

# *Browser Support for XML*

**Microsoft IE5**  
**Netscape Mozilla**

- Difficult to talk about browser support for XML...
  - end up talking about “Microsoft XML” or “Netscape XML” or...
  - *support is very fragmented at the moment*
- We'll look at some interesting features of IE5
  - this is not a “how-to” however...
- Plus a very quick look at Netscape 6/Mozilla
  - as above but more so !!

- XML doesn't stand alone in browsers
  - currently always teamed with HTML-originated technologies
    - *Cascading Style Sheets*
    - *JavaScript (or other scripting technology)*
  - this will gradually change
    - *XML and HTML support will merge into one*
    - *(hopefully...)*

- **IE5:**
  - (is a very different beast to IE4 as far as XML is concerned)
  - fully supports XML version 1.0
  - allows direct viewing of XML with XSL or CSS
    - *CSS support is only partial, however*
  - supports “data islands”
    - *embedded sections of XML in HTML*
  - has namespace support

- supports XML-Schemas
- provides a full DOM-based scripting/ programming interface
- supports Channel Definition Format and Open Software Distribution features
  - *XML applications*
- has a few Microsoft-specific features
  - *‘naturally’...*
  - *behaviours*
    - a technique for separating code and document content
  - *etc.*

- **Mozilla:**

- is Netscape's Open Source browser project
  - *released recently as Netscape 6...*
- is far more standards-oriented
  - *hence doesn't have everything yet ☺*
  - *Netscape gives 65 XML/HTML-related tests where IE5 fails and Mozilla passes*
    - out of 91
    - Netscape says: "...we salute Microsoft's leadership in this area. But..."
- Gecko engine uses XML for its own "skins" architecture
  - *XUL = XML User Interface Language*

- fully supports XML 1.0
- fully supports DOM level 1.0
  - *CSS Object Model and Event Model from level 2*
- supports simple XLink
  - *including transclusion*
- supports elements from the HTML namespace
- supports CSS 1.0 and parts of CSS 2.0
  - *tables, positioning, visual effects*
- supports Resource Definition Framework
  - *for many aspect, from bookmarks to searching*
- doesn't fully support XSL
  - *considers it unready*
  - *recommends you use DOM and scripting*
  - *XSLT can be installed as a module for Mozilla*

*"...how much...compatibility does XML offer?...the answer is both simple and complex.*

*The simple answer is that compatibility can occur on any platform that implements XML according to the published (W3C) standards....there are many XML-based technologies being implemented in current browsers before they have the W3C stamp as recommendations. This is because software companies typically move faster than the W3C. Another difficulty is that as technology advances, it is no longer compatible with the published specifications.*

*This scenario becomes even more complex when different software vendors introduce features and functionality on the software side that do not exist on other platforms."*



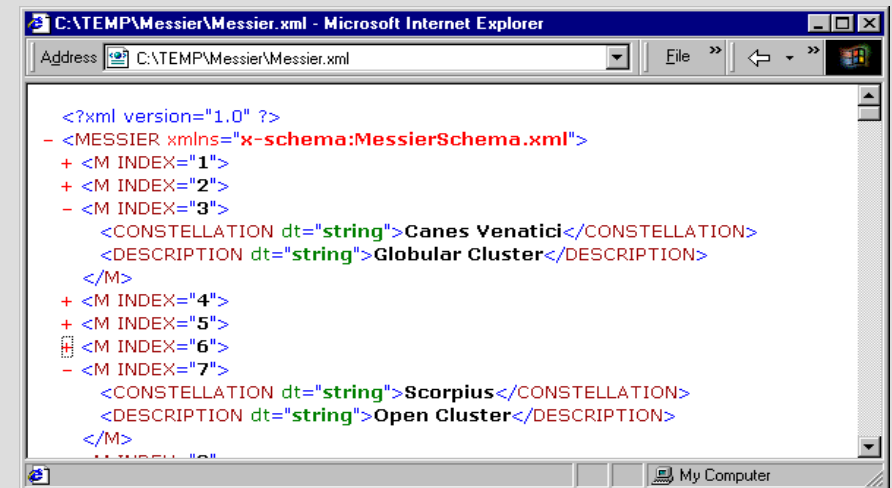
- **...the times, they are a'changin'**
  - and for a while it is going to be difficult to keep abreast of the changes
  - “The message here is that when you are creating XML-based applications or Web pages, it is important to understand clearly what technologies you are using and who your intended audience is.”

- **Data Source Objects**

- Bind XML data to a control on a page

- *much more streamlined than walking a DOM tree or applying XSL/CSS to a document*
    - *can work with data en-masse or with individual nodes*
    - *can also work with an XML data island*

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<HTML>
  <HEAD>
    <TITLE>The Messier Objects</TITLE>
    <SCRIPT FOR="window" EVENT="onload">
      var xmlDso = xmldso.XMLDocument;
      xmlDso.load("Messier.xml");
      countMess.innerHTML = " " + xmlDso.documentElement.childNodes.length + " ";
    </SCRIPT>
  </HEAD>
  <BODY BACKGROUND="M81.gif">
    <OBJECT WIDTH="0" HEIGHT="0" ID="xmldso"
      CLASSID="clsid:550dda30-0541-11d2-9ca9-0060b0ec3d39">
    </OBJECT>
    <H1 STYLE="color:blue">The <SPAN ID="countMess"></SPAN> Messier Objects</H1>
    <TABLE WIDTH="100%">
      <TR>
        <TD>
          <TABLE BORDER="1" WIDTH=75% DATASRC=#xmldso>
            <TR>
              <TD BGCOLOR="lightYellow" STYLE="font-size:18; font-weight:bold">
                M<SPAN DATAFLD="INDEX"></SPAN>
              </TD>
            </TR>
            <TR>
              <TD>
                <TABLE BORDER="1" WIDTH=100%>
                  <THEAD ALIGN="left" BGCOLOR="silver">
                    <TH>Constellation</TH>
                    <TH>Description</TH>
                  </THEAD>
                  <TR ALIGN="left" BGCOLOR="lightBlue">
                    <TD><DIV DATAFLD="CONSTELLATION"></DIV>
                    <TD><DIV DATAFLD="DESCRIPTION"></DIV>
                  </TR>
                </TABLE>
              </TD>
            </TR>
          </TABLE>
        </TD>
      </TR>
    </TABLE>
  </BODY>
</HTML>
```

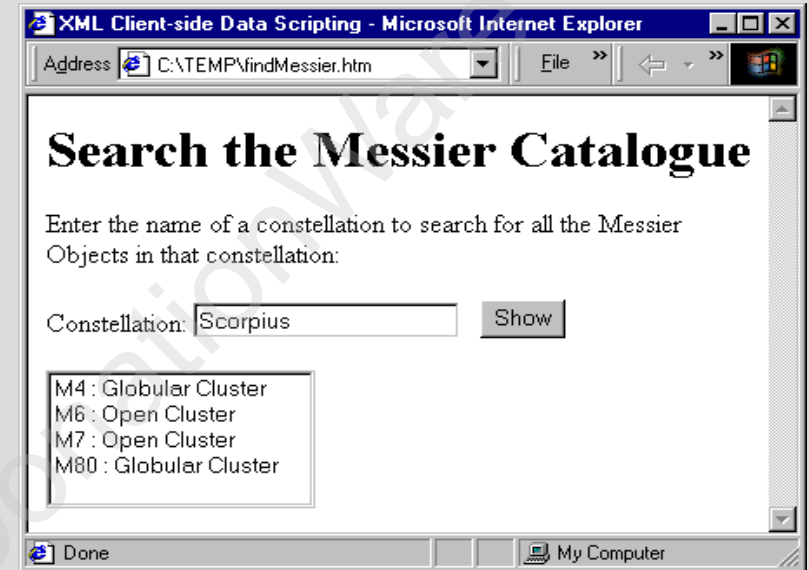


- XML embedded within a page
  - may break the usual separation of data and display
    - *sometimes data is self-contained...*
    - *often very useful...*
  - IE5-specific
  - island could be used as a DSO

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<HTML>
  <HEAD>
    <TITLE>Data Island</TITLE>
  </HEAD>
  <BODY>
    <XML ID="email">
      <EMAIL ENCRYPTED="128" PRIORITY="HIGH">
        <TO>king@asert.com.au</TO>
        <FROM>wizard@asert.com.au</FROM>
        <CC>queen@asert.com.au</CC>
        <SUBJECT>My XML Data Island</SUBJECT>
        <BODY>Isn't life GRAND!</BODY>
      </EMAIL>
    </XML>
    <!-- More Stuff Here -->
  </BODY>
</HTML>
```

- Using a script language (PERL, JavaScript, VBScript, etc.) to perform processing
- The following example shows:
  - JavaScript
  - XML Data Island linked to an external XML file
  - navigating the HTML Document Model
    - *to access the text box and list, etc.*

```
<HTML>
<HEAD>
<TITLE>XML Client-side Data Scripting</TITLE>
<XML ID="dsoMessierCatalog" SRC="Messier.xml"></XML>
<SCRIPT LANGUAGE="JavaScript">
function showAccordingTo()
{
    strCriteria =
        document.forms[0].txtCriteria.value.toLowerCase();
    objListBox = document.forms[0].selListBox;
    objListBox.options.length = 0;
    messierRS = dsoMessierCatalog.recordset;
    messierRS.MoveFirst();
    while (!messierRS.EOF)
    {
        StrConstell = '' + messierRS("CONSTELLATION");
        StrConstell = StrConstell.toLowerCase();
        if (StrConstell.indexOf(strCriteria) >= 0)
        {
            objListBox.options.length += 1;
            objListBox.options[objListBox.options.length - 1].text =
                "M" + messierRS("INDEX") + " : " + messierRS("DESCRIPTION");
        }
        messierRS.MoveNext();
    }
};
</SCRIPT>
</HEAD>
<BODY>
<H1>Search the Messier Catalogue</H1>
Enter the name of a constellation to search for all the Messier
Objects in that constellation:
<FORM>
    Constellation: <INPUT TYPE="TEXT" NAME="txtCriteria"> &nbsp;
    <INPUT TYPE="BUTTON" VALUE="Show" ONCLICK="showAccordingTo()"><P>
    <SELECT NAME="selListBox" SIZE="5">
        <OPTION>Enter a Constellation name...
    </SELECT>
</FORM>
</BODY>
</HTML>
```

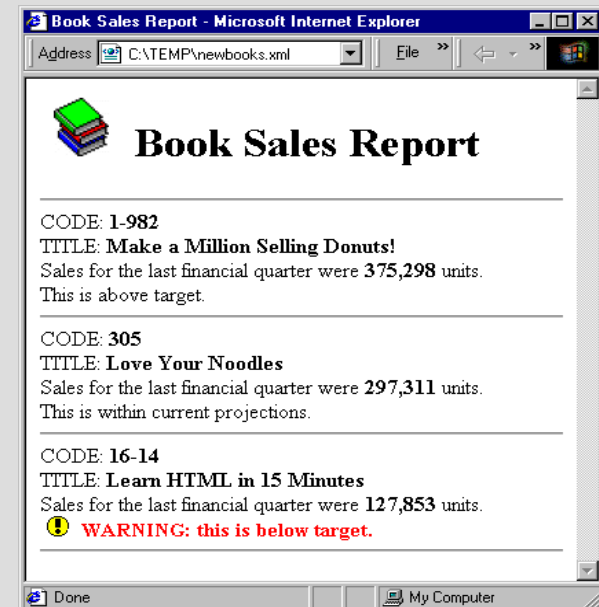


XML + XSL + CSS

- The following example illustrates the complementary nature of these three technologies:
  - XML is used to deliver structured data
  - XSL is used to structure the display
    - *select what to display, etc.*
  - CSS is used to give the displayed output a distinct ‘style’
    - *determine fonts, colours, etc.*

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/TR/WD-xsl">
<xsl:template match="/">
  <HTML>
  <HEAD>
    <TITLE>Book Sales Report</TITLE>
    <STYLE TYPE="text/css">
      .under {font-size:11pt; color:red;
              font-weight:bold}
    </STYLE>
  </HEAD>
  <BODY>
    <H1><IMG SRC="books.gif" ALIGN="BOTTOM" HSPACE="10" />
      Book Sales Report</H1>
    <HR></HR> <!-- make it legal by adding dummy end tag -->
    <xsl:for-each select="//ITEM" order-by="-SALES">
      <xsl:apply-templates select="CODE" />
      <xsl:apply-templates select="TITLE" />
      <xsl:apply-templates select="SALES" />
    </xsl:for-each>
  </BODY>
</HTML>
</xsl:template>
<xsl:template match="*">
  <xsl:node-name />: <B><xsl:value-of /></B><BR />
</xsl:template>
<xsl:template match="SALES">
  Sales for the last financial quarter were<B>
  <xsl:eval>formatNumber(this.text, "#,###,##0")</xsl:eval>
  </B>units.<BR />
  <xsl:choose>
    <xsl:when match="*[. $gt$ 350000]">
      <SPAN>This is above target.</SPAN><BR />
    </xsl:when>
    <xsl:when match="*[. $lt$ 150000]">
      <SPAN CLASS="under">
        <IMG SRC="under.gif" ALIGN="BOTTOM" HSPACE="5" />
        WARNING: this is below target.</SPAN><BR />
      </xsl:when>
    <xsl:otherwise>
      This is within current projections.<BR />
    </xsl:otherwise>
  </xsl:choose>
</xsl:template>
</xsl:stylesheet>
```

```
<?xml version="1.0"?>
<?xml-stylesheet
  type="text/xsl" href="newbooks.xml"?>
<BOOKLIST>
  <ITEM>
    <CODE>1-982</CODE>
    <TITLE>Make a Million Selling Donuts!</TITLE>
    <SALES>375298</SALES>
  </ITEM>
  <ITEM>
    <CODE>305</CODE>
    <TITLE>Love Your Noodles</TITLE>
    <SALES>297311</SALES>
  </ITEM>
  <ITEM>
    <CODE>16-14</CODE>
    <TITLE>Learn HTML in 15 Minutes</TITLE>
    <SALES>127853</SALES>
  </ITEM>
</BOOKLIST>
```





```
<?xml version="1.0"?>
<?xml-stylesheet href="common.css" type="text/css"?>
<?xml-stylesheet href="classic.css" type="text/css"?>
<?xml-stylesheet href="list.css" type="text/css"?>
<!DOCTYPE SearchResult>
<SearchResult xmlns:html="http://www.w3.org/1999/xhtml" xmlns:xlink="http://www.w3.org/1999/xlink">
  <Header>
    <Toolbar>
      <html:form>
        <Label>Sort by:</Label>
        <html:input type="button" id="Author" onclick="sort(books,this.id)" value="Author"/>
        <html:input type="button" id="Title" onclick="sort(books,this.id)" value="Title"/>
        <html:input type="button" id="ISBN" onclick="sort(books,this.id)" value="ISBN #"/>
        <html:input type="button" id="toggle" value="Toggle Style" onclick="initiateToggle(); return false;"/>
      </html:form>
    </Toolbar>
    <SearchTitle>Amazon.com Search Results for: the keywords include "Road"</SearchTitle>
  </Header>
  <BookSet>
    <Book>
      <BookCover>
        <html:img src="kerouac.gif" width="100" height="140"/>
      </BookCover>
      <Title xlink:type="simple" xlink:show="replace"
xlink:href="http://www.amazon.com/exec/obidos/ASIN/0140042598/002-5969498-2733628">On the Road</Title>
      <Author>Kerouac, Jack</Author>
      <ISBN>0140042598</ISBN>
      <Synopsis>
        On The Road, the most famous of Jack Kerouac's works, is not only the soul of the Beat movement and literature, but
        one of the most important novels of the century. Like nearly all of Kerouac's writing, On The Road is thinly
        fictionalized autobiography, filled with a cast made of Kerouac's real life friends, lovers, and fellow travelers.
        Narrated by Sal Paradise, one of Kerouac's alter-egos, On the Road is a cross-country bohemian odyssey that not only
        influenced writing in the years since its 1957 publication but penetrated into the deepest levels of American thought
        and culture.
      </Synopsis>
      <ListPrice>$12.95</ListPrice>
      <Price>$10.36</Price>
    </Book>
    <Book>
      [...]
    </Book>
  </BookSet>
  <html:script src="books.js"/>
</SearchResult>
```

Note the xlink!!

# Netscape 6 & XML...

```
<html>
<body>
<pre>
```

```
BookSet {
    display: block;
}
```

```
Book {
    float: left;
    display: block;
    width: 300px;
    background-color: #f1f1f1;
    border: 1px solid #cccccc;
    margin-right: 8px;
    margin-top: 8px;
}
```

```
BookCover {
    display: inline;
    float: left;
    margin-right: 10px;
    padding: 5px;
    width: 100px;
    height: 140px;
}
```

```
Title {
    display: block;
    font-weight: bold;
    color: blue;
    text-decoration: underline;
    cursor: pointer;
}
```

```
Author {
    display: block;
    font-style: italic;
}
```

```
Synopsis {
    display: block;
    overflow: scroll;
    height: 100px;
    width: 155px;
    padding: 3px;
    background-color: #ddddff;
}
```

```
ListPrice {
    display: block;
    text-align: right;
    padding-right: 15px;
    text-decoration: line-through;
}
```

```
Price {
    display: block;
    color: rgb(20,100,0);
    text-align:right;
    padding-right: 15px;
    font-weight: bold;
}
```

```
Price:before {
    content: "Our Price: ";
}
```

```
ISBN {
    display: block;
    font-family: monospace;
    font-size: 8pt;
}
```

```
</pre>
</body>
</html>
```

classic.css

# Netscape 6 & XML...

```
<html>
<body>
<pre>
```

```
BookSet {
    display: block;
}
```

```
Book {
    display: list-item;
    float: none;
    border: none;
    background: inherit;
}
```

```
BookCover {
    display: none;
}
```

```
Title {
    display: block;
    font-weight: bold;
    color: blue;
    text-decoration: underline;
    cursor: pointer;
}
```

```
Author {
    display: block;
    font-style: italic;
}
```

```
Synopsis {
    display: none;
}
```

```
ListPrice {
    display: none;
    text-align: right;
    padding-right: 15px;
    text-decoration: line-through;
}
```

```
Price {
    display: block;
    color: rgb(20,100,0);
    font-weight: bold;
    text-align:left;
}
```

```
Price:before {
    content: "Our Price: ";
}
```

```
ISBN {
    display: block;
    font-family: monospace;
    font-size: 8pt;
}
```

```
</pre>
</body>
</html>
```

list.css

# Netscape 6 & XML...

```
<html>
<body>
<pre>
```

```
// An inefficient, but effective bubble sort
function sort(collection, key)
```

```
{
  var i, j;
  var count = collection.length;
  var parent, child;

  for (i = count-1; i >= 0; i--) {
    for (j = 1; j <= i; j++) {
      if (collection[j-1][key] > collection[j][key]) {
        // Move the item both in the local array and
        // in the tree
        child = collection[j];
        parent = child.parentNode;

        collection[j] = collection[j-1];
        collection[j-1] = child;

        parent.removeChild(child);
        parent.insertBefore(child, collection[j]);
      }
    }
  }
}
```

```
// XXX This is a workaround for a bug where
// changing the disabled state of a stylesheet can't
// be done in an event handler. For now, we do it
// in a zero-delay timeout.
```

```
function initiateToggle()
{
  setTimeout(toggleStyleSheet, 0);
}
```

```
// Set user properties on the nodes in the collection
// based on information found in its children. For example,
// make a property "AUTHOR" based on the content of the
// "Author" element found in the childNode list of the node.
// This makes later sorting more efficient
function collectInfo(nodes, propNames)
```

```
{
  var i, j, k;
  var ncount = nodes.length;
  var pcount = propNames.length;

  for (i = 0; i < ncount; i++) {
    var node = nodes[i];
    var childNodes = node.childNodes;
    var ccount = childNodes.length;

    for (j = 0; j < ccount; j++) {
      var child = childNodes[j];

      if (child.nodeType == Node.ELEMENT_NODE) {
        var tagName = child.tagName;

        for (k = 0; k < pcount; k++) {
          var prop = propNames[k];
          if (prop == tagName) {
            node[prop] = child.firstChild.data;
          }
        }
      }
    }
  }
}
```

```
var enabled = true;
function toggleStyleSheet()
{
  if (enabled) {
    document.styleSheets[2].disabled = true;
  }
  else {
    document.styleSheets[2].disabled = false;
  }
}
```

```
enabled = !enabled;
}
```

```
var sortableProps = new Array("Author", "Title", "ISBN");
var books = new Array();
```

```
// We upcase the tagName as a workaround for a bug
// that loses the original case of the tag.
var bookset =
document.getElementsByTagName("Book");
```

```
// We need to create a "non-live" array to operate on.
Since
// we'll be moving things around in this array, we can't use
// the read-only, live one returned by
// getElementsByTagName.
for (var i=0; i < bookset.length; i++) {
  books[i] = bookset[i];
}
```

```
collectInfo(books, sortableProps);
```

```
</pre>
</body>
</html>
```

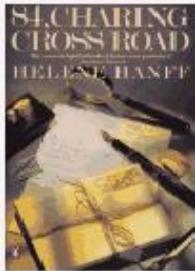

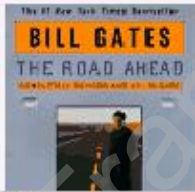
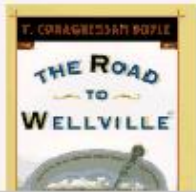
Amazon.com: buying info: The Road Ahead (CD ROM included) - Netscape 6

File Edit View Search Go Bookmarks Tasks Help

Back Forward Reload Stop http://www.mozilla.org/newlay Search Print

**Amazon.com Search Results**  
for the keywords include "Road"

Sort by: Author Title ISBN #  
Toggle Style

 <p><b>84 Charing Cross Road</b> <i>Hanff, Helene</i> 0140143505</p> <p>84, Charing Cross Road is a charming record of bibliophilia, cultural difference, and imaginative sympathy.</p> <p>\$9.95 <b>Our Price: \$7.96</b></p>	 <p><b>AAA Road Atlas 1999</b> <i>A. A. A.</i> 1562512625</p> <p>The definitive road atlas to the United States, Canada and Mexico.</p> <p>\$9.95 <b>Our Price: \$9.95</b></p>
 <p><b>Bethlehem Road</b> <i>Perry, Anne</i> 0449219143</p> <p>He might be elegant, but there's no mistaking it--the gentleman tied to the lamp post on...</p> <p>\$6.99 <b>Our Price: \$5.59</b></p>	 <p><b>On the Road</b> <i>Kerouac, Jack</i> 0140042598</p> <p>On The Road, the most famous of Jack Kerouac's works, is not only the soul of the Post movement and...</p> <p>\$12.95 <b>Our Price: \$10.36</b></p>
 <p><b>The Road Ahead</b> <i>Gates, Bill</i> 666666666</p> <p>In a study, the founder of Microsoft presents his vision for the future.</p> <p>Document: Done</p>	 <p><b>The Road to Wellville</b> <i>Boyle, T. Coraghessan</i> 0140167188</p> <p>A snobbish wife and her henpecked husband travel to Dr...</p>

Business Tech Fun Interact

Amazon.com: buying info...

File Edit View Search Go Book

Back Forward Reload Stop

**Amazon.com Search Results**  
for the keywords include "Road"

**On the Road**  
*Kerouac, Jack*  
0140042598  
**Our Price: \$10.36**

**84 Charing Cross Road**  
*Hanff, Helene*  
0140143505  
**Our Price: \$7.96**

**The Road to Wellville**  
*Boyle, T. Coraghessan*  
0140167188  
**Our Price: \$10.36**

**Bethlehem Road**  
*Perry, Anne*  
0449219143  
**Our Price: \$5.59**

**AAA Road Atlas 1999**  
*A. A. A.*  
1562512625  
**Our Price: \$9.95**

**The Road Ahead**  
*Gates, Bill*  
666666666  
**Our Price: \$12.76**

Document: Done

Business Tech