# Dynamic Web Development

Lecture 5

More on XML 2

# *XML Technologies*

There are a range of languages which enable you to process XML documents.

XPATH

XSLT

Lecture 4 - More on XML 1

XSL-FO

XML-DOM

Lecture 5 - More on XML 2

# XSL-FO

## Extensible Stylesheet Language Formatting Objects

XSL-FO describes the formatting of XML data for output to screen, paper or other media.

# XSL-FO

This gives you more control than CSS over the appearance of the document.

CSS solves problems of cross-browser rendering.

XSL-FO is more able to deal with other ways of displaying the document, which require more precision.

It allows you to specify the exact dimensions and margin sizes of a printed page for example.

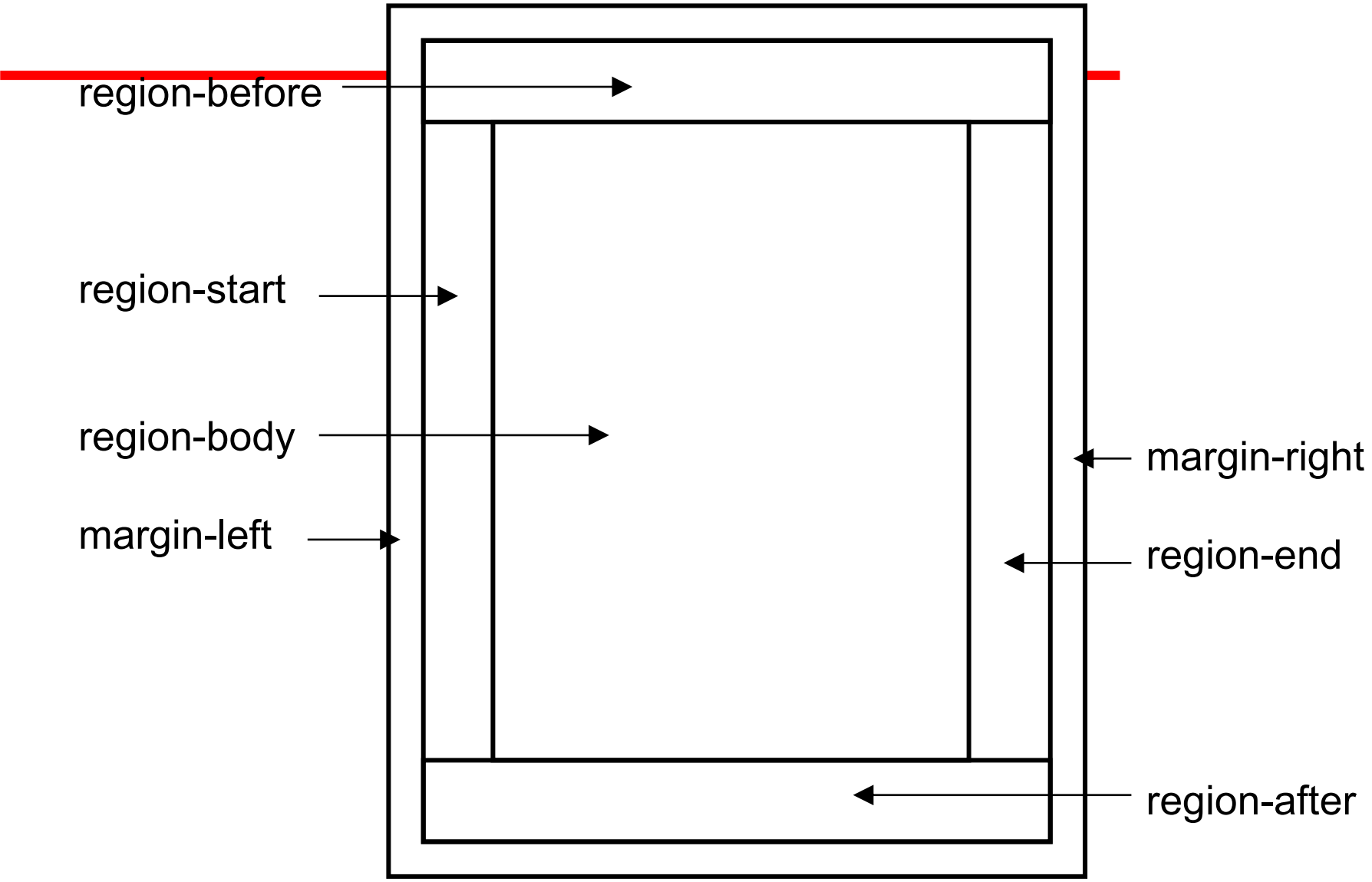
# *Laying out the Master Pages*

You can specify the size of various regions of the page.

You can also specify the start and end region:

English - Left to Right
Hebrew - Right to Left
Chinese - Top to Bottom

You could set up one <simple-page-master> element for the cover, and another for the inside pages.


region-before
region-start
region-body
margin-left
margin-right
region-end
region-after

```
<fo:layout-master-set>
    <fo:simple-page-master master-name="cover"
                           page-width="8.5in"
                           page-height="11in"
                           margin-top="0.5in"
                           margin-bottom="0.5in">
    </fo:simple-page-master>

    <fo:simple-page-master master-name="main"
                           page-width="8.5in"
                           page-height="11in"
                           margin-left="1in"
                           margin-right="1in"
                           margin-top="0.5in"
                           margin-bottom="0.5in">
        <fo:region-body margin-top="0.25in"
                        margin-bottom="0.25in"/>
        <fo:region-before extent="0.25in" />
        <fo:region-after extent="0.25in" />
    </fo:simple-page-master>
</fo:layout-master-set>
```

# *Flowing Content into the Pages*

<fo:page-sequence> elements contain <fo:flow> elements.

These act as containers for content that is allowed to flow from one page to another, depending upon its length

<fo:flow> elements can contain different types of elements. These include:

- <fo:block>
- <fo:block-container>
- <fo:list-block>
- <fo:table>
- <fo:table-and-caption>

# <fo:block>

A block can contain contain a combination or raw text and formatting elements, such as:

- <fo:external-graphic>
- <fo:inline>
- <fo:page-number>
- <fo:footnote>

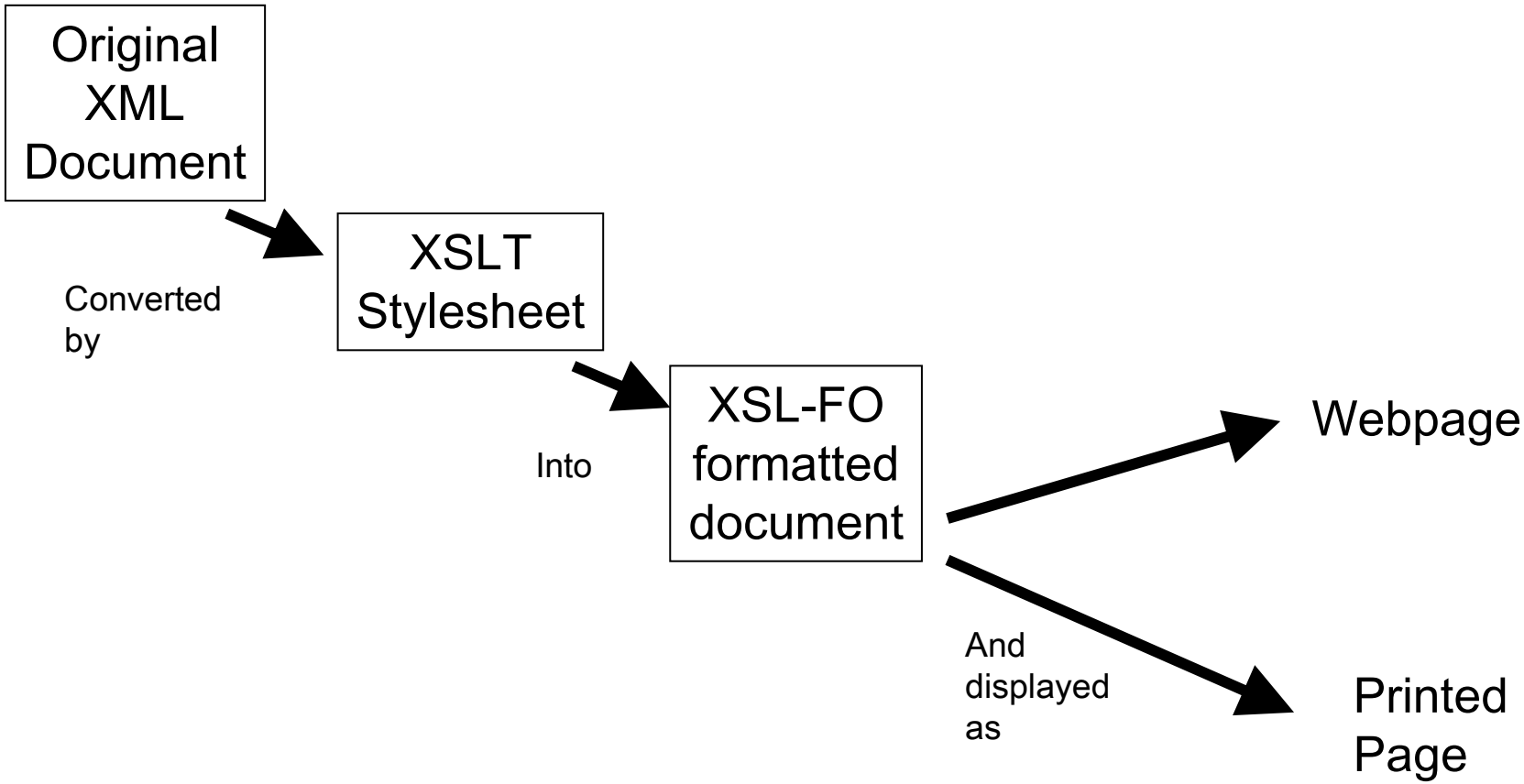
and

other nested <fo:block> elements

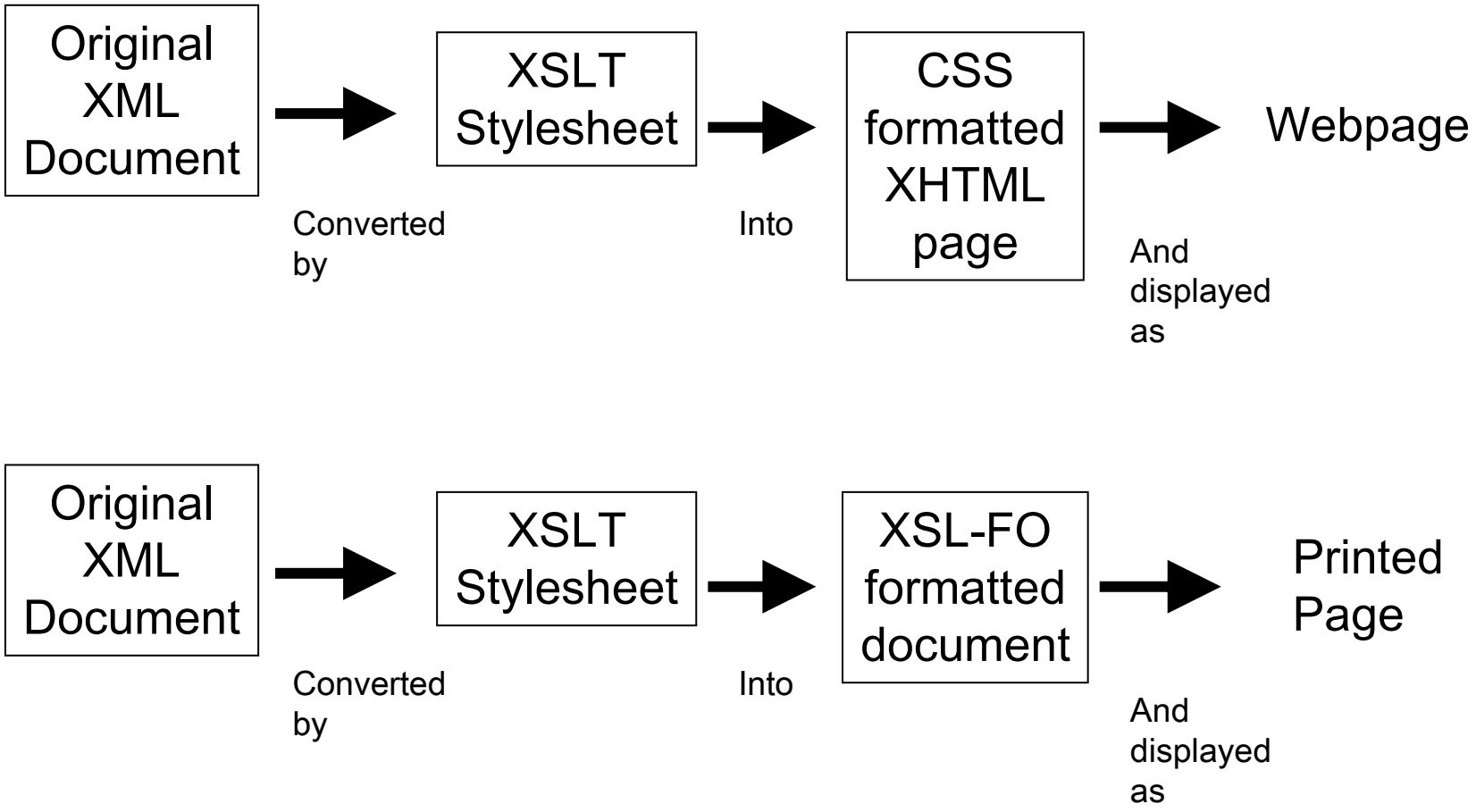
A block has the following attributes:

```
<fo:flow flow-name="xsl-region-body">
      <fo:block font-size="18pt"
                font-family="sans-serif"
                line-height="26pt"
                background-color="black"
                color="white"
                text-align="center"
                padding-top="0pt"
                space-after="5pt">
        Great Sporting Events
      </fo:block>
      <fo:block font-size="12pt"
                font-family="sans-serif"
                line-height="15pt"
                text-align="start"
                space-after="3pt">
        Sports have always seemed to be a large part of my life either
as an athlete participating or as a fan. I was raised in a very
sports- oriented environment.
      </fo:block>
</fo:flow>
```

# What the W3C wanted…



# What has actually happened …



XSL-FO hasn't taken off in the way that was hoped.

It is not yet supported by any web browsers.

It is therefore only used for the specialised role of laying out certain types of printed document.

The current practice is to convert an XSL-FO document into a more widely available format, such as an Adobe pdf file.

There are some commercial products which will carry out this last step.

# XML-DOM

# *What is the DOM?*

DOM stands for Document Object Model.

A way of viewing an XML document as a tree shaped data structure.

This makes it easier to write applications which can access the data within an XML document.

If you wanted to write such an application, what would it have to do?

# The DOM

If an application program is able to view an XML document as a tree structure, it makes it easier to do the following:

- Find and extract data from the document.
- Change data in the document
- Change the structure of the document itself.

# *DOM parsers*

Remember that a application cannot see the overall structure of an XML file.

It must read it in character by character, and slowly build up a picture of the hierarchy of the elements.

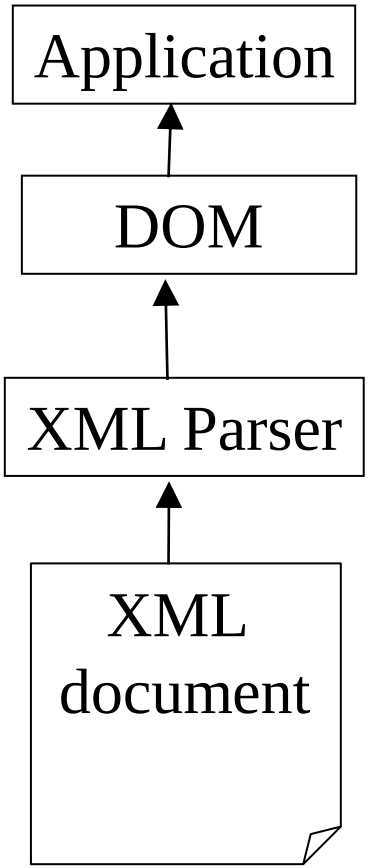
A program called a DOM parser is used to create the tree structure.

# Interfaces

The DOM can be thought of as a layer between the Application and the XML parser.

It provides a set of interfaces that the application can make use of.

These interfaces appear to the programmer as a set of commands that can be used to access the tree structure.



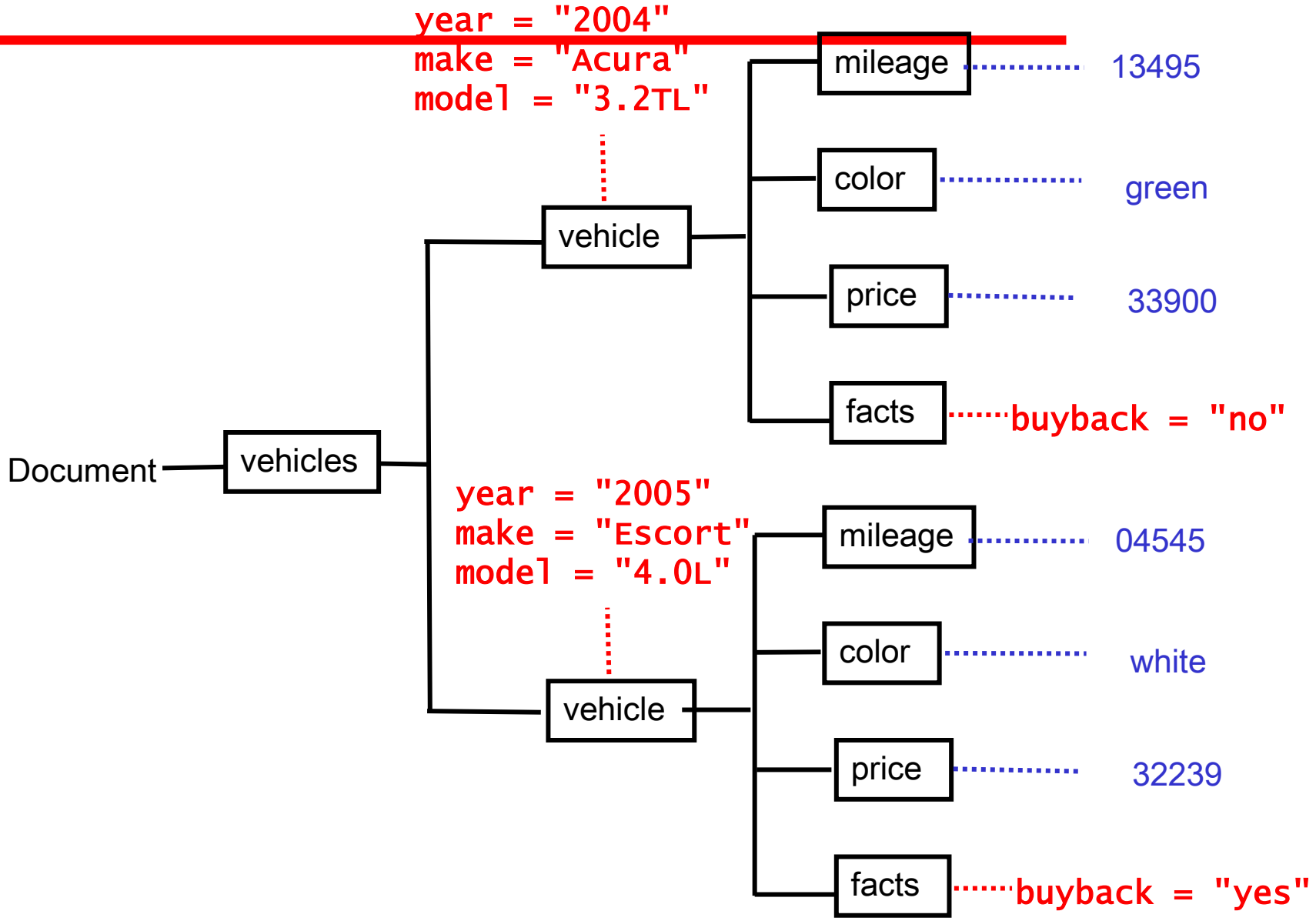
# Vehicles.xml

```
<vehicles>

  <vehicle year="2004" make="Acura" model="3.2TL">
    <mileage>13495</mileage>
    <color>green</color>
    <price>33900</price>
    <carfax buyback="no" />
  </vehicle>

  <vehicle year="2005" make="Escort" model="4.0L">
    <mileage>04545</mileage>
    <color>white</color>
    <price>32239</price>
    <carfax buyback="yes" />
  </vehicle>

</vehicles>
```


year = "2004"
make = "Acura"
model = "3.2TL"
mileage
13495
color
green
vehicle
price
33900
facts
buyback = "no"
Document
vehicles
year = "2005"
make = "Escort"
model = "4.0L"
mileage
04545
color
white
vehicle
price
32239
facts
buyback = "yes"

# *Language Bindings*

A programming language will normally have a set of commands which allow you to access a DOM tree.

This is known as the language binding. It is a standard interface between the language and the DOM tree.

The DOM specification includes bindings for Java and ECMAScript (Another name for Javascript).

Other languages are having them included as time goes on.