UTF-X Doc

Unit Testing Framework - XSLT Documentation System

Jacek Radajewski

0.1.0
April 2007

Table of Contents

1 Introduction

- 1.1 Wat is UTF-X?
- 1.2 What is UTF-X Doc?

2 Formatting XML

3 WSDL Example

4 UTF-X Document Structure

- 4.1 Document metadata
- 4.2 Namespace context
- 4.3 Nested Sections
- 4.4 Block Elements

1 Introduction

XML source available here

1.1 Wat is UTF-X?

UTF-X is an extension to the JUnit Java unit testing framework and provides functionality for unit testing XSLT stylesheets. UTF-X strongly supports the test-first-design principle with test rendition and test validation features allowing you to visually design your test before you start working on the on stylesheet. UTF-X was originally built to test XSLT stylesheet used in an XML publishing system so it has good support for DTD validation, XHTML and XSL:FO stylesheets.

1.2 What is UTF-X Doc?

UTF-X Doc is a simple application for documenting XML systems and apps. UTF-X Doc documents are authored in XML and then rendered into desired formats such as HTML or PDF. UTF-X Doc was originally written to demonstrate how UTF-X tests are applied to real XML applications and to document the UTF-X system itself. This document for example has been created using UTF-X Doc.

2 Formatting XML

UTF-X Doc allows you format XML fragments like this:

```
<u:sect1 id="formattingxml1">
    <u:title>Formatting XML</u:title>
    <u:para>UTF-X Doc allows you format XML fragments like this:</u:para>
    <u:figure>
        <u:xml>
            <more>
                <xml />
            </more>
        </u:xml>
        <u:caption>XML Fragment</u:caption>
    </u:figure>
</u:sect1>
```

Figure 2.1: XML Fragment

3 WSDL Example

The following example is a WSDL document with `wsdl:binding` highlighted by XPath `.//wsdl:binding`. Each prefix is highlighted in it's own colour.

```
  <wsdl:definitions xmlns:tns="http://example.com/stockquote.wsdl"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd1="http://example.com/stockquote.xsd"
```

```
http://schemas.xmlsoap.org/wsdl/soap/" name="StockQuote"
targetNamespace="http://example.com/stockquote.wsdl">
    <wsdl:types>
      <xsd:schema targetNamespace="http://example.com/stockquote.xsd">
        <xsd:element name="TradePriceRequest">
          <xsd:complexType>
            <xsd:all>
              <xsd:element name="tickerSymbol" type="xsd:string" />
            </xsd:all>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="TradePriceResponse">
          <xsd:complexType>
            <xsd:choice>
              <xsd:element name="price" type="xsd:float" />
              <xsd:element name="invalidTickerSymbol" type="xsd:string" />
            </xsd:choice>
          </xsd:complexType>
        </xsd:element>
      </xsd:schema>
    </wsdl:types>
    <wsdl:message name="GetLastTradePriceInput">
      <wsdl:part element="xsd1:TradePriceRequest" name="body" />
    </wsdl:message>
    <wsdl:message name="GetLastTradePriceOutput">
      <wsdl:part element="xsd1:TradePriceResponse" name="body" />
    </wsdl:message>
    <wsdl:portType name="StockQuotePortType">
      <wsdl:operation name="GetLastTradePrice">
        <wsdl:input message="tns:GetLastTradePriceInput" />
        <wsdl:output message="tns:GetLastTradePriceOutput" />
      </wsdl:operation>
    </wsdl:portType>
    <wsdl:binding name="StockQuoteSoapBinding" type="tns:StockQuotePortType">
      <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http" />
      <wsdl:operation name="GetLastTradePrice">
        <soap:operation soapAction="http://example.com/GetLastTradePrice" />
        <wsdl:input>
          <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
    </wsdl:binding>
  </wsdl:definitions>
```

4 UTF-X Document Structure

4.1 Document metadata

```
<u:document
xsi:schemaLocation="http://utf-x.sourceforge.net/xsd/utfxdoc_1_0/utfxdoc.xsd
http://utf-x.sourceforge.net/xsd/utfxdoc_1_0/utfxdoc.xsd">
  <u:title>UTF-X Doc</u:title>
  <u:subtitle>Unit Testing Framework - XSLT Documentation System</u:subtitle>
  <u:author>Jacek Radajewski</u:author>
  <u:version>1.0.0</u:version>
  <u:date>February 2006</u:date>
</u:document>
```

Figure 4.1: Document metadata

4.2 Namespace context

In order for all the funky features (e.g. element highliting) to work you will need to define a namespace context which bind prefixes to corresponding namespace URIs.

```
  <u:namespace-context>
    <u:namespace>
      <u:prefix>u</u:prefix>
      <u:uri>http://utf-x.sourceforge.net/xsd/utfxdoc_1_0/utfxdoc.xsd</u:uri>
    </u:namespace>
    <u:namespace>
      <u:prefix>xsd</u:prefix>
      <u:uri>http://www.w3.org/2001/XMLSchema</u:uri>
    </u:namespace>
    <u:namespace>
      <u:prefix>xsi</u:prefix>
      <u:uri>http://www.w3.org/1999/XMLSchema-instance</u:uri>
    </u:namespace>
    <u:namespace>
      <u:prefix>wsdl</u:prefix>
      <u:uri>http://schemas.xmlsoap.org/wsdl/</u:uri>
    </u:namespace>
    <u:namespace>
      <u:prefix>soap</u:prefix>
      <u:uri>http://schemas.xmlsoap.org/wsdl/soap/</u:uri>
    </u:namespace>
    <u:namespace>
      <u:prefix>soap-env</u:prefix>
      <u:uri>http://schemas.xmlsoap.org/soap/envelope/</u:uri>
    </u:namespace>
  </u:namespace-context>
```

Figure 4.2: Defining namespace context. Highlighted is the W3C Schema namespace to prefix binding.

4.3 Nested Sections

```
<!--~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~-->
<!-- nested sections -->
<!--~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~-->
<u:sect1 id="s1">
    <u:title>Level 1</u:title>
    <u:para>Paragraph in level 1 section</u:para>
    <u:sect2 id="s2">
        <u:title>Level 2</u:title>
        <u:para>Some more content in level 2 section.</u:para>
        <u:sect3 id="s3">
            <u:title>Level 3</u:title>
            <u:para>And the last, level 3 paragraph.</u:para>
        </u:sect3>
    </u:sect2>
</u:sect1>
```

Figure 4.3: Three levels of nesting available

4.4 Block Elements

```
<xsd:group name="block">
    <xsd:sequence>
        <xsd:choice maxOccurs="unbounded" minOccurs="0">
            <xsd:element ref="u:para" />
            <xsd:element ref="u:note" />
            <xsd:element ref="u:figure" />
            <xsd:element ref="u:pre" />
            <xsd:element ref="u:xml" />
            <xsd:element ref="u:ordered_list" />
            <xsd:element ref="u:unordered_list" />
        </xsd:choice>
    </xsd:sequence>
</xsd:group>
```

Figure 4.4: Block elements