

# XML / RSS



# Inhaltsverzeichnis



- XML
  1. Einführung
  2. Syntax
  3. Eigenschaften
  4. Validierung
  5. Fazit
- RSS
  1. Einführung
  2. Struktur
  3. Fazit

# Definition

- XML = eXtensible Markup Language
- Metasprache zur Beschreibung von Sprachen (anwendungsspezifische Sprachen)
- Metasprache/Auszeichnungssprache  $\Rightarrow$  formatierte Ausgabe von Dokumenten (wie sieht der Text aus?) z.B. HTML
- SGML (Standard Generalized Markup Language), ebenfalls eine Metasprache



- XML**
- Einführung
- Syntax
- Eigenschaften
- Validierung
- Fazit
- RSS**
- Einführung
- Struktur

# Wo zu XML?

- XML definiert/bestimmt die Umschreibung von Informationen
  - Einheitliche Struktur von Informationen vereinfacht weitere Verarbeitung
- „XML schließt semantische Lücke“
  - Informationen nicht nur von Menschen sondern auch von Maschinen lesbar / bzw. weiter bearbeitbar
- XML nicht nur in Web
  - Dient als Transportwerkzeug zwischen verschiedenen Anwendungen
  - Dies kann alles mögliche sein



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# XML-Syntax



- Ein kleines Beispiel zum Einstieg:

```
<?xml version="1.0"?>
<bibliography>
  <book id="11912319">
    <title>Foundations of DBs</title>
    <authors>
      <author>Abiteboul</author>
      <author>Hull</author>
      <author>Vianu</author>
    </authors>
    <publisher>Addison-Wesley</publisher>
    ...
  </book>
  <book>
    ...
    <editor>Chomicki</editor>
    <release date="2007"/>
    ...
  </book> ...
</bibliography>
```

Annotations:

- Versions Deklaration: points to `<?xml version="1.0"?>`
- Attributinhalt: points to `id="11912319"`
- Attributname: points to `id`
- Elementinhalt: points to `Abiteboul`
- Elementname: points to `<book>`

XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

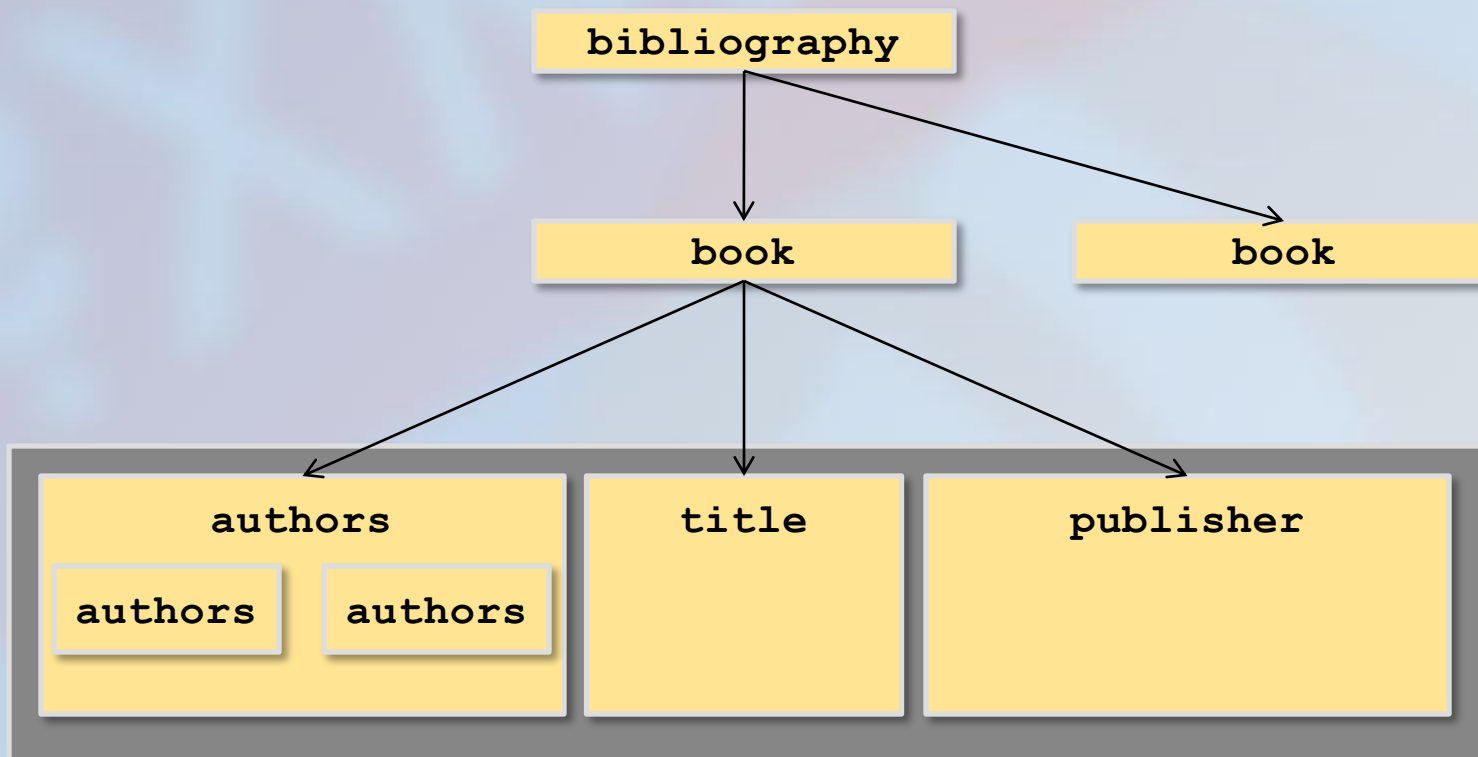
Einführung

Struktur

Verlauf

# Logischer Aufbau

- Gut als Baum visualisierbar



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# Element / Attribut



- Element

- Knoten im Baum
- Kann selbst einen Teilbaum beinhalten
- Ist ein wichtiger Bestandteil des Dokuments

```
<authors>
  <author>Abiteboul</author>
  <author>Hull</author>
  <author>Vianu</author>
</authors>
```

- Attribut

- Gehört immer zu einem Element
- Kann nur einmal pro Element verwendet werden
- Dient „nur“ zum einbinden zusätzlicher Informationen
  - Später hilfreich bei der automatisierten Informationsverarbeitung
  - u.U. leichter anzusprechen

```
<book id=„11912319“>
  ...
</book>
```

XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# Namensräume

- Problem
  - Grad  $\Rightarrow$  Temperatur?
  - Grad  $\Rightarrow$  Winkel?
- Lösung des Problems mit Hilfe von Namensräumen

```
<bspl>  
  <grad>...</grad>  
  <grad>...</grad>  
</bspl>
```

```
<?xml version="1.0" ?>  
<bestellung xmlns:produkt="http://localhost/XML/produkt"  
  xmlns:kunde="http://localhost/XML/kunde">  
  <produkt:nummer>p49393</produkt:nummer>  
  <produkt:name>JXY Rasierer VC100</produkt:name>  
  <produkt:menge>1</produkt:menge>  
  <produkt:preis>69,--</produkt:preis>  
  <kunde:nummer>k2029</kunde:nummer>  
  <kunde:name>Meier, Fritz</kunde:name>  
  <kunde:lieferadresse>Donnerbalkenstr.14, 80111  
    München</kunde:lieferadresse>  
</bestellung>
```



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf



# Wohlgeformte Dokumente

- XML-Dokument muss bestimmte Eigenschaften erfüllen (neben der eigentlichen Syntax)
  1. Gültige Deklaration des XML-Dokuments
  2. Mindestens ein Datenelement
  3. Korrekte Schachtelung von Elementen

```
<?xml version="1.0"?>
...
<book id=„11912319“>
  <title>Foundations of DBs</title>
  <publisher>Addison-Wesley</publisher>
  ...
</book>
...
```

- Dieses Dokument ist wohlgeformt (well-formed)



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# Gültige Dokumente

- XML-Dokument heißt gültig wenn gilt:
  1. Dokument ist wohlgeformt
  2. Regeln eines eingebundenen DTD wird beachtet
- DTD
  - Ist eine Dokumententypdefinition
  - Nur durch eine Dokumententypdefinition erhält XML-Dokument ihren Sinn

```
<?xml version="1.0"?>
<tür>
  <grad gehen="true">
    <feucht>rot</feucht>
  </grad>
  <grad>hand</grad>
</tür>
```



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# Dokumenttypen - Bspl.

- DTD für das Syntax-Beispiel

```
<!ELEMENT bibliography (paper)*>
<!ELEMENT paper (authors, release?, title, booktitle)>
<!ELEMENT authors (author)+>
<!ELEMENT author (#PCDATA)>
<!ELEMENT fullPaper EMPTY>
<!ELEMENT title (#PCDATA)>
<!ELEMENT booktitle (#PCDATA)>
<!ATTLIST fullPaper source ENTITY #REQUIRED>
<!ATTLIST paper id ID #REQUIRED>
```

- Verschiedene Möglichkeiten Elemente und Attribute zu setzen



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# DTD - Elemente



## Anzahl

(element)	1 mal element
(element)?	0 oder 1 mal
(element)+	Beliebig viele (mind. eins)
(element)*	Beliebig viele (auch keins)

## Reihen und Alternativen

(element1, element2, ...)	Genau diese Folge, alle, 1 mal
(element1   element2   ...)	Eins von den Elementen

## Inhalt

(#PCDATA)	Zeichenkette (Elementinhalt)
Empty	Kein Inhalt
(#PCDATA   element2   ...)*	Mix aus Zeichenkette und Elementen
ANY	Alles Kombinationen möglich

XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

# DTD - Attribute



## Vorgaben

#REQUIRED	notwendig
#IMPLIED	möglich
#FIXED	Finaler Wert
leer	Standardwert, nicht notwendig

## Attributtypen

CDATA	Um Markup Zeichenketten benutzen zu können z.B. <, >, ...
ID	XML-Globaler Identifier
IDREF	Referenz zu elementen mit eindeutiger ID
NMTOKEN	Zur Benutzung „exklusiver“ Zeichen
ENTITY	z.B. Einbindung externer Ressourcen
NOTATION	Identifier zur Erkennung von externen Daten
(muster)	Nur Zeichenkette „muster“
(muster1   muster2   ...)	Eins von den Mustern

XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# DTD - Einbindung

- Die DTD kann man auf zwei verschiedene Arten verwenden

Intern

```
<?xml version="1.0"?>
<!DOCTYPE bibliography [
<!ELEMENT bibliography (paper)*>
...
]>
<bibliography>...</bibliography>
```

Extern

```
<?xml version="1.0"?>
<!DOCTYPE bibliography SYSTEM "bib.dtd">
<bibliography>...</bibliography>
```

- Kombination auch möglich
  - Reihenfolge wichtig da z.B. keine Vererbung o.ä.



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# DTD - Alternativen

- DTD stammt noch aus SGML Zeiten
  - Komplex, eigene Syntax
- Nützliche Funktionen fehlen
  - Individuelle Einschränkungen
  - Namensräume fehlen
  - Vererbung
- Alternative: XML-Schema (XSD)
  - XML-Syntax
  - Erweiterbar (Wiederverwendung)
  - Vererbung
- Mächtiger als DTDs



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# XML - Schema

- Vorteil bzgl. DTDs da XML-Syntax
- Besser strukturiert
- Besserer Einsatz von Datenbanken möglich
  
- Einige unterstützte Datentypen
  - `xs:string`
  - `xs:integer`
  - `xs:decimal`
  - `xs:boolean`
  - `xs:date`



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf



# XML-Schema - Syntax

- Hierarchien bleiben erhalten durch explizite Deklarierung des Wurzel-Elements als „complexType“

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="artikel">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="titel" type="xs:string"/>
        <xs:element name="teaser" type="xs:string"/>
        <xs:element name="inhalt" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# XML-Schema - Syntax

- Hierarchien bleiben erhalten durch explizite Deklarierung des Wurzel-Elements als „complexType“

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" >
  <xs:element base="string" type="complexType" name="artikel" >
    <xs:complexType base="string" name="complexType" >
      <xs:sequence >
        <xs:element type="string" name="titel" />
        <xs:element type="string" name="teaser" />
        <xs:element type="string" name="inhalt" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
artikel.xml
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE artikel SYSTEM "artikel.dtd">
  <artikel>
    <titel>Der Titel</titel>
    <teaser>Der Teaser</teaser>
    <inhalt>Der Inhalt</inhalt>
  </artikel>

artikel.dtd
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" >
  <xs:element base="string" type="complexType" name="artikel" >
    <xs:complexType base="string" name="complexType" >
      <xs:sequence >
        <xs:element type="string" name="titel" />
        <xs:element type="string" name="teaser" />
        <xs:element type="string" name="inhalt" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# XML - Verarbeitung

- Validierung von XML-Dokumenten mittels Parser
  - Nichtvalidierend
  - Validierend
- Ergebnisse
  - DOM (Document Object Model)
  - SAX (Simple API for XML)

## DOM

- OO / Sprachunabhängig
- Hierarchie (Baumstruktur)
- Einlesen des gesamten Dokuments (Ressourcen?)

## SAX

- Teilauslesung des Dok.
- Speicherschonend
- Eher bei maschinell erstellten Dokumenten



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# Fazit

- Textbasiertes, Halbstrukturiertes Datenmodell
- Hierarchische Darstellung des Dokuments
- Wichtig für uns:
  - Teile des Dokuments können exakt angesprochen werden
  - Informationen liegen im Idealfall gut strukturiert vor
  - Einfache Weiterverarbeitung möglich
    - Automatisierbar



**XML**

Einführung

Syntax

Eigenschaften

Validierung

Fazit

**RSS**

Einführung

Struktur

Verlauf

# RSS

- Inzwischen „Really Simple Syndication“
- Früher „RDF Site Summary“ (Basierend auf RDF)
- Elektronisches Nachrichtenformat
- Dient zum verbreiten von Inhalten bzgl. einer Website
  - Informationen gelangen automatisch an den RSS-Abonnenten
  - Wie Rundmail aber:
    - Clientseitiger call
    - Keine Auswahl von Abonnenten möglich
- Basiert inzwischen auf XML 1.0



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# RSS - Bspl.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<rss version="2.0">
<channel>
  <title>Titel des Feeds</title>
  <link>Adresse der Webpräsenz</link>
  <description>Kurze Beschreibung des Feeds</description>
  <language>de-de</language>
  <copyright>urheberrechtliche Informationen</copyright>
  <pubDate>Datum der Erstellung</pubDate>
  ...
  <item>
    <title>Titel des ersten Artikels</title>
    <description>Eine kurze Zusammenfassung des
  Artikels</description>
    <link>Adresse zur Gesamtansicht des Artikels</link>
    <author>Autor des Artikels &lt;E-Mail-Adresse></author>
    <guid>Eindeutige Nummer des Artikels, muss für jeden Eintrag
anders sein</guid>
    </item>
    ...
  </channel>
</rss>
```



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# RSS - Bspl.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<rss version="2.0">
<channel>
  <title>Titel des Feeds</title>
  <link>Adresse der Webpräsenz</link>
  <description>Kurze Beschreibung des Feeds</description>
  <language>de-de</language>
  <copyright>urheberrechtliche Informationen</copyright>
  <pubDate>Datum der Erstellung</pubDate>
  ...
  <item>
    <title>Titel des ersten Artikels</title>
    <description>Eine kurze Zusammenfassung des
  Artikels</description>
    <link>Adresse zur Gesamt
    <author>Autor des Artike
    <guid>Eindeutige Nummer
  anders sein</guid>
  </item>
  ...
</channel>
</rss>
```

```
<![CDATA[
  <h1>Hier kann auch der
  vollständige HTML-Inhalt
  des Artikels stehen</h1>
  <p>...</p>
]]>
```



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf

# Besonderheiten

- Gute Suchmaschinenkompatibilität
  - Warum? Preisfrage!
- Integration von verschiedenen RSS-Feeds auf eine Seite möglich
- Struktur maschinell leicht zu lesen und zu separieren
- Auslesen von RSS-Feeds durch geeignete Reader
  - AmphetaDesk
  - Newsmonster
  - ...



XML

Einführung

Syntax

Eigenschaften

Validierung

Fazit

RSS

Einführung

Struktur

Verlauf



**Danke  
für eure  
Aufmerksamkeit**