04 XML Schema XML Schema

CSCI235/MCS9235/CSCI835 Databases, SCIT, Autumn 2015

04 XML Schema

XML Schema? What is it?

XML Schema is a formal notation for defining a schema for a class of XML documents

XML Schema is a vocabulary for expressing data dependencies

XML Schema document models are designed to define the usage and relationships of various schema components, such as the following:

Datatypes

Elements and their content

Attributes and their values

Reusable components and their content

Notations

© Janusz R. Getta

CSCI235/MCS9235/CSCI835 Databases, SCIT, Autumn 2015

```
04 XML Schema
                                           Example
Is the following data valid?
 <location>
                 <latitude>32.904237</latitude>
<longitude>73.620290</longitude>
<uncertainty units="meters">2</uncertainty>
 </location>
Constraints:
(1) the location must be comprised of a latitude, followed by a longitude
followed by an indication of the uncertainty of the lat/lon
```

- measurements,
 (2) the latitude must be a decimal with a value between -90 to +90,
- the longitude must be a decimal with a value between -180 to +180, for both latitude and longitude the number of digits to the right of the
- decimal point must be exactly six digits,
- (5) the value of uncertainty must be a non-negative integer,
 (6) the uncertainty units must be either meters or feet.

© Janusz R. Getta CSCI235/MCS9235/CSCI835 Databases, SCIT, Autumn 2015

04 XML Schema

XML Schema: an overview

Properties:

Enhanced data types (44+)

Type definition mechanisms

The same syntax as XML documents

Object-orientation (inheritance)

Create type constructor

Ability to specify uniqueness (keys or content) constraints Ability to define the multiple elements with the same name but different content

Ability to define the elements with nil contents

Ability to define substitutable elements, e.g. the "Book" element is substitutable for the "Publication" element.

© Janusz R. Getta CSCI235/MCS9235/CSCI835 Databases, SCIT, Autumn 2015

04 XML Schema

© Janusz R. Getta

DTD versus XML Schema

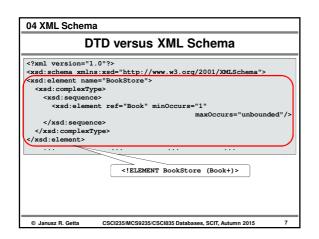
```
<?xml version="1.0"?>
<?xml version="1.0"?>
<!DOCTYPE BookStore [
<!ELEMENT BookStore (Book+)>
<!ELEMENT Book (Title, Author, Date, ISBN, Publisher)>
<!ELEMENT Title (#PCDATA)>
<!ELEMENT Author (#PCDATA)>
<!ELEMENT Date (#PCDATA)>
<!ELEMENT ISBN (#PCDATA)>
<!ELEMENT ISBN (#PCDATA)>
 <!ELEMENT Publisher (#PCDATA) > 1>
<BookStore>
       <Title>Java my way !</Title>
<Author>Janusz R. Getta</Author>
       <Date>05-APR-2005
<ISBN>09-345673</ISBN>
       <Publisher>Addison Wesley Ltd. Pty.</Publisher>
</BookStore>
 © Janusz R. Getta
                            CSCI235/MCS9235/CSCI835 Databases, SCIT, Autumn 2015
```

04 XML Schema

DTD versus XML Schema

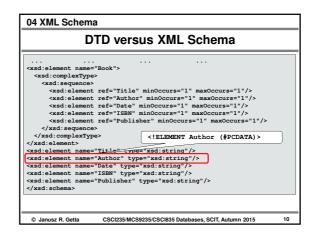
```
<?xml version="1.0"?>
   NML velock:
lookStore
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="bookstore.xsd">
   <Book>
      <Title>Java my way !</Title>
<Author>Janusz R. Getta</Author>
<Date>05-APR-2005</Date>
      <ISBN>09-345673</ISBN>
      <Publisher>Addison Wesley Ltd. Pty.</Publisher>
</BookStore>
```

© Janusz R. Getta CSCI235/MCS9235/CSCI835 Databases, SCIT, Autumn 2015



```
DTD versus XML Schema

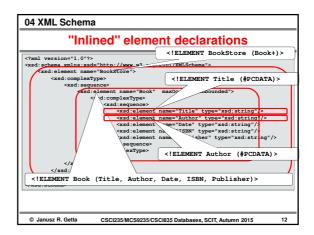
| Xsd:element name="Book">
| < xsd:complexType>
| < xsd:element ref="Title" minOccurs="1" maxOccurs="1"/>
| < xsd:element ref="Date" minOccurs="1" maxOccurs="1"/>
| < xsd:element ref="TSBN" minOccurs="1" maxOccurs="1"/>
| < xsd:element ref="TSBN" minOccurs="1" maxOccurs="1"/>
| < xsd:element ref="FBND minOccurs="1" maxOccurs="1"/>
| < xsd:element ref="FBND minOccurs="1" maxOccurs="1"/>
| < xsd:element ref="SBN" minOccurs="1" maxOccurs="1"/>
| < xsd:element name="Author" type="xsd:string"/>
| < xsd:element name="Author" type="xsd:string"/>
| < xsd:element name="Date" type="xsd:string"/>
| < xsd:element name="Publisher" type="xsd:string"/>
| < xsd:element name="Publisher" type="xsd:string"/>
| < xsd:element name="Date" type="xsd:string"/>
| < xsd:element name="Date"
```

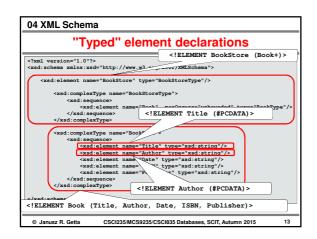


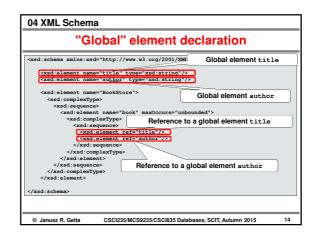
```
XML Schema styles

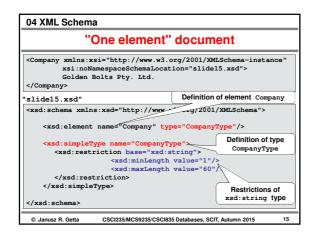
XML Schema can be created using one of the following 3 styles:
Inlined element declarations:
Element declaration precisely follow an order of elements in a document. No new types are defined
Typed element declarations:
A hierarchy of types is defined and the elements are declared using new types
Global element declarations:
Hypothetical global elements are declared and referenced in either Inlined or Typed declarations:

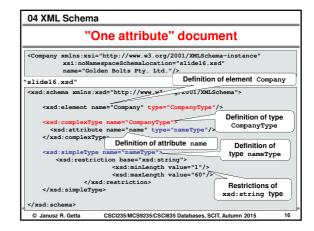
© Janusz R. Getta CSCI235MCS9225/CSCI835 Databases, SCIT, Autumn 2015 11
```

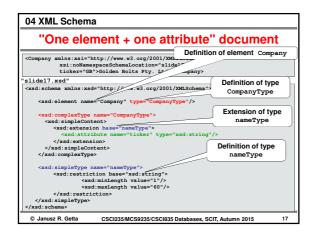








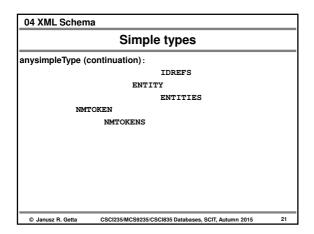


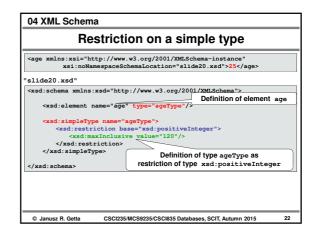


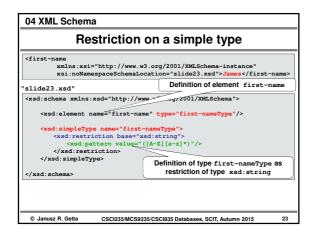
```
04 XML Schema
                     Simple types
anvsimpleType:
   duration
   dateTime
   time
   date
   gYearMonth
   gYear
   qMonth
   gDay
   gMonth
   boolean
   base64binary
   hexbinary
                CSCI235/MCS9235/CSCI835 Databases, SCIT, Autumn 2015
```

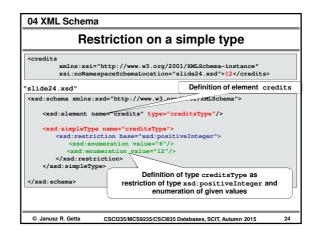
```
04 XML Schema
                     Simple types
anysimpleType:
   float
   double
   decimal
           integer
                  {\tt nonPositiveInteger}
                        negativeInteger
                  nonNegativeInteger
                        unsignedLong
                              unsignedInt
                                    unsignedShort
                                          unsignedByte
                        positiveInteger
 © Janusz R. Getta
                CSCI235/MCS9235/CSCI835 Databases, SCIT, Autumn 2015
```

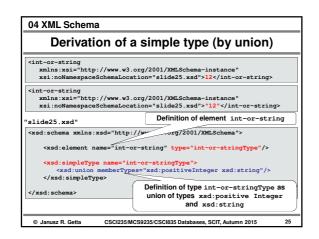
```
04 XML Schema
                      Simple types
anysimpleType (continuation):
                  long
                        int
                               short
string
   normalizedString
     token
            language
            NAME
                  NCName
                        ID
                         IDREF
 © Janusz R. Getta
                CSCI235/MCS9235/CSCI835 Databases, SCIT, Autumn 2015
```

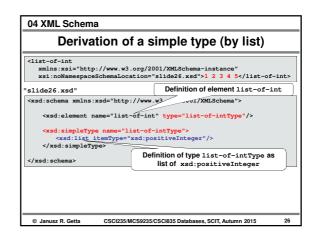


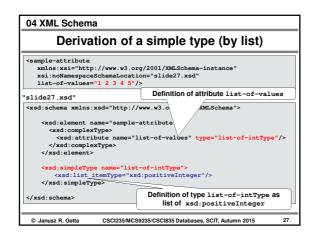


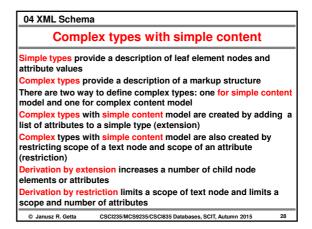


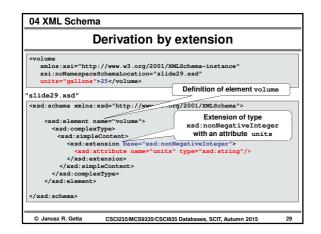


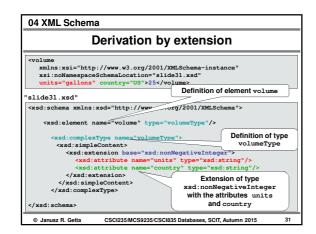


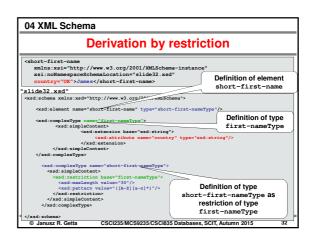












04 XML Schema References Elmasri R., Navathe S., Fundamentals of Database Systems, 6th edition, chapter 12 XML: Extensible Markup Language, pp. 420-448 http://www.uow.edu.au/~jrg/235/HOMEWORK 8.4 How to create XML Schemas and how to validate XML documents against XML Schemas ? (1) http://www.uow.edu.au/~jrg/235/HOMEWORK 8.5 How to create XML Schemas and how to validate XML documents against XML Schemas ? (2) http://www.uow.edu.au/~jrg/235/SLIDES/XML/ Appendix-2-XMLSchema-1-Reference.pdf http://www.uow.edu.au/~jrg/235/SLIDES/XML/ Appendix-3-XMLSchema-2-Reference.pdf © Janusz R. Getta CSCI235/MCS9235/CSCI835 Databases, SCIT, Autumn 2015