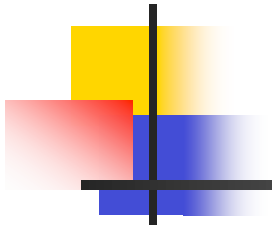
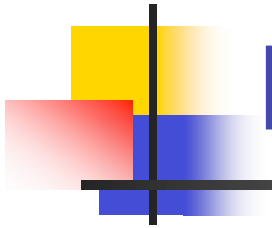


ICT336 Internet Systems Programming

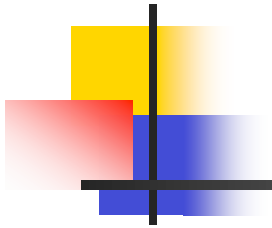
Example XML Applications

(Week 9 Lecture 2b)



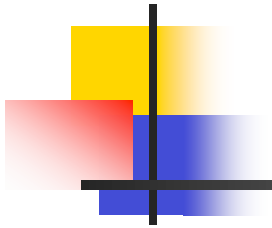
Learning Objectives

- Learn from some existing example XML applications on how to create XML solutions.



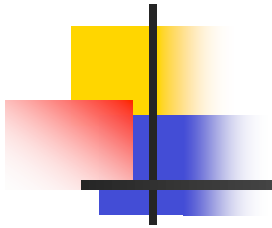
Learning Objectives

- In the scheme of what we are doing in this unit:
  - We are studying how to use XML as an important set of Internet technologies to use as solutions in different areas.
  - There are a few key considerations we should always keep in mind when designing XML solutions.
  - We can learn a lot about how to effectively use XML technologies by looking at some examples of how they have been applied to other problems.



Lecture Outline

- Some examples of existing XML applications:
  - Synchronized Multimedia Integration Language (SMIL)
    - for streaming multimedia
  - Mathematical Markup Language (MathML)
    - for manipulating mathematical expressions
  - XML in Bioinformatics and Genetic Research
  - XHTML
    - the XML version of HTML



SMIL

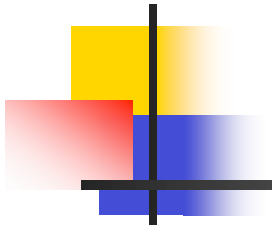
- Synchronized Multimedia Integration Language (SMIL) is a language defined by W3C
  - to "simple authoring of interactive audiovisual presentations…used for "rich media"/multimedia presentations which integrate streaming audio and video with images, text or any other media type.”.
  - The language contains mark-ups to define information about how the multimedia components (eg. images, sounds, videos) are to be synchronized.

An Example SMIL Document

```
<?xml version="1.0">
<smil>
  <head>
    <meta name="title" content="A SMIL Document"/>
    <region id="image_region"
       left="0" top="0" height="200" width="200" />
  </head>
  <body>
    <seq>
      <img src="header.gif"
         region="image_region" fill="remove" />
      <par>
        <audio src="track.wav" />
        <img src="picture.gif"
           region="image_region" fill="remove" />
      </par>
    </seq>
  </body>
</smil>
```

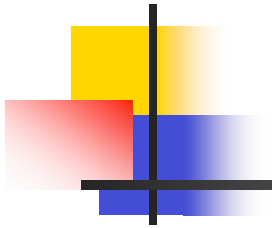
Present in sequence

Present in parallel



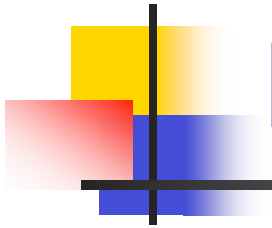
Creating SMIL Documents

- Some example authoring tools for creating SMIL documents:
  - Macromedia's HomeSite and ColdFusion.
  - RealNetworks' SMILGen Authoring Kit
  - Adobe GoLive.
- These software generate SMIL documents by allowing users to use a WYSIWYG graphical interface to define a streaming multimedia presentation.



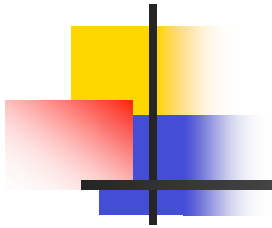
SMIL Browsers

- Example popular multimedia players now have support for SMIL (ie. they can show the presentation as described by a SMIL document)
  - Apple Quicktime 4.1+
  - RealNetworks' RealOne player.
  - Internet Explorer 5.5+
- We also have browsers which are specifically created for SMIL presentations
  - X-SMILES from TML laboratory.
  - Grins by Oratrix.



Reference

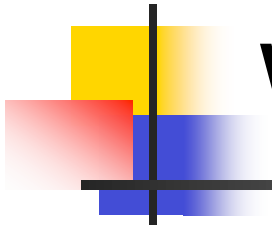
- For a more complete look of SMIL technologies, start at W3C's SMIL page:
  - http://www.w3.org/AudioVideo/



MathML

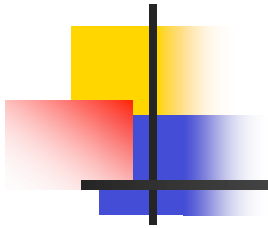
- MathML is intended to facilitate the use and re-use of mathematical and scientific content on the Web, and for other applications such as computer algebra systems, print typesetting, and voice synthesis.

- It can describe both
  - the presentation of mathematical notation for high-quality visual display, and
  - the actual mathematical content.



Why MathML?

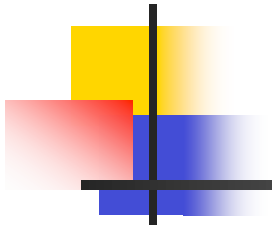
- HTML does not support display of mathematical expressions very well.
  - Besides the few ASCII symbols and operators we find on the keyboard
- With so much information moved onto the Web, the scientific community finds it very restrictive not being able to share basic mathematical descriptions.
  - They resorted to doing things like scanning a page with the mathematical expressions into images files and exporting the images.



Example MathML

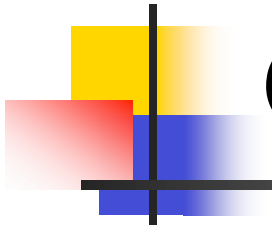
$$x^2 + 4x + 4$$

```
<apply>
  <plus/>
  <apply>
    <power/>
    <ci>x</ci>
    <cn>2</cn>
  </apply>
  <apply>
    <times/>
    <cn>4</cn>
    <ci>x</ci>
  </apply>
  <cn>4</cn>
</apply>
```



Example Software with MathML Support

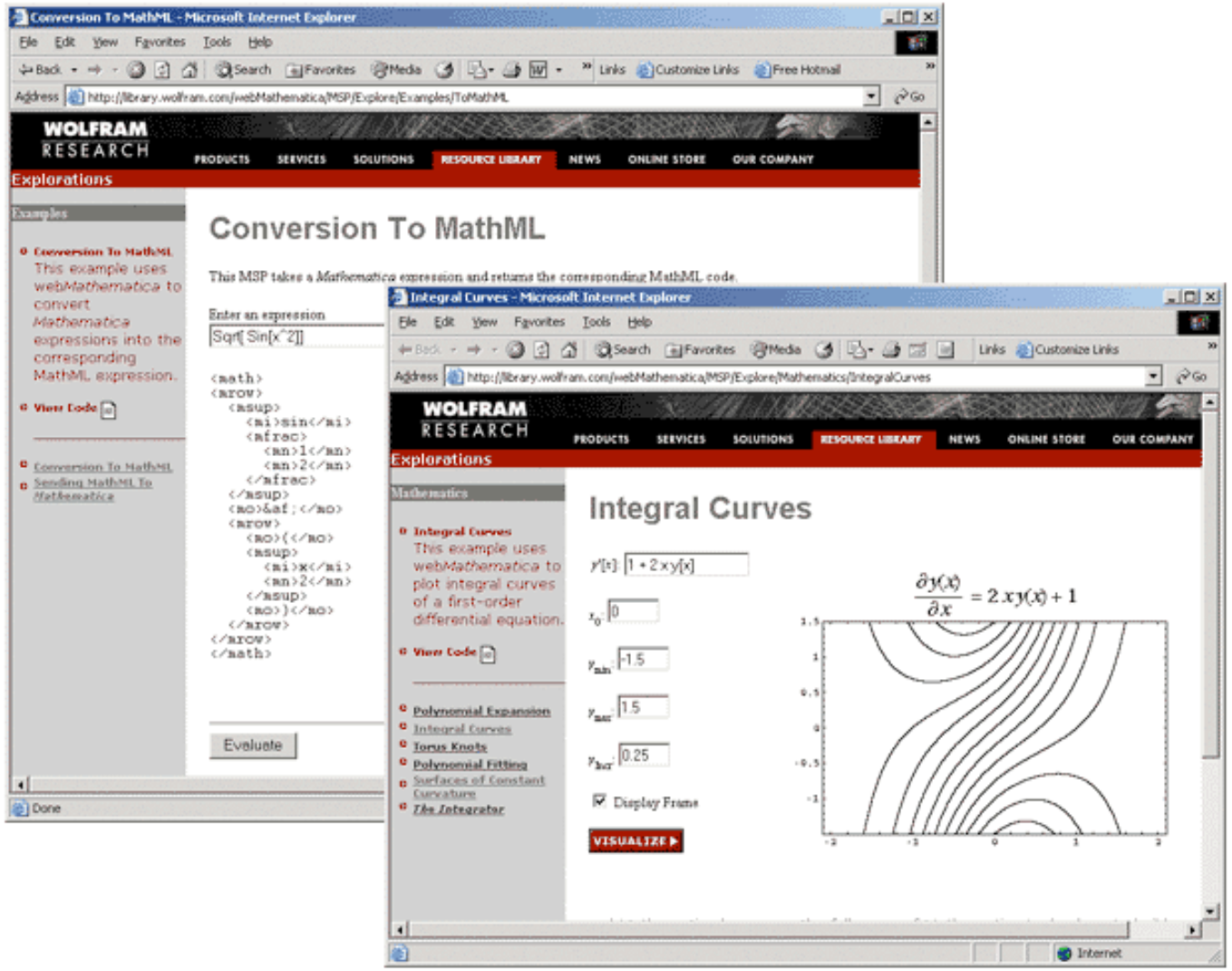
- Amaya
  - W3C's web browser - can display MathML and includes an editor for MathML.
- Maple:
  - a popular computer algebra system
  - import, export and evaluate MathML content
  - support generation and rendering of MathML
- Mathematica:
  - another popular maths manipulation system
  - contains a visual typesetting and authoring tool which both renders and exports MathML.



Other Example MathML Software

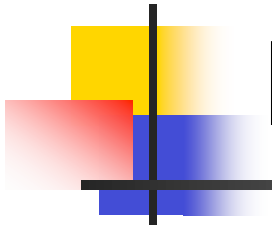
- MathType:
  - a stand-alone equation editor and authoring tool for MathML
- LaTeX2HTML MathML package:
  - LaTeX to MathML conversion
- LiveMath:
  - an interactive mathematics plug-in
- Mozilla browser:
  - native support for MathML from version 0.9.9

Example Software: The **webMathematica** package

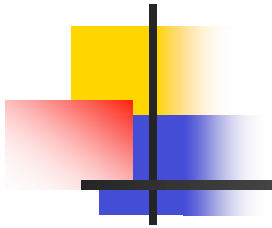


Source: http://www.w3.org/Math/implementations.html

MathML in Standard Web Browsers

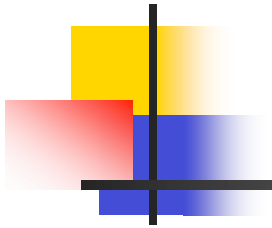


- One of the aims of MathML is for it to be natively render in all standard web browsers using style sheets (CSS and XSL).



Reference

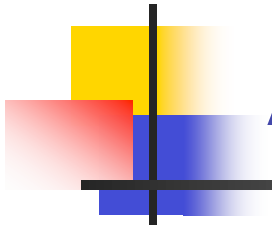
- For an overview of MathML, see
  - MathML technologies
    - http://www.w3.org/Math/
  - MathML Software, including screen shots
    - http://www.w3.org/Math/implementations.html



XML for Bioinformatics

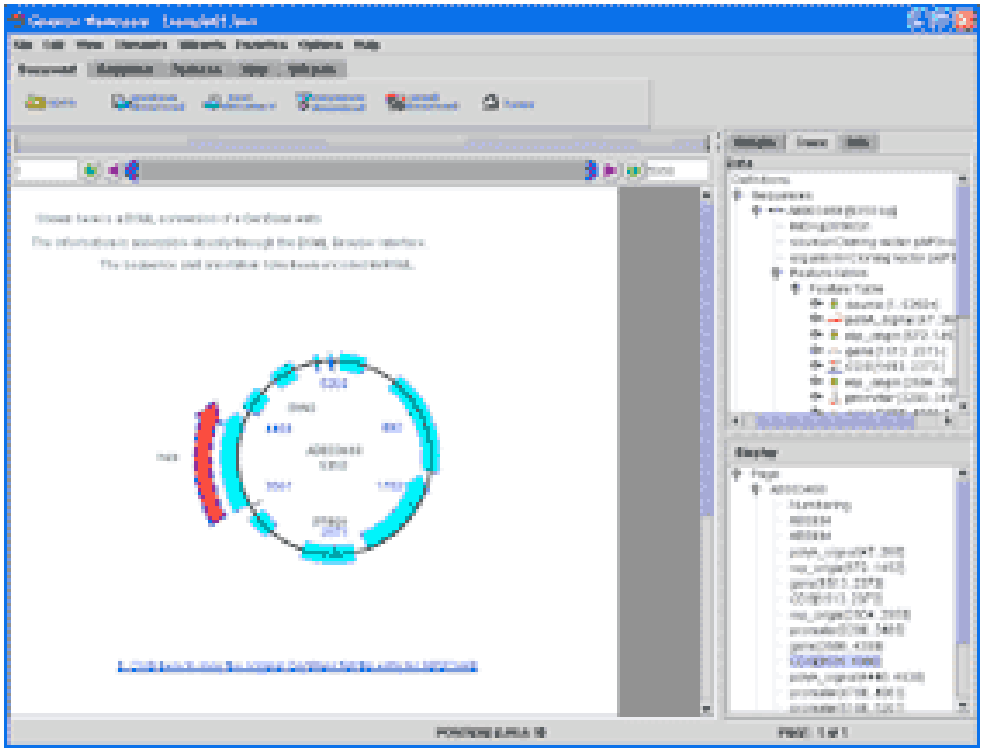
- Bioinformatics is the application of computing to molecular biology
  - mostly in analyzing genomic and genetic data.
- Good data management and processing tools is now critical due to the amount of data coming out of DNA sequencing efforts such as the Human Genome Project.
- XML provides a good platform to formulate these data management solutions.

Some Example XML Applications for Bioinformatics

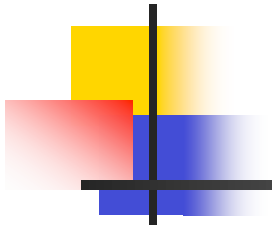


- The following example languages describe information in molecular biology and bioinformatics:
  - BIOpolymer Markup Language (BIOML)
    - for the annotation of biopolymer sequence information.
    - Browser: BioBrow – The BIOML Browser
    - Ref: http://65.219.84.5/BioML.html
  - Bioinformatic Sequence Markup Language (BSML)
    - for the representation of molecular biological data
    - BSML browsers, converters,
    - Ref: http://www.bsml.org/

Example Software: Genomic XML Viewer

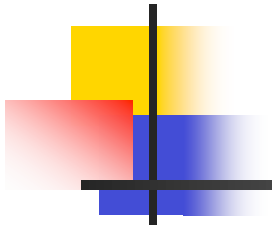


Source:
http://www.bsml.org/download/xmlviewer.asp



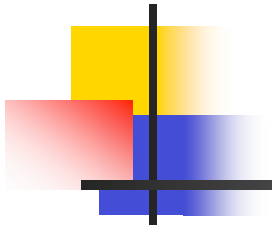
Bio.Perl.Org

- A site and project dedicated to the development of PERL tools for molecular biology computing.
  - Maintains a bio.perl package
- The project is looking to incorporate XML technology into the BIO.PERL toolkit
  - Create new modules based on XML::Parser, XML::Parser.DOM etc which support bioinformatics data manipulation
- Ref: http://bio.perl.org/Projects/XML



Reference

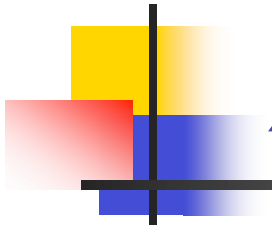
- For an overview of available XML technologies in bioinformatics
  - http://www.xml.com/pub/rg/Bioinformatics



XHTML

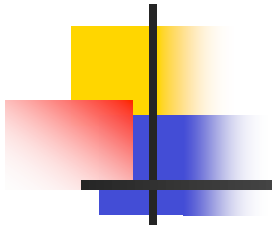
- The eXtensible HTML (XHTML) is W3C's recommendation for the latest version of HTML.

- XHTML 1.0 is a reformulation of HTML 4.01 in XML, and combines the strength of HTML 4 with the power of XML.



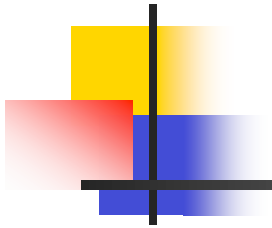
XHTML Syntax

- The syntax for XHTML should be very familiar to you who know HTML. It just adds all the restrictions placed by XML. Eg
  - All opening tags must have a closing tag, or a "/" at the end of the opening tag.
  - No overlapping tags.
  - One root element (the <html> tag).
  - Etc.



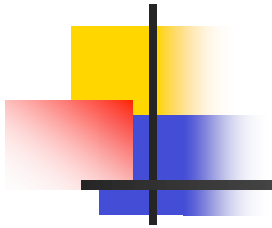
Why XHTML?

- Examples of what you can do with XHTML that you can't do with HTML.
  - Extend XHTML to include new elements for specific purposes.
    - I can choose to create a XHTML document containing a new tag <special_table>. XHTML browsers will be able to handle (it may ignore it if it chooses to), but HTML browsers will not.
  - I can write a parser that extracts non-empty elements in a documents.
    - In HTML, I run into problems because elements like <br> and <hr> are really empty, but it is not properly identified as such.



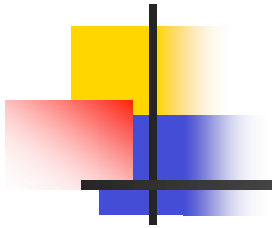
Why XHTML?

- The principal aim of XHTML is to provide richer and more manipulatable Web pages, so that it can be passed to on an ever increasing range of browser platforms including cell phones, televisions, cars, wallet sized wireless communicators, kiosks, and desktops.



Reference

- See Sybex textbook chapters 9-11 for XHTML syntax and use.
- For an overview of XHTML development
  - http://www.w3.org/MarkUp/



Common Threads

- From all these examples, we can see some common threads:
  - The creation of a rich standard description for the data in their problem domains
  - Having a good description mechanism allows people and software to easily exchange information
  - The common XML platform facilitates the creation and sharing of software created to increase the productivity of people in these fields.