3]
- Gib mir die Ortsnamen, die mehr als 6 Zeichen lang sind
  - Lösung: //ort[string-length()>6]

# XPath: gemeinsame Übungen (Weber)

- Wieviele Briefe enthält die Datei?
  - count(//TEI)
- Gib mir die Textanfänge, die ‚Du', ‚Ihr' oder ‚Sie' enthalten (Textanfänge stehen im Element incipit)
  - //incipit[contains(.,'Du') or contains(.,'Ihr') or contains(.,'Sie')]
- Wieviele verschiedene Bearbeiter gibt es? (Bearbeiter werden im Element *change* im Attribut *who* genannt )
  - count(distinct-values(//change/@who))
- Welche Änderungen wurden im Jahr 2012 gemacht?
  - //change(starts-with(@when,'2012'))
- Wieviele Änderungen sind für jeden Brief im Durchschnitt festgehalten? (Änderungen stehen in revisionDesc/change)
  - count(//revisionDesc/change) div count(//TEI)

# XPath: der zweite Baukasten, Funktionen

- substring(string,number,number?)
  - Gib mir von *string* ab Zeichen *number* soviele Zeichen, wie im zweiten *number* angegeben (sonst bis zum Ende)
- substring-before(string,string), substring-after(string,string)
  - Gib mir eine Teilzeichenkette vor oder hinter einer bestimmten Zeichenkette
- translate(string,string,string)
- matches(string,pattern)
- replace(string,pattern,pattern)
- distinct-values(*sequenz*)
  - ... Verschiedene Werte, jeweils nur einmal
- name()
- max() – min() – sum() – avg()

# XPath: Einzelübungen (Weber-Briefe)

- In welchen Repositorien liegen die Briefe eigentlich?
  (schauen Sie sich mal msIdentifier an ...)
- Welche verschiedenen Arten von Anmerkungen gibt es?
  (Element *node* mit Attribut *type)*
- Geben Sie in den Brieftexten (und nur da) erwähnte Orte aus, die mit B anfangen
- Extrahieren Sie die Kurztitel der Werke, in denen Briefe bereits gedruckt vorliegen
  (schauen Sie sich additional/listBibl/bibl an; Erstdrucke sind mit dem Wert *firstPrint* im Attribut *n* gekennzeichnet; der Haupttitel ist der Teil vor dem Komma)

- Sie brauchen: substring-before(), starts-with(), distinct-values()

# XPath: Weiterführende Hinweise

- XPath bei WP: http://de.wikipedia.org/wiki/XPath
- Ein Tutorial: http://www.w3schools.com/Xpath/
- Die Heimat von XPath: http://www.w3.org/TR/xpath