

Proseminar: XML- basierte Technologien

XPath

Bearbeiter:

Christian Vogel

03.06.04

TU Chemnitz

Gliederung:

- Allgemeines
- Knoten- Arten
- Ausdrücke
- Pfade
- Ausblick: XPath 2.0

Allgemeines (1)

- Warum XPath?
 - Adressierung einzelner Teile eines XML-Dokumentes
 - Auswahl von Elementen oder Elementgruppen
 - gezielt Werte verfügbar machen

Allgemeines (2)

- Überblick:
 - aktuell: Version 1.0 von Nov. 1999
 - standardisierte Anfragesprache für XML- Dokumente
 - eigenständige Syntax (keine XML- Syntax)
 - nur als Hilfsmittel anwendbar :
 - hauptsächlich bei XSLT („match“-Attribut)
 - aber auch XPointer, XML Schema

Allgemeines (3)

- operiert auf abstrakter, logischer Struktur
- XML- Dokument baumartig aufgebaut
- Verzweigungen sind Knoten (node)
- Gruppen von Knoten sind Knotensätze (node-set)
- 7 verschieden Knoten

Knoten- Arten:

- Wurzelknoten (root)
 - Elementknoten (elem)
 - Kommentarknoten (comm)
 - Textknoten (text)
 - Verarbeitungsanweisungsknoten (proc)
 - Attributknoten (attr)
 - Namensraumknoten (name)
-

Knoten- Arten:

- Knoten hat Format: *index (typ) expanded-name „string-value“*

XML- Dokument	Baum
<pre><?xml version="1.0"?> <?lala bla=blu?> <!DOCTYPE foo SYSTEM "foo.dtd"> <ns:foo xmlns:ns="http://www.TEST.de/"> <ns:bar ns:attr1="0815"> Welch seltsames </ns:bar> <!-- nicht einschlafen! --> XML-Dokument </ns:foo></pre>	<pre>1 (root) "Welch seltsames XML-Dokument" 2 (proc) lala "bla=blu" 3 (elem) http://www.TEST.de:/foo "Welch seltsames XML-Dokument" 4 (name) ns "http://www.TEST.de/" 5 (elem) http://www.TEST.de:/bar "Welch seltsames" 6 (name) ns "http://www.TEST.de/" 7 (attr) http://www.TEST.de:/attr1 "0815" 8 (text) "Welch seltsames" 9 (comm) "nicht einschlafen!" 10 (text) "XML-Dokument"</pre>

Ausdrücke:

- *expressions:*
 - primäre, syntaktische Konstrukt
 - oft in XML- Attributen
 - Sonderzeichen in XML- Syntax (`<` = <)
 - Resultate von folgenden Datentypen:
 - node-set (Menge von Knoten)
 - boolean (boolescher Wert)
 - number (Gleitkommazahlen)
 - string (Zeichenkette)

Ausdrücke:

strings	'lala', 'foobar'
Zahlen	Gleitkommazahlen
Variablen	\$var
logische Operatoren	'<' '>' '<=' '>=' '!=' '=' 'and' 'or'
arithmetische Op.	'+' '-' '*' div mod
Gruppierungen	'(' ')'
Vereinigung	' '
Funktionen	position(), not(),...
Lokalisierungspfade	/child::*

Lokalisierungspfade (1):

- wichtigste Art eines Ausdrucks
- Auswahl bestimmter Knoten aus Baum
- schrittweises Vorgehen
- ist Pfaden im Dateisystem ähnlich
- Arten:
 - relative LP

schritt1/schritt2

 - absolute LP

/schritt1/schritt2

Lokalisierungspfade (2):

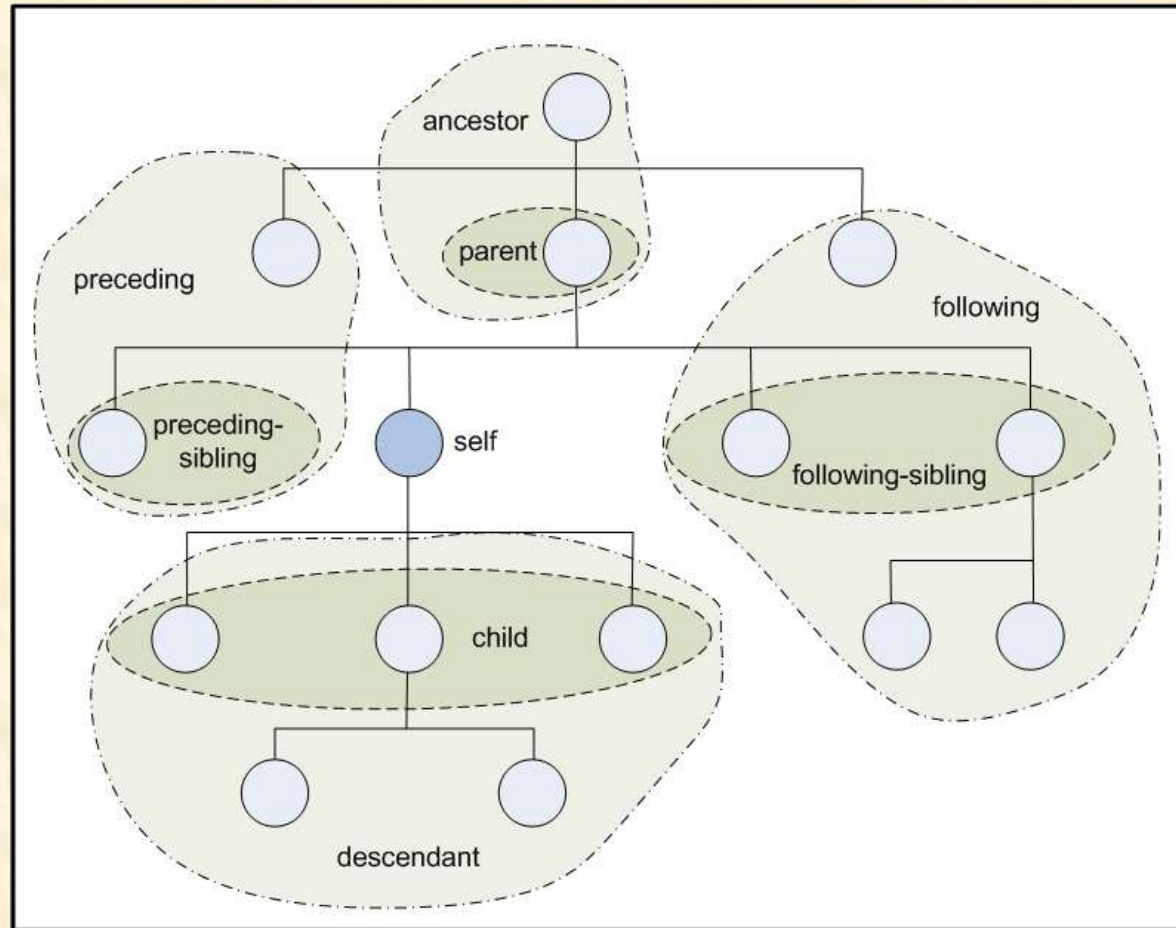
- Aufbau eines *Schrittes*:

achse::knotentest[prädikat1][prädikat2]...

achse::

- gibt Verhältnis der gewünschten Knoten zum Kontextknoten an

Lokalisierungspfade (3):



Lokalisierungspfade (4):

- zusätzlich noch:
 - descendant-or-self
 - ancestor-or-self
 - namespace
 - attribute

Lokalisierungspfade (5):

- Aufbau eines *Schrittes*:

achse::knotentest[prädikat1][prädikat2]...

achse::knotentest

- Prüfung der durch achse ausgewählten Knoten, ob vom Typ *knotentest*

Lokalisierungspfade (6):

<u><i>knotentest</i></u>	<u><i>erfüllt für...</i></u>
*	...jeden Knoten des Hauptknotentyps
text()	...jeden Textknoten
comment()	...jeden Kommentarknoten
processing-instruction()	...jeden Verarbeitungsanweisungsknoten
node()	...jeden Knoten
ns:name	...jeden Knoten aus <i>ns</i> mit <i>name</i>

Lokalisierungspfade (7):

– Beispiele:

*child::**

/descendant-or-self::node()

attribute::ns:href

Lokalisierungspfade (8):

- Aufbau eines *Schrittes*:

achse::knotentest[prädikat1][prädikat2]...

achse::knotentest[prädikat]

- zusätzliche Filterung der Knotenmenge

Lokalisierungspfade (9):

– Beispiele:

```
child::para[position()=last()]
```

```
/child::doc/child::chapter[position()=5]/child::section[position()=2]
```

```
child::para[attribute::type=' warning'][position()=5]
```

```
child::*[self::chapter or self::appendix]
```

Lokalisierungspfade (10):

<u>Schritt</u>	<u>Abkürzung</u>
child::	
attribute::	@
self::node()	.
parent::node()	..
/descendant-or-self::node()/	//

Lokalisierungspfade (11):

– Beispiele:

./para

`self::node()/descendant-or-self::node()/child::para`

Ausblick: XPath 2.0

- neuer Typ: *Sequenztyp*
 - jede Operation liefert Sequenz
 - enthält Werte einfacher Typen oder Knoten
 - Aufzählung durch Komma
 - keine Schachtelung
 - Operatoren:
 - *for , to*

for \$x in (1 to 10) return \$x*\$x

Ausblick: XPath 2.0

- bedingte Ausdrücke:
 - if (@level) then @level else 5
- Einführung starker Typisierung:
 - XML Schema basierend
- erweiterte Pfadausdrücke
- Operatoren für Differenz / Durchschnitt
- viele neue Funktionen
 - (Zeichenketten-, Knoten-, Datumfunktionen)

Quellen:

- deutsche Übersetzung der W3C Empfehlung „XML Path Language (XPath)“
- „XML Lösungen“ von Jake Sturm
- Artikel von Oliver Becker im JAVASPEKTRUM 6/2003