

The background of the slide features an abstract, textured pattern in warm tones of orange, red, and yellow at the top. Below this, a solid light gray area contains the title and author information. A thin horizontal line with a colorful, pixelated pattern separates the title from the author's name.

DTD en XML Schema

Veerle Ongenae

Overzicht Onderwerpen XML

- XML

- XML en namespaces

- XML Technologieën

- Structureren en datatypes: DTD en XML Schema
 - Presentatie: CSS en XSL
 - Manipulatie (o.a. transformatie): XPath, DOM en SAX
 - Diverse

Structureren en datatypes

- Structuur van een XML-document beschrijven
- Twee technologieën
 - DTD (Document Type Definition)
 - XML Schema
- Een geldig XML-document (valid)
 - De structuur van het document is conform de opgegeven DTD of het opgegeven XML Schema
- Verschil “geldig” en “welgevormd”

DTD

- Voorbeelden
 - boekendtd.xml
 - boeken.dtd
 - query.xml
 - query.dtd
 - reply.xml
 - reply.dtd

DTD

- Beschrijft de structuur van een XML-document
 - Welke elementen en attributen
 - Volgorde elementen
 - Frequentie elementen
 - Relatie tussen elementen
 - Inhoud element/attribuut
- Kan intern (in XML-document) of extern (in een apart bestand)

DTD

- Link tussen XML-bestand en DTD
- Inhoud DTD
 - Beschrijving elementen
 - Beschrijving attributen
 - Entiteiten

DTD in XML-file

<?xml version="1.0"?>

<!DOCTYPE basiselement [
 elementdeclaraties, attributen, ...
]>

<basiselement>

...

</basiselement>

DTD aangeven in XML-file

<?xml version="1.0"?>

**<!DOCTYPE basiselement SYSTEM
"bestand">**

<basiselement>

...

</basiselement>

DTD aangeven in XML-file

<?xml version="1.0"?>

<!DOCTYPE basiselement PUBLIC
"naam" "url">

<basiselement>

...

</basiselement>

DTD

- Link tussen XML-bestand en DTD
- Inhoud DTD
 - Beschrijving elementen
 - Beschrijving attributen
 - Entiteiten

Beschrijving van elementen

■ Syntax

<!ELEMENT naamelement inhoud>

■ Inhoud

■ **(#PCDATA)**: tekst

- parsable character data
- geen speciale tekens!!

<!ELEMENT isbn (#PCDATA)>

Beschrijving van elementen

- Nul, één of meerdere kindelementen
 - Aangeven hoeveel keer een kind voorkomt
 - ? (0 of 1)
 - + (1 of meer)
 - * (0 of meer)
 - Volgorde

<!ELEMENT books (book*)>

**<!ELEMENT author (name, initial?,
firstname+)>**

Beschrijving van elementen

- Keuze

<!ELEMENT catalogus (boek | cd | tijdschrift | dvd | video)>

<!ELEMENT catalogus (boek | cd | tijdschrift | dvd | video)*>

- Leeg: geen kindelementen

<!ELEMENT query EMPTY>

DTD

- Link tussen XML-bestand en DTD
- Inhoud DTD
 - Beschrijving elementen
 - Beschrijving attributen
 - Entiteiten

Beschrijving van attributen

■ Syntax

```
<!ATTLIST naamelement  
    attr1 type1 standaard1  
    attr2 type2 standaard2  
    ... >
```

- Naam van het attribuut
- Type van het attribuut
- Standaard
 - Verplicht?
 - Vaste waarde?

Beschrijving van attributen

■ Voorbeelden

<!ATTLIST query

isbn NMTOKEN #REQUIRED

zipcode (75230 | 75233 | 75235) #REQUIRED>

<!ATTLIST reply

time CDATA #REQUIRED

price NMTOKEN #REQUIRED>

Attribuuttypes

- **CDATA**: lettertekens, geen speciale tekens
- **NMTOKEN**: combinatie van letters, cijfers, ., -, _, : (geen spaties, ...)
- Opsomming: lijst van mogelijkheden
- **ID**: een unieke id
- **IDREF**: een unieke id van een ander element
- ...

Standaarden voor attributen

- #IMPLIED: optioneel
- #REQUIRED: vereist
- #FIXED “string”: vaste waarde
- “string”: de standaardwaarde

DTD

- Valideren XML-bestanden
- Bestaat uit
 - Beschrijving van elementen
 - Beschrijving van attributen
 - Entiteiten

Structureren en datatypes

- Structuur van een XML-document beschrijven
- Twee technologieën
 - DTD (Document Type Definition)
 - XML Schema
- Een geldig XML-document (valid)
 - De structuur van het document is conform de opgegeven DTD of het opgegeven XML Schema
- Verschil “geldig” en “welgevormd”

XML Schemas

■ Voorbeelden

- queryxsd.xml
- query.xsd
- replyxsd.xml
- reply.xsd
- boekenxsd.xml
- boeken.xsd

XML Schema

- Link tussen XML-document en XML Schema
- XML Schema
 - Structuur
 - Beschrijving van elementen
 - Types

XML Schema aangeven in XML-file

<?xml version="1.0"?>

<basiselement

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

**xsi:noNamespaceSchemaLocation="basis
element.xsd">**

...

</basiselement>

XML Schema aangeven in XML-file

- Het attribuut **noNamespaceSchemaLocation**
 - URI schema
 - Van de elementen die niet tot een namespace behoren

XML Schema aangeven in XML-file

<?xml version="1.0"?>

<basiselement

xmlns="namespaceBasiselement"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="namespaceBasiselement basiselement.xsd">

...

</basiselement>

XML Schema

- Link tussen XML-document en XML Schema
- XML Schema
 - Structuur
 - Beschrijving van elementen
 - Types

Structuur van een XML Schema

- XML-document
- Extensie .xsd of .xs
- Basiselement
 - schema
 - Namespace
<http://www.w3.org/2001/XMLSchema>

Basiselement XML Schema

```
<xs:schema elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

■ Of

```
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://www.boeken.be"
  targetNamespace="http://www.boeken.be"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
```

qualified $\leftrightarrow$ unqualified

■ qualified

- Behoort tot namespace
- Uniek in namespace

■ unqualified

- Enkel binnen een element gekend
- Hoeft niet in uniek te zijn in namespace
 - Bv. 2 verschillende elementen met een zelfde attribuut
 - Bv. 2 elementen titel:
 - Aanspreektitel binnen een persoon-element
 - Titel van een boek binnen een boek-element

Inhoud schema-element

- Elementbeschrijving
 - Altijd qualified, indien onmiddellijk onder schema-element
- Type declaraties
- ...

XML Schema

- Link tussen XML-document en XML Schema
- XML Schema
 - Structuur
 - Beschrijving van elementen
 - Types

Declaratie van elementen

- Voorbeelden declaraties
 - Elementen van een bestaand type zonder kindelementen of attributen
 - Lege elementen met attributen
 - Elementen met kindelementen
 - Element met inhoud en attributen

Elementen van een ingebouwd type zonder kindelementen of attributen

■ Voorbeeld

- `<xs:element name="isbn" type="xs:string" />`

■ Voorbeelden van ingebouwde datatypes

- token
- NMTOKEN
- int
- date
- duration
- ...

Declaratie van elementen

- Voorbeelden declaraties
 - Elementen van een bestaand type zonder kindelementen of attributen
 - Lege elementen met attributen
 - Elementen met kindelementen
 - Element met inhoud en attributen

Lege elementen met attributen

■ Voorbeeld

```
<xs:element name="query" type="queryType"/>
<xs:complexType name="queryType">
  <xs:attribute name="isbn" type="xs:NMTOKEN"/>
  <xs:attribute name="zipcode" type="zips"/>
</xs:complexType>
```

■ Alternatief

```
<xs:element name="query">
  <xs:complexType>
    <xs:attribute name="isbn" type="xs:NMTOKEN"/>
    <xs:attribute name="zipcode" type="zips"/>
  </xs:complexType>
</xs:element>
```

Declaratie van elementen

- Voorbeelden declaraties
 - Elementen van een bestaand type zonder kindelementen of attributen
 - Lege elementen met attributen
 - Elementen met kindelementen
 - Element met inhoud en attributen

Elementen met kindelementen

■ Voorbeeld

```
<xs:element name="book">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="isbn"/>
      <xs:element ref="title"/>
      <xs:element ref="author" minOccurs="1"
        maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

Elementen met kindelementen

```
<xs:element name="author">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="name" type="xs:string"/>
      <xs:element name="initial"
        type="xs:NMTOKEN" minOccurs="0"
        maxOccurs="1"/>
      <xs:element name="firstname"
        type="xs:token" minOccurs="1"
        maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

Declaratie van elementen

- Voorbeelden declaraties
 - Elementen van een bestaand type zonder kindelementen of attributen
 - Lege elementen met attributen
 - Elementen met kindelementen
 - Element met inhoud en attributen

Een element met attributen en inhoud

■ Voorbeeld

```
<xsd:element name="directeur" type="directeurtype"/>
<xsd:complexType name="directeurtype">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute default="neen" name="wnd"
type="janeentype" use="optional"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
```


Syntax elementdeclaratie

- Element

- Naam
- Type

- Mogelijke syntax

`<xs:element name="..." type = ".." />`

- of

`<xs:element name="...">`

typedeclaratie

`</xs:element>`

XML Schema

- Link tussen XML-document en XML Schema
- XML Schema
 - Structuur
 - Beschrijving van elementen
 - Types

Types

■ Twee soorten

- Eenvoudige → simpleType
 - Inhoud bestaat alleen uit tekst of waarden
- Complexe → complexType
 - Inhoud bestaat niet alleen uit tekst en/of heeft attributen
 - Een complex type kan een rij van elementen bevatten (in volgorde!) → sequence

Eenvoudig type

- Bestaand type
- Beperking van een bestaand type (restriction)
 - Getallen: interval, aantal cijfers, aantal cijfers na komma
 - Strings: patroon (reguliere uitdrukkingen – regular expressions), lengte, aantal waarden
- Unie van eenvoudige types (union)
- Lijst $\approx$ array (list)

Voorbeeld van een eenvoudig type

```
<xs:simpleType name="zips">  
  <xs:restriction base="xs:string">  
    <xs:pattern value="75230" />  
    <xs:pattern value="75233" />  
    <xs:pattern value="75235" />  
  </xs:restriction>  
</xs:simpleType>
```

Voorbeeld van een eenvoudig type

```
<xsd:simpleType name="diplomatype">  
  <xsd:restriction base="xsd:string">  
    <xsd:pattern value="HOBU | UNIV | MO | HOKT" />  
  </xsd:restriction>  
</xsd:simpleType>
```

Voorbeeld van een eenvoudig type

```
<xsd:simpleType name="initiaalttype">  
  <xsd:restriction base="xsd:string">  
    <xsd:pattern value="[a-z] | [A-Z]\."/>  
  </xsd:restriction>  
</xsd:simpleType>
```

Voorbeeld van een eenvoudig type

```
<xsd:simpleType name="postcodetype">  
  <xsd:restriction base="xsd:string">  
    <xsd:pattern value="[1-9][0-9]{3}" />  
  </xsd:restriction>  
</xsd:simpleType>
```


Complex type

- Rij (sequence) → volgorde belangrijk
- Keuze (choice)
- Uitbreidingen van andere complexe types
 - `complexContent` → extension
- Beperkingen van andere complexe types
 - `complexContent` → restriction
- Uitbreiding van eenvoudige types met attributen
 - `simpleContent` → extension
- Opmerking ivm attributen
 - Attributen worden laatst gedeclareerd

Voorbeeld rij en attributen

```
<xsd:element name="meting">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="plaats"/>
      <xsd:element ref="tijdstip"/>
      <xsd:element ref="hemelcoördinaat"/>
    </xsd:sequence>
    <xsd:attribute name="toestel" type="xsd:IDREF"/>
    <xsd:attribute name="persoon" type="xsd:IDREF"/>
    <xsd:attribute name="hemellichaam" type="xsd:IDREF"/>
  </xsd:complexType>
</xsd:element>
```

Voorbeeld uitbreiding

```
<xs:complexType name="personName">
  <xs:sequence>
    <xs:element ref="title"
      minOccurs="0"/>
    <xs:element ref="forename"
      minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element ref="surname"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="extendedName">
  <xs:complexContent>
    <xs:extension base="personName">
      <xs:sequence>
        <xs:element ref="generation"
          minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Elementdeclaraties in types

■ Element

- Naam (name)
- Type (type)
- Verwijzing naar declaratie (ref)
- Minimum aantal (minOccurs)
- Maximaal aantal (maxOccurs)

Attribuutdeclaraties in types

- Attribuut
 - Naam (name)
 - Type (type)
 - Gebruik (use)
 - optional (standaard)
 - required
 - ...

Structureren en datatypes

■ DTD

- Structuur: welke elementen en attributen?
- Beperkte gegevenstypes
- Oorsprong in SGML (Standard Generalized Markup Language)

■ XML Schema

- Recenter
- Uitgebreidere beschrijving structuur en datatypes mogelijk
- XML syntax

Informatie

- DTD <http://www.w3.org/TR/2004/REC-xml-20040204/#NT-doctypedecl>
- XML Schema <http://www.w3.org/TR/xmlschema-0/>
- Reguliere uitdrukkingen <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/datatypes.html#regexs>