

## XML Tools and Technologies



**Submitted To:** Program Manager  
GeoConnections  
Victoria, BC, Canada

**Submitted By:** Jody Garnett  
Brent Owens  
Refractions Research Inc.  
Suite 400, 1207 Douglas Street  
Victoria, BC, V8W-2E7  
jgarnett@refractions.net  
Phone: (250) 885-0632  
Fax: (250) 383-2140

# TABLE OF CONTENTS

---

<b>XML TOOLS AND TECHNOLOGIES.....</b>	<b>1</b>
<b>TABLE OF CONTENTS.....</b>	<b>2</b>
<b>1 INTRODUCTION.....</b>	<b>3</b>
<b>2 XML PROCESSING.....</b>	<b>4</b>
2.1 XML PROCESSING STANDARDS.....	4
<b>3 REVIEW OF EXISTING TOOLS.....</b>	<b>5</b>
3.1 XML PROCESSING TOOLS .....	5
3.2 DTD VALIDATION .....	5
3.3 XML SCHEMA VALIDATION .....	5
3.4 XML SPY.....	6
3.5 OPENGIS LITERATURE.....	7
3.6 OPEN SOURCE EXAMPLES.....	7
<b>4 REFERENCES.....</b>	<b>8</b>
4.1 OPEN GIS CONSORTIUM, INC. ....	8
4.2 THE GEOSERVER PROJECT .....	8
4.3 XML TOOLS.....	8
4.3.1 <i>The Apache XML Project</i> .....	8
4.3.2 <i>Other</i> .....	8
<b>APPENDIX A: EXAMPLE XML SCHEMA FILE.....</b>	<b>9</b>

# 1 INTRODUCTION

---

This review of XML tools is focused on the requirements for creating a Validating Web Feature Server. This product will be tightly integrated with the existing GeoServer project.

The Validating Web Feature Server will require XML processing for:

- Configuration of geo-spatial validation plug-ins
- Web Feature Server Update requests
- Web Feature Server Update response
- Validation of XML Schema

The XML Processing tools, currently in use by the GeoServer project, meet these requirements and no change is recommended.

Spatial constraint expression is featured in the OpenGIS Web Feature Server specification. Constraints are expressed in XML and open source parsing code is available as part of the GeoServer and Geotool2 codebase.

## 2 XML PROCESSING

---

There are several common approaches to parsing XML documents.

- **Push Model:**  
XML parser is configured with a series of callbacks, defined as functions or events, which are invoked as the parser encounters elements of the XML document. Although efficient the difficulty of tracking state is placed on the application programmer.
- **Tree Model:**  
An in memory tree based data structure is produced from the XML document, an API is provided to allow navigation between nodes of the tree. Keeping XML documents in memory is prohibitive given the volume of data in a geospatial application.
- **Pull Model:**  
An effective compromise between the previous two approaches. A pull-based parser is queried for the next response, allowing the application programmer the use of stack to track parser state.

A variety of query and transformation languages exist that make use of, or define, hierarchical based operations.

In addition there have been several attempts to provide direct data mapping of XML documents to Java Objects.

### 2.1 XML Processing Standards

XML processing tools are based on the following standards:

- Document Object Model (DOM) - W3C standard for tree based XML parsing
- Simple API for XML (SAX) - de-facto standard for push based XML parsing
- XSL Transformations (XSLT) - W3C standard for XML to XML transformations
- Data Type Definition - defines an XML documents validity using a text file
- XMLSchema - defines an XML documents validity using an XML file

Many smaller standards are used to fill out the XML landscape:

- XQuery - a query language similar in spirit to SQL
- XPath - used to query hierarchical data structure
- XPointer - used to provided links between XML documents

In addition Java XML processing tools reference the following:

- Java API for XML Processing (JAXP) - allows the use of different XML processing implementations to be used without source code changes.

## 3 REVIEW OF EXISTING TOOLS

---

This review is limited to open source tools, given the level of cooperation required with the GeoServer and geotools2 projects.

### 3.1 XML Processing Tools

XML Processing Tools used by the GeoServer project:

- Xerces
  - Validating XML parser
  - SAX support (used by GeoServer)
  - DOM support
  - XML Schema support
  - JAXP support
- Xalan
  - XSLT engine
  - XPath support
- JDOM
  - Java interface to SAX or DOM based parsers

Other popular XML Processing Tools:

- Saxon – popular XSLT engine

Java XML Data Binding Tools:

- Castor – open source, although lacks complete schema support
- JAXB – Java Community Process based, provides schema support

### 3.2 DTD Validation

The WFS Specification mandates the use of XML Schema as a minimum for all operations. Other output formats for GetFeature, such as DTD, are possible if specified by DescribeFeatureType and GetCapabilities.

DTD Validation Tools:

- XSDValid – contains dtdvalid DTD validation tool

The GeoServer project does not currently uses DTD schema definitions.

### 3.3 XML Schema Validation

GeoServer services make use of the Geographic Markup Language (GML) as an XML based transportation encoding. GML is validated against a series of XML Schemas customized with application schema definitions.

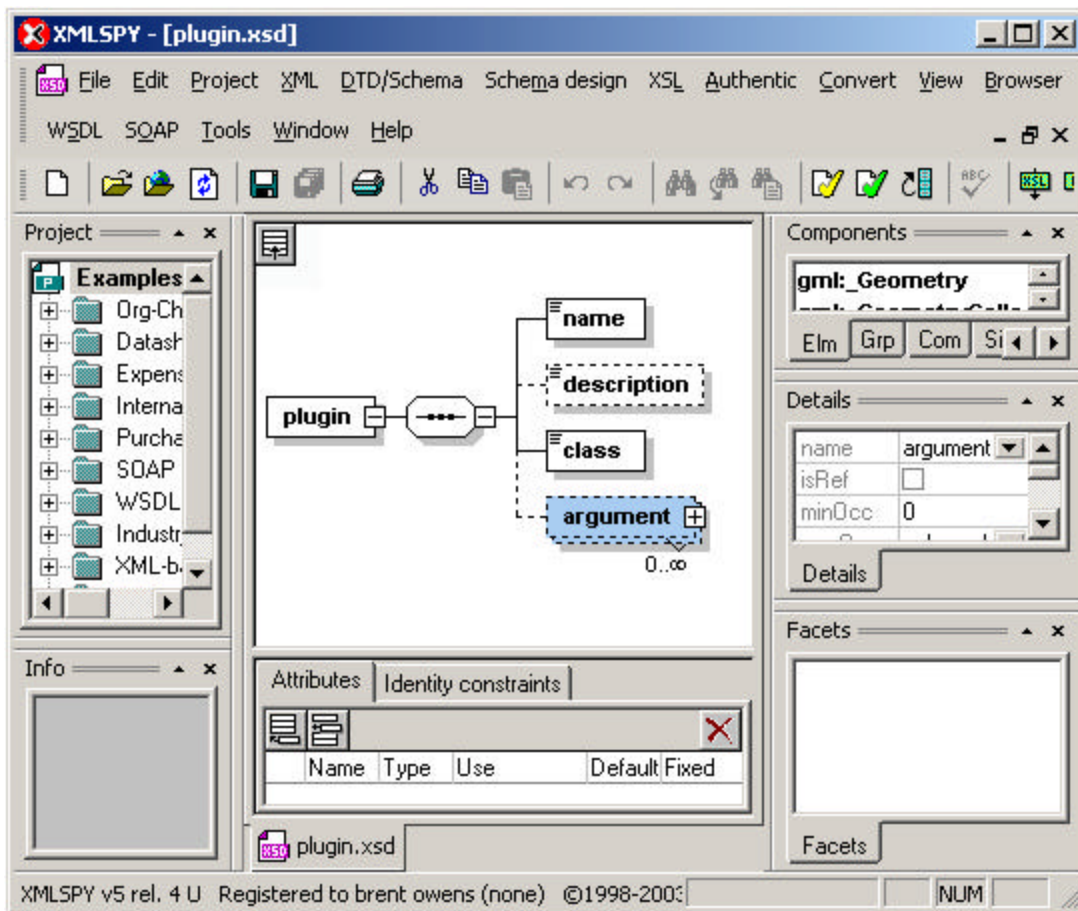
The following tools were found for XML Schema validation:

- XSV – web based user interface
- SQC – IBM XML Schema Quality Checker

- XSDValid – contains csdvalid XML Schema validation tool
  - Decisionsoft XML validation - <http://tools.decisionsoft.com/schemaValidate.html>
  - GML4J – open source GML Schema Parser
- SQC in particular is well suited to QA integration.

### 3.4 XML SPY

XML Spy is a wonderful tool produced by Altova. It is an XML development environment for designing and editing XML, XML Schema, XSL/XSLT, SOAP, WSDL and Web Service technologies. It has proved to be very useful in our development of XML and XML schemas.



### 3.5 OpenGIS Literature

OpenGIS describes in detail how to specify spatial constraints for the WFS GetFeature Operation.

Spatial Constraints are specified as part of a Filter on a Query operation, however a short cut exists for filtering based on a Bounding Box:

```
<QuerytypeName="myns:HYDROGRAPHY">
  <ogc:PropertyName>myns:GEOTEMP</ogc:PropertyName>
  <ogc:PropertyName>myns:DEPTH</ogc:PropertyName>
  <ogc:Filter><ogc:Not><ogc:Disjoint>
    <ogc:PropertyName>myns:GEOTEMP</ogc:PropertyName>
    <gml:Box>
      <gml:coordinates>-57.9118,46.2023-46.6873,51.8145</gml:coordinates>
    </gml:Box>
  </ogc:Disjoint></ogc:Not></ogc:Filter>
</Query>
```

The Filter Encoding Implementation Specification defines spatial operations using the following DTD fragment:

```
<!ENTITY % spatial_ops "( Equals |
                          Disjoint |
                          Intersects |
                          Touches |
                          Crosses |
                          Within |
                          Contains |
                          Overlaps |
                          Beyond |
                          BBOX) ">
<!ELEMENT Equals (PropertyName,%GeometryClasses;)>
<!ELEMENT Disjoint (PropertyName,%GeometryClasses;)>
<!ELEMENT Intersects (PropertyName,%GeometryClasses;)>
<!ELEMENT Touches (PropertyName,%GeometryClasses;)>
<!ELEMENT Crosses (PropertyName,%GeometryClasses;)>
<!ELEMENT Within (PropertyName,%GeometryClasses;,Distance)>
<!ELEMENT Contains (PropertyName,%GeometryClasses;)>
<!ELEMENT Overlaps (PropertyName,%GeometryClasses;)>
<!ELEMENT Beyond (PropertyName,%GeometryClasses;,Distance)>
<!ELEMENT BBOX (PropertyName,gml:Box)>
<!ELEMENT Distance (#PCDATA)>
<!ATTLIST Distance units CDATA #REQUIRED>
```

### 3.6 Open Source Examples

The current GeoServer 0.96 implementation relies on geotools2 for parsing spatial constraints.

## 4 REFERENCES

---

### 4.1 *Open GIS Consortium, Inc.*

An international industry consortium developing publicly available geoprocessing specifications

Open GIS Consortium	<a href="http://www.opengis.org">Www.opengis.org</a>
WFS Specification	<a href="http://www.opengis.org/techno/specs/02-058.pdf">Www.opengis.org/techno/specs/02-058.pdf</a>
Filter Encoding Implementation Specification	<a href="http://www.opengis.org/techno/RFC13a.pdf">Www.opengis.org/techno/RFC13a.pdf</a>

### 4.2 *The GeoServer Project*

A Java implementation of the Web Feature Server specification:

GeoServer	<a href="http://Geoserver.sourceforge.net">Geoserver.sourceforge.net</a>
-----------	--

### 4.3 *XML Tools*

#### 4.3.1 *The Apache XML Project*

XML Schema and DTD Validation:

The Apache XML Project	<a href="http://Xml.apache.org">Xml.apache.org</a>
Xerces	<a href="http://Xml.apache.org/xerces2-j">Xml.apache.org/xerces2-j</a>
Xalan	<a href="http://Xml.apache.org/xalan-j">Xml.apache.org/xalan-j</a>

#### 4.3.2 *Other*

XML Spy	<a href="http://www.xmlspy.com">www.xmlspy.com</a>
JAXP	<a href="http://java.sun.com/xml/jaxp">java.sun.com/xml/jaxp</a>
Crimson	<a href="http://gump.cocoondev.org/xml-crimson.html">gump.cocoondev.org/xml-crimson.html</a>
JDOM	
KXML	

#### **XML Schema Validation Resources**

The W3C XML Schema web site (<http://www.w3.org/XML/Schema>) has a list of validation tools.



## APPENDIX A: EXAMPLE XML SCHEMA FILE

### GeoServer featureType.xsd (XML schema file)

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:complexType name="LatLonBoundingBoxType">
    <xs:attribute name="minx"
      type="xs:string" use="required"/>
    <xs:attribute name="miny"
      type="xs:string" use="required"/>
    <xs:attribute name="maxx"
      type="xs:string" use="required"/>
    <xs:attribute name="maxy"
      type="xs:string" use="required"/>
  </xs:complexType>
  <xs:complexType name="MetadataURLType">
    <xs:sequence>
      <xs:element name="url" type="xs:string"/>
      <xs:element name="type" type="xs:string"/>
      <xs:element name="format" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="featureTypeType">
    <xs:sequence>
      <xs:element name="Name" type="xs:string"/>
      <xs:element name="Title" type="xs:string"/>
      <xs:element name="Abstract" type="xs:string"/>
      <xs:element name="Keywords" type="xs:string"/>
      <xs:element name="SRS" type="xs:string"/>
      <xs:element name="Operations" type="xs:string"/>
      <xs:element name="LatLonBoundingBox"
        type="LatLonBoundingBoxType"/>
      <xs:element name="MetadataURL"
        type="MetadataURLType"/>
      <xs:element name="Host" type="xs:string"/>
      <xs:element name="Port" type="xs:string"/>
      <xs:element name="DatabaseName"
        type="xs:string"/>
      <xs:element name="User" type="xs:string"/>
      <xs:element name="Password" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="featureType"
    type="featureTypeType"/>
</xs:schema>
```