

XSLT - 在服务器端

由于并非所有的浏览器都支持 XSLT，另一种解决方案是在服务器上完成 XML 至 XHTML 的转化。

跨浏览器解决方案

在前面的章节，我们讲解过如何在浏览器中使用 XSLT 来完成 XML 到 XHTML 的转化。我们创建了一段使用 XML 解析器来进行转换的 JavaScript。JavaScript 解决方案无法工作于没有 XML 解析器的浏览器。

为了让 XML 数据适用于任何类型的浏览器，我们必须在服务器上对 XML 文档进行转换，然后将其作为 XHTML 发送回浏览器。

这是 XSLT 的另一个优点。XSLT 的设计目标之一是使数据在服务器上从一种格式转换到另一种格式成为可能，并向所有类型的浏览器返回可读的数据。

XML 文件和 XSLT 文件

请看这个在前面的章节已展示过的 XML 文档：

```
<?xml version="1.0" encoding="UTF-8"?>
<catalog>
  <cd>
    <title>Empire Burlesque</title>
    <artist>Bob Dylan</artist>
    <country>USA</country>
    <company>Columbia</company>
    <price>10.90</price>
    <year>1985</year>
  </cd>
  .
  .
</catalog>
```

[查看 XML 文件。](#)

以及附随的 XSL 样式表：

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

  <xsl:template match="/">
    <h2>My CD Collection</h2>
    <table border="1">
      <tr bgcolor="#9acd32">
        <th style="text-align:left">Title</th>
```

```
<th style="text-align:left">Artist</th>
</tr>
<xsl:for-each select="catalog/cd">
<tr>
  <td><xsl:value-of select="title" /></td>
  <td><xsl:value-of select="artist" /></td>
</tr>
</xsl:for-each>
</table>
</xsl:template>

</xsl:stylesheet>
```

[查看 XSL 文件。](#)

请注意，这个 XML 文件没有包含对 XSL 文件的引用。

重要事项：上面这句话意味着，XML 文件可使用多个不同的 XSL 样式表来进行转换。

在服务器把 XML 转换为 XHTML

这是用于在服务器上把 XML 文件转换为 XHTML 的源代码：

使用 PHP 代码转换：

```
<?php
// 载入 XML 文件
$xml = new DOMDocument;
$xml->load('cdcatalog.xml');

// 载入 XSL 文件
$xsl = new DOMDocument;
$xsl->load('cdcatalog.xsl');

// 设置转换
$proc = new XSLTProcessor;

// 添加 xsl 规则
$proc->importStyleSheet($xsl);

echo $proc->transformToXML($xml);
?>
```

提示：假如您不了解如何编写 php，您可以学习我们的 [PHP 教程](#)。

使用 ASP 代码转换：

```
<%
'载入 XML 文件
```

```
set xml = Server.CreateObject("Microsoft.XMLDOM")
xml.async = false
xml.load(Server.MapPath("cdcatalog.xml"))

'载入 XSL 文件
set xsl = Server.CreateObject("Microsoft.XMLDOM")
xsl.async = false
xsl.load(Server.MapPath("cdcatalog.xsl"))

'转换文件
Response.Write(xml.transformNode(xsl))

%>
```

提示：假如您不了解如何编写 ASP，您可以学习我们的 [ASP 教程](#)。

第一段代码创建了微软的 XML 解析器（XMLDOM）的一个实例，并把 XML 文件载入了内存。第二段代码创建了解析器的另一个实例，并把这个 XSL 文件载入了内存。最后一行代码使用 XSL 文档转换了 XML 文档，并把结果作为 XHTML 发送到您的浏览器。太好了！

[它是如何工作的。](#)

[← XSLT 在客户端](#)

[XSLT – 编辑 XML →](#)

[点我分享笔记](#)