Disclaimer

- The material provided in this document is not my original work and is a summary of some one else’s work(s).
- A simple Google search of the title of the document will direct you to the original source of the material.
- I do not guarantee the accuracy, completeness, timeliness, validity, non-omission, merchantability or fitness of the contents of this document for any particular purpose.
- Downloaded from najeebkhan.github.io

MusicXML

By

Najeeb

2013-8-14

Outline

- XML
- MusicXML
- Example

eXtensible Markup Language

- XML is a markup language designed to describe data
- XML has only a couple of predefined tags
  - All the rest are defined by the document designer.
  - XML can be used to create languages
- XML uses a Document Type Definition (DTD) or XML Schema to define an XML document type.
  - DTD or Schema define allowable tags, attributes, and value types

- An XML document is defined by a standard opening element:
  - <?xml version=“1.0” ?>
- The XML body starts with a single root element.
- An element is text of the form:

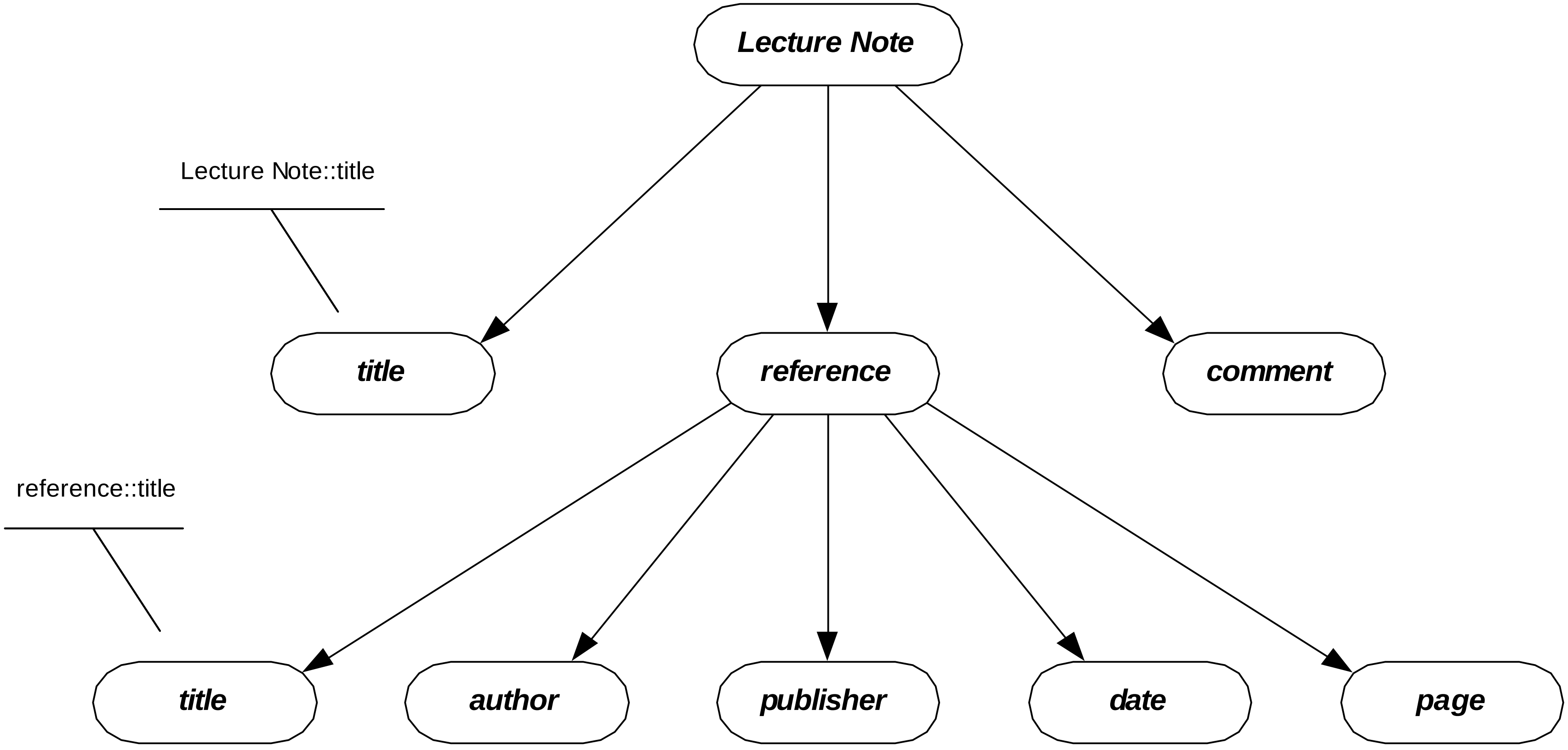
  <someTag anAttribute=“someValue”>payload text</someTag>

  where the payload may be one or more child elements or simply text or both.

XML Example

-
  ```
  <?xml version="1.0" ?>
   <LectureNote>
     <title>XML Example #1</title>
     <reference>
       <title>C#.Net Web Developer's Guide</title>
       <author>Turtschi et. al.</author>
       <publisher>Syngress</publisher>
       <date>2002</date>
       <page>433</page>
     </reference>
     <comment>First example in Chap 9</comment>
   </LectureNote>
  ```

XML Example


Lecture Note
Lecture Note::title
title
reference
comment
reference::title
title
author
publisher
date
page

XML Presentation

- There are several ways XML data can be presented to a user:
  - XML file interpreted by script in an HTML page
  - XML styled with an XSL style sheet
  - Essentially, the XSL sheet creates viewable HTML
  - Read, interpreted, and modified by an application

MusicXML

- There are many fine computer music programs in the world. Unfortunately, sharing music between them used to be difficult
- MIDI is a wonderful format for performance applications like sequencers, but it is not so wonderful for other applications like music notation
- The goal is to create a universal format for common Western music notation

- Say you have 100 music applications, each with its own format
- For each application to communicate with the other, 10,000 separate programs would need to be written without a common interface language
- With a common interface language, each application writes only one program, so only 100 separate programs would be required

- MusicXML has two types of structures
- Partwise DTD: The root element is <score-partwise>
  - The musical part is primary, and measures are contained within each part
- timewise DTD: The root element is <score-timewise>
  - The measure is primary, and musical parts are contained within each measure

“Hello World” in MusicXML

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE score-partwise PUBLIC
    "-//Recordare//DTD MusicXML 3.0 Partwise//EN"
    "http://www.musicxml.org/dtds/partwise.dtd">
<score-partwise version="3.0">
  <part-list>
    <score-part id="P1">
      <part-name>Music</part-name>
    </score-part>
  </part-list>
  <part id="P1">
    <measure number="1">
      <attributes>
        <divisions>1</divisions>
        <key>
          <fifths>0</fifths>
        </key>
        <time>
          <beats>4</beats>
          <beat-type>4</beat-type>
        </time>
        <clef>
          <sign>G</sign>
          <line>2</line>
        </clef>
      </attributes>
      <note>
        <pitch>
          <step>C</step>
          <octave>4</octave>
        </pitch>
        <duration>4</duration>
        <type>whole</type>
      </note>
    </measure>
  </part>
</score-partwise>
```

- <?xml version="1.0" encoding="UTF-8" standalone="no"?>
  - Setting the value of standalone to "no" means that we are defining the document with an external definition in another file

- <!DOCTYPE score-partwise PUBLIC "-//Recordare//DTD MusicXML 3.0 Partwise//EN" "http://www.musicxml.org/dtds/partwise.dtd">
  - We are using MusicXML, specifically a partwise score where measures are contained within parts. We use a PUBLIC declaration including an Internet location for the DTD

-
  ```
  <part-list>
      <score-part id="P1">
          <part-name>Part 1</part-name>
      </score-part>
  </part-list>
  ```
  - A MusicXML file starts off with a header that lists the different musical parts in the score

- <part id="P1">
  - beginning the first part within the document
- <measure number="1">
  - Starting the first measure in the first part
- <attributes>
  - <divisions>1</divisions>
  - <key>
    <fifths>0</fifths>
    </key>

-
  ```
  <time>
        <beats>4</beats>
        <beat-type>4</beat-type>
  </time>
  ```

```
<note>
        <pitch>
                <step>C</step>
                <octave>4</octave>
        </pitch>
        <duration>4</duration>
        <type>whole</type>
</note>
```