

XPath 约束

1 简介

约束可添加到任何 Xpath 查询中，以筛选检索的数据。它应始终呈现为有效表达式形式。该表达式应由一个或多个变量与运算符、函数、关键字或系统变量组合而成。

例如，该查询将检索名字等于 Jansen 的所有客户：

```
//Sales.Customer[Name = 'Jansen']
```

查询的前半部分负责定义要检索的实体，而后半部分（位于方括号之间）则将数据约束到某个属性。请注意，约束由（且应始终由）方括号括起。

可将多个约束添加到单个查询中，这适用于除 id 查询之外的所有查询。该操作通常通过关闭第一组方括号后再打开一组新方括号这一简单快捷的方法来完成。

在 Studio Pro 中，不用写入完整查询，只需写入约束。实体由上下文隐式确定。因此，只需在客户上下文中编写 [Name='Jansen']，而不是 //Sales.Customer[Name='Jansen']。

在 Java 中，的确需要写入整个查询，包括双斜杠 (//) 和实体名称。

2 示例

该查询会检索名字等于 Jansen 且在鹿特丹生活的所有客户：

```
//Sales.Customer[Name = 'Jansen'][Sales.Customer_Address/Sales.Address/City = 'Rotterdam']
```

此外，也可将约束与运算符 and 或 or 组合。该查询会检索名字等于 Jansen 且在鹿特丹生活的所有客户：

```
//Sales.Customer[Name = 'Jansen' and Sales.Customer_Address/Sales.Address/City = 'Rotterdam']
```

该查询会检索名字等于 Jansen 或在鹿特丹生活的所有客户。

```
//Sales.Customer[Name = 'Jansen' or Sales.Customer_Address/Sales.Address/City = 'Rotterdam']
```

利用圆括号，可将约束分组以定义优先级。该查询会检索不仅名字等于“Jansen”或“Smit”，而且在鹿特丹生活的所有客户：

```
//Sales.Customer[( Name = 'Jansen' or Name = 'Smit' ) and Sales.Customer_Address/Sales.Address/City = 'Rotterdam']
```

在某些情况下，定义子约束来限制被约束的数据可能也很有用。这可以通过将子约束添加到原始约束的方括号中来轻松实现。请勿将此与两个单独的约束混淆，因为子约束仅适用于元约束，而不适用于实际查询。因此，方括号不会一个接一个地打开和关闭；子约束应完全处于元约束内部。在足够复杂的查询中，这可能会就一个约束的结束位置和另一个约束的开始位置而导致混淆。确保仔细跟踪方括号集，以防止这种情况发生。

该查询会检索具有“管理员”角色的所有用户：

```
//Sales.User[id = '%UserRole_Administrator%']]]
```

该查询会检索在鹿特丹或洛斯登生活的所有客户：

```
//Sales.Customer[Sales.Customer_Address/Sales.Address[City = 'Rotterdam' or City = 'Losdun']]]
```

该查询会检索在圭亚那新阿姆斯特丹生活的所有客户（而不是在例如印第安纳州新阿姆斯特丹生活的那些客户）：

```
//Sales.Customer[Sales.Customer_Address/Sales.Address[City = 'New Amsterdam'] /Sales.Adress_Country/Sales.Country/Name = 'Guyana']
```

避免在单个约束中多次使用同一路径。譬如，关于鹿特丹和洛斯登的示例也可以这样建立：

```
//Sales.Customer[Sales.Customer_Address/Sales.Address/City = 'Rotterdam' or Sales.Customer_Address/Sales.Address/City = 'Losdun']
```

不过，该查询的执行效率低下，因此会显著减慢查询过程。