

# Using XML for Legends and Map Surround

By Victor Dohar

Natural Resources Canada  
601 Booth Street  
Ottawa, Ontario, Canada K1A 0E8  
Telephone: (613) 943-2693  
Fax: (613) 952-7308  
e-mail: vdohar@NRCan.gc.ca

## WHAT IS XML?

XML is an acronym for **E**xtensible **M**arkup **L**anguage. Basically, it is a readable text file used to store information in a structured manner. Just as HTML (Hypertext Markup Language) was designed to display data on web pages, XML was designed to store data. It is important to note however, that an XML document by itself does not do anything. It cannot be executed, or perform any function. It is simply a means of storing information and passing it from application to application. Thus, it is widely accepted as a means to allow for the exchange of data between incompatible systems.

The structure and syntax rules of an XML document are fairly straightforward. The information conveyed in an XML document must be enclosed between standard markups, or more commonly known as tags or nodes. The result is a start and end tag with a value in between, forming an element. The start tag can also include element attributes, which are used to describe the value between

the tags. The use of tags is important as they allow a computer application (or human) to quickly locate a piece of information, much like a directory structure on a hard disk. Unlike HTML where tags are predefined, XML tags are defined and named by the user or the application that creates the XML document. The syntax rules are not very complicated. Listed below are a few to help you understand the basic rules of an XML document:

1. All XML documents must contain a declaration and one unique root element,
2. All elements must have matching start and end tags,
3. Tag names are case sensitive,
4. All elements must be properly nested,
5. Element attribute values must always be double quoted.

An XML document is considered to be well-formed when none of these syntax rules are broken.

The following is a sample XML document, displaying one root element `<Paper>` containing three additional elements with some information about this paper. For legibility reasons in this paper, values between the tags are displayed in bold, and nested tags are indented.

```
<Paper>
  <Title>Using XML for Legends and Map Surround</Title>
  <Author>Vic Dohar</Author>
  <Organization>Natural Resources Canada</Organization>
</Paper>
```

The following is a similar XML document with more information:

```
<Conference>
  <Name>DMT '05</Name>
  <Papers>
    <Paper>
      <Title>Geologic quadrangle mapping at the ISGS</Title>
      <Author>
        <Surname>Domier</Surname>
        <GivenName>Jane</GivenName>
```

```

    </Author>
    <Organization>Illinois State Geological Survey</Organization>
</Paper>
<Paper>
  <Title>Using XML for Legends and Map Surround</Title>
  <Author>
    <Surname>Dohar</Surname>
    <GivenName>Vic</GivenName>
  </Author>
  <Organization>Natural Resources Canada</Organization>
</Paper>
</Papers>
</Conference>

```

The two examples above contain the same type of information, yet some information is stored differently. This variance in structure is driven and controlled by an XML Schema. An XML schema is used to define the structure or elements that exist in an XML document. They are the legal building blocks of an XML document as defined by the originator. Schemas define each element, the data type for each element, each element's attributes, the number of occurrences of an element, whether or not an element is optional or mandatory, its child elements, and the order of elements, just to list a few. XML schemas are also written as an XML document, but are saved with the .xsd file extension, thus they are at times referred to as XSD documents. At the top of an XML document, a reference is usually made to a schema in order to validate the content and structure of the XML document.

The diagram in Figure 1 is a graphic representation of a schema for the above XML document, produced using the software XMLSpy by Altova (<http://www.altova.com>). This software allows schemas to be created graphically, much like UML (Unified Modeling Language) diagrams. The diagram basically states (from left to right) that the root element is called Conference, and it must contain elements called Name and Papers. Name contains a text string representing the name of the conference, and Papers must contain any number of Paper elements. Each Paper element must contain a Title, an Author, and an Organiza-

tion element. Finally, each Author element must contain a GivenName and Surname element, along with an optional MiddleInitial element.

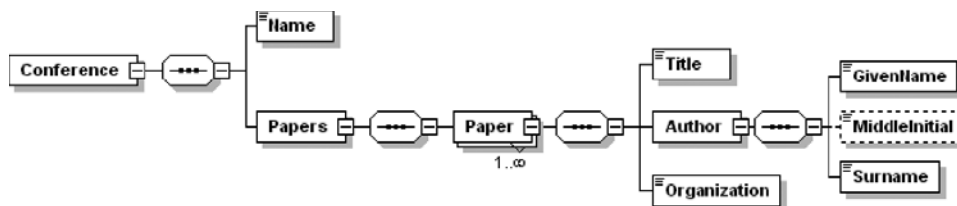
## XML Resources

The above should provide a basic level of understanding when discussing the use of XML for map surround and legend creation. There are many resources available for you to get a better understanding of XML. Two that I use often when creating applications utilizing XML are W3 Schools (<http://www.w3schools.com/xml/default.asp>) and the Microsoft Development Network (<http://msdn.microsoft.com/xml/>). In addition to learning XML, you will also need software to manage, view, and edit XML documents in human-friendly form. Some are free like Peter's XML Editor (<http://www.iol.ie/~pxe/>) with limited capabilities, whereas others such as Altova's XMLSpy charge a fee and have many bells and whistles.

## APPLYING XML FOR MAKING MAPS

### Using XML for Map Surround Elements

The Publication Process and Integration (PPI) is an electronic web-based system to manage each Geological Survey of Canada (GSC) publication through its vari-



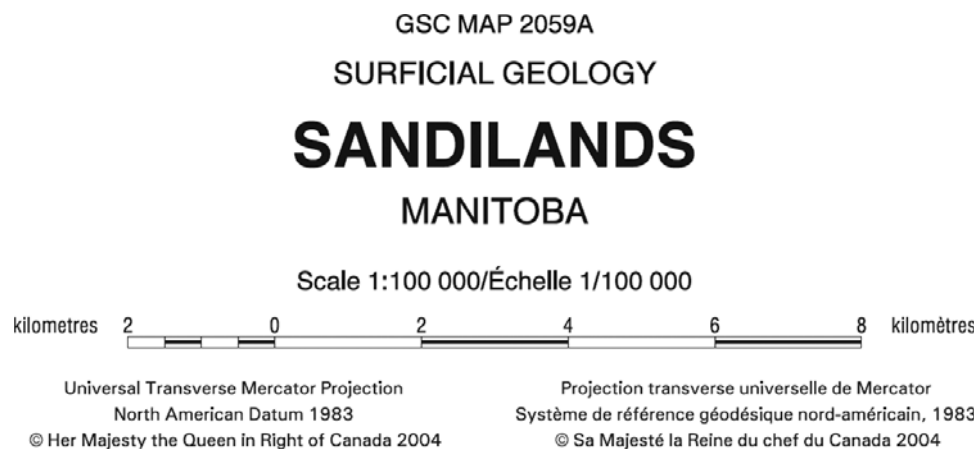
**Figure 1.** A graphic representation of an XML schema based on the sample XML document produced using XMLSpy software (Altova, Inc.). It clearly displays the relationships between the elements, the order of elements, and the element type. Each element can be dragged and edited in order to create schema variations.

ous stages. The system replaces with web-based forms the many paper submission forms that were required of authors in order to publish reports, open files, bulletins, and maps. The information entered in these web forms is stored in an Oracle database, where it can be extracted to

an XML document. Some of the information that is entered is metadata which can be used for generating various map surround elements such as title block and recommended citation.

The following sample XML document generated from Oracle is then used in an ArcMap VBA (Visual Basic for Applications) application to display in ArcMap the title block shown below (see display in Figure 2).

```
<PublicationInformation>
  <Authors>
    <Author>
      <Surname>Smith</Surname>
      <Initial>L</Initial>
    </Author>
  </Authors>
  <Language>english</Language>
  <Bilingual>no</Bilingual>
  <Publication>
    <Series>A-series map</Series>
    <Number>2059</Number>
    <Title>Sandilands</Title>
  </Publication>
  <Map>
    <Feature>surficial geology</Feature>
    <Coverage>
      <District></District>
      <Province>Manitoba</Province>
    </Coverage>
    <ScaleDenominator>100 000</ScaleDenominator>
  </Map>
</PublicationInformation>
```



**Figure 2.** Image of title block from a geological map. The content was extracted from an Oracle database as an XML document. A VBA script in ArcMap generates the title block along with a second XML document that stores the GSC design specifications.

In addition to the above XML document containing the information for the title block, another XML document stored on a central server is used for storing the GSC Design Specifications or the rendering of these elements in ArcMap. This XML document is used to store the

properties of these elements; such as font name, font size, colour, justification, indentation, and line spacing. Should a change in design be required, only the values in this XML document need to be updated, without the need to modify the VBA script.

Shown below is an excerpt from the GSC Design Specifications XML document for the map title element of the title block. The same XML schema exists for other surround elements.

```
<GSCDesignSpecifications>
  <TitleBlock>
    <MapTitle>
      <Font>
        <Name>Arial</Name>
        <Style>Regular</Style>
      </Font>
      <Size units="points">24</Size>
      <Colour>
        <Cyan>0</Cyan>
        <Magenta>0</Magenta>
        <Yellow>0</Yellow>
        <Black>100</Black>
      </Colour>
      <LeadingFactor>1.25</LeadingFactor>
      <HorizontalAlignment>HaCenter</HorizontalAlignment>
      <VerticalAlignment>VaBaseline</VerticalAlignment>
      <LineSpacings>
        <LineSpacing>
          <FromElement>default</FromElement>
          <Distance units="points">32</Distance>
        </LineSpacing>
      </LineSpacings>
      <LineLimit units="picas">36</LineLimit>
      <Indent units="picas">0</Indent>
    </MapTitle>
  </TitleBlock>
</GSCDesignSpecifications>
```

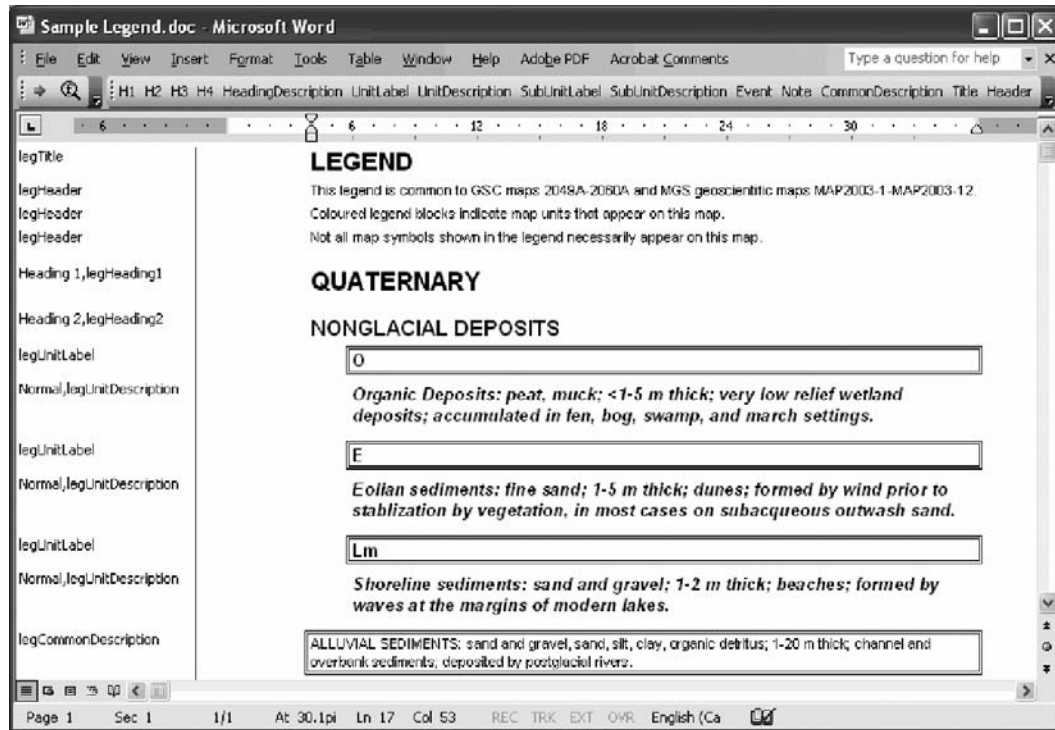
The use of these XML documents and the VBA application in ArcMap provides an efficient means of adding this information to maps thereby ensuring quality and consistency in all the maps published at the GSC. The key benefits are that this approach reduces errors and omissions by reducing the need for user intervention, and provides consistent rendering of the information based on established design specifications.

## Using XML for Geological Legends

A similar approach utilizing XML documents is used for rendering geological legends in ArcMap. In most instances, the text of a geological legend is initially created by the author/geologist as a Microsoft Word docu-

ment. By utilizing the styles and formatting capabilities of paragraphs in Microsoft Word, custom formatting styles are created and applied to each paragraph. The custom formatting styles reflect the content of a geological legend (i.e., geological unit description) as well as resembling the geological legend XML schema.

VBA scripting and a toolbar in Microsoft Word allow the user with the click of a mouse to apply the desired custom formatting style to each paragraph. Paragraphs are then formatted visually according to the settings of each style; however it is only meant as a visual aid and has no bearing on the final appearance of the legend in ArcMap (see Figure 3). The important aspect is that each paragraph is formatted correctly. Based on the formatting style applied to each paragraph, a VBA script in Microsoft



**Figure 3.** Screenshot of Microsoft Word document, displaying a sample geological legend (shown on right side) that has its paragraphs formatted to custom styles (shown on left side). Also shown is the toolbar for applying a custom formatting style to each paragraph. Based on the formatting, the content of each paragraph is written to an XML document accordingly.

Word transfers the content in each paragraph to an XML document, placing the content within the corresponding element tags (see XML document below that has been translated from the Word document in Figure 3). The

XML document in turn is validated against the legend content schema XSD document (see Figure 4) before being processed in ArcMap.

```
<LegendContent>
  <LegendTitle>
    <Title legID="1">LEGEND</Title>
    <Header legID="2">This legend is common to GSC maps 2049A - 2060A, and MGS
      geoscientific maps MAP2003-1 - MAP2003-12.</Header>
    <Header legID="3">Coloured legend blocks indicate map units that appear on
      this map.</Header>
    <Header legID="4">Not all map symbols shown in the legend necessarily ap-
      pear on this map.</Header>
  </LegendTitle>
  <UnitLegend>
    <Heading>
      <HeadingLabel legID="5" level="1">QUATERNARY</HeadingLabel>
    </Heading>
  </UnitLegend>
  <UnitLegend>
    <Heading>
      <HeadingLabel legID="6" level="2">NONGLACIAL DEPOSITS</HeadingLabel>
    </Heading>
  </UnitLegend>
```

```

<UnitLegend>
  <Units>
    <Unit boxID="1">
      <UnitLabel legID="7">O</UnitLabel>
      <UnitDescription legID="8">Organic deposits: peat, muck; &lt;1-5 m
        thick; very low relief wetland deposits; accumulated in fen, bog,
        swamp, and marsh settings.</UnitDescription>
    </Unit>
  </Units>
</UnitLegend>
<UnitLegend>
  <Units>
    <Unit boxID="2">
      <UnitLabel legID="9">E</UnitLabel>
      <UnitDescription legID="10">Eolian sediments: fine sand; 1-5 m thick;
        dunes; formed by wind prior to stabilization by vegetation, in most
        cases on subaqueous outwash sand.</UnitDescription>
    </Unit>
  </Units>
</UnitLegend>
<UnitLegend>
  <Units>
    <Unit boxID="3">
      <UnitLabel legID="11">Lm</UnitLabel>
      <UnitDescription legID="12">Shoreline sediments: sand and gravel; 1-2
        m thick; beaches; formed by waves at the margins of modern lakes.</
        UnitDescription>
    </Unit>
  </Units>
</UnitLegend>
<UnitLegend>
  <CommonDescription legID="13">ALLUVIAL SEDIMENTS: sand and gravel, sand,
    silt, clay, organic detritus; 1-20 m thick; channel and overbank
    sediments; deposited by postglacial rivers.</CommonDescription>
</UnitLegend>
</LegendContent>

```

In ArcMap, a VBA script is used to generate the geological legend (see Figure 5) using three XML documents. The content of the legend is extracted from the XML document generated from Microsoft Word (described above). The rendering or design specifications of the legend (i.e., fonts, colours, legend box sizes, line spacing) is obtained from the GSC Design Specifications XML document noted above. A third XML document is used to control the layout of the legend on the paper. This is used primarily for the legend's location on the paper, number of columns, and aligning geological units chronologically in multiple columns. In addition, when the VBA script generates the legend, the symbology used for each of the geological units in ArcMap is transferred to the legend.

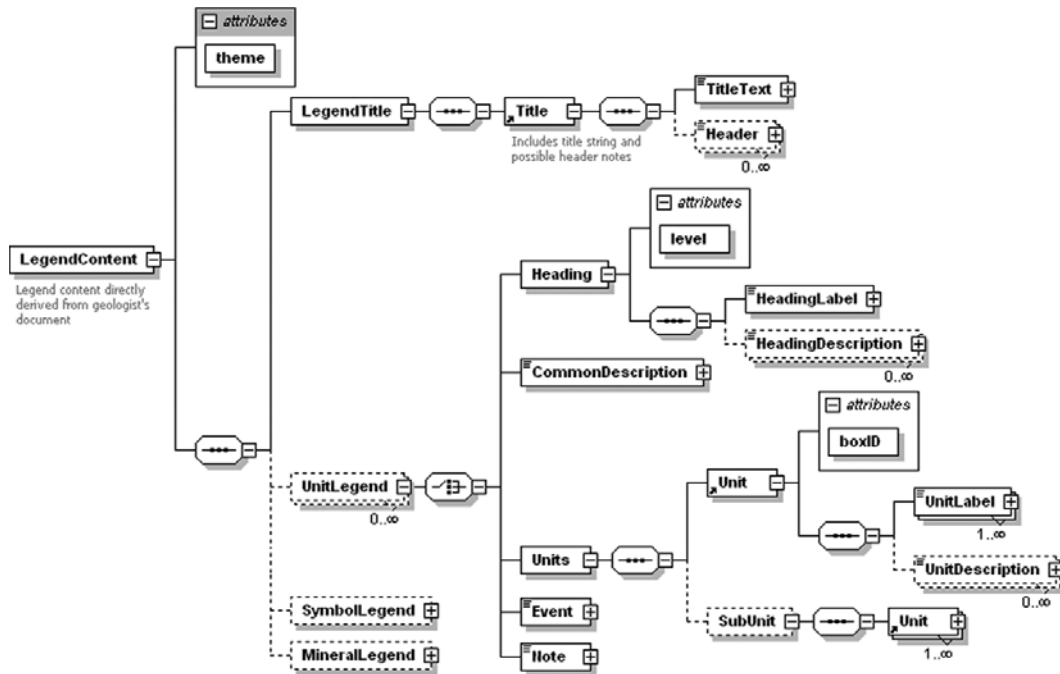
It is important to note that the legend created by this method is not dynamically linked to the ArcMap table of contents (TOC). If any edits are required to the legend, ei-

ther to the content in the Word document, or the symbology of a geological unit in ArcMap, the simplest task is to delete the current legend from ArcMap and regenerate the legend with the updated XML documents and ArcMap symbology. This method utilizing three XML documents ensures a consistent level of quality and output from ArcMap.

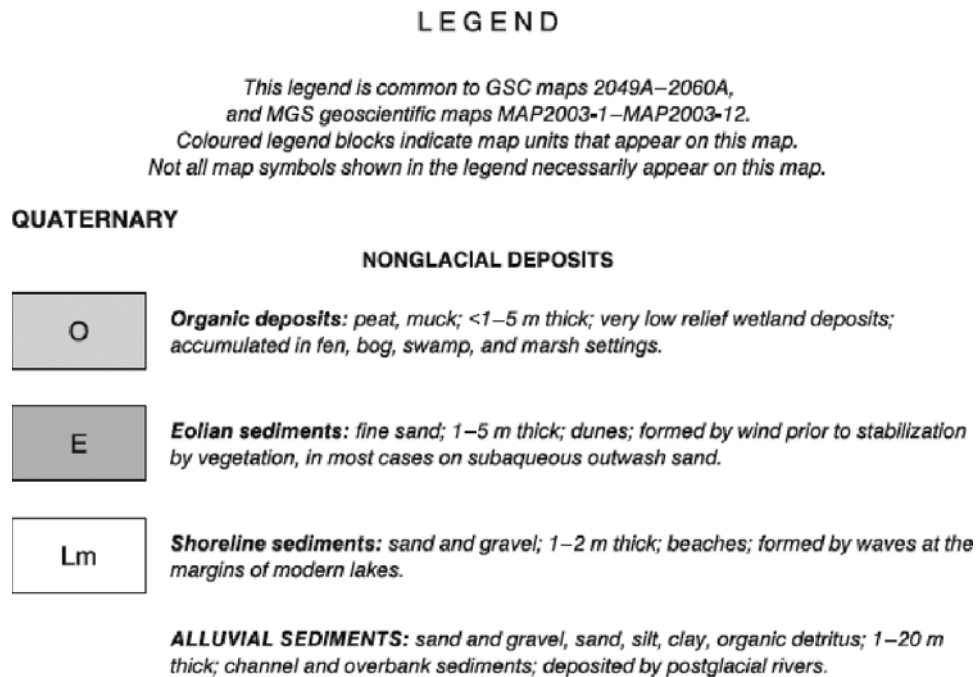
## Next Steps

The next steps in using XML documents for geological legend generation is to complete and fine tune the VBA scripting in ArcMap. After doing so, the XML schema for the legend will be expanded to include geological and mineral symbols that also occur on maps. Since the content of the legend exists in an established XML schema, other applications can be developed, such as a





**Figure 4.** XML schema representing the structure of XML documents for legend content schema, generated using XMLSpy. XML documents that are generated from Microsoft Word are validated against this schema before being processed in ArcMap. This schema diagram states which elements are required (boxes with solid outline), those that are optional (boxes with dashed outline), the number of occurrences of each element (0..∞), and the lineage between elements (symbols between elements indicating either a choice, or a sequence).



**Figure 5.** Top portion of a legend for a published surficial geological map. (NOTE: The legend in this image was not produced in ArcMap, as the VBA script is still in beta testing status. The goal is to achieve results similar to current production methods.)

customized query tool either for ArcMap or web mapping. By having data stored in a structured manner and widely accessible, the possibilities are limitless.

## REFERENCES

- ArcMap and ArcGIS, ESRI Inc., <http://www.esri.com>.  
Microsoft Development Network, XML Development Center, <http://msdn.microsoft.com/xml/>.  
NADM Data Interchange Technical Team, 2003, XML Encoding of the North American Data Model, in D.R. Soller, ed., Digital Mapping Techniques '03—Workshop Proceedings: U.S. Geological Survey Open-File Report 03-471, p. 215-221, available at <http://pubs.usgs.gov/of/2003/of03-471/boisvert/index.html>.  
Peter's XML Editor, Peter Reynolds, <http://www.iol.ie/~pxe/>.  
W3 Schools, XML Reference, <http://www.w3schools.com/xml/default.asp>.  
XMLSpy, Altova, Inc., <http://www.altova.com>.