

E-Business und digitale Prozesse

Übung

Grundkurs XML I:

Grundelemente wohlgeformter XML-Dateien

Prof. Dr. Thomas Myrach
Universität Bern
Institut für Wirtschaftsinformatik
Abteilung Informationsmanagement

Stand: 18. September 2022

- Sie kennen die XML-Grundkonstrukte Element und Attribut.
- Sie können XML-Dokumente als eine hierarchische Struktur beschreiben.
- Sie wissen, wie man einen Prolog und Kommentare formuliert.
- Sie können XML von HTML abgrenzen.
- Sie wissen, unter welchen Bedingungen ein XML-Dokument als wohlgeformt gilt.
- Sie wissen, was unter der Gültigkeit eines XML-Dokuments zu verstehen ist.
- Sie können aus XML-Dokumenten heraus XML-Schemas referenzieren.
- Sie wissen, wie XML-Dokumente validiert werden können.

Agenda: Übersicht

Aufbau eines XML-Dokuments

Abgrenzung XML und HTML

Wohlgeformtheit und Gültigkeit

Logischer Aufbau eines XML-Dokuments

- Im XML-Dokument können verschiedene Informationseinheiten unterschieden werden.
- Die Informationseinheiten bilden, hierarchisch angeordnet, die logische Struktur eines XML-Dokuments (Baumstruktur).
- Die W3C-Empfehlung unterscheidet insgesamt elf Typen von Informationseinheiten:
 - Element
 - Attribut
 - Kommentar
 - Verarbeitungsanweisung
 - Zeichen
 - Namensraum
 - usw.

Beispiel einer XML-Datei (1/2)

Textansicht

```

<?xml version="1.0" encoding="UTF-8"?>
<teilnehmer>
  <!-- Diese Teilnehmerliste ist unvollständig -->
  <student>
    <name>Bigler Tim</name>
    <matrikelnummer>06-100-333</matrikelnummer>
    <promotion />
    <kontakt>
      <fon kategorie="mobile">+41 (0)76 666 77 88</fon>
      <mail kategorie="privat">tim.bigler@gmx.net</mail>
      <mail kategorie="studium">tim@students.unibe.ch</mail>
    </kontakt>
  </student>
  <student>
    ...
  </student>
  ...
</teilnehmer>

```

Prolog (points to `<?xml version="1.0" encoding="UTF-8"?>`)

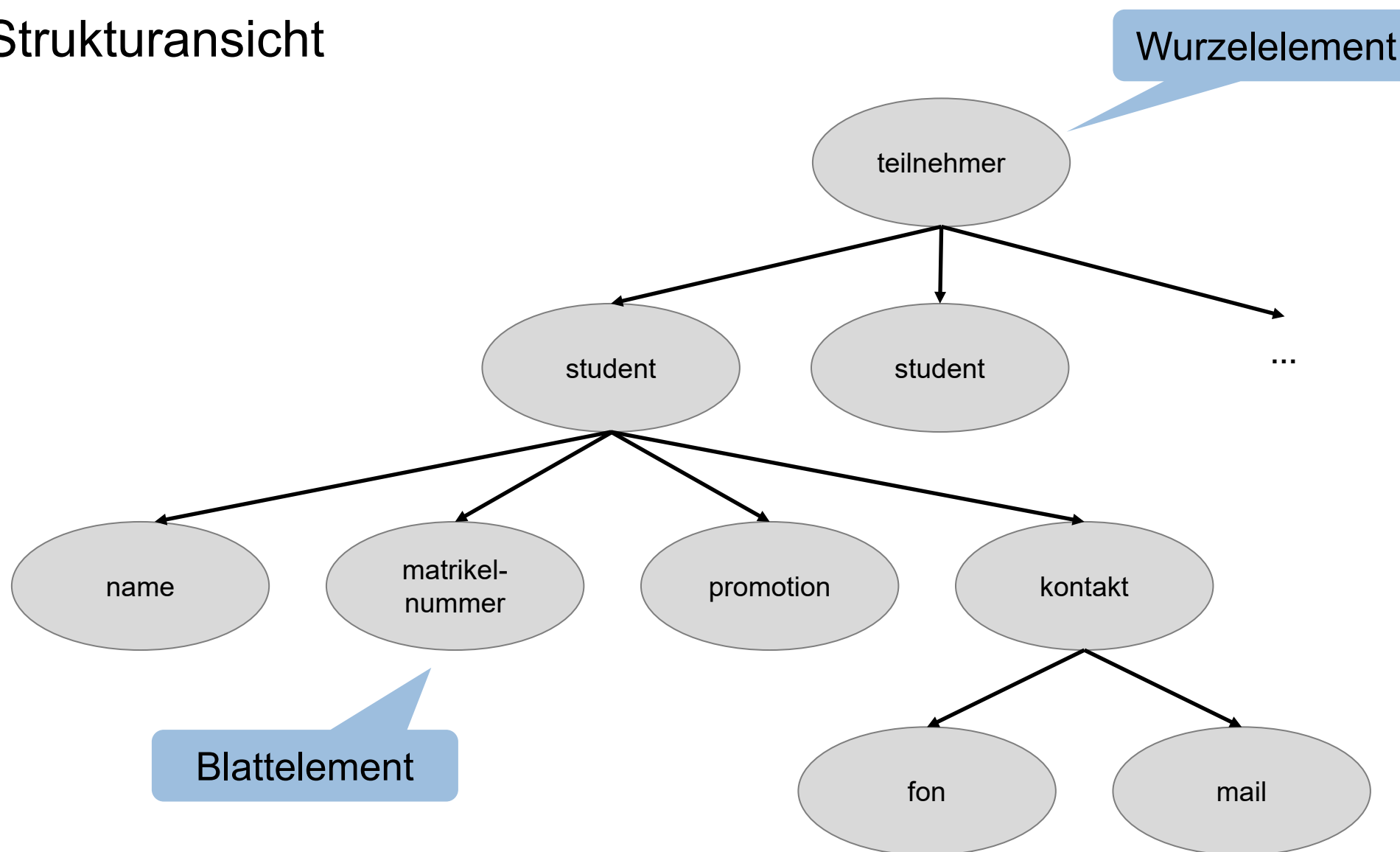
Kommentar (points to `<!-- Diese Teilnehmerliste ist unvollständig -->`)

Element (points to `<student>`)

Attribut (points to `kategorie="mobile"` in `<fon kategorie="mobile">`)

Beispiel einer XML-Datei (2/2)

Strukturansicht



Element (1/3)

Hierarchie

- Das Element ist die grundsätzliche Strukturierungseinheit innerhalb eines XML-Dokuments.
- Das erste Element im XML-Dokument (nach dem Prolog) ist das Wurzelement.
- Ein XML-Dokument muss genau ein Wurzelement haben.
- Ausser dem Wurzelement hat jedes Element genau ein Elternelement.
- Ausser den Blattelementen hat ein Element ein bis mehrere Kindelemente.
- Kindelemente müssen in die übergeordneten Elternelemente hierarchisch geschachtelt werden.
- Überlappungen zwischen den einzelnen Elementen sind ausgeschlossen.

Element (2/3)

Aufbau

- Ein vollständiges Element setzt sich zusammen aus dem Start-Tag, dem Inhalt und dem End-Tag.
 - Beispiel:
`<elementname>inhalt</elementname>`
wobei
 - Start-Tag: `<elementname>`
 - Elementinhalt: `inhalt`
 - End-Tag: `</elementname>`
- Wenn ein Element keinen Inhalt hat, können Start- und End-Tag zusammen gezogen werden (Leeres-Element-Tag).
 - Beispiel:
`<elementname></elementname>` ist gleichbedeutend mit `<elementname />`

Element (2/3)

Inhalt

- Der Inhalt eines Elements unterscheidet sich je nach Position in einer Elementhierarchie.
- Blattelemente:
 - Inhalte betreffen die eigentlichen Nutzdaten.
 - Je nach Datentyp sind das z.B. Texte, Zahlen, Datumsangaben.
 - Beispiel: `<name>Bigler Tim</name>`
- Sonstige Elemente:
 - Inhalte betreffen vor allem hierarchisch untergeordnete Elemente.
 - Beispiel: `<teilnehmer>`
 - `<student> ... </student>`
 - `<student> ... </student>`
 - `...``</teilnehmer>`

- Attribute können nur im Kontext von Elementen benutzt werden.
- Ein Attribut ermöglicht, zu einem Element spezifische Eigenschaften zu formulieren.
- Attribute bestehen aus einem Attributnamen und einem diesem zugeordneten Attributwert.
- Dargestellt werden Attribute in der Form `attributname="attributwert"`.
- Die für ein Attribut zugelassenen Werte können beschränkt oder vorgegeben sein.
- Attribute können ähnlich wie Unter-/Kindelemente benutzt werden.

- Ein Attribut wird innerhalb des Start-Tags spezifiziert.

- Beispiel:

```
<elementname attributname="attributwert">  
inhalt</elementname>
```

- Ein Element kann mehrere Attribute haben. Die einzelnen Attribute werden dann ohne spezielles Trennzeichen aufgeführt.

- Beispiel:

```
<elementname attributname1="attributwert"  
attributname2="attributwert">inhalt </elementname>
```

- Zum einzelnen Element müssen die Attributnamen eindeutig sein.

Sondertypen

Prolog

- Dem eigentlichen, hierarchisch strukturierten XML-Dokument wird ein Prolog vorangestellt.
- Der Prolog ist nicht zwingend, jedoch empfohlen um das Dokument sofort als XML-Dokument zu identifizieren.
- Im Rahmen eines Prologs können verschiedene Attribute angegeben werden.
- Beispiele:
 - `<?xml version="1.0"?>`
 - `<?xml version="1.0" encoding="UTF-8"?>`
- Das Attribut "version" im Prolog wird auch als XML-Deklaration bezeichnet.
- Das Attribut "encoding" gibt den zu verwendenden Zeichensatz an, d. h. wie Bits und Bytes als Zeichen dargestellt werden sollen.

Sondertypen

Kommentar

- Ein Kommentar ist ebenfalls eine Informationseinheit, welche oft anzutreffen ist.
- Kommentare können überall geschrieben werden, wo ein normales Element auch stehen könnte.
- Ein Kommentar beginnt mit „<!--“ und endet mit „-->“.
- Beispiel:

```
<!-- Jetzt fängt das Dokument an -->
<wurzel>
  <!-- Hier ist der Inhalt -->
  ...
</wurzel>
```

Agenda: Übersicht

Aufbau eines XML-Dokuments

Abgrenzung XML und HTML

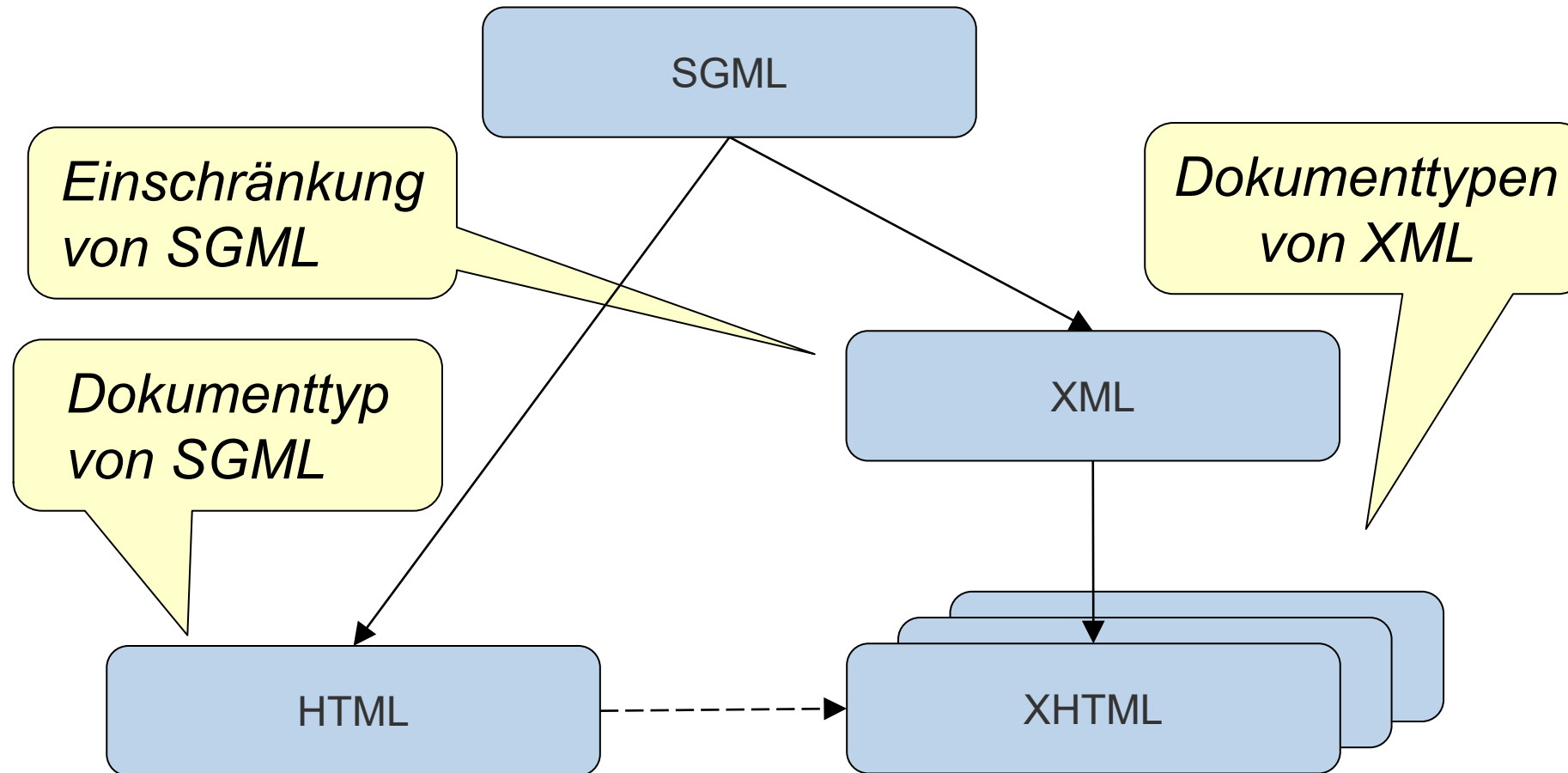
Wohlgeformtheit und Gültigkeit

HTML und XML

Grundlagen

- Die Auszeichnungssprachen HTML und XML sind eng verwandt.
- HTML und XML haben prinzipiell die gleiche Auszeichnungslogik.
- Beide werden im Kontext des Internets eingesetzt.
- HTML ist die Auszeichnungssprache, die zur Strukturierung von Texten eingesetzt wird, die in einem Web-Browser dargestellt werden sollen.
- Auch XML-Dateien sind textbasiert und können in modernen Web-Browsern dargestellt werden.
- XML wird für vielfältige Zwecke einsetzen.
- Ein wichtiger Anwendungszweck ist der Datenaustausch.

Abgrenzung XML und HTML



HTML vs. XML

Verfügbare Elemente

- In HTML existiert eine definierte Menge von Tags, mit denen ein Dokument strukturiert werden muss.
- Alle anderen Tags werden nicht als HTML interpretiert.
- Üblicherweise ignorieren Web-Browser nicht bekannte Tags.
- In XML können beliebige Tags definiert werden.
- Die zu verwendenden Tags lassen sich separat festlegen
 - Dokumenttypdefinition (DTD)
 - XML-Schema.
- Dokumente lassen sich nach den spezifischen Erfordernissen einer Problemstellung strukturieren.

HTML vs. XML

Beispiel für verfügbare Elemente

HTML-Beispiel Adresse

```
<p>  
Institut für Wirtschaftsinformatik<br>  
Engenhaldenstrasse 8<br>  
CH-3012 Bern<br>  
</p>
```

XML-Beispiel Adresse

```
<adresse>  
<name>Institut für Wirtschaftsinformatik</name>  
<strasse>Engenhaldenstrasse 8</strasse>  
<plz>CH-3012</plz><ort>Bern</ort>  
</adresse>
```

HTML vs. XML

Gross- und Kleinschreibung

- In HTML wird die Gross- und Kleinschreibung nicht unterschieden.
- Start- und End-Tag können unterschiedlich geschrieben werden.
- Beispiel:
`<P>Dies ist ein Absatz</p>`
(korrektes HTML)
- In XML wird zwischen Gross- und Kleinschreibung von Tags unterschieden (case sensitiv).
- Eine unterschiedliche Schreibweise von Start- und End-Tag ist in XML unzulässig.
- XHTML-Tags werden prinzipiell klein geschrieben.
- Beispiel:
`<p>Dies ist ein Absatz</p>`
(korrektes XHTML)

HTML vs. XML

Start- und End-Tag

- In HTML existieren Elemente, die nur einen Start-Tag haben.
- Beispiel:
 - `
` für break
 - `` für image
- Jedes XML-Element muss ein Start- und End-Tag haben.
 - `<tag></tag>`
- Hat ein XML-Element keinen Inhalt, so können Start und End-Tag zusammengezogen werden:
 - `<tag/>`
- Beispiel XHTML:
 - `
` für break
 - `` für image

HTML vs. XML

Überprüfung der Syntax

- Bei HTML wird mit Verstössen gegen die Syntax relativ locker umgegangen.
- Web-Browser zeigen für gewöhnlich auch syntaktisch falsche HTML-Dokumente ohne weiteres an.
- Allerdings kann es zu unerwarteten und ungewünschten Darstellungen kommen.
- Bei XML wird strikte auf die Einhaltung der Syntax geachtet.
- Auch Web-Browser zeigen nicht korrekte XML-Dokumente nicht an, sondern machen auf Fehler aufmerksam.
- Bei XML-Dokumente ist die Gewähr grösser, dass sie korrekt sind und richtig interpretiert werden können.

Agenda: Übersicht

Aufbau eines XML-Dokuments

Abgrenzung XML und HTML

Wohlgeformtheit und Gültigkeit

Korrektheit von XML-Dokumenten

Wohlgeformtheit

- Wohlgeformtheit oder wohlgeformtes Dokument bedeutet, dass die Auszeichnungen die grundsätzlichen Regeln der XML-Syntax nicht verletzen.
- Die Anforderungen der Wohlgeformtheit bestehen unabhängig von der Gültigkeit (DTD oder XML-Schema sind dazu also nicht erforderlich).
- Wohlgeformtheit beinhaltet die grundlegende XML-Syntax wie z. B.
 - genau ein Wurzelement,
 - hierarchische Verschachtelung der Elemente (Baumstruktur),
 - Start- und End-Tag (allenfalls zusammengezogen).
- Verstösse der Wohlgeformtheit führen dazu, dass ein Dokument im Browser nicht angezeigt und stattdessen eine Fehlermeldung ausgegeben wird.

Korrektheit von XML-Dokumenten

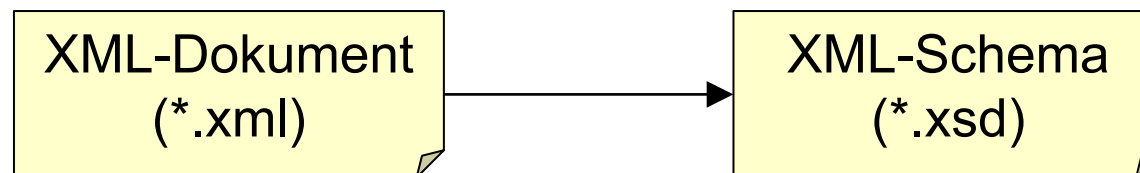
Gültigkeit (Validität)

- Mit Gültigkeit oder validem XML-Dokument wird ausgedrückt, ob ein Dokument alle erforderlichen Informationen und in der korrekten Reihenfolge und Form enthält.
- Soll ein XML-Dokument einem vorgegebenen Strukturmodell genügen, sind die Vorgaben explizit zu definieren.
- Es existieren zwei Optionen, wie die Vorgaben definiert werden können:
 - die Dokumenttyp-Definition (DTD) und
 - das XML-Schema.
- Gemäss W3C-Standard ist ein XML-Dokument dann valide,
 - wenn es wohlgeformt ist,
 - auf eine DTD bzw. ein XML-Schema verweist und
 - das darin beschriebene Format einhält.

Gültigkeit von XML-Dokumenten

Referenzierung eines XML-Schemas (1/2)

- Ein XML-Dokument kann mit einem XML-Schema verbunden werden.
- Dies geschieht, indem aus dem XML-Dokument auf das XML-Schema referenziert wird.
- Die Referenzierung erfolgt durch spezielle Attribute.
- Diese werden typischerweise im Wurzelement des XML-Dokuments definiert.



Gültigkeit von XML-Dokumenten

Referenzierung eines XML-Schemas (2/2)

```
<wurzel  
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  xsi:noNamespaceSchemaLocation="schema.xsd">  
</wurzel>
```

1. Deklaration des Dokuments als Instanz eines Schemas:
 - Definition eines Namespace «xsi» für die Deklaration als Schema-Instanz.
 - Die URL verweist auf eine Schema-Definition für Schema-Instanzen.
 - In ihnen werden Attribute für Schema-Instanzen definiert.
2. Verweis auf den Speicherort des Schemas:
 - Definition von Schema-Instanz-Attributen mit dem Präfix «xsi».
 - Vor allem der Verweis auf das referenzierte XML-Schema.
 - Alternative: Attribute «noNamespaceSchemaLocation» oder «SchemaLocation».

- Über ein (verbundenes) XML-Schema lassen sich XML-Dokumente validieren.
- Web-Browser unterstützen typischerweise keine Validierung.
- Sie zeigen XML-Dokumente auch dann an, wenn sie gegen die Vorgaben des XML-Schemas verstossen.
- XML-Editoren verfügen dagegen über Validierfunktionen.
- Mit ihnen lässt sich prüfen, ob ein XML-Dokument den Vorgaben des XML-Schemas entspricht.
- Sie unterstützen durch entsprechende Fehlermeldungen das Auffinden der Fehler und deren Korrektur.
- Auf die Art kann sichergestellt werden, dass nur korrekte XML-Dokumente verwendet werden.

Ende der Lektion

Wie geht es weiter?



- Zu dieser Lektion stehen Übungsaufgaben zur Verfügung.
- Diese Aufgaben können in Ilias bearbeitet werden.
- Die Lösung der Aufgaben wird besprochen.