Techniques for Data Analysis of AnIML Files

Stuart Chalk
University of North Florida

Overview

- Data analysis issues
- Data structures in AnIML
- Reading XML files
- Knowing where the data is
- Getting to the data
- XPATH
- XSLT
- XML Parsing

Data Analysis Issues

- Large amounts of digital data
- Where is the data in AnIML?
- How do I reference the data?
- Getting the data
- Large files

- Data integrity checking

Data Structures in AnIML

- IndividualValueSet

```
<Series name="Absorbance" dependency="dependent" seriesID="absorbance1" seriesType="Float32">
    <IndividualValueSet>
        <Float32>-0.000162</Float32>
        <Float32>-0.000166</Float32>
        <Float32>-0.000157</Float32>
        <Float32>-0.000157</Float32>
        <Float32>-0.000157</Float32>
        <Float32>-0.000141</Float32>
        <Float32>-0.000128</Float32>
        <Float32>-0.000137</Float32>
        <Float32>-0.000141</Float32>
        <Float32>-0.000137</Float32>
        <Float32>-0.000128</Float32>
        <Float32>-0.000128</Float32>
        <Float32>-0.00012</Float32>

        <Float32>-0.002199</Float32>
    </IndividualValueSet>
    &Abs;
</Series>
```

350 characters
(126 data characters)

Data Structures in AnIML

- AutoIncrementedValueSet

```
<SeriesSet name="Spectrum" length="2001">
    <Series name="Wavelength" dependency="independent" seriesID="wavelength1" seriesType="Float32">
        <AutoIncrementedValueSet>
            <StartValue>
                <Float32>400</Float32>
            </StartValue>
            <Increment>
                <Float32>-0.1</Float32>
            </Increment>
        </AutoIncrementedValueSet>
        &nm;
    </Series>

</SeriesSet>
```

- EncodedValueSet

```
<Series name="Time" dependency="independent" seriesID="time1" seriesType="Float32">
    <EncodedValueSet>
        i94puUkQLrleoCS5XqAkuV6gJLlm2RO5vTcGuainD7lm2RO5qKcPub03Brm9Nwa5gqj7uA==
    </EncodedValueSet>
</Series>
```

72 characters

Reading XML Files

- Any text application
- XML Reading Languages
  - C++, Java, .NET
  - PHP, Ruby, Python
- XML Reading Applications
  - XMLSpy
  - OxygenXML
  - Excel (via VBA)

Knowing where the data is

- AnIML Schema
  - Series
  - Parameters
  - Attributes

- Add unique identifiers
  - seriesID attribute
  - id attribute
  - Parameter name

Getting the data

- Traverse the document tree
  - Only practical for smaller files
  - Must have standard format for data

```
<Result name="Spectrum">
    <SeriesSet name="Spectrum" length="2001">
        <Series name="Wavelength" dependency="independent" seriesID="wavelength1" seriesType="Float32">
        <Series name="Time" dependency="independent" seriesID="time1" seriesType="Float32"> [4 lines]
        <Series name="Absorbance" dependency="dependent" seriesID="absorbance1" seriesType="Float32"> [2
    </SeriesSet>
</Result>
```

```
<?php
  $animl = new simplexml_load_file('animl.xml);
  $wave=$animl->ExperimentStepSet->Result->SeriesSet->Series[0];
  $time=$animl->ExperimentStepSet->Result->SeriesSet->Series[1];
  $abs=$animl->ExperimentStepSet->Result->SeriesSet->Series[2];
?>
```

- Better is to use...

XPATH

- Syntax for accessing specific nodes in an XML file

```
<Result name="Spectrum">
    <SeriesSet name="Spectrum" length="2001">
        <Series name="Wavelength" dependency="independent" seriesID="wavelength1" seriesType="Float32">
        <Series name="Time" dependency="independent" seriesID="time1" seriesType="Float32"> [4 lines]
        <Series name="Absorbance" dependency="dependent" seriesID="absorbance1" seriesType="Float32"> [2
    </SeriesSet>
</Result>
```

```
<?php
  $xml = new SimpleXMLElement($expt['content']);
  $xml->registerXPathNamespace("a","urn:org:astm:animl:schema:core:draft:0.37");
  $wave=$xml->xpath("//a:Series[@seriesID='wavelength1']";
  $time=$xml->xpath("//a:Series[@seriesID='time1']";
  $abs=$xml->xpath("//a:Series[@seriesID='absorbance1']";
?>
```

XSLT

- Xml Stylesheet Language Transformation
- XML document that takes xml from another file and reformats it based on the stylesheet rules
- Currently version 2.0
- Version 2.0 is much more versatile than Version 1.0 and incorporates better xpath support
- http://www.w3.org/TR/xslt20/

XML Parsing

- Tree parsers
  - Load the whole file into memory
  - Can access any part of the file at any time
  - Can modify the file
  - Slow for large files
- Stream parsers
  - Load only part of the file into memory at a time
  - Can only access the part of the file in memory
  - Cannot modify file
  - Fast for large files

PHP http://www.ibm.com/developerworks/xml/library/x-xmlphp2/index.html
.NET http://support.softartisans.com/kbview_674.aspx

Resources

- http://animl.sourceforge.net
- http://www.w3.org/standards/xml/
- http://chalk.coas.unf.edu/animl