XML Schema

IT 4153 Advanced Database

J.G. Zheng

Spring 2012

XML Schema

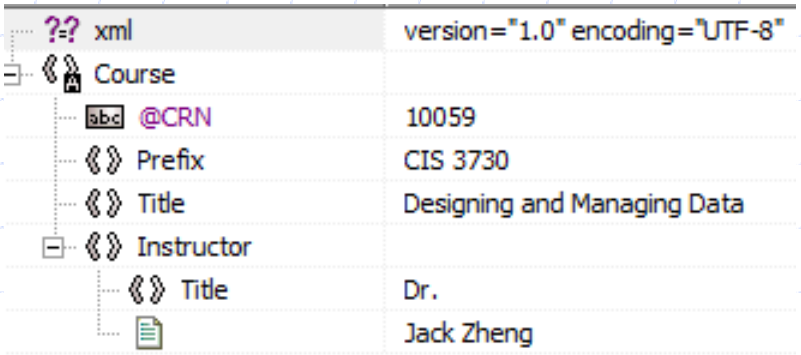
- XML Schema is a standard for describing the structure of an XML document.
  - Documents that conform to an XML Schema are called schema-valid.
  - It's much like the meta-data in the database, where tables, columns, keys, and other constraints are defined – the difference is metadata in database are usually enforced, whereas in XML documents, schema-invalid XML document can still exist.
- An XML schema file usually ends in ".XSD"
- Schema defines elements and attributes that should be in an XML file:
  - Their names, structures, occurrences, and other constraints

XML Elements Defined in Schemas

- There are two type of elements
- *Simple type* elements have a single data value (text node), and do not have attributes or other child elements.
- *Complex type* elements can have (by order)
  - Zero to more child elements (either simple type or complex type): these elements are arranged in a "sequence" or other structure.
  - Zero to more attributes.
  - Text content (use "mixed" attribute)

Example: A Sample XML Document

```
<?xml version="1.0" encoding="UTF-8" ?>
<Course CRN="10059">
   <Prefix>CIS 3730</Prefix>
   <Title>Designing and Managing Data</Title>
   <Instructor><Title>Dr.</Title>
   Jack Zheng
   </Instructor>
</Course>
```

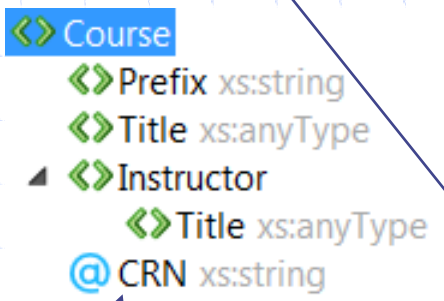


Example: XML Schema

Every schema starts with the schema root element "xs:schema". "xs" is the namespace defined in the same line.

XML declaration: XML schema is also an XML file.

This defines the root element of the XML document. Note that all elements should have starting tag (<tagname>) and closing tag</tagname>.



A tree view of the schema

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Course">
  <xs:complexType>
      <xs:sequence>
      <xs:element name="Prefix" type="xs:string"/>
      <xs:element name="Title"/>
      <xs:element name="Instructor">
      <xs:complexType mixed="true">
          <xs:sequence minOccurs="0" maxOccurs="1">
          <xs:element name="Title"/>
          </xs:sequence>
      </xs:complexType>
      </xs:element>
      </xs:sequence>
      <xs:attribute name="CRN" type="xs:string"/>
  </xs:complexType>
  </xs:element>
</xs:schema>
```

Example: Complex Type

"Course" root element is a complex type. It consists of a sequence of child elements, and an attribute.

"sequence" means the elements in it should be in this order.

Simple type elements, which does not have child elements or attributes (can have text content)

The "mixed" attribute means text and elements are mixed.

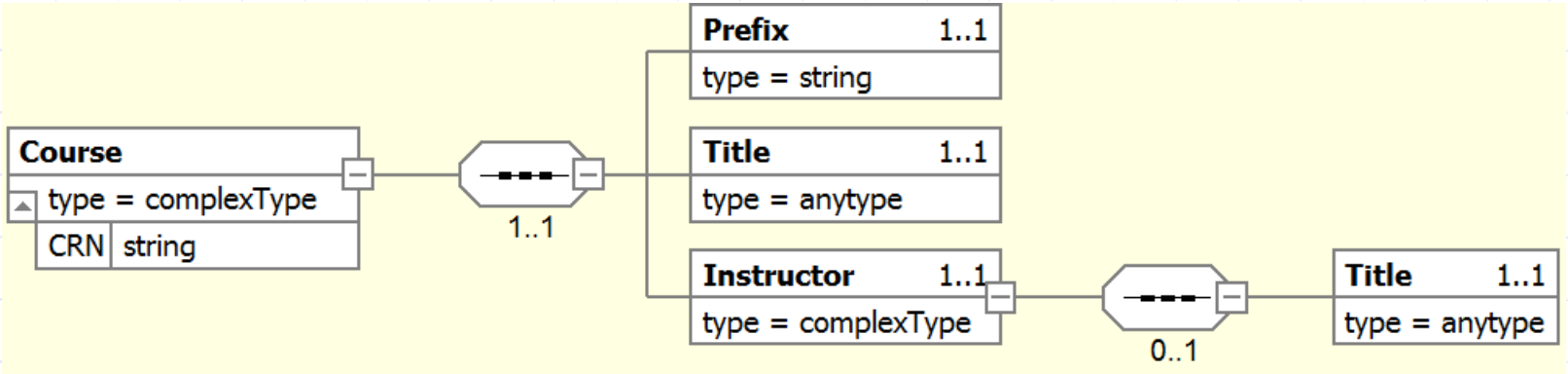
Note that the attribute is part of a complex type, and must be defined after the definition of "sequence" other elements.

Notice how elements and attributes are in a hierarchical structure.

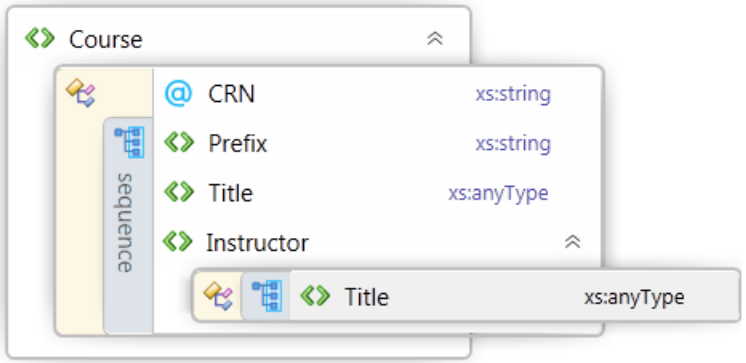
```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Course">
  <xs:complexType>
      <xs:sequence>
      <xs:element name="Prefix" type="xs:string"/>
      <xs:element name="Title"/>
      <xs:element name="Instructor">
        <xs:complexType mixed="true">
          <xs:sequence minOccurs="0" maxOccurs="1">
           <xs:element name="Title"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      </xs:sequence>
      <xs:attribute name="CRN" type="xs:string"/>
  </xs:complexType>
  </xs:element>
</xs:schema>
```

Other Graphic Views

- In XMLPad 3.0



- In Visual Studio 2010



Validating XML Files and XML Schemas

- Documents that conform to an XML Schema are called schema-valid.
  - Some web sites to validate XML files against schemas
    - http://www.xmlvalidation.com/
    - http://tools.decisionsoft.com/schemaValidate/
- XML schema documents are themselves XML documents
  - It can be validated against a meta schema maintained by W3C
  - Validate any schema using this service
    - http://www.w3.org/2001/03/webdata/xsv

Relational Data and XML

- XML Document vs. XML Data
  - Document centric XML file
    - Focus on content
    - Fewer tags, less structured
  - Data centric XML file
    - Focus on data and structure
    - More tags, more structured
- Relational data (table) can be transformed to XML format (data centric XML file)
  - Write an XML schema
  - Transform relational data to XML format which conforms to the schema

A Simple Relation-to-XML Example

- Transforming a single table
  - The "Shippers" table in the "Northwind" database.
- General guideline
  - The table becomes the root element (a complex type): may use the table name as the root element name.
  - Each row (record) becomes direct child elements (complex types) under the root element.
  - Each value in the row becomes (two choices)
    - an attribute of the row element (the column name becomes the attribute name, and the data becomes the attribute value), or
    - an third level child element (simple type) under the row element: the column name becomes the element name and the data becomes the text node under the element.

Example: Shippers Table

- Table structure

	Column Name	Data Type	Allow Nulls
🔑	ShipperID	int	☐
	CompanyName	nvarchar(40)	☐
	Phone	nvarchar(24)	☑

- Data

	ShipperID	CompanyName	Phone
1	1	Speedy Express	(503) 555-9831
2	2	United Package	(503) 555-3199
3	3	Federal Shipping	(503) 555-9931

XML Schema based on Attributes

```
<?xml version="1.0" encoding="utf-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<xsd:element name="Shippers">
<xsd:complexType>
    <xsd:sequence>
    <xsd:element name="Shipper" maxOccurs="unbounded">
    <xsd:complexType>
        <xsd:attribute name="ShipperID" />
        <xsd:attribute name="CompanyName" />
        <xsd:attribute name="Phone" />
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

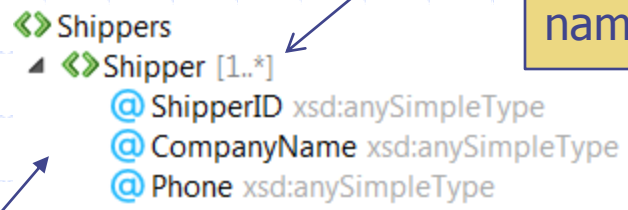
"xsd" is the namespace, a standard declaration.

The root element has the table name as the element name. It is a complex type.

Each row (record) becomes a direct child element under the root element. It is also a complex type.

"unbounded" means there can be multiple "Shipper" elements.

The column name becomes the attribute name.

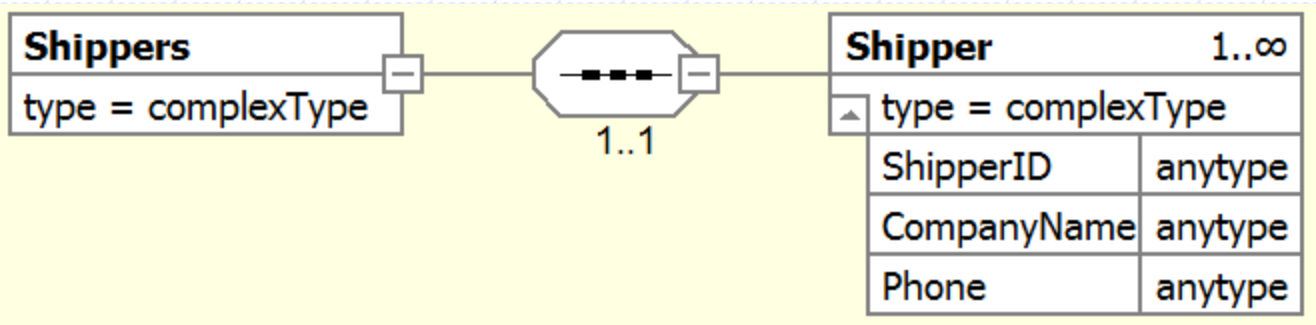


A tree view of the table/XML schema.

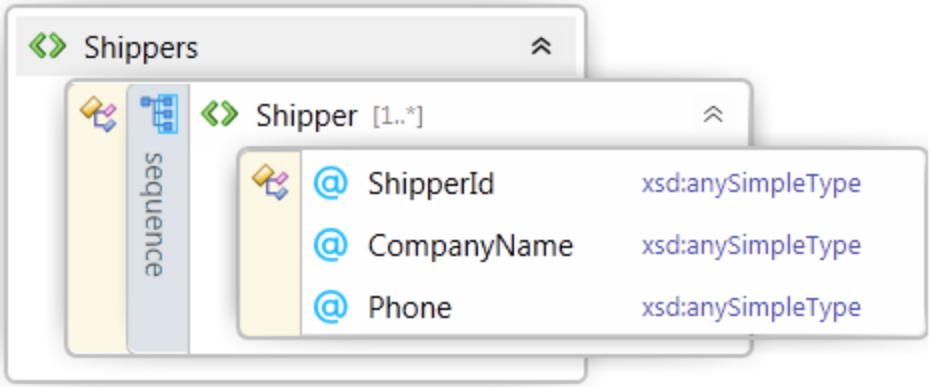
"@" means attribute.

Other Graphic Views

- In XMLPad 3.0



- In Visual Studio 2010



Corresponding XML Data File

```
<?xml version="1.0" encoding="utf-8"?>
<Shippers>
  <Shipper ShipperID="1" CompanyName="Speedy Express"
   Phone="(503)555-9831" />
  <Shipper ShipperID="2" CompanyName="United Package"
   Phone="(503)555-3199" />
  <Shipper ShipperID="3" CompanyName="Federal Shipping"
   Phone="(503)555-9931" />
</Shippers>
```

The root element has the table name as the element name. It is a complex type.

Each row (record) becomes a direct child element under the root element. There are 3 records hence 3 "Shipper" elements.

Values of the row (record) become attributes of the row element: the column name becomes the attribute name. The data becomes the attribute value.

XML Schema based on Elements

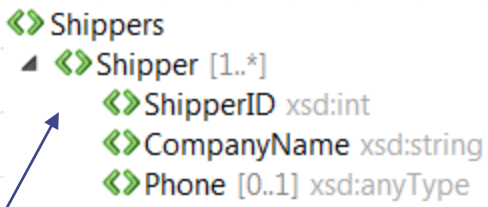
```
<?xml version="1.0" encoding="utf-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <xsd:element name="Shippers">
    <xsd:complexType>
       <xsd:sequence>
       <xsd:element name="Shipper" maxOccurs="unbounded">
       <xsd:complexType>
             <xsd:sequence>
             <xsd:element name="ShipperID" minOccurs="1" type="xsd:int"/>
             <xsd:element name="CompanyName" type="xsd:string"/>
             <xsd:element name="Phone" minOccurs="0"/>
             </xsd:sequence>
       </xsd:complexType>
       </xsd:element>
       </xsd:sequence>
    </xsd:complexType>
    </xsd:element>
</xsd:schema>
```

The root element has the table name as the element name.

The root element is a complex type with a sequence of other elements

Each row (record) becomes a direct child element under the root element. It is a complex type.

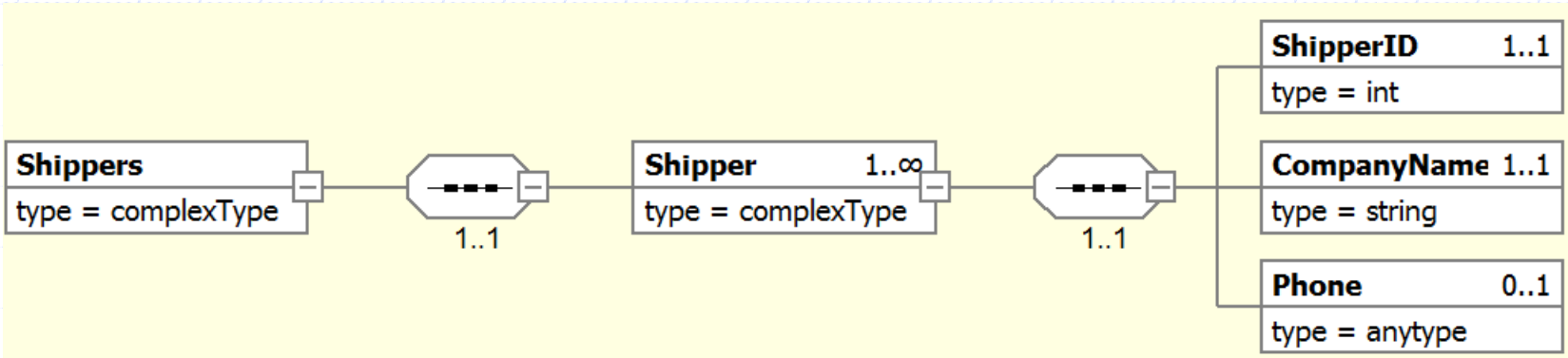
Third level child elements under the row element: the column name becomes the element name. All are simple type elements arranged in a "sequence" with corresponding data types.



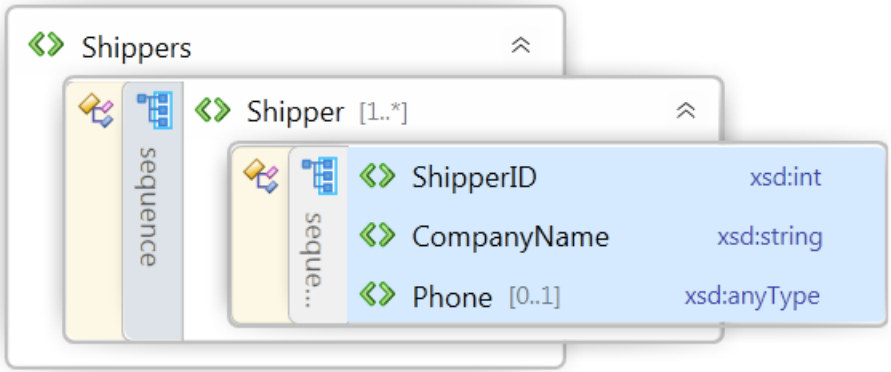
A tree view of the table/XML schema.

Other Graphic Views

- In XMLPad 3.0



- In Visual Studio 2010



Corresponding XML Data File

```
<?xml version="1.0" encoding="utf-8" ?>
<Shippers>
    <Shipper>
        <ShipperID>1</ShipperID>
        <CompanyName>Speedy Express</CompanyName>
        <Phone>(503) 555-9831</Phone>
    </Shipper>
    <Shipper>
        <ShipperID>2</ShipperID>
        <CompanyName>United Package</CompanyName>
        <Phone>(503) 555-3199</Phone>
    </Shipper>
    <Shipper>
        <ShipperID>3</ShipperID>
        <CompanyName>Federal Shipping</CompanyName>
        <Phone>(503) 555-9931</Phone>
    </Shipper>
</Shippers>
```

The root element has the table name as the element name.

Each row (record) becomes a direct child element under the root element. There are three occurrences.

Third level child elements under the row element: the column name becomes the element name; the data becomes the text node.

Summary

- Key concepts
  - XML schema
  - Validation
- Key skills
  - Create XML schema for simple data
  - Create XML files based on schemas and validate the file against the schema