UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft

Edirom Summer School 2010

Workshop: XML und XML-Technologien
21.-24. November 2010

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Begriff XML

XML ist eine Auszeichnungssprache für Textdokumente

- 'X' für „extensible“, für beliebige Aufgaben erweiterbar
- 'M' für „markup“, Auszeichnung: Beschreibung von Semantik eines Dokuments
- 'L' für „language“, Dialekt einer Sprachenfamilie (SGML)
- Anwendung ausschliesslich auf menschlich lesbare Textdokumente, also ohne Binärdaten

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

XML kann ...

- Für unterschiedliche Anwendungsbereiche konfiguriert werden: Chemiker, Musiker, Philologen … können ein für ihre Anforderungen optimales Schema konzipieren
- Von unterschiedlichen Systemen gelesen werden: Menschen, Linux-Servern, Windows-PCs, Smartphones …
- Für lange Zeit aufbewahrt werden: Informationen zur Dekodierung sind im Schema enthalten, ermöglichen das Auslesen noch in langer Zukunft (Gegenbeispiel: Binärdaten)

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

XML kann nicht ...

- „Ausgeführt“ werden: Steuerungslogik ist nicht vorgesehen (Ausnahme: XSLT)
- Daten senden: Netzwerkprotokolle können zur Kodierung XML verwenden, doch XML bleibt in passiver Rolle
- Als vollständige Datenbank dienen: Es gibt Datenbanken mit Optimierung auf XML (eXist...) und Anfragesprachen auf XML-Dokumente (XPath...), doch zum Betrieb einer Datenbank ist eine Programmiersprache mit Ausführungslogik nötig

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

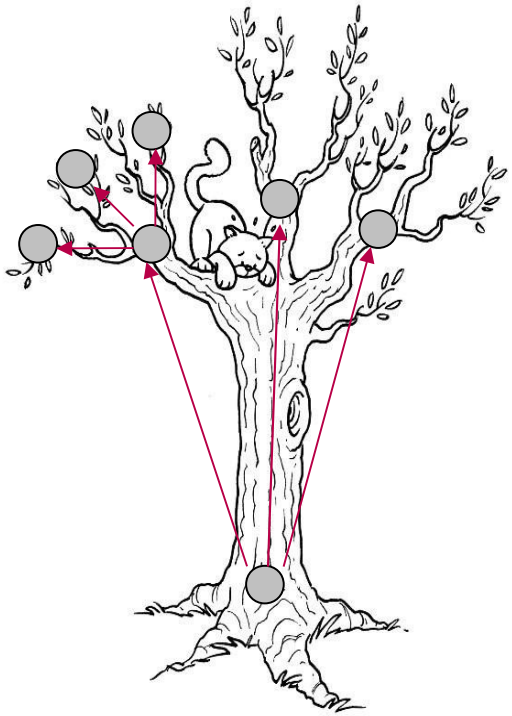
Verwandtschaft von XML

- Vorfahre ist Sprache SGML mit Wurzeln in den 1970ern, konnte sich jedoch aufgrund der Komplexität nicht bei Anwendern durchsetzen
- Webseiten sind in HTML beschrieben: Gleicher Vorfahre, gleiche Struktur, anderer Anwendungszweck (Präsentation)
- Wichtiger Dialekt von HTML ist XHTML: Präsentationsdaten von HTML in der Form von XML (nur kleinere Änderungen)

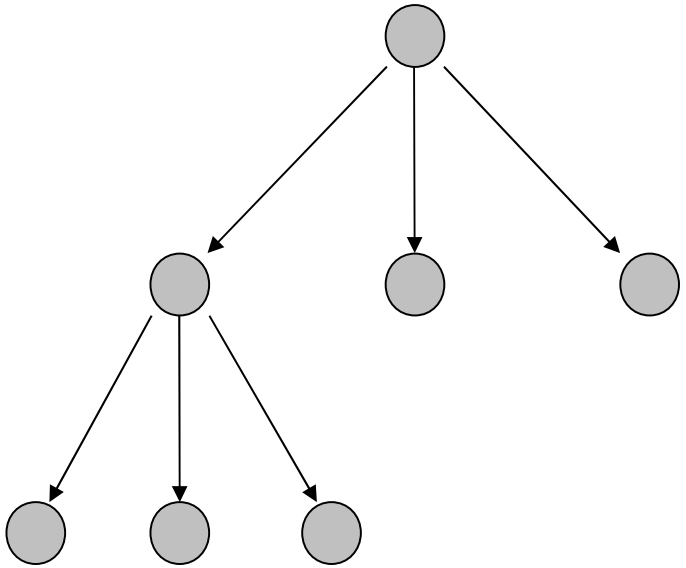
M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Baumprinzip Natur - Mathematik



Abstraktion



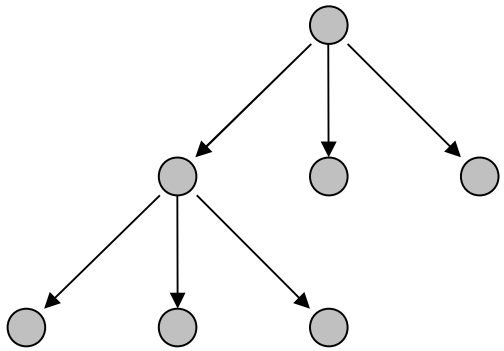
M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Baumkonzept von XML

Jedes XML Dokument ist logischer n-ärer Baum:

- Baum hat einen eindeutigen Wurzelknoten
- Ein Knoten hat beliebig viele („n“) Kinder,
  jeder Knoten genau einen Elternknoten (Ausnahme: Wurzel)
- Knoten sind entweder der Wurzelknoten, ein innerer Knoten (mit Kindern) oder Blätter (ohne Kindknoten)



M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Rekursivität

Bäume sind eine rekursive Struktur

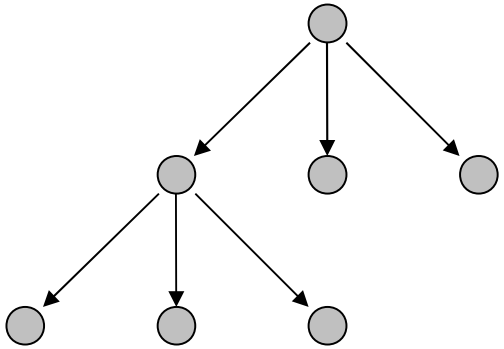
- Informelle Erklärung für Rekursion:
  „Ketchup besteht aus Zucker, Tomaten und Ketchup“
- Ein Baum besteht selbst aus Bäumen (Rekursion)
  oder einzelnem Wurzelknoten (Rekursionsabbruch)

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Baumkonzept von XML

Übung: Finden Sie 3 Beispiele für Baumstrukturen in der Natur (Bäume verstehen)



M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Klammerung

XML Dokumente sind Strukturen aus Klammerpaaren

- Jede öffnende Klammer braucht eine schliessende Klammer
- Keine Kreuzverschachtelung zulässig
- Jede Klammer trägt semantische Information
- Korrekte Klammerung notwendig für „*Wohlgeformtheit*“

Beispiel:
```
<klammer>
    Inhalt
</klammer>
```

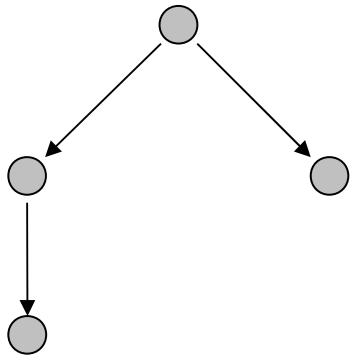
M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Klammerung - Baum

Klammerung reflektiert die Baumstruktur

```
<baum>
   <wurzel>
      <knoten>
         <blatt>
         </blatt>
      </knoten>
      <blatt>
      </blatt>
   </wurzel>
</baum>
```



M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

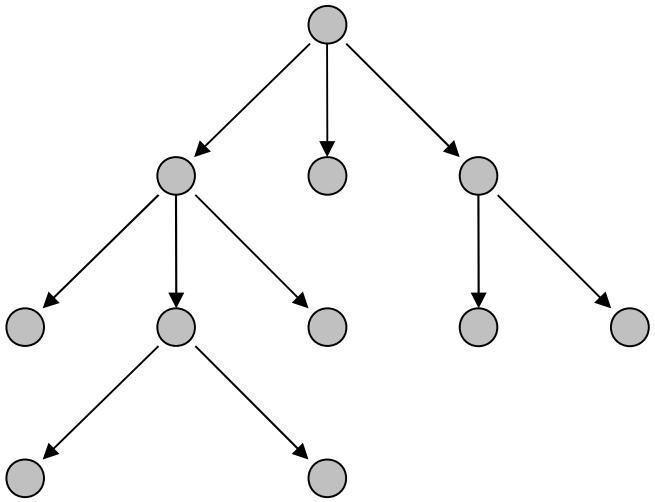
Block 2: XML Grundlagen

Übung: Klammerung - Baum

Baumstruktur auf Klammerung übertragen

Elemente: baum, wurzel, knoten, blatt

```
<baum>
        [...]
</baum>
```

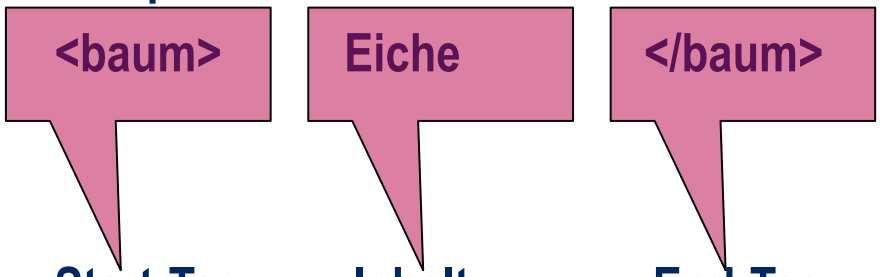


M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Tags und Elemente

- Beispiel-XML:



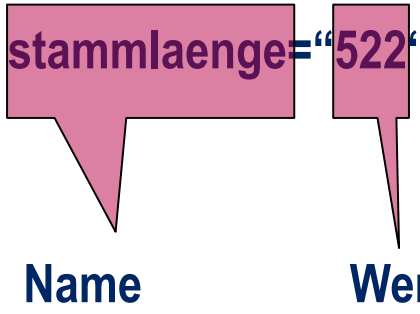
- Ein Tag-Paar klammert den Inhalt (content) eines Elements ein
- Ein Tag beginnt ('<') und endet ('>') immer mit einer spitzen Klammer. Ein Slash ('/') deutet auf das Ende des Elements hin.
- Tags und ihr Inhalt bilden ein Element.

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Attribute

- Ein Attribut ist ein Name-Wert Paar im Start-Tag eines Elements (<baum stammlaenge="522" alter = "alt">)



- Ein Element kann beliebig viele Attribute tragen
- Leerstellen (whitespace) zwischen Name, Gleichheitszeichen und Wert werden nicht beachtet → persönliche Ästhetik
- Ein Attribut kann genutzt werden, um die Baumtiefe zu verringern (Blätter in Elternknoten hochziehen)

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Übung: Elemente und Attribute

Stellen Sie den folgenden Baum einmal maximal und einmal minimal dar, indem Sie Attribute in Elemente bzw. Elemente in Attribute wandeln!

```
<person>
        <name first="Alan" last="Turing"> </name>
        <profession value="mathematician/>
        <id-number>
            12345
        </id-number>
        <build_machines/>
    </person>
```

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Begriffe für die Struktur

- „Tag“: Klammerphrase (<baum>)
- „Start-Tag“: Öffnendes Tag (<baum>)
- „End-Tag“: Schliessendes Tag (</baum>
- „Element“: Alles zwischen und inklusive zweier Tags (<baum> Eiche </baum>)
- „Leer-Element“: Element ohne Inhalt (<baum> </baum>; Kurzschreibweise: <baum/>)
- „Attribut“: Name-Wert Paar im Start-Tag (<baum stammlaenge=“522“>)

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Rechtschreibregeln

- XML beachtet Gross- und Kleinschreibung:
  <baum> ist etwas anderes als <Baum> oder <BAUM>!
- Tags dürfen nicht mit „xml“ beginnen (Schlüsselwort)
- Tags dürfen nicht mit Ziffern oder Punktsetzungszeichen beginnen:
  <1mal> <einmal>; <!ZuTeuer!> <ZuTeuer!>
- Tags dürfen keine Leerzeichen enthalten:
  <rote eiche> <rote_eiche>, <roteEiche>
- Alles andere ist erlaubt!

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Übung: Begriffe für die Struktur

XML Struktur eines Beispiels benennen

```
<person>
    <name first="Alan" last="Turing"> </name>
    <profession value="mathematician/>
    <id-number>
        12345
    </id-number>
    <build_machines/>
</person>
```

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

XML Parser

XML ist für Menschen und für Computer lesbar

- Menschen können XML wie ASCII Text lesen
- Computer lesen XML-Dateien über Parser (“Zerteiler“):
  Aus der Klammerstruktur wird der logische Baum aufgebaut
- Fürs Einlesen muss das Dokument wohlgeformt sein, sonst bricht der Parser ab
- Um den Sinn von Dokumenten zu „verstehen“, muss der Parser die Bildungsvorschriften der Struktur kennen (Schema)

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Bildungsvorschriften für XML

XML Dokumente sind nach einem Schema aufgebaut

- Welche / wieviele Elemente und Attribute kommen vor?
- Welcher Datentyp ist im Inhalt zu erwarten?
  (Ziffern / Zeichenkette / Wahrheitswert)
- Welche Kinderelemente sind unter einem Element erlaubt?
- Entspricht ein XML Dokument dem von ihm verwiesenen Schema, so ist es „*valide*“
- Verweist es auf kein Schema, so steht es für sich: „standalone“
- Mehr dazu in Block 3: Schemata

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Weitere XML Strukturen

Kommentare

- (<!-- Dies ist ein Kommentar -->)
- Eingeschlossen in <!-- und -->
- Dienen dem Verständnis menschlicher Leser
- Werden vom Parser ignoriert

XML-Deklaration

- (<? xml version="1.0" encoding="UTF-8" ?>)
- Steht am Beginn eines XML Dokuments (und ist optional)
- Dient dem Verständnis durch Parser (Kodierung)
- Wird vom menschlichen Leser ignoriert

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Wohlgeformtheit, Validität

Bedingungen für Wohlgeformtheit

- Jedes Start-Tag hat ein entsprechendes End-Tag
- Elemente dürfen verschachtelt sein, aber nicht kreuzweise
- Es muss exakt ein Wurzelelement geben
- Attributwerte müssen in Anführungszeichen stehen
- Ein Element darf nicht zwei Attributwerte mit demselben Namen haben
- Kommentare und Verarbeitungsanweisungen dürfen nicht in Tags vorkommen
- Im Inhalt eines Elements dürfen keine '<' oder '&' Zeichen sein

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Textencoding: Unicode u.a.

Zeichen in XML werden als Symbol-Zeiger gespeichert

- Computer versteht Buchstaben 'Z' als solchen nicht.
- Aber Computer verstehen Zahlen: „90“ ist als Binärzahl aus 0 und 1 darstellbar (Exkurs: wie sieht die aus?)
- Weiss der Computer, wie lang die Zahl ist (Pointer: 4*Hex), kann er auf einer Tabelle nachsehen, welches Zeichen gemeint ist und dieses darstellen
- Verschiedene Tabellen: UTF-8, ASCII, ISO-8859-1 usw.
- Unicode ist der Standard, wenn in der XML-Deklaration kein anderes Encoding angegeben wird

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Übung: Struktur eines Beispieldokument benennen (Tonleiter.mei)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<mei xmlns:ns2="http://www.w3.org/1999/xlink" version="1.9b">
    <meihead>
        <filedesc>
            <titlestmt>
                <title>Tonleiter</title>
            </titlestmt>
            <pubstmt/>
            <sourcedesc>
                <source xml:id="Tonleiter.mei_src.2">
                    <titlestmt>
                        <title>etude de julian</title>
                        <respstmt/>
                    </titlestmt>
                    <pubstmt/>
                </source>
            </sourcedesc>
        </filedesc>
        <encodingdesc/>
    </meihead>
    <music>
        <body>
            <mdiv>
                <score>
                    <scoredef meter.unit="4" meter.count="4" key.sig="2s"
key.mode="major">
                        <staffgrp>
                            <staffdef n="1" midi.div="1" label.full="Sopran"
key.sig="2s" key.mode="major" clef.shape="G" clef.line="2"/>
                            <staffdef n="2" midi.div="1" label.full="Alt"
key.sig="2s" key.mode="major" clef.shape="F" clef.line="2"/>
                        </staffgrp>
```

M.Sc. Julian Dabbert
Fakultät für Kulturwissenschaften der Universität Paderborn

Block 2: XML Grundlagen

Übung: XML Dokument schreiben (Personenbeschreibung)

Verfassen Sie eine Personenbeschreibung von sich in XML

- Benutzen Sie die Daten ihres Personalausweises
- Das Schema ist beliebig, denken Sie sich etwas Sinnvolles aus

Ende des Blocks 2