

关联

1 简介

关联描述实体之间的关系。在域模型中，关联以两个实体之间的连线或箭头表示。

来自同一数据源的两个外部实体之间的关联在发起应用程序中定义，因此在模型中使用实体时会自动建立。有关更多详细信息，请参见外部实体的“关联”部分。

1.1 所有权

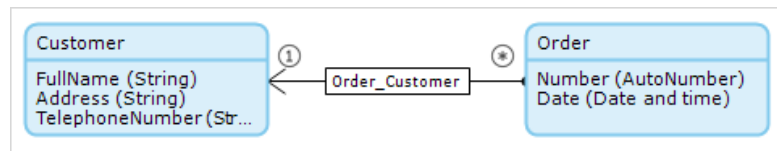
应从作为关联所有者的实体的对象查看和编辑关联值。关联中的所有权由箭头表示（注意，箭头并不表示方向）。一个实体或两个实体都可以是关联的所有者。如果一个实体是所有者的所有者，则有一个箭头从所有者指向另一个实体。如果两个实体都是所有者，则两个实体之间有一根连线，但没有箭头。这是可以控制箭头的唯一方式。

了解为什么存在所有权很重要。所有权在 Mendix 中实现，这样就可以动态更改关系，而不是受困于第一个设计。例如，如果设计某个内容作为一对多关联，然后需要变成具有默认所有权的多对多关联，则无需重新构建数据库，因为 Mendix 可代为处理。

1.2 多重性

在关联任意一侧，关联的多重性（或引用对象的数量）由数字一 (1) 或一个星号 (*) 表示。

在以下示例中，箭头指示**订单**为关联的所有者，1 和 * 表示一个客户与多个订单相关联：



持久实体与非持久实体之间的关联必须在非持久实体中开始，且所有者为**默认所有者**。关于持久和非持久实体的更多信息，请参见持久性。

2 创建关联

创建关联的最快方法是绘制域模型中两个实体之间的关联。默认情况下，这将创建一对多关联，从关联的所有者/多侧开始，在关联一侧结束。用下划线将两个实体的名称连接在一起，可对关联命名。然后，可按照下一节所述对关联进行编辑。

还可在应用程序的不同模块中的实体之间创建关联。这种情况下，无法绘制关联。可在拥有关联的实体的**关联**选项卡中创建一个新关联，以此创建与另一模块的域模型中实体的关联。然后可以选择应用程序中的任何实体作为关联目标。有关更多信息，请参见“关联选项卡属性”。

只能创建和编辑外部实体与本地实体之间的关联。但是，外部实体不能是与本地实体的关联的所有者。

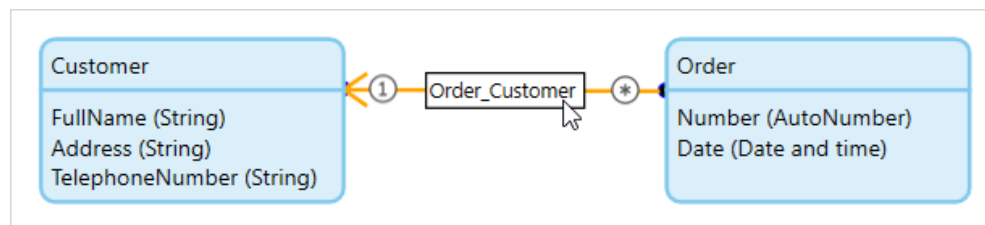
如果需要连接两个外部实体，可考虑添加本地实体，并将该本地实体与两个外部实体相连。在这种情况下，本地实体必须是这两个关联的所有者。

3 编辑关联

有两种编辑关联的方法。

3.1 直接编辑关联

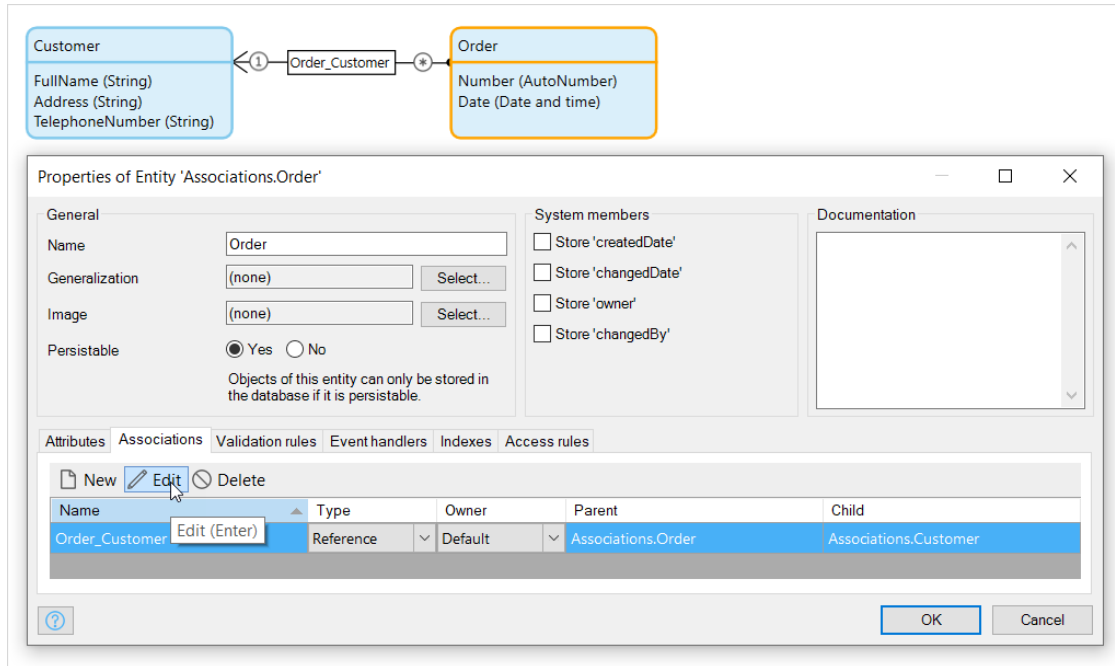
可以编辑关联本身。这种情况下，将使用多重性和可导航性来定义关联。



有关更多信息，请参见“关联属性”。

3.2 从实体中的关联进行编辑

可以实体成员的形式编辑关联。这种情况下，将使用类型和所有者来定义关联。

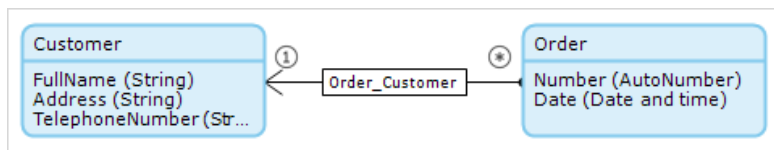


有关更多信息，请参见“关联选项卡属性”。

4 关联示例

4.1 一对多关联

本示例中绘制了一个从**订单**实体到**客户**实体的关联，结果如下所示：



类型属性的默认值为引用，所有者（订单实体）为默认。这等同于将多重性设为一个“客户”对象与多个“订单”对象关联，这样一个客户可以有多个订单，但一个订单只能有一个客户。

在 XML 中，这些实体及其关联的实例如下所示（请注意，关联仅存储在**订单**元素中）：

```
<Order id="101">
  <number>1</number>
  <date>9/30/2008</date>
  <Order_Customer>id_201</Order_Customer>
</Order>
```

```
<Customer id="201">
  <fullname>Apple Inc.</fullname>
  <address>1 Infinite Loop</address>
  <telephonenumber>1-800-MY-APPLE</telephonenumber>
</Customer>
```

4.2 具有默认所有权的多对多关联

通过绘制关联，然后将类型属性设为引用集，并将所有者保留为默认，可创建具有默认所有权的多对多关联。

在本示例中，一个**客户**可以有多个**组**，而一个**组**可以有多个**客户**。这等同于将多重性设为多个“组”对象与多个“客户”对象关联，同时将可导航性设为“客户”对象引用“组”对象：



在 XML 中，这些实体及其关联的实例如下所示（请注意，关联仅存储在**客户**元素中）：

```
<Customer id="201">
  <fullname>Apple Inc.</name>
  <address>1 Infinite Loop</address>
  <telephonenumber>1-800-MY-APPLE</telephonenumber>
  <Customer_Group>id_301 id_302</Customer_Group>
</Customer>
```

```
<Group id="301">
  <name>Multinational corporations</name>
</Group>
```

```
<Group id="302">
  <name>Hardware suppliers</name>
</Group>
```

4.3 一对一关联

通过将所有者属性设为两者（同时将类型属性保留为默认值引用），可创建一对一关联。

在本示例中，一个**客户**可以有一个**配置文件**，而一个**配置文件**可以有一个**客户**。这等同于将多重性设为一个“客户”对象与一个“配置文件”对象关联：



在 XML 中，这些实体及其关联的实例如下所示（请注意，关联同时存储在**配置文件**元素和**客户**元素中）：

```
<Profile id="401">
  <religion>Buddhism</religion>
  <job>Chief Executive Officer</job>
  <website>http://www.apple.com/ </website>
  <Customer_Profile>id_201</Customer_Profile>
</Profile>
```

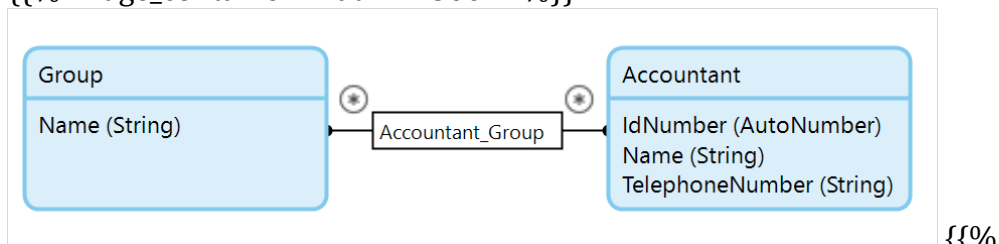
```
<Customer id="201">
  <fullname>Steve Jobs</fullname>
  <address>1 Infinite Loop</address>
  <telephonenumber>1-800-MY-APPLE</telephonenumber>
  <Customer_Profile>id_401</Customer_Profile>
</Customer>
```

4.4 具有双重所有权的多对多关联

通过将所有者属性设为两者，将类型属性设为引用集，可创建两个实体均为所有者的多对多关联。

在本示例中，一位**会计师**可以有多个**组**，而一个**组**可以有多位**会计师**。这等同于将多重性设为多个“组”对象与多个“会计师”对象关联，同时将可导航性设为“会计师”和“组”对象互为引用：

```
{{% image_container width= "500" %}}
```



```
/image_container %}}
```

在 XML 中，这些实体及其关联的实例如下所示（请注意，关联同时存储在**会计师**元素和**组**元素中）：

```
<Accountant id=""501"">  
<idnumber> 1</idnumber>  
<name> 浅灰色</name>  
<telephonenumber>1-800-EARL-灰</telephonenumber>  
<Accountant_Group> id_301 id_302</Accountant_Group>
```

```
<Accountant id=""502"">  
<idnumber> 2</idnumber>  
<name> Scrooge McDuck</name>  
<telephonenumber>1-800-SCROOGE-MCDUCK</telephonenumber>  
<Accountant_Group> id_301 id_302</Accountant_Group>
```

```
<Group id=""301"">  
<name> 多</name>  
<Accountant_Group> id_501 id_502</Accountant_Group>
```

```
<Group id=""302"">  
<name> 软件供应商</name>  
<Accountant_Group> id_501 id_502</Accountant_Group>  
</Group> </Group> </Accountant> </Accountant>
```