Wang XML Notation

The XML notation is chosen to represent the ground truth which is the Augmented Wang Notation for the table. The XML file contains 7 main sections that detail various parts of a table. This document describes the Wang XML [1] notation using the table in Figure 1 as an example.

Region and State Information

Location	Population* (2000)	Longitude†
Northeast	3.120	
Maine	1.275	69°14.0′W
New Hampshire	1.236	71°34.3′W
Vermont	0.609	72°40.3′W
Northwest	9.315	
Washington	5.894	120°16.1′W
Oregon	3.421	120°58.7′W

*Population in Millions

†Geographic Center

Fig. 1 Region and State Information

Section1: The XML begins with:

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

Fig. 2 XML Section 1

The first line of the document states the version and notifies the reader that it is a valid XML document. The second line informs the reader that this XML is for a table ontology.

Section2: This section contains basic information about the table:

<Table TableOID="T13967" Title="Region and State Information" Caption="Sample Table" DocumentCitation="Lynn, S. and Embley, D.W., Semantically Conceptualizing and Annotating Tables, Technical Report, Brigham Young University, July 2008, www.deg.byu.edu/papers/TableConceptualization.pdf" Number="1">
<CategoryRootNodes>
<CategoryRootNode CategoryRootNodeOID="C1"/>
<CategoryRootNode CategoryRootNodeOID="C2"/>
</CategoryRootNodes>
</Table>

Fig. 3 XML Section 2

The table element contains multiple attributes like *TableOID*, *Title*, *Caption*, *DocumentCitation*, and *Number*. The table element contains the *CategoryRootNodes* element. This element contains the *CategoryRootNodeOID* attribute. The example table has a Wang dimensionality of 2 so there are two category root nodes. The *CategoryRootNodeOID* values are obtained from the indented notation sheets and are ordered based on the location of the cells in the sheet as below:

Consider a simulated table shown in Figure 4.

STUB		C21	
		C11	C12
R21	R11	XX	XX
	R12	XX	XX
	R13	XX	XX

Fig. 4 Simulated Table

Wang's category trees for tables are represented using the indented notation constructed from the list-row notation. In the indented notation, nodes at the same level of the tree appear in the same column. The children of a particular node in the tree appear below the node in the next column. The indented notation is formed by looping through the list-row notation and printing the values of the rows in such a way that no paths in the tree are repeated. Figures 1 and 2 represent the indented notation for Simulated Table 1.

R21	
	R11
	R12
	R13

Fig. 5 Indented Notation for Row Category

C21	
	C11
	C12

Fig. 6 Indented Notation for Column Category

The OID notation for the above categories is:

C1	
	C1.1
	C1.2
	C1.3

Fig. 7 OID Notation for Row Category

C2	
	C2.1
	C2.2

Fig. 8 OID Notation for Column Category

The C21, C11 and C12 nodes are just used to represent column categories as in previous Figures. But the Cx.x notation in the OID notation is a notation for categories used for every table in the XML notation.

Section3: The third section lists every category node in the table:

<CategoryNodes>

<CategoryNode CategoryNodeOID="C1"/>

<CategoryNode CategoryNodeOID="C1.1" Label="Population"/>

<CategoryNode CategoryNodeOID="C1.2" Label="Longitude"/>
<CategoryNode CategoryNodeOID="C2" Label="Location"/>
<CategoryNode CategoryNodeOID="C2.1" Label="Northeast"/>
<CategoryNode CategoryNodeOID="C2.1.1" Label="Maine"/>
<CategoryNode CategoryNodeOID="C2.1.2" Label="New Hampshire"/>
<CategoryNode CategoryNodeOID="C2.1.3" Label="Vermont"/>
<CategoryNode CategoryNodeOID="C2.2" Label="Northwest"/>
<CategoryNode CategoryNodeOID="C2.2.1" Label="Washington"/>
<CategoryNode CategoryNodeOID="C2.2.2" Label="Oregon"/>
</CategoryNodes>

Fig. 9 XML Section 3

The category nodes element contains a list of every category node in the table. Every category node element contains an attribute *CategoryNodeOID* which is the category's operational id number. From the example one can see that if a category node belongs to category root C1, it will share the same prefix in its OID. The following diagram may help explain how the id scheme works.

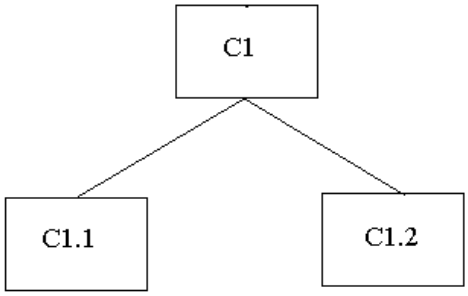

Fig. 10 Category Tree 1 for table in Fig. 1

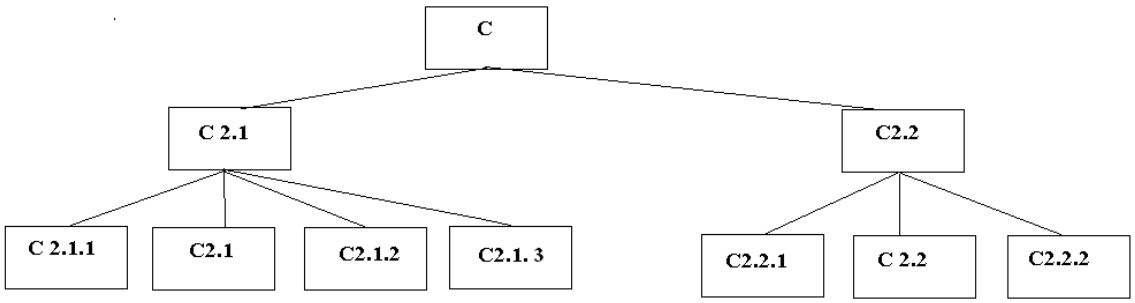

Fig. 11 Category Tree 2 for table in Fig. 1

C2.1 and C2.2 are aggregates which is why they are repeated in the tree. While they are shown twice in the tree and table they are not listed twice in the XML. If a category has a label, which should be every category node except possibly category root nodes, the label will also be added as an attribute.

Section4: The next section lists the category nodes with their children:

<CategoryParentNodes>
<CategoryParentNode CategoryParentNodeOID="C1">
<CategoryNodes>
<CategoryNode CategoryNodeOID="C1.1" />
<CategoryNode CategoryNodeOID="C1.2" />
</CategoryNodes>
</CategoryParentNode>
<CategoryParentNode CategoryParentNodeOID="C2">
<CategoryNodes>
<CategoryNode CategoryNodeOID="C2.1" />
<CategoryNode CategoryNodeOID="C2.2" />
</CategoryNodes>
</CategoryParentNode>
<CategoryParentNode CategoryParentNodeOID="C2.1">
<CategoryNodes>
<CategoryNode CategoryNodeOID="C2.1.1"/>

<CategoryNode CategoryNodeOID="C2.1.2"/>

<CategoryNode CategoryNodeOID="C2.1.3"/>

</CategoryNodes>
</CategoryParentNode>

<CategoryParentNode CategoryParentNodeOID="C2.2">

<CategoryNodes>

<CategoryNode CategoryNodeOID="C2.2.1"/>

<CategoryNode CategoryNodeOID="C2.2.2"/>

</CategoryNodes>

</CategoryParentNode>

</CategoryParentNodes>

Fig. 12 XML Section 4

This section begins with the element *CategoryParentNodes* which contains a list of all the category nodes that have children. Each element in the list is of type *CategoryParentNode* and has a *CategoryParentNodeOID* attribute whose value is the same as its *CategoryNodeOID* value. Each of these elements contains an element *CategoryNodes*. This element contains a list of the current category parent node's direct children of element type *CategoryNode.*

Section5: DataCells are listed in this section:

<DataCells>

<DataCell DataCellOID="D1,1" DataValue="3.120">

<HeaderNodes>

<HeaderNode HeaderNodeOID="C2.1"/>

</HeaderNodes>

<CategoryLeafNodes>

<CategoryLeafNode CategoryLeafNodeOID="C1.1" />

</CategoryLeafNodes>

</DataCell>

<DataCell DataCellOID="D1,2">

<HeaderNodes>

<HeaderNode HeaderNodeOID="C2.1"/>

</HeaderNodes>

<CategoryLeafNodes>

<CategoryLeafNode CategoryLeafNodeOID="C1.2" />

</CategoryLeafNodes>

</DataCell>

<DataCell DataCellOID="D2,1" DataValue="1.275">

<CategoryLeafNodes>

<CategoryLeafNode CategoryLeafNodeOID="C1.1" />

<CategoryLeafNode CategoryLeafNodeOID="C2.1.1" />

</CategoryLeafNodes>

</DataCell>

.

.

.

</DataCells>

Fig. 13 XML Section 5

This section consists of the *DataCells* element. Within this element is a list of elements that detail each data cell within the table. Each data cell has its own *DataCell* element with its *DataCellOID* attribute which is its x and y coordinates in the Excel table with (1,1) being the data cell in the top left corner. The x-coordinate increases to the right and the y-coordinate increases from top to bottom. *DataCells* usually contain a second attribute, *DataValue,* whose value is the textual or numerical content of that data cell within the table. Within each *DataCell* there may be a list of *HeaderNodes, CategoryLeafNodes,* or both. *HeaderNodes* are category nodes that are aggregates. For example C2.1 is an aggregate since it contains the accumulated information for Maine, New Hampshire and Vermont. It has children and it is not a leaf node, which is why it is put into the *HeaderNodes* list. Each *HeaderNode* element has a *HeaderNodeOID* attribute which is its *CategoryNodeOID*. If on the other hand the category associated with the data cell is a leaf node it will be listed in the *CategoryLeafNodes* element.

Each of these will be listed as an element type *CategoryLeafNode*. Similar to the *HeaderNode* elements they also have a *CategoryLeafNodeOID* attribute that is the same as their category node OID.

Section6: The next important section lists all the augmentations that occur within the table:

<Augmentations>
<Augmentation AugmentationOID="A1" AugmentationText="2000">
<CategoryNode CategoryNodeOID="C1.1"/>
</Augmentation>
<Augmentation AugmentationOID="A2" AugmentationText="Population in Millions"
FootnoteReference="%ampersand%number42;">
<CategoryNode CategoryNodeOID="C1.1"/>
</Augmentation>
<Augmentation AugmentationOID="A3" AugmentationText="Geographic Center"
FootnoteReference="%ampersand%dagger;">
<CategoryNode CategoryNodeOID="C1.2"/>
</Augmentation>
<Augmentation AugmentationOID="A4" AugmentationText="Geographic Center"
FootnoteReference="%ampersand%dagger;">
<CategoryNode CategoryNodeOID="C1.3"/>
</Augmentation>
</Augmentations>

Fig. 14 XML Section 6

In this section we have the *Augmentations* element. Within this element we see a list of *Augmentation* elements. Each *Augmentation* element has an *AugmentationOID* attribute. The numbering is sequential. If the Augmentation is a footnote it will contain the *AugmentationText* attribute and may also contain a *FootnoteReference* attribute. The *AugmentationText* is the comment about the cell and *FootnoteReference* is the type of symbol used in the cell to point to the footnote. If the *Augmentation* is a *Unit* it will have an attribute *AugmentationType* with value

"Units". Lastly if the *Augmentation* is of type "Other" the element will have an attribute *AugmentationText*. Within each *Augmentation* element there is another element. This element will be of type *CategoryNode* or *DataCell* and will contain the OID of the cell(s) the augmentation corresponds to.

Section7: The last part of the XML is the closing label for the Table Ontology:

</TableOntology>

Fig. 15 XML Section 7

References:

R. Padmanabhan, R. C. Jandhyala, M. Krishnamoorthy, G. Nagy, S. Seth, W. Silversmith, Interactive Conversion of Large Web Tables, Proceedings of Eighth International Workshop on Graphics Recognition, GREC 2009, Published by City University of La Rochelle, La Rochelle, France, July 22-23, 2009.