XSL

Peter Chapin
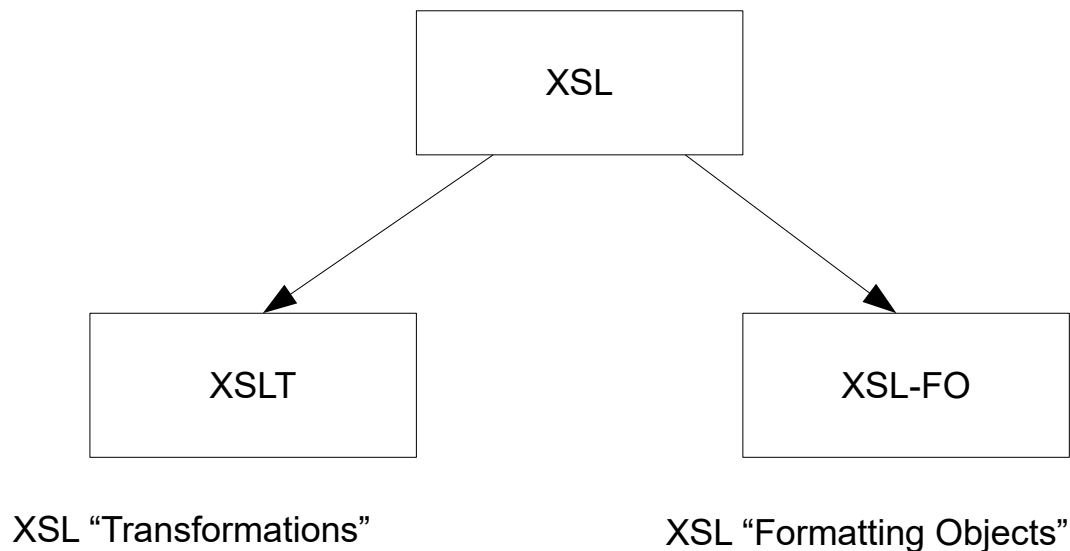Vermont State University

eXtensible Style Language

- Problem...
  - How can one display arbitrary XML languages on the web?
    - Web browsers can't be expected to understand every custom XML language ever created!
- Solution #1...
  - Use CSS.
    - Define CSS styles on every element so the browser knows exactly what to do with each.
    - This is possible but tedious.

Another Way

- Solution #2...
    - Transform an XML language into something the browser does understand: XHTML.
    - This is what XSL is about.
- XSL is quite general.
    - Can be used to transform any XML language to any other.
        - Can be used for document type conversions.
            - OpenDocument format to/from XHTML
            - Recipe Markup Language to Office Open XML (MS2007)
            - etc...

Two Parts to XSL

- XSL is actually two separate, but related standards.

```
                    ┌─────────────┐
                    │     XSL     │
                    └─────────────┘
                     /           \
                    /             \
                   v               v
         ┌─────────────┐   ┌─────────────┐
         │    XSLT     │   │   XSL-FO    │
         └─────────────┘   └─────────────┘

    XSL "Transformations"     XSL "Formatting Objects"
```
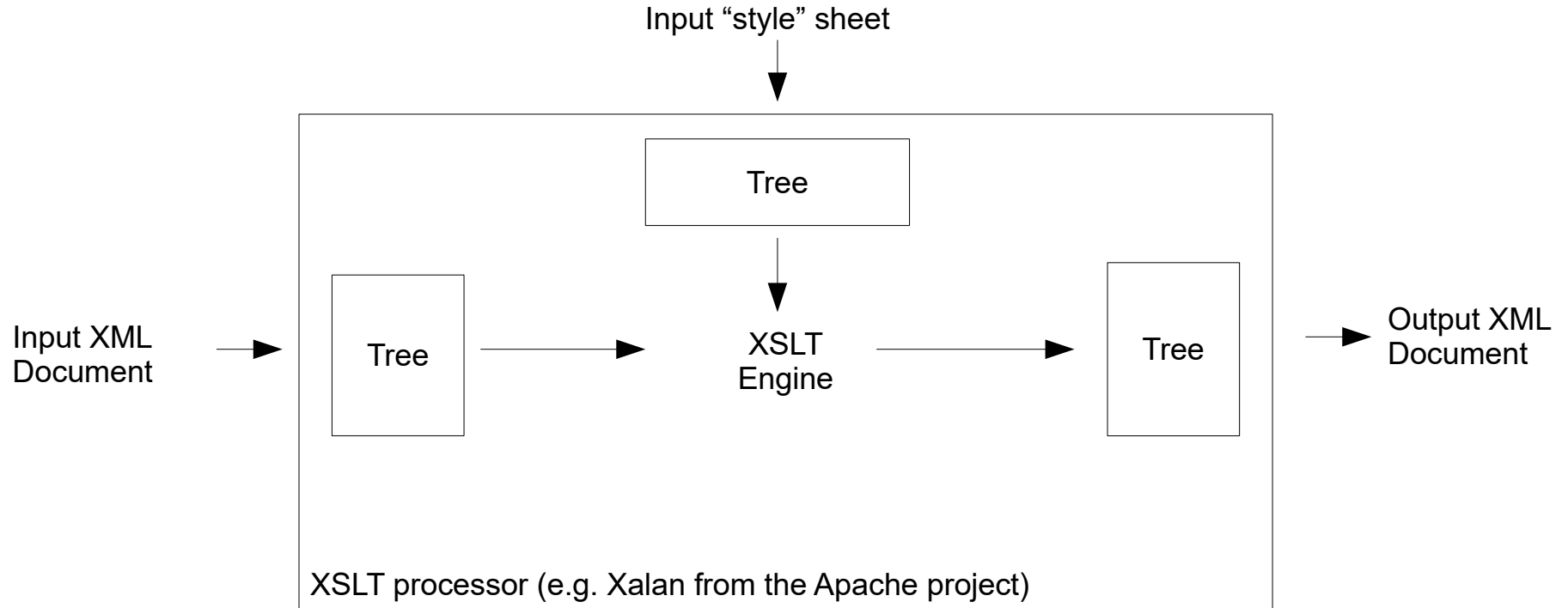
XSL-FO

- An XML language for page layout.
    - Plays a role similar to PostScript or PDF
        - Allows precise specification of a page's appearance.
        - Gives details of fonts, spacing, positioning, etc.
        - BUT... a pure XML language
            - Unlike PostScript or PDF
            - Thus benefits from XML tools and techniques.
    - Conceivably...
        - One could build a printer that prints accepts XSL-FO documents directly.
        - In practice one converts XSL-FO to PDF, etc.
            - See Apache FOP: http://xmlgraphics.apache.org/fop/

XSLT

- An XML language for transformations.
  - Allows conversion of any XML language to another.
    - For example, one might convert XHTML to XSL-FO
    - Gives print quality output from a web page!
  - This is what most people think of when they say "XSL."
- XSLT is powerful.
  - It's a Turing complete functional programming language!
  - Can do arbitrary computation on the input document if necessary to generate the output.

XSLT Block Diagram

Input "style" sheet

Tree

Input XML Document → Tree → XSLT Engine → Tree → Output XML Document

XSLT processor (e.g. Xalan from the Apache project)

XPath

- To understand XSLT you first need XPath.

- XPath is a way to specify particular "nodes" in an XML document.

  - XML documents are trees.

  - Thus one can write paths through these trees

    – Just like with a file system (well, almost)

- XSLT uses XPath to "talk about" parts of the input document.

  - "... copy this part of the input to the output, but skip that part ..."

XML Document Model

- XML Documents are trees.

  - ```
    <recipe-list>
      <recipe>
        <ingredients>
          <ingredient quantity="1"
                      units="cup">Flour</ingredient>
        <ingredients>
        <instructions>Eat</instructions>
      </recipe>
    </recipe-list>
    ```

  - Nodes created for each element.

    – Child nodes created for each child element, attribute, or text content.

  - Root node represents entire document.

    – Has a child node for the root element.