XML

Authored by : Asfiya Khan Presented by : Asfiya Khan

Cybage®
Delivering Value Scientifically

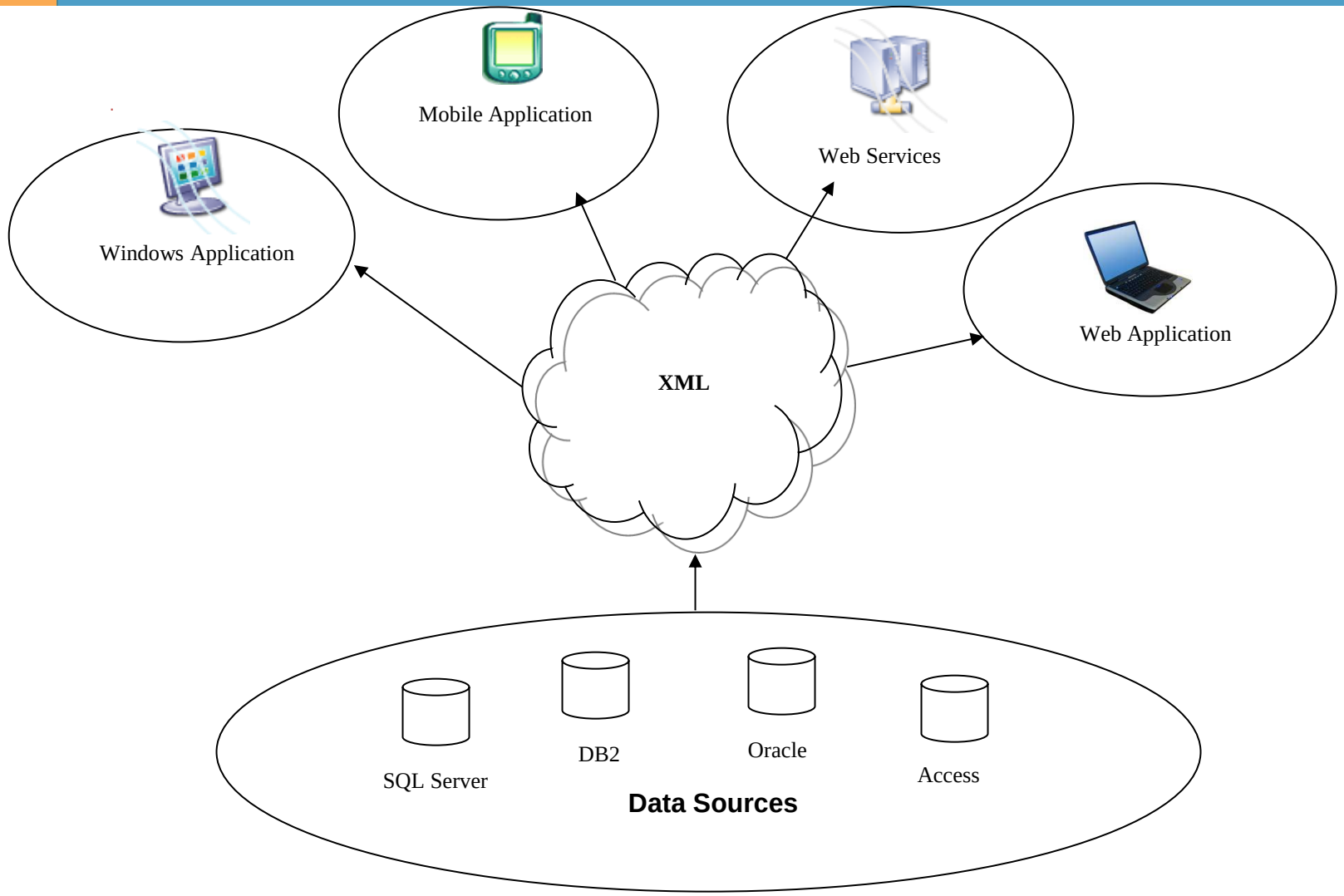
Agenda

- XML Introduction
- Advantages
- Components of XML
- Rules of XML Document
- Difference between SGML and HTML
- XML schema
- Attributes in an XML schema
- XML namespaces

XML Introduction

- Text-based markup language enables storage of data in a structured format.
- Cross-platform.
- Hardware and software independent markup language.
- Enables structured data transfer between heterogeneous systems.
- Common data interchange format in a number of applications.

Introducing XML (Contd.)



Advantages of XML

- Domain-specific vocabulary
- Data interchange
- Smart searches
- Granular updates
- User-selected view of data
- Message transformation

Components of XML Document

- Processing Instruction (PI) <?xml version="1.0" encoding="UTF-8"?>
- Tags
- Elements <STOREDATA> </STOREDATA>
- Content
- Attributes
- Entities
- Comments

Cybage®
Delivering Value Scientifically

Sample Code

```
<?xml version="1.0" encoding="UTF-8"?>   ⬅ Processing Instruction (PI)

<STOREDATA>   ⬅ Root Element

<!--STOREDATA is the root element-->   ⬅ Comments

<STORE STOREID="S101">   ⬅ Child Elements   Attributes
        <PRODUCTNAME>Toys</PRODUCTNAME>   ⬅ Content
        <QUANTITY>100</QUANTITY>
        <DISPLAY>The price of this toy is &lt; 200 </DISPLAY>   ⬅ Entities
</STORE>

</STOREDATA>
```

Rules of XML Document

- Every start tag must have an end tag.
- Empty tags must be closed using a forward slash (/).
- All attribute values must be given in double quotation marks.
- Tags must have proper nesting.
- XML tags are case sensitive.

Difference Between SGML, HTML, and XML

- **Standard Generalized Markup Language** (SGML) allows documents to describe their grammar
- **Hypertext Markup Language** (HTML) is used for data presentation.
- **Extensible Markup Language** is used for data description and definition.

Difference Between SGML, HTML, and XML

Difference Between SGML, HTML, and XML (Contd.)

	SGML	HTML	XML
Extensibility	Yes	Yes	Yes
Structure	Yes	No	Yes
Validation	Yes	No	Yes
Browser Dependency	Low	High	Low
Cost/Benefit	Poor	Good	Medium

XML Schema

- Defines the list of elements and attributes to be used in an XML.
- Specifies the order and datatype of elements.
- Microsoft has developed the **XML Schema Definition (XSD)** language to define the schema.

Data Types in XML Schemas(XSD)

- **Primitive**
- **Derived**
- **Atomic**
- **List**
- **Union**

Cybage®
Delivering Value Scientifically

Elements in XML Schemas

There are two types of element in XSD Simple and Complex.

Simple Element:

```
<xsd:element name= "element-name"
  type="datatype"
  minOccurs="nonNegativeInteger"
  maxOccurs="nonNegativeInteger
  |unbounded"/>
```

The name attribute specifies the name of the element declared.

The type attribute specifies the data type of the element declared.

minOccurs specifies the minimum number of times the element can occur.

maxOccurs specifies the maximum number of times the element can appear

Elements in XML Schemas

- Complex Element:

```
<xsd:complexType name="datatype”>
  Content model declaration
  </xsd:complexType>
```

The complexType element is used to declare a new complex data type.

The name attribute specifies the name of the new complex data type.

The Content model declaration contains the declaration for the elements and attributes that make up the content of the complex type.

XML Attributes

- Attributes in an XML schema are declared in the same way as elements.

- Attribute declarations can be defined in two ways:
  - Simple type definitions: Facilitates local validation of the attribute information.
  - Global attribute declarations: Enables reuse of attributes.

The syntax for declaring an attribute in XSD is:

<attribute name="attributename" ref="attribname“ type="datatypename" use="value" value="value">
</attribute>

- The attribute element consists of the following attributes:
  - name
  - ref
  - type
  - use

XML Namespaces

- Its a virtual space that is assigned or recognized by a Uniform Resource Identifier (URI).
- A namespace is a string that uniquely identifies the elements and attributes from different schemas.
- A namespace is a unique identifier used to resolve conflicts between elements that have the same names.
- The following guidelines ensure the uniqueness of a URI:
  - Using a URI that is controlled by the developer.
  - Using a relative URI.

XML Namespaces

- A namespace can be declared in an XSD document by using the xmlns keyword.

- The general form of the xmlns keyword is:
- xmlns:prefix="URI"

- There are two types of namespace declarations:
  - **Default Declaration**: Declares a default namespace for a document without specifying the prefix for a default namespace.
  - **Explicit Declaration**: Enables xmlns keyword to associate a prefix with a namespace.

Any Questions?

Thank you!