

XSD 限定 / Facets

限定 (restriction) 用于为 XML 元素或者属性定义可接受的值。对 XML 元素的限定被称为 facet。

对值的限定

下面的例子定义了带有一个限定且名为 "age" 的元素。age 的值不能低于 0 或者高于 120：

```
<xs:element name="age">
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="120"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

对一组值的限定

如需把 XML 元素的内容限制为一组可接受的值，我们要使用枚举约束 (enumeration constraint)。

下面的例子定义了带有一个限定的名为 "car" 的元素。可接受的值只有：Audi, Golf, BMW：

```
<xs:element name="car">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Audi"/>
      <xs:enumeration value="Golf"/>
      <xs:enumeration value="BMW"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

上面的例子也可以被写为：

```
<xs:element name="car" type="carType"/>

<xs:simpleType name="carType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Audi"/>
    <xs:enumeration value="Golf"/>
    <xs:enumeration value="BMW"/>
  </xs:restriction>
</xs:simpleType>
```

注意：在这种情况下，类型 "carType" 可被其他元素使用，因为它不是 "car" 元素的组成部分。

对一系列值的限定

如需把 XML 元素的内容限制定义为一组可使用的数字或字母，我们要使用模式约束 (pattern constraint)。

下面的例子定义了带有一个限定的名为 "letter" 的元素。可接受的值只有小写字母 a - z 其中的一个：

```
<xs:element name="letter">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="[a-z]"/>
    </xs:restriction>
  </xs:simpleType>
```

```
</xs:restriction>
</xs:simpleType>
</xs:element>
```

下一个例子定义了带有一个限定的名为 "initials" 的元素。可接受的值是大写字母 A - Z 其中的三个：

```
<xs:element name="initials">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="[A-Z][A-Z][A-Z]" />
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

下一个例子也定义了带有一个限定的名为 "initials" 的元素。可接受的值是大写或小写字母 a - z 其中的三个：

```
<xs:element name="initials">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="[a-zA-Z][a-zA-Z][a-zA-Z]" />
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

下一个例子定义了带有一个限定的名为 "choice" 的元素。可接受的值是字母 x, y 或 z 中的一个：

```
<xs:element name="choice">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="[xyz]" />
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

下一个例子定义了带有一个限定的名为 "prodid" 的元素。可接受的值是五个阿拉伯数字的一个序列，且每个数字的范围是 0-9：

```
<xs:element name="prodid">
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:pattern value="[0-9][0-9][0-9][0-9][0-9]" />
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

对一系列值的其他限定

下面的例子定义了带有一个限定的名为 "letter" 的元素。可接受的值是 a - z 中零个或多个字母：

```
<xs:element name="letter">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="([a-z])*" />
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

下面的例子定义了带有一个限定的名为 "letter" 的元素。可接受的值是一对或多对字母，每对字母由一个小写字母后跟一个大写字母组成。举个例子，"sToP"将会通过这种模式的验证，但是 "Stop"、"STOP" 或者 "stop" 无法通过验证：

```
<xs:element name="letter">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="([a-z][A-Z])+" />
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

```
</xs:restriction>
</xs:simpleType>
</xs:element>
```

下面的例子定义了带有一个限定的名为 "gender" 的元素。可接受的值是 male 或者 female :

```
<xs:element name="gender">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="male|female"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

下面的例子定义了带有一个限定的名为 "password" 的元素。可接受的值是由 8 个字符组成的一行字符，这些字符必须是大写或小写字母 a - z 亦或数字 0 - 9 :

```
<xs:element name="password">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="[a-zA-Z0-9]{8}"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

对空白字符的限定

如需规定对空白字符 (whitespace characters) 的处理方式，我们需要使用 whiteSpace 限定。

下面的例子定义了带有一个限定的名为 "address" 的元素。这个 whiteSpace 限定被设置为 "preserve"，这意味着 XML 处理器不会移除任何空白字符：

```
<xs:element name="address">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:whiteSpace value="preserve"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

这个例子也定义了带有一个限定的名为 "address" 的元素。这个 whiteSpace 限定被设置为 "replace"，这意味着 XML 处理器将移除所有空白字符（换行、回车、空格以及制表符）：

```
<xs:element name="address">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:whiteSpace value="replace"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

这个例子也定义了带有一个限定的名为 "address" 的元素。这个 whiteSpace 限定被设置为 "collapse"，这意味着 XML 处理器将移除所有空白字符（换行、回车、空格以及制表符会被替换为空格，开头和结尾的空格会被移除，而多个连续的空格会被缩减为一个单一的空格）：

```
<xs:element name="address">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:whiteSpace value="collapse"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

```
</xs:simpleType>
</xs:element>
```

对长度的限定

如需限制元素中值的长度，我们需要使用 length、maxLength 以及 minLength 限定。

本例定义了带有一个限定且名为 "password" 的元素。其值必须精确到 8 个字符：

```
<xs:element name="password">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:length value="8"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

这个例子也定义了带有一个限定的名为 "password" 的元素。其值最小为 5 个字符，最大为 8 个字符：

```
<xs:element name="password">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:minLength value="5"/>
      <xs:maxLength value="8"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

数据类型的限定

限定	描述
enumeration	定义可接受值的一个列表
fractionDigits	定义所允许的最大的小数位数。必须大于等于0。
length	定义所允许的字符或者列表项目的精确数目。必须大于或等于0。
maxExclusive	定义数值的上限。所允许的值必须小于此值。
maxInclusive	定义数值的上限。所允许的值必须小于或等于此值。
maxLength	定义所允许的字符或者列表项目的最大数目。必须大于或等于0。
minExclusive	定义数值的下限。所允许的值必需大于此值。
minInclusive	定义数值的下限。所允许的值必需大于或等于此值。
minLength	定义所允许的字符或者列表项目的最小数目。必须大于或等于0。
pattern	定义可接受的字符的精确序列。
totalDigits	定义所允许的阿拉伯数字的精确位数。必须大于0。

whiteSpace	定义空白字符（换行、回车、空格以及制表符）的处理方式。
← XML Schema 属性	XML Schema 复合元素 →
<div>点我分享笔记</div>	