<http://blog.csdn.net/carefree31441/article/details/3235849>

"xmlns"是XHTML namespace的缩写，叫做"名字空间"声明。名字空间是什么作用呢？我的理解是：

由于xml允许你自己定义自己的标识，你定义的标识和其他人定义的标识有可能相同，但表示不同的意义。当文件交换或者共享的时候就容易产生错误。为了避免这种错误发生，XML采用名字空间声明，允许你通过一个网址指向来识别你的标识。

比如在一个文档<table>wood table</table>中<table>表示桌子，而在另一个文档<table>namelist</table>中<table>表示表格。如果我需要同时处理这两个文档，就会发生名字冲突。为了解决这个问题，我们引进了namespaces这个概念。namespaces通过给标识名称加一个网址(URL)定位的方法来区别这些名称相同的标识。   
Namespaces同样需要在XML文档的开头部分声明，声明的语法如下：   
<document xmlns:yourname='URL'>   
其中yourname是由你定义的namespaces的名称，URL就是名字空间的网址。   
假设上面的"桌子<table>"文档来自http://www.zhuozi.com，我们就可以声明为   
<document xmlns:zhuozi='http://www.zhuozi.com'>;   
然后在后面的标识中使用定义好的名字空间：   
<zhuozi:table>wood table</table>   
这样就将这两个<table>区分开来。注意的是：设置URL并不是说这个标识真的要到那个网址去读取，仅仅作为一种区别的标志而已。

**命名冲突**请看下面两个例子

**这个 XML 文档在table元素中携带了水果的信息:**

**<table>  
<tr>  
<td>Apples</td>  
<td>Bananas</td>  
</tr>  
</table>**

**这个 XML 文档在table元素中携带了桌子的信息(家具，不能吃的哦):**

**<table>  
<name>African Coffee Table</name>  
<width>80</width>  
<length>120</length>  
</table>**

**如果上面两个XML文档片断碰巧在一起使用的话，那么将会出现命名冲突的情况。因为这两个片断都包含了<table>元素，而这两个table元素的定义与所包含的内容又各不相同。**

**--------------------------------------------------------------------------------**

**使用前缀解决命名冲突问题  
下面的XML文档在table元素中携带了信息:**

**<h:table>  
<h:tr>  
<h:td>Apples</h:td>  
<h:td>Bananas</h:td>  
</h:tr>  
</h:table>**

**下面的XML文档携带了家具table的信息:**

**<f:table>  
<f:name>African Coffee Table</f:name>  
<f:width>80</f:width>  
<f:length>120</f:length>  
</f:table>**

**现在已经没有元素命名冲突的问题了，因为这两个文档对各自的table元素使用了不同的前缀，table元素在两个文档中分别是(<h:table> 和<f:table>)。**

**通过使用前缀，我们创建了两个不同的table元素。**

**--------------------------------------------------------------------------------**

**使用命名空间  
下面的XML文档在table元素中携带了信息:**

**<h:table xmlns:h="http://www.w3.org/TR/html4/">**  
**<h:tr>  
<h:td>Apples</h:td>  
<h:td>Bananas</h:td>  
</h:tr>  
</h:table>**

**下面的XML文档携带了家具table的信息:**

**<f:table xmlns:f="http://www.w3schools.com/furniture">**  
**<f:name>African Coffee Table</f:name>  
<f:width>80</f:width>  
<f:length>120</f:length>  
</f:table>**

**在上面两个例子中除了使用前缀外，两个table元素都使用了xmlns属性，使元素和不同的命名空间关联到一起。**

**--------------------------------------------------------------------------------**

**命名空间属性  
命名空间属性一般放置在元素的开始标记处，其使用语法如下所示:**

**xmlns:namespace-prefix="namespace"**

**在上面的例子中，命名空间定义了一个Internet 地址:**

**xmlns:f="**[**http://www.w3schools.com/furniture"**](http://www.w3schools.com/furniture%22)

**W3C 命名规范声明命名空间本身就是一个统一资源标示符，Uniform Resource Identifier (URI)。**

**当我们在元素的开始标记处使用命名空间时，该元素所有的子元素都将通过一个前缀与同一个命名空间相互关联。**

**注意：用来标识命名空间的网络地址并不被XML解析器调用，XML解析器不需要从这个网络地址中查找信息，该网络地址的作用仅仅是给命名空间一个唯一的名字，因此这个网络地址也可以是虚拟的，然而又很多公司经常把这个网络地址值象一个真实的Web页面，这个地址包含了关于当前命名空间更详细的信息。  
可以访问**[**http://www.w3.org/TR/html4/.**](http://www.w3.org/TR/html4/)

**--------------------------------------------------------------------------------**

**统一资源标识符  
通用资源标识符（A Uniform Resource Identifier (URI)）是一个标识网络资源的字符串。最普通的URI应该是统一资源定位符Uniform Resource Locator (URL)。URL用于标识网络主机的地址。另一方面，另一个不常用的URI是通用资源名字Universal Resource Name (URN)。在我们的例子中，一般使用的是URLs。**

**既然前面的例子使用的URL地址来标识命名空间，我们可以确信这个命名空间是唯一的。**

**--------------------------------------------------------------------------------**

**默认的命名空间  
定义一个默认的XML命名空间使得我们在子元素的开始标记中不需要使用前缀。他的语法如下所示:**

**<element xmlns="namespace">**

**下面的XML文档在table元素中包含了水果的信息:**

**<table xmlns="**[**http://www.w3.org/TR/html4/">**](http://www.w3.org/TR/html4/%22%3E)  
**<tr>  
<td>Apples</td>  
<td>Bananas</td>  
</tr>  
</table>**

**下面的XML文档包含了家具table的信息:**

**<table xmlns="**[**http://www.w3schools.com/furniture">**](http://www.w3schools.com/furniture%22%3E)  
**<name>African Coffee Table</name>  
<width>80</width>  
<length>120</length>  
</table>**

**--------------------------------------------------------------------------------**

**使用命名空间  
档开始使用XSL的时候,就会发现命名空间使用的是如此频繁。XSL样式单主要用于将XML文档转换成类似于HTML文件的格式。**

**如果看一下下面的XSL文档，就会发现有很多标记都是HTML标记。那些标记并不是HTML标记，是加了前缀的XSL,这个XSL前缀由命名空间"**[**http://www.w3.org/TR/xsl"**](http://www.w3.org/TR/xsl%22)**所标识:**

**<?xml version="1.0" encoding="ISO-8859-1"?>  
<xsl:stylesheet xmlns:xsl="**[**http://www.w3.org/TR/xsl">**](http://www.w3.org/TR/xsl%22%3E)  
**<xsl:template match="/">  
<html>  
<body>  
<table border="2" bgcolor="yellow">  
<tr>  
<th>Title</th>  
<th>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>  
</body>  
</html>  
</xsl:template>  
</xsl:stylesheet>**

版权声明：本文为博主原创文章，未经博主允许不得转载。