Wine collector

Final Project Report

Pallavi Murthy, Kartik Adur  
July 30, 2014

Indiana University

Z603 – XML Workshop

# Introduction:

Every sophisticated person dreams of one day having their own wine collection in their own cellar. But as we discovered while researching for this project, wines are hard, to track, to use, to talk about, or even to understand. They have a lot of metadata attached to them: origin of the grapes, type of grapes, location of bottling, year of bottling, the aroma, taste, etc. to name just a few. And not all metadata is associated with every wine. So we decided to find the most common elements in most wines and then use them as descriptors for all the wines any discerning wine connoisseur would wish to keep in their cellar.

The wine industry uses certain jargon or wine specific terminology that we have (faithfully) used so as to keep the aura that is associated with wines, rather than bring it to a base level with other drinks. For instance ‘Vintage’ specifies the year of bottling the wine. The region encapsulates the area where the grapes (for the wine) were grown. And the wine descriptors describe the aromas or tastes experienced while imbibing wine.

The structure used in this XML schema was done specifically to differentiate data based on semantics. The vineyard, the region where the grapes were grown, the year when the wine was bottled, and the location you can purchase another bottle are the main descriptors of the wine bottle. The wine contained within the bottle on the other hand has its own set of metadata, such as its taste, aroma, color, alcohol content, and the type of grapes used for the wine. A specific section was also created to indicate what foods best pair with the bottle. Normally foods are paired with the wine, but our research indicated that some bottles contain a wine mixture of more than one type of grape variety, hence the separation.

We have strived to make collecting, labeling, and tracking wine in your own cellar slightly easier with our efforts. This data may be slightly incomplete since we are unaware of all possible data variants, but hopefully this will get you started.

# Schema:

<?xml version="1.0" encoding="UTF-8"?>  
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">  
 <xs:element name="winecollection">  
 <xs:complexType>  
 <xs:sequence>  
 <xs:element name="bottle" minOccurs="1" maxOccurs="10">  
 <xs:complexType>  
 <xs:all>  
 <xs:element name="producer" type="producerInfo"></xs:element>  
 <xs:element name="vintage" type="xs:gYear"></xs:element>  
 <xs:element name="wine" type="wineInfo"></xs:element>  
 <xs:element name="serve" type="serveInfo"></xs:element>  
 <xs:element name="sellers" type="sellerInfo" minOccurs="0"></xs:element>  
 </xs:all>  
 </xs:complexType>  
 </xs:element>  
 </xs:sequence>  
 </xs:complexType>  
 </xs:element>  
   
 <xs:complexType name="producerInfo">  
 <xs:sequence>  
 <xs:element name="vineyard" type="xs:string"></xs:element>  
 <xs:element name="address" type="xs:string"></xs:element>  
 <xs:element name="country" type="xs:string"></xs:element>  
 <xs:element name="geocode">  
 <xs:complexType>  
 <xs:sequence>  
 <xs:element name="latitude" minOccurs="1" maxOccurs="1"></xs:element>  
 <xs:element name="longitude" minOccurs="1" maxOccurs="1"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
 </xs:element>  
 <xs:element name="label" type="xs:anyURI" minOccurs="0"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
   
 <xs:complexType name="wineInfo">  
 <xs:sequence>  
 <xs:element name="variety" type="xs:string"></xs:element>  
 <xs:element name="type" type="xs:string"></xs:element>  
 <xs:element name="color">  
 <xs:simpleType>  
 <xs:restriction base="xs:string">  
 <xs:enumeration value="Almost Clear"></xs:enumeration>  
 <xs:enumeration value="Green Yellow"></xs:enumeration>  
 <xs:enumeration value="Platinum Yellow"></xs:enumeration>  
 <xs:enumeration value="Pale Yellow"></xs:enumeration>  
 <xs:enumeration value="Pale Gold"></xs:enumeration>  
 <xs:enumeration value="Deep Gold"></xs:enumeration>  
 <xs:enumeration value="Pale Salmon"></xs:enumeration>  
 <xs:enumeration value="Deep Pink"></xs:enumeration>  
 <xs:enumeration value="Deep Salmon"></xs:enumeration>  
 <xs:enumeration value="Pale Ruby"></xs:enumeration>  
 <xs:enumeration value="Deep Violet"></xs:enumeration>  
 <xs:enumeration value="Deep Purple"></xs:enumeration>  
 <xs:enumeration value="Tawny"></xs:enumeration>  
 </xs:restriction>  
 </xs:simpleType>  
 </xs:element>  
 <xs:element name="alcoholbyvolume" type="xs:decimal"></xs:element>  
 <xs:element name="descriptors" type="descriptorInfo"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
   
 <xs:complexType name="descriptorInfo">  
 <xs:sequence>  
 <xs:element name="acidity" type="xs:string" minOccurs="0" maxOccurs="5"></xs:element>  
 <xs:element name="alcohol" type="xs:string" minOccurs="0" maxOccurs="5"></xs:element>  
 <xs:element name="body" type="xs:string" minOccurs="0" maxOccurs="5"></xs:element>  
 <xs:element name="flower" type="xs:string" minOccurs="0" maxOccurs="5"></xs:element>  
 <xs:element name="fruit" type="xs:string" minOccurs="0" maxOccurs="5"></xs:element>  
 <xs:element name="herb" type="xs:string" minOccurs="0" maxOccurs="5"></xs:element>  
 <xs:element name="inorganic" type="xs:string" minOccurs="0" maxOccurs="5"></xs:element>  
 <xs:element name="oak" type="xs:string" minOccurs="0" maxOccurs="5"></xs:element>  
 <xs:element name="spice" type="xs:string" minOccurs="0" maxOccurs="5"></xs:element>  
 <xs:element name="style" type="xs:string" minOccurs="0" maxOccurs="5"></xs:element>  
 <xs:element name="tannin" type="xs:string" minOccurs="0" maxOccurs="5"></xs:element>  
 <xs:element name="yeast" type="xs:string" minOccurs="0" maxOccurs="5"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
   
 <xs:complexType name="serveInfo">  
 <xs:sequence>  
 <xs:element name="best" type="xs:string" minOccurs="1" maxOccurs="3"></xs:element>  
 <xs:element name="good" type="xs:string" minOccurs="0" maxOccurs="3"></xs:element>  
 <xs:element name="temperature" type="xs:int"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
   
 <xs:complexType name="sellerInfo">  
 <xs:sequence>  
 <xs:element name="name" type="xs:string"></xs:element>  
 <xs:element name="price" type="xs:decimal"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
</xs:schema>

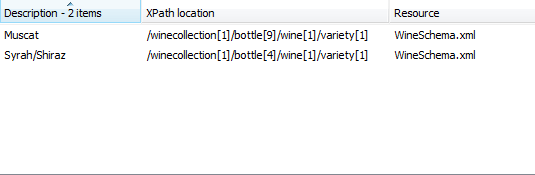
# Sample Data:

<winecollection>  
 <bottle>  
 <producer>  
 <vineyard>Kemblefield Estate Winery Ltd</vineyard>  
 <address>Aorangi Rd, Hastings 4171</address>  
 <country>NZ</country>  
 <geocode>  
 <latitude>-39.592077</latitude>  
 <longitude>176.507468</longitude>  
 </geocode>  
 </producer>  
 <vintage>2012</vintage>  
 <wine>  
 <variety>Sauvignon Blanc</variety>  
 <type>White Wine</type>  
 <color>Green Yellow</color>  
 <alcoholbyvolume>13.0</alcoholbyvolume>  
 <descriptors>  
 <acidity>Crisp</acidity>  
 <body>Light-Bodied</body>  
 <fruit>Tropical Fruit</fruit>  
 <fruit>Citrus</fruit>  
 <herb>GooseBerry</herb>  
 <herb>Vegetal</herb>  
 <herb>Asparagus</herb>  
 <herb>Bell Pepper</herb>  
 </descriptors>  
 </wine>  
 <serve>  
 <best>Green Vegetables</best>  
 <good>Potato</good>  
 <temperature>43</temperature>  
 </serve>  
 </bottle>  
 <bottle>  
 <producer>  
 <vineyard>Sextant Wines</vineyard>  
 <address>2324 California 46, Paso Robles, CA 93446</address>  
 <country>US</country>  
 <geocode>  
 <latitude>35.633555</latitude>  
 <longitude>-120.6566579</longitude>  
 </geocode>  
 </producer>  
 <vintage>2012</vintage>  
 <wine>  
 <variety>Cabernet Sauvignon</variety>  
 <type>Red Wine</type>  
 <color>Deep Purple</color>  
 <alcoholbyvolume>14.5</alcoholbyvolume>  
 <descriptors>  
 <body>Full-Bodied</body>  
 <body>Finesse</body>  
 <fruit>Cherry</fruit>  
 <inorganic>Smoky</inorganic>  
 <oak>Smoky</oak>  
 <spice>Pepper</spice>  
 <style>Smoky</style>  
 <tannin>Chocolate</tannin>  
 </descriptors>  
 </wine>  
 <serve>  
 <best>Red Meat</best>  
 <good>Pork</good>  
 <temperature>62</temperature>  
 </serve>  
 </bottle>  
 <bottle>  
 <producer>  
 <vineyard>Martin Ray Winery</vineyard>  
 <address>2191 Laguna Rd, Santa Rosa, CA 95401</address>  
 <country>US</country>  
 <geocode>  
 <latitude>38.469055</latitude>  
 <longitude>-122.850627</longitude>  
 </geocode>  
 </producer>  
 <vintage>2011</vintage>  
 <wine>  
 <variety>Angeline Pinot Noir</variety>  
 <type>Red Wine</type>  
 <color>Pale Ruby</color>  
 <alcoholbyvolume>13.9</alcoholbyvolume>  
 <descriptors>  
 <fruit>Strawberry</fruit>  
 <fruit>Raspberry</fruit>  
 <fruit>Cherry</fruit>  
 <spice>Tea</spice>  
 <style>Smooth</style>  
 <style>Supple</style>  
 </descriptors>  
 </wine>  
 <serve>  
 <best>Poultry</best>  
 <good>Onions</good>  
 <temperature>54</temperature>  
 </serve>  
 </bottle>  
</winecollection>

# Query Examples:

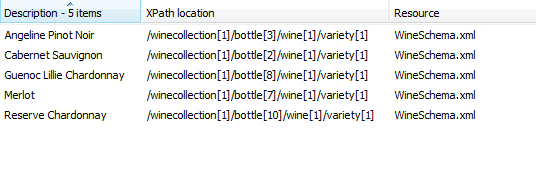
## XPath 1

**List all variety of wines with vintage year of 2008**//bottle[vintage="2008"]/wine/variety



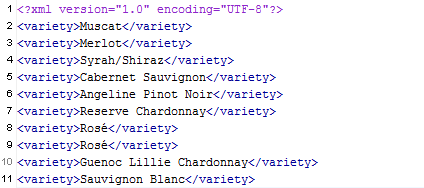
## XPath 2

**List all wine varieties made in the US**//bottle/producer[country="US"]/ancestor::bottle/wine/variety

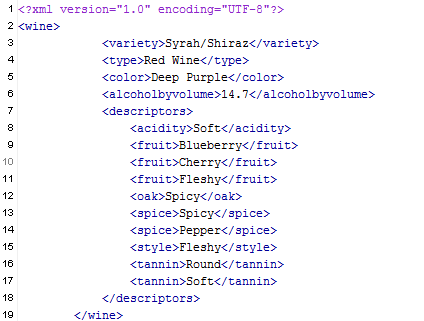


## XQuery 1

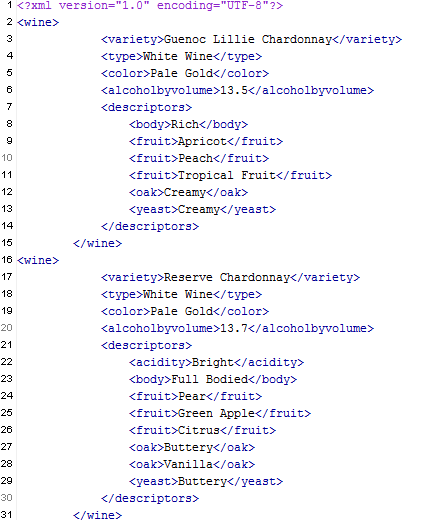
**List all wines ordered by the alcohol content by volume**for $x in doc('WineSchema.xml')//bottle  
where $x/vintage  
order by $x/wine/alcoholbyvolume descending  
return $x/wine/variety



## XQuery 2

**List all wines with a hint of Blueberry**for $x in doc('WineSchema.xml')//bottle  
where $x/wine/descriptors/fruit/text () = 'Blueberry'  
return $x/wine  


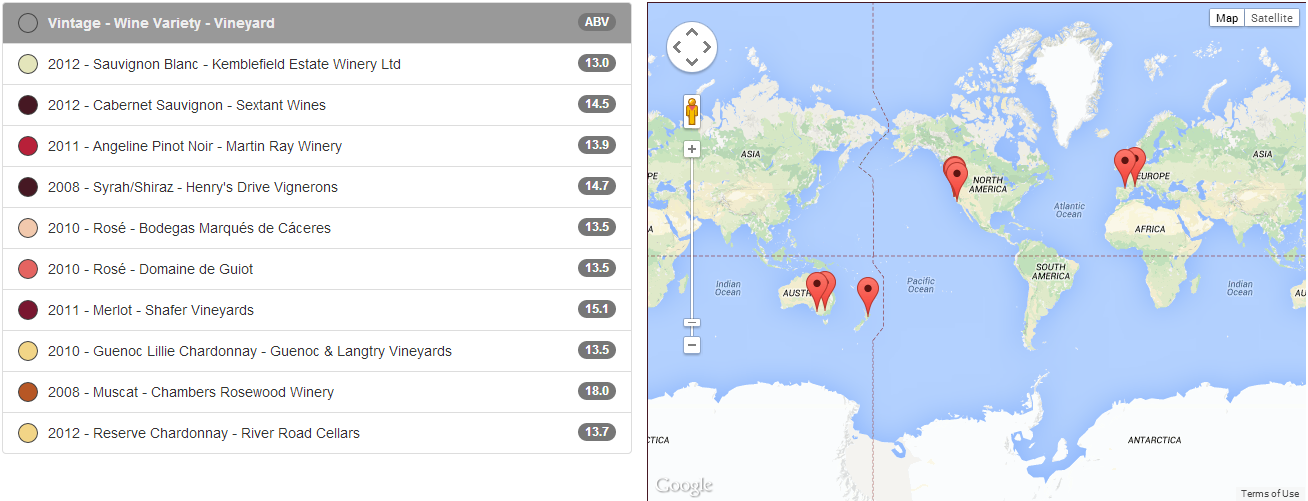
## XQuery 3

**List the descriptions for all Chardonnays**for $x in doc('WineSchema.xml')//bottle  
where $x/wine[contains(variety, ‘Chardonnay')]  
return $x/wine  


# XSLT Description:

People who collect wines usually look for a few data elements such as the vintage, color, type of grapes, and location. While alcohol by volume is not strictly a discerning criteria there are times when it can be helpful. So the wine collection is displayed as a list of bottles, indicating their color, vintage, variety, and producer. An accompanying map also indicates where the wine was produced for easy reference.

<?xml version="1.0" encoding="UTF-8"?>  
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">  
 <xsl:variable name="lowercase" select="'abcdefghijklmnopqrstuvwxyz'"/>  
 <xsl:variable name="uppercase" select="'ABCDEFGHIJKLMNOPQRSTUVWXYZ'"/>  
 <xsl:variable name="dash" select="'-'"/>  
 <xsl:variable name="space" select="' '"/>  
  
 <xsl:template match="/">  
 <xsl:for-each select="//bottle">  
 <li class="list-group-item bottle clearfix" data-lon="{producer/geocode/longitude}"  
 data-lat="{producer/geocode/latitude}">  
 <span  
 class="color {translate(translate(wine/color, $uppercase, $lowercase), $space, $dash)}"  
 ><blank/></span>  
 <xsl:value-of select="vintage"/> &#45; <xsl:value-of select="wine/variety"/> &#45;  
 <xsl:value-of select="producer/vineyard"/>  
 <span class="badge" style="float:right;"><xsl:value-of select="wine/alcoholbyvolume"/></span>  
 </li>  
 </xsl:for-each>  
 </xsl:template>  
</xsl:stylesheet>

****

# Conclusion:

This project is a work in progress, and while there is an attempt to capture a majority of the metadata related to wines, there is still data that was hard to capture and faithfully reproduce. We attempted to figure what data would fit where in our xml schema by creating the smallest units of indivisible data collections.

There is a vast amount of literature available for a wine connoisseur. But at the end of the day there are few things that concern a person willing to have some wine. Does it taste good for the appropriate occasion or food? Is it chilled to the right temperature? What year the wine was bottled? And how much was paid for it? The last two are more for the collectors boasting but important none the less.

As stated before this version work (barely) and would require a whole lot more in terms of data and functionality before it can be considered complete. For instance, by clicking on listed bottles the user should be able to view detailed information for a bottle. Adding new bottles, removing bottles that have been consumed, and archiving bottles that have been consumed would be really good features.

But for now this version shows promise as an application for wine collectors who would like to catalogue their wine collection.