

HTML & XML

Andreas Mertens

Kursinhalte I

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- XML Einführung
- Werkzeuge und Quellen
- Überprüfung von XML-Dokumenten
 - (XML DTDs und XML-Schemata)
- XSL-Stylesheets und Transformationen
- Xlinks, Xpath
- SAX (Simple API for XML)
- Literatur

Kursinhalte II

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- Übungen, Übungen, Übungen
- Im Rahmen des 3tägigen Seminars bauen wir eine XML-Anwendung

XML-Einführung I

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions
(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur

- XML ist eine Auszeichnungssprache
- XML steht für eXtensible Markup Language
- XML ist eine Untermenge von SGML (Standard Generalized Markup Language)
- SGML (Standard Generaliyed Markup Language(Mitte 80er als ISO-Std. verabschiedet) erlebte nie einen „Hype“ wie XML

XML-Einführung II

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- Es gibt viele Auszeichnungssprachen
 - SGML
 - PDF
 - RTF, ...
- Herstellerabhängige Formate wie RTF (Microsoft) oder PDF (Adobe) konnten sich nicht durchsetzen
- SGML galt als zu Umfangreich und zu kompliziert

Warum XML I ?

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- Dokument besteht aus
 - Inhalt
 - Struktur
 - Repräsentation (kann verschiedene haben)
 - HTML
 - PDF
 - DOC
- Inhalt, Struktur und Repräsentation soll trennbar sein
 - Auszeichnungssprache => XML

Warum XML II ?

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions
(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

1. Eine Dot.com-Company betreibt einen Online-Shop mit Produkten (Bsp. Bücher wie amazon.com)
2. Die Dot.com-Company möchte aber ihren Produktkatalog nicht nur online offerieren, sondern auch als Print-Prospekt
3. Außerdem möchte Dot.com Daten mit einem anderen Partner austauschen können

Lösung

Einführung

Werkzeuge

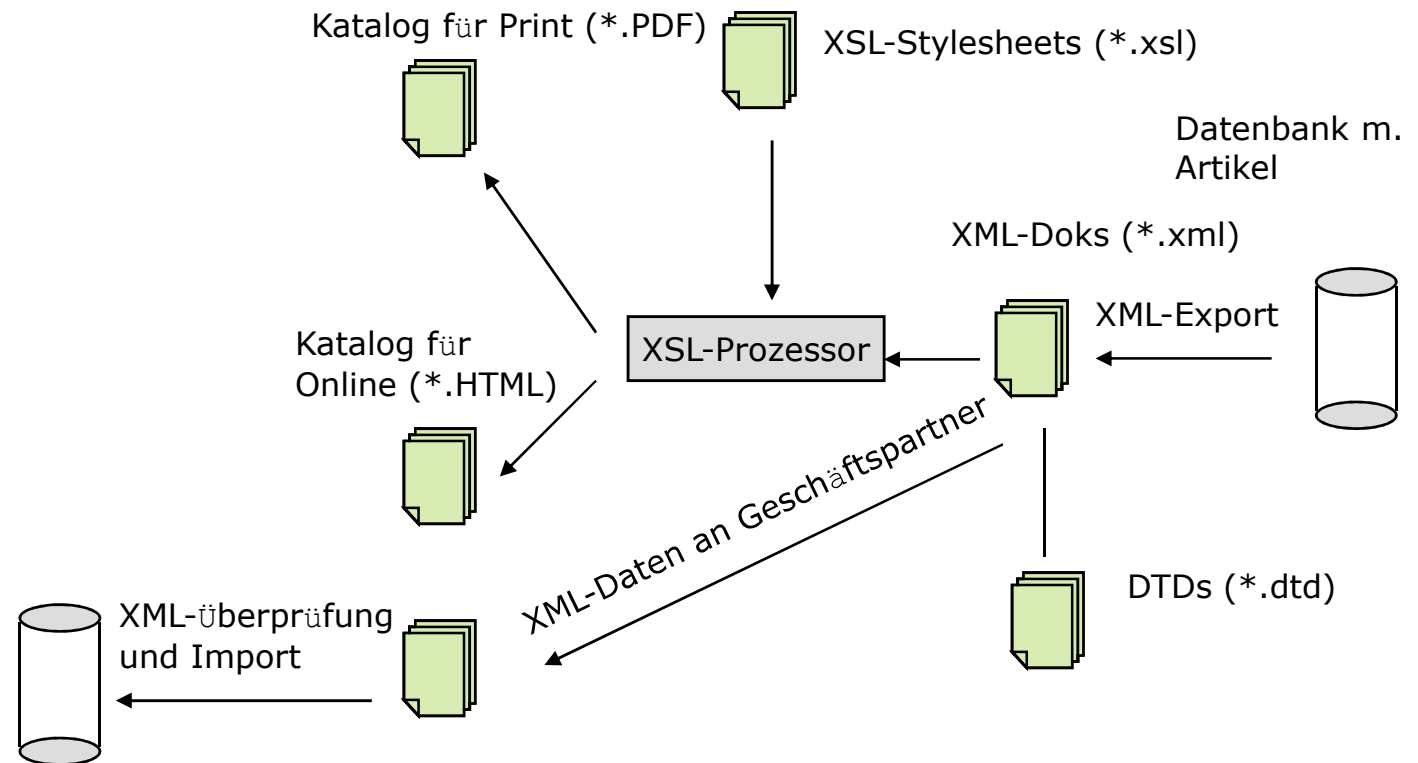
Überprüfung

Document Type
Defintions
(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur



XML-Entwurfsziele I

Einführung

Werkzeuge

Überprüfung

Document Type
Definitions
(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur

- Die Markup-Language soll erweiterbar sein (X in XML für eXtensible)
- Es soll einfach sein, Anwendungen zu schreiben die XML-Dokumente verwenden
- XML-Dokumente sollen leicht zu erstellen sein

XML-Entwurfsziele II

Einführung

Werkzeuge

Überprüfung

Document Type
Definitions
(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur

- XML-Dokumente sollen formal überprüfbar sein (PARSER)
- Der XML-Entwurf soll schnell gehen und einfach

XML-Werkzeuge I

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions
(DTDs)

XML-Schemata

eXtensible
Style Sheet

Language

Literatur

- Microsoft MSXML (Parser, XSL-Prozessor, frei)
 - www.microsoft.com
- Apache-Projekt (Parser, XSL-Prozessor, Xerces und Xalan, frei)
 - www.apache.org
- XMLSpy (XML-Editor, nicht frei)
 - www.xmlspy.com
 - Benutzt MSXML

XML-Werkzeuge II

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions
(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- James Clark XML-Ressources
 - <http://www.jclark.com>
- IBM Alphaworks Parser, XSL-Prozessor
 - <http://www.alphaworks.ibm.com/xml>
- Jedit unter <http://www.jedit.org> mit PlugIns
- Stylus Studio von Excelon unter <http://www.exln.com>

XML (MSXML)

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions
(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- Installation
 - Kopieren der Dateien
 - Msxml4.dll
 - Msxml4a.dll
 - Msxml4r.dll
 - nach c:\winnt\system32
 - Registrieren der „Haupt-DLL“:
 - Regsvr32 msxml4.dll

XML-Überprüfung

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions
(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- Well-Formed (wohlgeformt), ist ein XML-Dokument, wenn es
 - zu Beginn des Dokumentes eine XML-Deklaration vorliegt
 - es mind. 1 Datenlement gibt
 - container (Tags) korrekt angeordnet sind
- Valid (gültig) ist ein XML-Dokument, wenn
 - es mittels einer DTD oder eines XML-Shemas und einem XML-Parser überprüft werden kann (Grammatik, formal korrekt).

XML-Überprüfung

Einführung

Werkzeuge

Überprüfung

Document Type
Definitions
(DTDs)

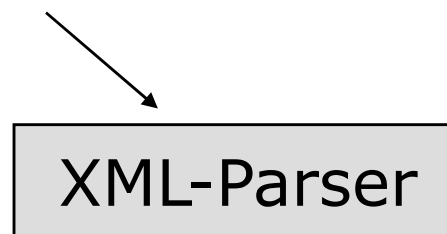
XML-Schemata

eXtensible
Style Sheet
Language

Literatur

book.xml

```
<?xml version="1.0"?>  
<!DOCTYPE book SYSTEM „book.dtd“>  
<book>  
.  
.  
</book>
```



XML-Überprüfung

Einführung

Werkzeuge

Überprüfung

Document Type
Defintions
(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur

book.xml

```
<?xml version="1.0"?>  
<!DOCTYPE book SYSTEM „book.dtd“>  
<book>  
.  
.  
</book>
```

XML-Parser

Im XML-File
wird optional
der Bezug zu
einer DTD
hergestellt

XML-Überprüfung

Einführung

Werkzeuge

Überprüfung

Document Type
Defintions
(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur

book.xml

```
<?xml version="1.0"?>
<!DOCTYPE book SYSTEM „book.dtd“>
<book>
.
.
</book>
```

book.dtd

```
<!ELEMENT book (#PCDATA)>
<!ELEMENT . >
<!ELEMENT . >
```

XML-Parser

wellformed?
Valid?

XML-DTD I

Einführung

Werkzeuge

Überprüfung

Document
Type
Defintions
(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- DTD = **D**ocument **T**ype **D**efintion
- „Eine DTD beschreibt die Art und Struktur der Auszeichnungselemente für einen Typ von Dokumenten“

XML-DTD II

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- Eine DTD kann
 - Intern eingebunden sein
 - DTD ist Bestandteil des XML-Dokumentes
 - Extern eingebunden werden
 - Durch den Verweis auf eine externe .dtd-Datei
 - muß gar nicht angegeben sein (Siehe html)

DTD-Beispiel (intern)

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- XML-File:

```
<?xml version="1.0" standalone="yes" ?>
<!DOCTYPE book [
<!ELEMENT book (chapter+)
<!ELEMENT chapter (#PCDATA) ]>
<book>
    <chapter>Dies ist Kapitel 1 ....</chapter>
    <chapter>Dies ist Kapitel 2 ....</chapter>
</book>
```

DTD-Beispiel (extern)

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- XML-File:

```
<?xml version="1.0" standalone="no" ?>
<!DOCTYPE book SYSTEM „book.dtd“>
<book>
    <chapter>Dies ist Kapitel 1 ....</chapter>
    <chapter>Dies ist Kapitel 2 ....</chapter>
</book>
```

XML-DTD Beispiel

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

Beispiel .dtd:

```
<!ELEMENT book (chapter+)>  
<!ELEMENT chapter (#PCDATA)>
```

Ein Buch besteht aus
ein oder mehreren
Kapiteln (+)

Ein Kapitel besteht
aus „Parsed
Charactern Data“.

CDATA und PCDATA

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- Parsed Character Data (PCDATA)
 - Wird „geparsed“. Darf somit keine Tags enthalten, wenn sie nicht in der DTD definiert sind.
- Character Data (CDATA) im Inhalt
 - Wird nicht mehr „geparsed“:

```
<![CDATA[<TITLE>MeinTitel</TITLE>]]>
```

EBNF

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

Symbol	Bedeutung	Beispiel
,	Sequenzen	Vorname, Nachname
	Oder	Vorname Nachname
*	Null oder mehrmals	Produkt*
+	Ein oder mehrmals	Produkt+
?	Optional Null oder Einmal	Produkt?
	Muß Einmal erscheinen	Produkt

Beispiel 1: Werkzeuge

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- Download Microsoft
www.microsoft.com/xml
 - XML Validation-Tool/Parser (xmlint.zip)
 - MSXML 4.0 (Microsoft XML Core Services 4.0 SP1)
- Command-Line-Parser:
 - Xmlint.exe

Beispiel 1: book.xml

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- XML-File:

```
<?xml version="1.0"?>
<book>
    <chapter>
        Dies ist Kapitel 1 ....
    </chapter>
    <chapter>
        Dies ist Kapitel 2 ....
    </chapter>
</book>
```

Beispiel 1: book.xml

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

XML-File überprüfen mit:

```
xmlint book.xml
```

Wenn das XML-File wohlgeformt ist
(wellformed), gibt es keine Meldung.

Aufgabe 1

Einführung	Entwickeln Sie eine XML-Struktur die folgende Daten strukturiert:
Werkzeuge	
Überprüfung	
<u>Document</u>	Alben für CDs, MCs, Vinyl
<u>Type</u>	
<u>Defintions</u>	
<u>(DTDs)</u>	Das Format soll den Typ (CD MC Vinyl),
XML-Schemata	die Interpreten (Vorname, Nachname),
eXtensible	Titel mit Gesamtdauer des Albums, sowie die
Style Sheet	Lieder mit Spieldauer beinhalten.
Language	
Literatur	

Beispiel 2: book2.xml

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- XML-File (malformed):

```
<?xml version="1.0"?>
<book>
    <chapter>
        Dies ist Kapitel 1 ....
    </chapter>
    <chapter>
        Dies ist Kapitel 2 ....
</book>
</chapter>
```

**Container
Chapter außerhalb
von book ge-
schlossen**



Beispiel 2: book2.xml

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

XML-File überprüfen mit:

```
xmlint book2.xml
```

```
Das Endtag 'book' stimmt nicht mit dem Starttag  
'chapter' überein.
```

```
URL:file:///D:/book2.xml
```

```
Line 00009: </book>
```

```
Pos 00003: --^
```

Beispiel 3: book3.xml

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

DTD book3.dtd:

```
<!ELEMENT book (chapter+)>  
<!ELEMENT chapter (#PCDATA)>
```

Beispiel 3: book3.xml

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

XML-File book3.xml:

```
<?xml version="1.0"?>
<!DOCTYPE book SYSTEM "book3.dtd">
<book>
    <chapter>
        Dies ist Kapitel 1 ....
    </chapter>
    <chapter>
        Dies ist Kapitel 2 ....
    </chapter>
</book>
```

Beispiel 3: book3.xml

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

XML-File überprüfen mit:

```
xmlint book3.xml
```

Keine Meldung, falls das XML-File
wohlgeformt (wellformed) und gültig (valid)
gem. book3.dtd ist.

Beispiel 4: book4.xml

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible


Style Sheet

Language

Literatur

XML-File book4.xml:

```
<?xml version="1.0"?>
<!DOCTYPE book SYSTEM "book3.dtd">
<book>
  <chapter>
    Dies ist Kapitel 1 ....
  </chapter>
  Text zwischen den Kapiteln
  <chapter>
    Dies ist Kapitel 2 ....
  </chapter>
</book>
```



Beispiel 4: book4.xml

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

XML-File überprüfen mit:

```
xmlint book4.xml
```

```
Gemäß dem DTD/Schema ist in diesem Zusammenhang  
kein Text zugelassen.
```

```
Erwartet: chapter.
```

```
URL:file:///D:/book4.xml
```

```
Line 00006: </chapter>
```

```
Pos 00012: -----^
```

Übung 1: book4.xml

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

Passen Sie book3.dtd so an, daß zwischen den chapter-tags normaler Text stehen kann.

```
<?xml version="1.0"?>
<!DOCTYPE book SYSTEM "book3.dtd">
<book>
    <chapter>
        Dies ist Kapitel 1 ....
    </chapter>
    Text zwischen den Kapiteln
    <chapter>
        Dies ist Kapitel 2 ....
    </chapter>
</book>
```

XML-DTD Attribute I

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

Beispiel .xml-File:

```
<chapter Author="sedat duman">  
    Im ersten Kapitel behandeln wir....  
</chapter>  
<chapter Author="eva schick" sex= "female">  
    Im ersten Kapitel behandeln wir....  
</chapter>
```

XML-DTD Attribute II

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- **#REQUIRED**
 - Das Attribut muß angegeben werden
- **#IMPLIED**
 - Das Attribut kann angegeben werden
- **#FIXED**
 - Fest, fix

XML-DTD Attribute III

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

.dtd-File:

```
<!ELEMENT book (chapter)* >  
<!ELEMENT chapter (#PCDATA)*>  
<!ATTLIST chapter Author CDATA #REQUIRED  
sex (male|female) #IMPLIED>
```

XML-DTD im XML-File

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

Interne DTD:

```
<?xml version="1.0"?>
<!DOCTYPE book [
  <!ELEMENT book (#PCDATA|chapter)* >
  <!ELEMENT chapter (#PCDATA)>
]>
<book>
  <chapter>
    Dies ist Kapitel 1 ....
  </chapter>
  Text zwischen den Kapiteln
  <chapter>
    Dies ist Kapitel 2 ....
  </chapter>
</book>
```

} DTD

Empty Elements

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

Bekanntes Empty-Element (auch Standalone-Element genannt) in HTML:

```
<IMG SRC="image.jpg" />
```

Einfaches Beispiel:

```
<MeinTag farbe="blau" />
```

Dazu die DTD:

```
<!ELEMENT MeinTag EMPTY >  
<!ATTLIST MeinTag farbe CDATA #REQUIRED>
```

Empty Elements Beispiel

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

EmptyElements.xml:

```
<?xml version="1.0"?>
<!DOCTYPE MyDemo [
<!ELEMENT MyDemo (MyTag)* >
<!ELEMENT MyTag (#PCDATA)>
<!ATTLIST MyTag farbe CDATA #REQUIRED>
]>
<MyDemo>
    <MyTag farbe="blau" />
    <MyTag farbe="geld" />
    <MyTag farbe="gruen" />
</MyDemo>
```

DTDs Zusammenfassung

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- DTDs
 - Verwenden selbst keine XML-Syntax (EBNF)
 - Erlauben keine ausreichende Spezifizierung von Datentypen und Wertebereichen
 - Alternative: XML-Schemata

Aufgabe 2

Einführung

Werkzeuge

Überprüfung

**Document
Type
Definitions
(DTDs)**

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

Bauen Sie für Ihre XML-Struktur aus Aufgabe 1 eine Document Type Definition (DTD) und testen Sie diese.

Entities I

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions
(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- Ähnlich wie #define in C/C++
- Wird vom Parser bearbeitet und ersetzt
- >, < und Umlaute wie ä aus HTML sind Entities

Entities II

Einführung

Werkzeuge

Überprüfung

Document

Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- Dokument mit Entities -->
<!DOCTYPE MyDemo [
<!ELEMENT MyDemo (autor)>
<!ELEMENT autor (#PCDATA)>
<!ENTITY trainer "Andreas Mertens">
<!ENTITY email „andreas.mertens@4th-arcanum.de"
]>
<MyDemo>
    <autor>Sie erreichen mich unter der E-
    Mail-Adresse &email;</autor>
</MyDemo>
```

XML Schema

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions
(DTDs)

XML-
Schemata

eXtensible
Style Sheet

Language

Literatur

- XML-Schemas
 - Benutzen eine XML-basierte Syntax um die Struktur von XML-Dokumenten zu beschreiben
 - Unterstützt die Validierung verschiedener Datentypen, einschließlich der Überprüfung von Wertebereichen
 - Ist erweiterbar
 - Unterstützt komplexe Elemente

Schema Alternativen

Einführung

- RELAX

Werkzeuge

- <http://www.xml.gr.jp/relax/>

Überprüfung

Document Type
Defintions
(DTDs)

- TREX

- <http://www.thaiopensource.com/trex>

XML-
Schemata

eXtensible
Style Sheet
Language

- SCHEMATRON

- <http://www.ascc.net/xml/resource/schematron/schematron.html>

Literatur

XML Schema Beispiel 1

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions
(DTDs)

XML-
Schemata

eXtensible
Style Sheet
Language

Literatur

- Einfaches XML-File mit XML-Schema (xmlschema.xml)

```
<?xml version="1.0"?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/
  XMLSchema">
  <xsd:element name="Text"
    type="xsd:string"/>
</xsd:schema>
<Text>
  Dies ist ein Text
</Text>
```

xsd=XML Schema Definition

XML Schema Datentypen

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions
(DTDs)

XML-
Schemata

eXtensible
Style Sheet
Language

Literatur

- 44 Build-In Datentypen

```
xsd:float  
xsd:integer  
xsd:string  
xsd:date  
xsd:time  
xsd:hexBinary  
.  
.  
.  
.
```

XML Schema Beispiel 2

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions

(DTDs)

XML-
Schemata

eXtensible

Style Sheet

Language

Literatur

- ComplexType, Sequence

```
<?xml version="1.0" ?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="Text">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="Abschnitt" type="xsd:string" />
        <xsd:element name="UAbschnitt" type="xsd:string" />
        <xsd:element name="Stand" type="xsd:date" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

XML Schema Beispiel 2

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions

(DTDs)

XML-
Schemata

eXtensible

Style Sheet

Language

Literatur

- Gültiges XML-File zum Schema

```
<?xml version="1.0"?>
<Text xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="d:\...\u3\xmlschema2.xsd">

  <Abschnitt>
    Dies ist der 1. Abschnitt dieses Textes
  </Abschnitt>
  <UAbschnitt>
    Dies ist einer Unterabschnitt
  </UAbschnitt>
  <Stand>2002-05-08</Stand>

</Text>
```

**Typenüberprüfung
YYYY-MM-DD mit DTD
Nicht möglich !!!!**

XML Schema Beispiel 3

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions

(DTDs)

XML-
Schemata

eXtensible

Style Sheet

Language

Literatur

- Eigene Datentypen ableiten/erweitern:

```
<?xml version="1.0" ?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="MiniText">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:minLength value="3" />
        <xsd:maxLength value="10" />
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>
</xsd:schema>
```

**Eigenen Typ MiniText mittels
Restriction „ableiten“ und
einschränken mit minLength und
maxLength**

XML Xlinks I

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions

(DTDs)

XML-
Schemata

eXtensible

Style Sheet

Language

Literatur

- XML Xlinks sind wesentlich mächtiger als HTML-Links:
 - Ein XLinks können auf mehrere „Targets“ linken
 - XLinks können jedes XML-Element als Verweis definieren
 - Zwei Typen: extended (Mehrfachverlinkung) und simple (Einfachverlinkung)
- Leider bisher wenige Anwendungen die Xlinks unterstützen

XML Xlinks II

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions
(DTDs)

XML-
Schemata

eXtensible
Style Sheet
Language

Literatur

- Attribut show (new|replace|embed|none|other)

```
<a xlink:type="simple"
  xlink:actuate="onRequest"
  xlink:show="replace"
  xlink:href="http://www.heise.de">
  Zu www.heise.de mit replace.
</a><br/>
```

```
<a xlink:type="simple"
  xlink:actuate="onRequest"
  xlink:show="new"
  xlink:href="http://www.heise.de">
  Zu www.heise.de mit replace
</a>
```

XSL

Einführung

Werkzeuge

Überprüfung

Document Type
Definitions
(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur

- XSL = eXtensible Stylesheet Language
- XSL besteht aus zwei Teilen
 - XSLT (Transformation Language) – Definiert Regeln für die Transformation von XML-Dokumenten
 - XSL-FO (Formatting Language) – Bietet Unterstützung zur Formatierung (PDF, DOC, ...)

XSL-Prozessor

Einführung

Werkzeuge

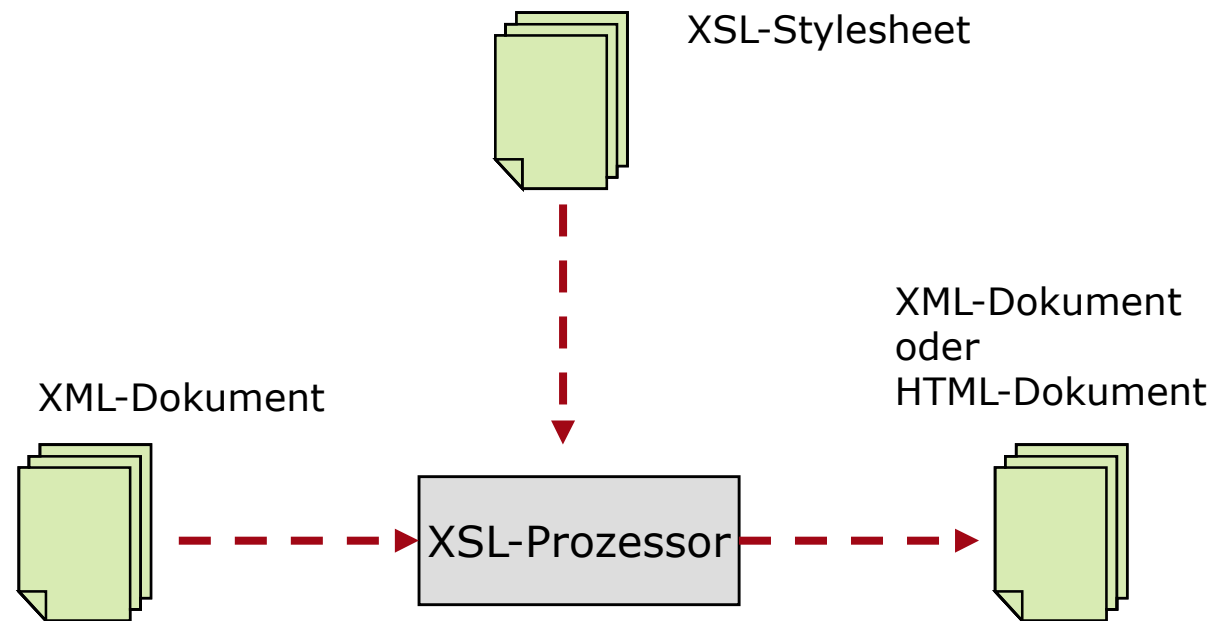
Überprüfung

Document Type
Defintions
(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur



XSL Beispiel

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- XSLT-Beispiel: test.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xml" href="test.xslt"?>
<GREETING>
    <MESSAGE subj="Hello!"> Hello John! </MESSAGE>
    <TO>John</TO>
    <MESSAGE subj="Hello again!"> Hello again John!
    </MESSAGE>
    <TO>John</TO>
</GREETING>
```

XSL Quellbaum

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions

(DTDs)

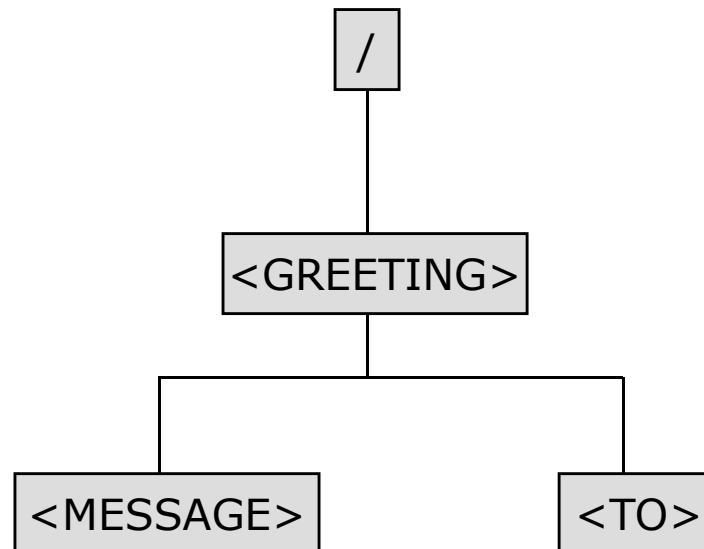
XML-Schemata

eXtensible

Style Sheet

Language

Literatur



XSL Beispiel

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions

(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur

- XSLT-Beispiel: test.html

```
<html>
  <head>
    <META http-equiv="Content-Type" content="text/html;
      charset=UTF-8">
    <title>Gruss</title>
  </head>
  <body>
    <p> Hello John! </p>John<p> Hello again John!
    </p>John
  </body>
</html>
```

XSL Zielbaum

Einführung

Werkzeuge

Überprüfung

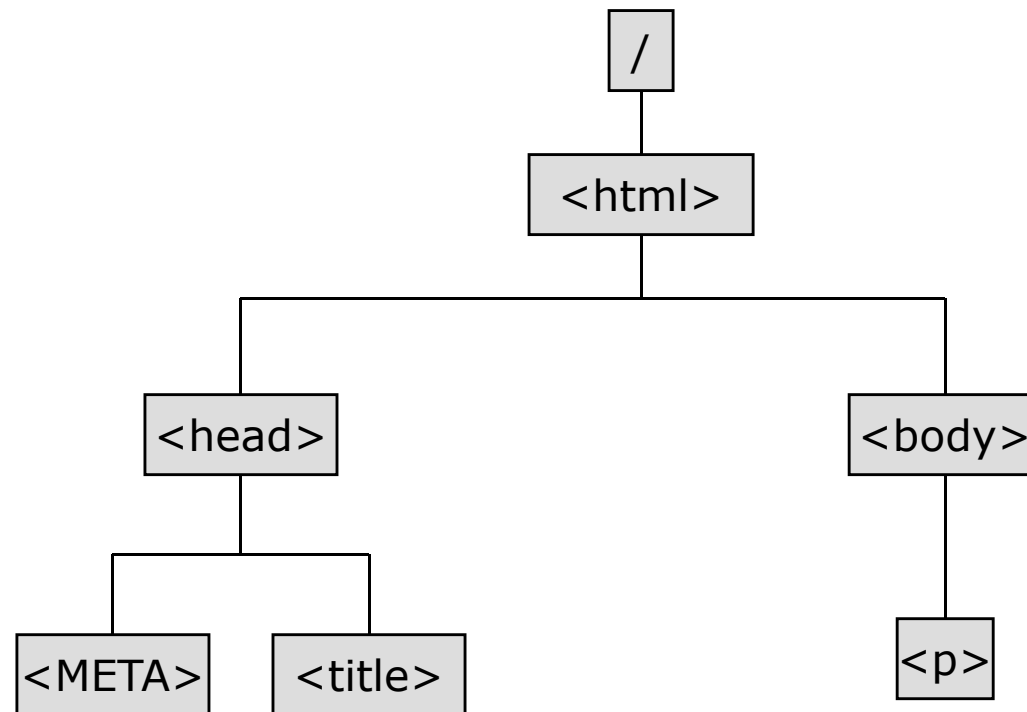
Document Type

Defintions
(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur



XSL Transformation

Einführung

Werkzeuge

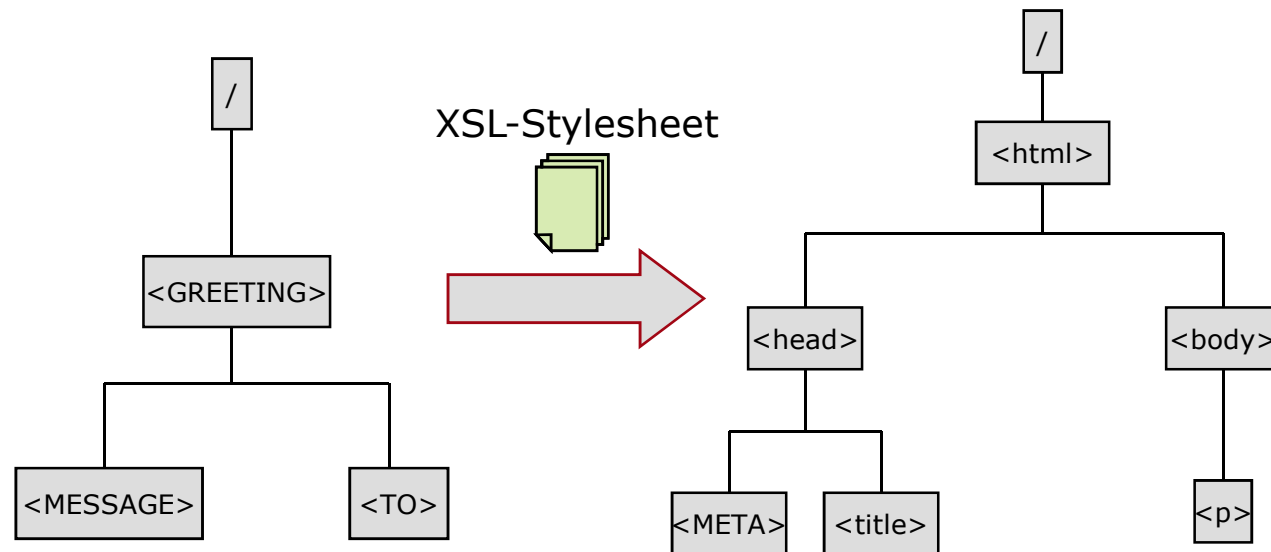
Überprüfung

Document Type
Definitions
(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur



XSL Beispiel (Stylesheet)

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions
(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur

- XSLT-Beispiel: test.xslt

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/
1999/XSL/Transform">
  <xsl:output method="html" version="1.0" encoding="UTF-8" />
    <xsl:template match="GREETING">
      <html><head><title>Gruss</title></head><body>
        <xsl:apply-templates/>
      </body></html>
    </xsl:template>
    <xsl:template match="MESSAGE">
      <p> <xsl:value-of select="."/> </p>
    </xsl:template>
  </xsl:stylesheet>
```

Unterschiede XSL vs CSS

Einführung

Werkzeuge

Überprüfung

Document Type
Definitions
(DTDs)

XML-Schemata

**eXtensible
Style Sheet
Language**

Literatur

- Cascading Stylesheets (css) führen keine echte Transformation aus und sind nicht so mächtig
- Xsl-Dateien führen eine echte Transformation aus und sind wesentlich mächtiger (können auch Informationen ausblenden).
- Umfangreiche CSS-Referenz: [SELFHTML](#)

XML Ausblick

- Einführung
 - Werkzeuge
 - Überprüfung
 - Document Type Definitions (DTDs)
 - XML-Schemata
 - eXtensible Style Sheet Language**
 - Literatur
- XML
 - ist sehr mächtig
 - sehr umfangreich
 - kann damit in diesem Seminar, daher nur Grundlagen vermitteln
 - Weitere XML-Themen
 - SAX (Simple API for XML, eventbasiert)
 - SOX (Schema for Object-oriented XML)
 - DOM (Document Object Model)
 - XML-Namespaces
 - XML-Stylesheets (XPath, ...)

DOM

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions
(DTDs)

XML-Schemata

eXtensible
Style Sheet
Language

Literatur

- DOM = Document Object Model
- Grundlegende Objekte
 - DOMException
 - DOMImplementation
 - DocumentFragment
 - Document
 - Node
 - NodeList
 - NamedNodeMap
 - CharacterData
 - Attr
 - Element
 - Text

Literatur I

- Einführung
- Werkzeuge
- Überprüfung
 - „XML Complete“, McGraw-Hill, ISBN 0-07-913702-4
 - Keine XML-Schemas
- Document Type Definitions (DTDs)
 - „JustXML“, Prentice Hall, ISBN 0-13-943417-8
- XML-Schemata
- eXtensible Style Sheet Language
 - „XML by Example“, QUE, ISBN 0-7897-2242-9
 - keine XML-Schemas behandelt (-)
- Literatur**

Literatur II

Einführung

Werkzeuge

Überprüfung

Document Type

Defintions

(DTDs)

XML-Schemata

eXtensible

Style Sheet

Language

Literatur

- „Professional XML“, 2nd Edition, ISBN 1-861005-05-9
 - Sehr umfangreich (ca. 1300 Seiten), XML-Schemas (++)