STRUCTURING WITH SCHEMAS

A Simple XML Document:

We'll talk about a shirt. There's actually a lot we can talk about with regard to a shirt: size, color, fabric, price, brand, and condition, among other properties. The Following example shows one possi- ble XML rendition of a document describing a shirt. Of course, there are many other possible ways to describe a shirt, but this example provides a foundation for our further discussions.

```
<?xml version="1.0"?>

<shirt>
    <model>Zippy Tee</model>
    <brand>Tommy Hilbunger</brand>
    <price currency="USD">14.99</price>
    <on_sale/>
    <fabric content="60%">cotton</fabric>
    <fabric content="40%">polyester</fabric>
    <options>
    <colorOptions>
    <color>red</color>
    <color>white</color>
    </colorOptions>
    <sizeOptions>
    <size>Medium</size>
    <size>Large</size>
    </sizeOptions>
    </options>
    <description> This is a <b>funky</b> Tee shirt similar to the Floppy Tee shirt </description>
</shirt>
```

XML Declaration:

The XML declaration is a processing instruction of the form <?xml ...?>. Although it is not required, the presence of the declaration explicitly identifies the document as an XML document and indicates the version of XML to which it was authored. In addition, the XML declaration indicates the presence of external markup declarations and character encoding. Because a number of document formats use markup similar to XML, the declaration is useful in establishing the document as being compliant with a specific version of XML without any doubt or ambiguity. In general, every XML document should use an XML declaration. As documents increase in size and complexity, this importance likewise grows.

Components of the XML Declaration:

Component	*Description*
<?xml	Startsthebeginningoftheprocessinginstruction(inthis case,fortheXMLdeclaration).
Version="xxx"	DescribesthespecificversionofXMLbeingusedinthe document(inthiscase,version1.0oftheW3Cspecifica-

	tion).Futureiterationscouldbe2.0,1.1,andsoon.
standalone=”xxx”	Thisstandaloneoptiondefineswhetherdocumentsare allowedtocontainexternalmarkupdeclarations.This optioncanbesetto“yes”or“no”.
encoding=”xxx”	Indicatesthecharacterencodingthatthedocumentuses. Thedefaultis“US-ASCII”butcanbesettoanyvaluethat XMLprocessorsrecognizeandcansupport.Themostcommonalternat e

ValidXMLDeclarations

```
<?xmlversion=”1.0”standalone=”yes”?>
<?xmlversion=”1.0”standalone=”no”?>
<?xmlversion=”1.0”encoding=”UTF-8”standalone=”no”?>
```

Document Type Declaration

A Document Type Declaration names the document type and identifies the internal con- tent by specifying the root element, in essence the first XML tag that the XML-process- ing tools will encounter in the document. A DOCTYPE can identify the constraints on the validity of the document by making a reference to an external DTD subset and/or include the DTD internally within the document by means of an internal DTD subset.

GeneralFormsoftheDocumentTypeDeclarations:

```
<!DOCTYPENAMESYSTEM‘‘file’’>
<!DOCTYPENAME[]>
<!DOCTYPENAMESYSTEM‘‘file’’[]>
```

ComponentsoftheDocumentTypeDeclaration

Component	*Description*
<	ThestartoftheXMLtag(inthiscase,thebeginningofthe DocumentTypeDeclaration).

Component	*Description*
!DOCTYPE	ThebeginningoftheDocumentTypeDeclaration.
NAME	Specifiesthenameofthedocumenttypebeingdefined. ThismustcomplywithXMLnamingrules.
SYSTEM	Specifiesthatthefollowingsystemidentifierwillberead andprocessed.
‘‘file’’	Specifiesthenameofthefiletobeprocessedbythesys- tem.
[	StartsaninternalDTDsubset.
]	EndstheinternalDTDsubset.
>	TheendoftheXMLtag(inthiscase,theendofthe

DocumentTypedeclaration).

MarkupandContent:

In general, six kinds of markup can occur in an XML document: elements, entity references, comments, processing instructions,marked sections, and Document Type Declarations.

XML BASED STANDARDS:

1) XPATH

XPath is a syntax for defining parts of an XML document. XPath uses path expressions to navigate in XML documents. XPath contains a library of standard functions. XPath is a major element in XSLT. XPath is a W3C Standard

2) XSD

It defines elements that can appear in a document. defines attributes that can appear in a document. It defines which elements are child elements. defines the order of child elements. It defines the number of child elements. It defines whether an element is empty or can include text. It defines data types for elements and attributes. It defines default and fixed values for elements and attributes

3) XSL

XSL describes how the XML document should be displayed! XSL consists of three parts:XSLT - a language for transforming XML documents, XPath - a language for navigating in XML documents, XSL-FO - a language for formatting XML documents

4) XSLT

A common way to describe the transformation process is to say that XSLT transforms an XML source-tree into an XML result-tree.XSLT stands for XSL Transformations. XSLT is the most important part of XSL. XSLT transforms an XML document into another XML document. XSLT uses XPath to navigate in XML documents. XSLT is a W3C Recommendation.

Source : http://nprcet.org/e%20content/Misc/e-Learning/IT/VIII%20Sem/IT1451%20-%20XML%20and%20Web%20Services.pdf