Novell Identity Manager

3

www.novell.com

策略构建器和驱动程序自定义指南

2006 年 7 月 21 日

Novell®

法律声明

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

联机文档：要访问本产品和其它 Novell 产品的联机文档或获取产品的更新，访问下列网址：
www.novell.com/documentation。

Novell 商标

DirXML 是 Novell, Inc. 在美国和其它国家 / 地区的注册商标。
eDirectory 是 Novell, Inc. 的商标。
Novell 是 Novell, Inc. 在美国和其它国家 / 地区的注册商标。
Nsure 是 Novell, Inc. 的商标。

第三方资料

所有第三方商标均属其各自所有者的财产。

关于本指南

Novell® Identity Manager 3.0 ® 是一项数据共享和同步服务，可实现应用程序、目录以及数据库之间的信息共享。它将分散的信息链接在一起，从而允许您通过创建策略来管理指定系统在身份发生更改时的自动更新。

Identity Manager 为帐户供应、安全性、一次签到、用户自助服务、鉴定、授权、自动工作流程和万维网服务提供了基础。它允许您集成、管理和控制分布式身份信息，以便将正确的资源安全地递送给适当的人员。

本指南提供了有关 Identity Manager 3.0 中策略构建器和驱动程序配置的详细参照信息。

读者

本指南供 Identity Manager 管理员使用。

反馈

欢迎您对本手册和本产品的其它文档提出意见和建议。请使用联机文档中每页底部的 " 用户意见 " 功能，或访问 www.novell.com/documentation/feedback.html 并输入您的意见。

文档更新

有关本文档的最新版本，请参见 Identity Manager 文档万维网站点 (http://www.novell.com/documentation/idm)

有关 Identity Manager 2.0 的文档，请访问 Identity Manager 文档万维网站点 (http://www.novell.com/documentation/idm)

其它文档

有关 Identity Manager 驱动程序使用方法的文档，请访问 Identity Manager 文档万维网站点 (http://www.novell.com/documentation/idmdrivers/index.html)。

文档约定

在本文档中，大于号 (>) 用于分隔同一步骤中的各项操作，以及交叉参照路径中的各个项目。

商标符号 （® 、 ™ 等）表示 Novell 商标。星号 (*) 表示第三方商标。

策略和过滤器

1

本节包含策略和过滤器的概述，及其在 Identity Manager 环境中的功能。包括以下主题：

1.1 什么是策略和过滤器？

使用策略可以在很大程度上自定义 Identity Manager 发送和接收更新的方式。

理解有关驱动程序 Shim 编写目的的某些详细信息有助于理解策略。

部署驱动程序的公司编写驱动程序 Shim 是为了同步可能使用到的任何内容。开发者编写驱动程序 Shim 以检测已连接系统中的所有相关更改，然后将检测到的更改传递给 Identity Vault。

在一个根据 Identity Manager 规范格式化的 XML 文档中包含检测到的更改。以下代码片段中包含其中一个 XML 文档：

```
<nds dtdversion="2.0" ndsversion="8.7.3">
<source>
   <product version="2.0">DirXML</product>
   <contact>Novell, Inc.</contact>
</source>

<input>
   <add class-name="User" event-id="0" src-dn="\ACME\Sales\Smith"
   src-entry-id="33071">
      <add-attr attr-name="Surname">
         <value timestamp="1040071990#3" type="string">Smith</value>
      </add-attr>
      <add-attr attr-name="Telephone Number">
         <value timestamp="1040072034#1" type="teleNumber">111-1111</
value>
      </add-attr>
   </add>
</input>
</nds>
```

驱动程序能够报告所有相关的更改，您可以据此过滤信息。过滤器用于阻止信息。您可以修改过滤器，仅允许所需的信息进入您的环境。有关哪些更改很重要以及如何处理这些更改的逻辑是在引擎中而不是驱动程序 Shim 中处理的。

如果某家公司并不怎么关心组，则可以使用在 Identity Vault 或已连接系统中阻止所有有关组的操作的过滤器。如果某家公司很重视用户和组，则可以使用允许在 Identity Vault 和已连接系统之间同步这两种类型对象的过滤器。

在驱动程序自定义中，第一步是定义过滤器，仅同步那些您关注的对象。

第二步是定义 Identity Manager 对过滤器允许的对象所执行的操作。上文的 XML 文档中的添加操作就是这样的例子，将姓名为 Smith、电话号码是 111-1111 的用户添加到已连接系统中。假定您允许此操作，则 Identity Manager 需要决定如何处理此用户。

Identity Manager 将以特定顺序应用一组策略，从而确定需要执行的操作。

首先，匹配策略将回答以下问题：" 此对象是否已在数据存储区中？ " 要回答此问题，您需要定义对象的唯一特性。因为电子邮件地址一般是唯一的，所以要检查的常用特性可以是电子邮件地址。您可以定义这样一个策略：如果两个对象的电子邮件地址相同，则它们是同一对象。

如果找到匹配项，则 Identity Manager 将该匹配项记录在称为关联的特性中。关联是一个唯一值， Identity Manager 可使用它将已连接系统中的对象关联起来。

未找到匹配项时将调用创建策略。创建策略将告诉 Identity Manager 您希望在何种情况下创建对象。您可以将某些特性设为创建规则中的必需项。如果这些特性不存在， Identity Manager 将阻止对象的创建，直到提供了所需的信息。

创建对象后，布局策略将通知 Identity Manager 放置对象的位置。您可以指定应该在与对象原来的系统相同的层次结构中创建对象，或者可以根据某个特性值将对象放置在完全不同的位置。

如果希望根据对象的位置特性按层次放置用户，并根据全名为它们命名，则可以将这些特性设为创建策略中的必需项。这样可以确保该特性存在，从而使布局策略正常工作。

还可以使用策略做很多其它的事情。使用策略构建器，可以轻松地生成唯一值、添加和去除特性、生成事件和命令、发送电子邮件等。甚至还可以进行更高级的转换，方法是使用 XSLT 直接转换 XML 文档（请记住，在 XML 文档中的 Identity Vault 之间发送更改）。

要记住的基本内容是使用策略可以控制 Identity Manager 如何处理更新。

请继续转到 “策略简介”在第 11 页 学习更多有关不同类型的策略的内容，然后转到第 2 章“通过使用带有 Designer 的策略构建器定义策略”在第 33 页 或第 3 章“在 iManager 中使用策略构建器定义策略”在第 195 页 学习如何使用策略构建器。

1.1.1 与较早版本相比的术语变更

在 DirXML® 1.1a 中，术语 " 规则 " 用来说明规则集、规则集中的各个规则或每个规则中的各种条件和操作，具体取决于环境。如果环境交待不清，这种重叠在某些情况下会导致混淆。

在 Identity Manager 2 中，现在使用 " 策略 " 代替以前使用的 " 规则 " 来描述所发生的高级转换。您现在要定义的是策略集，其中每项策略包含一个或多个规则。而术语 " 规则 " 现在仅用于描述单个条件和操作集。

下表说明了从 DirXML 1.1a 到 Identity Manager 2.x 的术语变更情况。

表 ***1-1*** 从 *DirXML 1.1a* 到 *Identity Manager 2.x* 的术语变更情况

概念	DirXML 1.1a 术语	Identity Manager 2.x 术语
转换集	规则	策略集
策略集中的单个转换	规则	策略

概念	DirXML 1.1a 术语	Identity Manager 2.x 术语
单个转换中的条件和操作	规则	规则

下表说明了从 Identity Manager 2.x 到 Identity Manager 3.0 的术语变更情况。

表 *1-2* 从 *Identity Manager 2.x* 到 *Identity Manager 3.0* 的术语变更情况

概念	Identity Manager 2.x 术语	Identity Manager 3 术语
产品	DirXML	Identity Manager
安装该产品的服务器	DirXML 服务器	Metadirectory 服务器
应用程序或数据库中正在同步数据的服务器	DirXML 已连接系统服务器	已连接系统服务器
对象的储存位置	eDirectory™	Identity Vault
处理部件	DirXML 引擎	Metadirectory 引擎

1.1.2 DirXML 底稿

DirXML 底稿是实施 Identity Manager 策略的主要方法。它说明了有序规则集实施的策略。规则包含要测试的条件集，以及条件符合时要执行的有序操作集。

使用策略构建器可以创建 DirXML 底稿，它提供了易于使用的 GUI 界面。

1.2 策略简介

本节介绍了各种可用策略和它们在 Identity Manager 中的职能，以及如何定义您自己的策略。包括以下主题：

- “策略”在第 11 页
- “定义策略”在第 30 页

1.2.1 策略

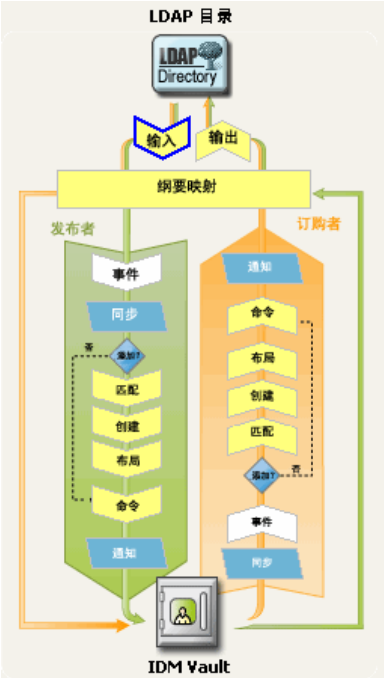
您可以在订购者通道和发布者通道上定义若干不同类型的策略。每种策略应用于数据转换中的不同步骤，某些策略仅在发生特定操作时才可以使用。例如，创建策略只能在创建新对象时使用。

在通道上执行策略的顺序是：

- “事件转换策略”在第 12 页
- “匹配策略”在第 15 页
- “创建策略”在第 16 页
- “布局策略”在第 19 页
- “命令转换策略”在第 22 页
- “纲要映射策略”在第 24 页

- “输出转换策略”在第 27 页
- “输入转换策略”在第 29 页

图 *1-1* 策略的执行顺序



事件转换策略

事件转换策略更改 Metadirectory 引擎中事件的显示，这些事件发生在 Identity Vault 或已连接的应用程序中。事件转换策略中执行的大部分常见任务是自定义过滤，如范围过滤和事件类型过滤。

范围过滤根据事件位置或特性值去除不需要的事件。例如，如果 department （部门）特性不等于特定的值或不是特定组的成员，则去除该事件。

事件类型过滤根据事件类型去除不需要的事件。例如，去除所有删除事件。

示例：

- 范围过滤
- 类型过滤

范围过滤：此示例中的 DirXML 底稿策略允许仅用于用户的事件，这些用户包含在用户子树中、未被禁止且其 Title （职务）特性中不包含文字 Consultant 和 Manager。它还生成状态文档，指示阻止操作的时间。

```
<policy>
      <rule>
            <description>Scope Filtering</description>
            <conditions>
                  <or>
                     <if-class-name op="equal">User</if-class-name>
                  </or>
```

```
                    <or>
                          <if-src-dn op="not-in-subtree">Users</if-
src-dn>
                           <if-attr name="Login Disabled"
op="equal">True</if-attr>
                        <if-attr mode="regex" name="Title"
op="equal">.*Consultant.*</if-attr>
                        <if-attr mode="regex" name="Title"
op="equal">.*Manager.*</if-attr>
                  </or>
            </conditions>
            <actions>
                  <do-status level="error">
                        <arg-string>
                              <token-text>User doesn’t meet required
conditions</token-text>
                        </arg-string>
                  </do-status>
                  <do-veto/>
               </actions>
      </rule>
</policy>
```

此 DirXML 底稿策略禁止对用户对象 （已关联的对象除外）进行修改操作。

```
<policy>
      <rule>
             <description>Veto all operation on User except modifies
of already associated objects</description>
            <conditions>
                  <or>
                        <if-class-name op="equal">User</if-class-name>
                  </or>
                  <or>
                        <if-operation op="not-equal">modify</if-
operation>
                         <if-association op="not-associated"/>
                  </or>
            </conditions>
            <actions>
                     <do-veto/>
            </actions>
      </rule>
</policy>
```

类型过滤 - 此示例中的 DirXML 底稿策略的第一条规则仅允许同步员工和合同工树枝中的对象。第二条规则阻止所有重命名和移动操作。

```
<policy>
       <rule>
            <description>Only synchronize the Employee and Contractor
subtrees</description>
```

```
            <conditions>
                        <and>
                               <if-src-dn op="not-in-
container">Employees</if-src-dn>
                               <if-src-dn op="not-in-
container">Contractors</if-src-dn>
                        </and>
            </conditions>
            <actions>
                        <do-status level="warning">
                           <arg-string>
                             <token-text>Change ignored: Out of
scope.</token-text>
                           </arg-string>
                         </do-status>
                          <do-veto/>
            </actions>
       </rule>
       <rule>
            <description>Don’t synchronize moves or renames</
description>
            <conditions>
                        <or>
                            <if-operation op="equal">move</if-
operation>
                            <if-operation op="equal">rename</if-
operation>
                        </or>
           </conditions>
            <actions>
                        <do-status level="warning">
                                <arg-string>
                                          <token-text>Change ignored:
We don’t like you to do that.</token-text>
                                </arg-string>
                  </do-status>
                  <do-veto/>
            </actions>
      </rule>
</policy>
```

此 DirXML 底稿策略阻止所有的 Add 事件。

```
<policy>
      <rule>
             <description>Type Filtering</description>
             <conditions>
                       <and>
                                <if-operation op="equal">add</if-
operation>
                       </and>
             </conditions>
             <actions>
```

```
                        <do-status level="warning">
                                 <arg-string>
                                        <token-text>Change ignored:
Adds are not allowed.</token-text>
                                  </arg-string>
                        </do-status>
                        <do-veto/>
                </actions>
        </rule>
</policy>
```

匹配策略

匹配策略 （如订购者匹配和发布者匹配）在目标数据存储区中寻找对应于源数据存储区中的非关联对象的对象。请注意并非始终需要匹配策略，这一点很重要。

例如，在下列情况下可能不需要匹配策略：

- 没有预先存在或相应的对象时执行初始迁移

必须认真构思匹配策略，以确保匹配策略不会找到错误的匹配项。

示例：

- 按因特网电子邮件地址匹配
- 按常用名匹配

按 **ID** 匹配：此示例中的 DirXML 底稿策略根据因特网电子邮件地址匹配用户。

```
<policy>
        <rule>
            <description>Match Users based on email address</
description>
            <conditions>
                      <and>
                        <if-class-name op="equal">User</if-class-name>
                      </and>
            </conditions>
            <actions>
                   <do-find-matching-object>
                         <arg-dn>
                           <token-text>ou=people,o=novell</token-text>
                         </arg-dn>
                  <arg-match-attr name="Internet EMail Address"/>
                    </do-find-matching-object>
            </actions>
       </rule>
</policy>
```

按名称匹配：此示例中的 DirXML 底稿策略根据组对象的 Common Name （常用名）特性对它进行匹配。

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<policy>
        <rule>
                <description>Match Group by Common Name</description>
                <conditions>
                          <or>
                                   <if-class-name op="equal">Group</
if-class-name>
                           </or>
                </conditions>
                <actions>
                        <do-find-matching-object scope="subtree">
                               <arg-match-attr name="CN"/>
                         </do-find-matching-object>
                </actions>
         </rule>
</policy>
```

创建策略

创建策略 （如订购者创建策略和发布者创建策略）定义在创建新对象必须满足的条件。如果没有创建策略，则意味着可以创建对象。

例如，在 Identity Vault 中创建新用户，但只为新用户对象分配了名称和 ID。在 eDirectory 树中可镜像该创建，但不会在与 Identity Vault 连接的应用程序中立即反映这一添加，因为您的创建策略指定只允许具有一个更完整定义的用户对象。

订购者和发布者的创建策略可以相同，也可以不同。

要在 eDirectory 中创建对象时，可指定在创建过程中使用的模板对象。

创建策略常用于以下情况：

- 禁止创建可能由于缺少特性而不合格的对象。
- 提供默认特性值。
- 提供默认口令。

示例：

- 必需的特性
- 默认特性值
- 默认口令
- 指定模板

必需的特性：此示例中的 DirXML 底稿策略的第一条规则要求在创建用户之前用户对象需要包含 CN、 Given Name （名）、 Surname （姓氏）和 Internet EMail Address （因特网电子邮件地址）特性。第二条规则要求所有组织单元对象都有 OU 特性。最后一条规则禁止所有名为 Fred 的用户对象。

```
<policy>
      <rule>
            <description>Veto if required attributes CN, Given Name,
Surname and Internet EMail Address not available</description>
```

```
            <conditions>
                  <or>
                          <if-class-name op="equal">User</if-class-
name>
                  </or>
            </conditions>
            <actions>
             <do-veto-if-op-attr-not-available name="CN"/>
             <do-veto-if-op-attr-not-available name="Given Name"/>
                  <do-veto-if-op-attr-not-available name="Surname"/>
                  <do-veto-if-op-attr-not-available name="Internet
EMail Address"/>
            </actions>
      </rule>
    <rule>
            <description>Organizational Unit Required Attributes</
description>
            <conditions>
                <or>
                        <if-class-name op="equal">Organizational
Unit</if-class-name>
                  </or>
            </conditions>
            <actions>
                  <do-veto-if-op-attr-not-available name="OU"/>
            </actions>
      </rule>
</policy>
```

默认特性值： 此示例中的 DirXML 底稿策略添加用户 Description （说明）特性的默认值。

```
<policy>
      <rule>
            <description>Default Description of New Employee</
description>
            <conditions>
                     <or>
                       <if-class-name op="equal">User</if-class-name>
                     </or>
            </conditions>
            <actions>
                   <do-set-default-attr-value name="Description">
                         <arg-value type="string">
                                 <token-text>New Employee</token-text>
                         </arg-value>
                   </do-set-default-attr-value>
            </actions>
       </rule>
</policy>
```

默认口令：此示例中的 DirXML 底稿策略将口令值创建为：名的前两个字母和姓的前六个字母，均为小写。

```
<policy>
       <rule>
                        <description>Default Password of [2]FN+[6]LN</
description>
            <conditions>
                  <and>
                        <if-class-name op="equal">User</if-class-name>
                         <if-password op="not-available"/>
                  </and>
            </conditions>
            <actions>
                  <do-set-dest-password>
                        <arg-string>
                              <token-lower-case>
                                    <token-substring length="2">
                                          <token-op-attr name="Given
Name"/>
                                    </token-substring>
                                 <token-substring length="6">
                                          <token-op-attr
name="Surname"/>
                                    </token-substring>
                                     </token-lower-case>
                           </arg-string>
                   </do-set-dest-password>
             </actions>
      </rule>
</policy>
```

指定模板：此示例中的 DirXML 底稿策略在用户的 Title （职务）特性指示该用户为经理（包含 "Manager"）时指定模板对象。

```
<policy>
         <rule>
                     <description>Assign Manager Template if Title
contains Manager</description>
                     <conditions>
                           <and>
                              <if-class-name op="equal">User</if-class-
name>
                               <if-op-attr name="Title" op="available"/
>
                                <if-op-attr mode="regex" name="Title"
op="equal">.*Manager.*</if-op-attr>
                           </and>
                      </conditions>
                      <actions>
                             <do-set-op-template-dn>
                                    <arg-dn>
                                          <token-text>Users\Manager
Template</token-text>
                                     </arg-dn>
```

```
                              </do-set-op-template-dn>
                        </actions>
                  </rule>
</policy>
```

布局策略

布局策略确定新对象在 Identity Vault 和已连接的应用程序中的放置位置及名称。

如果希望在 Identity Vault 中创建对象，则发布者通道需要布局策略。即使希望在已连接的应用程序中创建对象，订购者通道也可能不需要布局策略，这取决于目标数据存储区的本质。例如，与关系数据库同步时无需任何布局策略，因为关系数据库中的行没有位置或名称。

示例：

- 按特性值布局
- 按名称布局

按特性值布局：此示例中的 DirXML 底稿策略根据 Department （部门）特性的值在特定树枝中创建用户。

```
<policy>
        <rule>
             <description>Department Engineering</description>
             <conditions>
                  <and>
                        <if-class-name op="equal">User</if-class-name>
                        <if-op-attr mode="regex" name="Department"
op="equal">.*Engineering.*</if-op-attr>
                  </and>
            </conditions>
            <actions>
                  <do-set-op-dest-dn>
                        <arg-dn>
                              <token-text>Eng</token-text>
                          <token-text>\</token-text>
                                       <token-op-attr name="CN"/>
                        </arg-dn>
                  </do-set-op-dest-dn>
            </actions>
       </rule>
       <rule>
            <description>Department HR</description>
            <conditions>
                  <and>
                         <if-class-name op="equal">User</if-class-
name>
                        <if-op-attr mode="regex" name="Department"
op="equal">.*HR.*</if-op-attr>
                  </and>
            </conditions>
            <actions>
                  <do-set-op-dest-dn>
```

```
                        <arg-dn>
                              <token-text>HR</token-text>
                          <token-text>\</token-text>
                           <token-op-attr name="CN"/>
                        </arg-dn>
                  </do-set-op-dest-dn>
            </actions>
      </rule>
</policy>
```

此 DirXML 底稿策略按 src-dn 确定用户或组织单元在输入文档中的布局。

```
<policy>
       <rule>
             <description>PublisherPlacementRule</description>
             <conditions>
                       <or>
                            <if-class-name op="equal">User</if-class-
name>
                             <if-class-name op="equal">Organizational
Unit</if-class-name>
                       </or>
                       <or>
                             <if-src-dn op="in-subtree">o=people,
o=novell</if-src-dn>
                      </or>
               </conditions>
               <actions>
                    <do-set-op-dest-dn>
                        <arg-dn>
                              <token-text>People</token-text>
                              <token-text>\</token-text>
                              <token-unmatched-src-dn convert="true"/>
                        </arg-dn>
                     </do-set-op-dest-dn>
                  </actions>
        </rule>
</policy>
```

按名称布局：此示例中的 DirXML 底稿策略根据用户的姓的第一个字母在特定树枝中创建用户。如果用户姓氏的首字母为 A 到 I，则将用户放置在树枝 Users1 中；若为 J 到 R，则放置在 Users2 中；为 S 到 Z 的字母，则放置在 Users3 中。

```
<policy>
       <rule>
            <description>Surname - A to I in Users1</description>
            <conditions>
                       <and>
                             <if-class-name op="equal">User</if-
class-name>
                             <if-op-attr mode="regex" name="Surname"
op="equal">[A-I].*</if-op-attr>
```

```
                        </and>
            </conditions>
            <actions>
                <do-set-op-dest-dn>
                        <arg-dn>
                              <token-text>Users1</token-text>
                        <token-text>\</token-text>
                              <token-op-attr name="CN"/>
                        </arg-dn>
                  </do-set-op-dest-dn>
            </actions>
       </rule>
       <rule>
            <description>Surname - J to R in Users2</description>
            <conditions>
                     <and>
                             <if-class-name op="equal">User</if-class-
name>
                              <if-op-attr mode="regex" name="Surname"
op="equal">[J-R].*</if-op-attr>
                     </and>
            </conditions>
            <actions>
               <do-set-op-dest-dn>
                    <arg-dn>
                              <token-text>Users2</token-text>
                        <token-text>\</token-text>
                         <token-op-attr name="CN"/>
                        </arg-dn>
                  </do-set-op-dest-dn>
            </actions>
        </rule>
        <rule>
             <description>Surname - S to Z in Users3</description>
             <conditions>
                       <and>
                             <if-class-name op="equal">User</if-class-
name>
                              <if-op-attr mode="regex" name="Surname"
op="equal">[S-Z].*</if-op-attr>
                        </and>
             </conditions>
             <actions>
                  <do-set-op-dest-dn>
                     <arg-dn>
                              <token-text>Users3</token-text>
                              <token-text>\</token-text>
                              <token-op-attr name="CN"/>
                        </arg-dn>
                  </do-set-op-dest-dn>
            </actions>
      </rule>
</policy>
```

命令转换策略

命令转换策略通过替换或添加命令，更改 Identity Manager 发送到目标数据存储区的命令。截取 " 删除 " 命令并用 " 修改 "、" 移动 " 或 " 禁用 " 命令代替便是一个在命令转换策略中替换命令的示例。而根据 " 添加 " 命令的特性值创建 " 修改 " 命令便是一个在命令转换策略中添加命令的常见示例。

在最常用的情况下，命令转换策略用于更改 Identity Manager 执行的命令，这些命令是对提交至 Metadirectory 引擎的事件的默认处理方式。

通常的做法还有在此处添加与其它任何策略的说明均不完全适合的策略。

示例：

- 将 " 删除 " 转换为 " 修改 " 和 " 移动 "
- 创建附加操作
- 设置口令失效时间

将 " 删除 " 转换为 " 修改 "：此 DirXML 底稿策略将 Login Disabled （禁止登录）特性的 " 删除 " 操作转换为 " 修改 " 操作。

```
<policy>
      <rule>
            <description>Convert User Delete to Modify</description>
            <conditions>
                  <and>
                        <if-operation op="equal">delete</if-
operation>
                        <if-class-name op="equal">User</if-class-name>
                  </and>
            </conditions>
            <actions>
                  <do-set-dest-attr-value name="Login Disabled">
                        <arg-value type="state">
                              <token-text>true</token-text>
                        </arg-value>
                  </do-set-dest-attr-value>
                  <do-veto/>
            </actions>
      </rule>
</policy>
```

创建附加操作：此 DirXML 底稿策略确定用户的目标树枝是否已存在。如果该树枝不存在，则策略将创建 " 添加 " 操作以创建树枝对象。

```
<policy>
      <rule>
            <description>Check if destination container already
exists</description>
            <conditions>
                  <and>
                        <if-operation op="equal">add</if-operation>
                  </and>
```

```
            </conditions>
    <actions>
                  <do-set-local-variable name="target-container">
                        <arg-string>
                              <token-dest-dn length="-2"/>
                        </arg-string>
                  </do-set-local-variable>
                  <do-set-local-variable name="does-target-exist">
                        <arg-string>
                              <token-dest-attr class-
name="OrganizationalUnit" name="objectclass">
                                    <arg-dn>
                                          <token-local-variable
name="target-container"/>
                                    </arg-dn>
                          </token-dest-attr>
                        </arg-string>
                  </do-set-local-variable>
            </actions>
      </rule>
      <rule>
            <description>Create the target container if necessary</
description>
            <conditions>
                  <and>
                        <if-local-variable name="does-target-exist"
op="available"/>
                        <if-local-variable name="does-target-exist"
op="equal"/>
                  </and>
                </conditions>
            <actions>
                  <do-add-dest-object class-name="organizationalUnit"
direct="true">
                        <arg-dn>
                              <token-local-variable name="target-
container"/>
                        </arg-dn>
                  </do-add-dest-object>
                  <do-add-dest-attr-value direct="true" name="ou">
                        <arg-dn>
                              <token-local-variable name="target-
container"/>
                        </arg-dn>
                        <arg-value type="string">
                              <token-parse-dn dest-dn-format="dot"
length="1" src-dn-format="dest-dn" start="-1">
                                    <token-local-variable
name="target-container"/>
                              </token-parse-dn>
                                    </arg-value>
                  </do-add-dest-attr-value>
            </actions>
       </rule>
```

```
</policy>
```

设置口令失效时间：此 DirXML 底稿策略修改 eDirectory 用户的 Password Expiration Time（口令失效时间）特性。

```
<?xml version="1.0" encoding="UTF-8"?>
<policy xmlns:jsystem="http://www.novell.com/nxsl/java/
java.lang.System">
    <rule>
            <description>Set password expiration time for a given
interval from current day</description>
            <conditions>
                        <and>
                    <if-operation op="equal">modify-password</if-
operation>
                </and>
            </conditions>
            <actions>
                         <do-set-local-variable name="interval">
                    <arg-string>
                              <token-text>30</token-text>
                    </arg-string>
                </do-set-local-variable>
                <do-set-dest-attr-value class-name="User"
name="Password Expiration Time" when="after">
                                              <arg-association>
                          <token-association/>
                    </arg-association>
                    <arg-value type="string">
                                                                <token-
xpath expression="round(jsystem:currentTimeMillis() div 1000 +
(86400*$interval))"/>
                    </arg-value>
                </do-set-dest-attr-value>
            </actions>
      </rule>
</policy>
```

纲要映射策略

纲要映射策略包含 Identity Vault 与已连接系统之间的纲要映射定义。

可以从 eDirectory 中读取 Identity Vault 纲要。而已连接系统的 Identity Manager 驱动程序提供了已连接应用程序的纲要。这两个纲要均确定后，将在 Identity Vault 和目标应用程序之间创建一个简单映射。

在 Identity Manager 驱动程序配置中定义纲要映射策略后，可以映射相应的数据。

请注意，以下内容非常重要：

- 相同的策略可以双向应用。

- 在 Metadirectory 引擎和应用程序 Shim 之间的任一通道以任一方向传递的所有文档都通过纲要映射策略传递。

有关管理信息，请参见第 7 章 “管理纲要映射策略”在第 385 页 。

示例：

- 基本纲要映射策略
- 自定义纲要映射策略

基本纲要映射策略：此示例中的 DirXML 底稿策略说明了基本纲要映射策略的原始 XML 源。但是，通过 Designer for Identity Manager 编辑策略时，使用默认的纲要映射编辑器可以以图形方式显示和编辑策略。

```
<?xml version="1.0" encoding="UTF-8"?><attr-name-map>
      <class-name>
              <app-name>WorkOrder</app-name>
              <nds-name>DirXML-nwoWorkOrder</nds-name>
      </class-name>
      <class-name>
               <app-name>PbxSite</app-name>
               <nds-name>DirXML-pbxSite</nds-name>
      </class-name>
      <attr-name class-name="DirXML-pbxSite">
               <app-name>PBXName</app-name>
               <nds-name>DirXML-pbxName</nds-name>
      </attr-name>
      <attr-name class-name="DirXML-pbxSite">
                  <app-name>TelephoneNumber</app-name>
                 <nds-name>Telephone Number</nds-name>
      </attr-name>
      <attr-name class-name="DirXML-pbxSite">
               <app-name>LoginName</app-name>
               <nds-name>DirXML-pbxLoginName</nds-name>
       </attr-name>
       <attr-name class-name="DirXML-pbxSite">
               <app-name>Password</app-name>
               <nds-name>DirXML-pbxPassword</nds-name>
       </attr-name>
       <attr-name class-name="DirXML-pbxSite">
               <app-name>Nodes</app-name>
               <nds-name>DirXML-pbxNodesNew</nds-name>
       </attr-name>
      </attr-name-map>
```

自定义纲要映射策略：此示例中的 DirXML 底稿策略使用 DirXML 底稿执行自定义纲要映射。

```
<?xml version="1.0" encoding="UTF-8"?><policy>
      <rule>
            <!--
            The Schema Mapping Policy can only handle one-to-one
mappings.
```

```
            That Mapping Policy maps StudentPersonal addresses.
            This rule maps StaffPersonal addresses.
      -->
            <description>Publisher Staff Address Mappings</
description>
            <conditions>
                  <and>
                        <if-local-variable name="fromNds"
op="equal">false</if-local-variable>
                        <if-xpath op="true">@original-class-name =
'StaffPersonal'</if-xpath>
                  </and>
            </conditions>
            <actions>
                  <do-rename-op-attr dest-name="SA" src-name="Address/
Street/Line1"/>
                  <do-rename-op-attr dest-name="Postal Office Box"
src-name="Address/Street/Line2"/>
                  <do-rename-op-attr dest-name="Physical Delivery
Office Name" src-name="Address/City"/>
                  <do-rename-op-attr dest-name="S" src-name="Address/
StatePr"/>
                  <do-rename-op-attr dest-name="Postal Code" src-
name="Address/PostalCode"/>
            </actions>
      </rule>
      <rule>
            <description>Subscriber Staff Address Mappings</
description>
            <!--
            The Schema Mapping Policy has already mapped addresses to
StudentPersonal.
            This rule maps StudentPersonal to StaffPersonal.
      -->
            <conditions>
                  <and>
                        <if-local-variable name="fromNds"
op="equal">true</if-local-variable>
                        <if-op-attr name="DirXML-sifIsStaff"
op="equal">true</if-op-attr>
                  </and>
            </conditions>
            <actions>
                  <do-rename-op-attr dest-name="Address/Street/Line1"
src-name="StudentAddress/Address/Street/Line1"/>
                  <do-rename-op-attr dest-name="Address/Street/Line2"
src-name="StudentAddress/Address/Street/Line2"/>
                  <do-rename-op-attr dest-name="Address/City" src-
name="StudentAddress/Address/City"/>
                  <do-rename-op-attr dest-name="Address/StatePr" src-
name="StudentAddress/Address/StatePr"/>
                  <do-rename-op-attr dest-name="Address/PostalCode"
src-name="StudentAddress/Address/PostalCode"/>
            </actions>
```

```
      </rule>
</policy>
```

输出转换策略

输出转换策略主要处理数据格式的转换，即从 Metadirectory 引擎提供的数据转换到应用程序 Shim 需要的数据。这些转换的示例包括：

- 特性值格式转换
- XML 词汇转换
- 输出转换策略还可以自定义处理状态讯息，即处理从 Metadirectory 引擎返回到应用程序 Shim 的状态讯息。

Metadirectory 引擎在任一通道提供给应用程序 Shim 的所有文档都可以通过输出转换策略传递。因为输出转换发生在纲要映射之后，所以所有纲要名称都在应用程序名称空间中。

示例：

- 特性值格式转换
- 状态讯息的自定义处理

特性值转换：此示例中的 DirXML 底稿策略将电话号码格式由 (nnn) nnn-nnnn 重设为 nnn.nnn.nnnn。相反的转换可以在输入转换策略的示例中找到。

```
<policy>
      <rule>
            <description>Reformat all telephone numbers from (nnn)
nnn-nnnn to nnn.nnn.nnnn</description>
            <conditions/>
            <actions>
                        <do-reformat-op-attr name="telephoneNumber">
                            <arg-value type="string">
                                        <token-replace-first
regex="^\((\d\d\d)\) *(\d\d\d)-(\d\d\d\d)$" replace-with="$1.$2.$3">
                                                    <token-local-
variable name="current-value"/>
                                    </token-replace-first>
                            </arg-value>
                         </do-reformat-op-attr>
            </actions>
     </rule>
</policy>
```

状态讯息的自定义处理：此示例中的 DirXML 底稿策略检测到级别为未成功，且操作数据中包含 password-publish-status 子要素的状态文档，然后使用 " 从模板发送电子邮件 " 操作生成一条电子邮件讯息。

```
<?xml version="1.0" encoding="UTF-8"?>
      <policy>
                <description>Email notifications for failed password
publications</description>
```

```
            <rule>
                        <description>Send e-mail for a failed publish
password operation</description>
                     <conditions>
                            <and>
                                        <if-global-variable
mode="nocase" name="notify-user-on-password-dist-failure"
op="equal">true</if-global-variable>
                                        <if-operation
op="equal">status</if-operation>
                                        <if-xpath
op="true">self::status[@level != 'success']/operation-data/password-
publish-status</if-xpath>
                                  </and>
                     </conditions>
                     <actions>
                     <!-- generate email notification -->
                      <do-send-email-from-template notification-
dn="\cn=security\cn=Default Notification Collection" template-
dn="\cn=security\cn=Default Notification Collection\cn=Password Sync
Fail">
                           <arg-string name="UserFullName">
                                 <token-src-attr name="Full Name">
                                     <arg-association>
                                                 <token-xpath
expression="self::status/operation-data/password-publish-status/
association"/>
                                     </arg-association>
                                 </token-src-attr>
                            </arg-string>
                            <arg-string name="UserGivenName">
                                  <token-src-attr name="Given Name">
                                         <arg-association>
                                                  <token-xpath
expression="self::status/operation-data/password-publish-status/
association"/>
                                         </arg-association>
                                </token-src-attr>
                          </arg-string>
                          <arg-string name="UserLastName">
                               <token-src-attr name="Surname">
                                                 <arg-association>
                                            <token-xpath
expression="self::status/operation-data/password-publish-status/
association"/>
                                         </arg-association>
                                 </token-src-attr>
                           </arg-string>
                           <arg-string name="ConnectedSystemName">
                                   <token-global-variable
name="ConnectedSystemName"/>
                           </arg-string>
                           <arg-string name="to">
                                  <token-src-attr name="Internet Email
```

```
Address">
                                    <arg-association>
                                            <token-xpath
expression="self::status/operation-data/password-publish-status/
association"/>
                                    </arg-association>
                                </token-src-attr>
                            </arg-string>
                            <arg-string name="FailureReason">
                                    <token-text/>
                                       <token-xpath
expression="self::status/child::text()"/>
                             </arg-string>
                    </do-send-email-from-template>
            </actions>
      </rule>
</policy>
```

输入转换策略

输入转换策略主要处理数据格式的转换，即从应用程序 Shim 提供的数据转换到 Metadirectory 引擎需要的数据。这些转换的示例包括：

- 特性值格式转换
- XML 词汇转换
- 驱动程序心跳
- 输入转换策略还可以自定义处理状态讯息，即处理从应用程序 Shim 返回到 Metadirectory 引擎的状态讯息。

应用程序 Shim 在任一通道提供给 Metadirectory 引擎的所有文档都通过输入转换策略传递。

示例：

- 特性值格式转换
- 驱动程序心跳

特性值格式转换：此示例中的 DirXML 底稿策略将电话号码的格式由 nnn.nnn.nnnn 重设为 (nnn) nnn-nnnn。相反的转换可以在输出转换策略的示例中找到。

```
<policy>
      <rule>
            <description>Reformat all telephone numbers from
nnn.nnn.nnnn to (nnn) nnn-nnnn</description>
            <conditions/>
            <actions>
                  <do-reformat-op-attr name="telephoneNumber">
                        <arg-value type="string">
                                    <token-replace-first
regex="^(\d\d\d)\.(\d\d\d)\.(\d\d\d\d)$" replace-with="($1) $2-$3">

<token-local-variable name="current-value"/>
                                    </token-replace-first>
```

```
                              </arg-value>
                  </do-reformat-op-attr>
            </actions>
      </rule>
</policy>
```

驱动程序心跳：此 DirXML 底稿策略创建了一个状态心跳事件。驱动程序的心跳功能用于在每个心跳间隔发送成功讯息 (HEARTBEAT:$driver)。可以由 Novell Audit 监视此讯息。Identity Manager 驱动程序必须支持心跳，而且必须在驱动程序配置页中启用心跳。

```
<?xml version="1.0" encoding="UTF-8" ?>
 <policy>
      <rule>
      <description>Heartbeat Rule, v1.01, 040126, by Holger Dopp</
description>
            <conditions>
                  <and>
                        <if-operation op="equal">status</if-operation>
                        <if-xpath op="true">@type="heartbeat"</if-
xpath>
                  </and>
            </conditions>
            <actions>
                  <do-set-xml-attr expression="." name="text1">
                        <arg-string>
                              <token-global-variable
name="dirxml.auto.driverdn" />
             </arg-string>
                  </do-set-xml-attr>
                  <do-set-xml-attr expression="." name="text2">
                        <arg-string>
                              <token-text>HEARTBEAT</token-text>
                        </arg-string>
                  </do-set-xml-attr>
            </actions>
  </rule>
</policy>
```

1.2.2 定义策略

所有策略都可以使用以下两种方式之一进行定义：

- 使用策略构建器界面生成 DirXML 底稿。现有的非 XSLT 规则将在导入后自动转换为 DirXML 底稿。
- 使用 XSLT 样式表。

还可以（且通常是）使用纲要映射表定义纲要映射策略。

策略构建器和 DirXML 底稿

策略构建器界面用于定义可能实施的大多数策略。策略构建器界面使用图形环境，使您可以轻松地定义和管理策略。

策略构建器中规则创建的基本功能由一种称为 DirXML 底稿的自定义脚本语言提供。

DirXML 底稿包含多种可以测试的条件、执行的操作和添加到策略中的动态值。这些选项中的每一个都可以使用智能下拉列表显示，它们在每个点只提供有效的选择，还提供与常用值的快速链接。

借助策略构建器，可以不必直接使用 DirXML 底稿。

有关策略构建器的更多信息，请参见第 2 章 “通过使用带有 Designer 的策略构建器定义策略”在第 33 页 和第 3 章 “在 iManager 中使用策略构建器定义策略”在第 195 页 。有关 DirXML 底稿的更多信息，请参见 “DirXML 底稿”在第 11 页 。

提示 : 虽然并不是必须要使用策略构建器，但是您可以在 DirXML Driver Developer Kit Documentation （DirXML 驱动程序开发者工具文档）(http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/ref/index.html) 万维网站点获得完整的 DirXML 底稿参照。

XSLT 样式表

要定义更复杂的策略，可以使用 XSLT 样式表将一个 XML 文档直接转换为包含所需更改的另一个 XML 文档。

样式表提供了极大的灵活性，并在以下情况使用：转换不适合在使用策略构建器中的规则创建的预定义条件和可用操作中进行。

要创建 XSLT 样式表，您需要完全了解 XSLT、 nds.dtd 和传送到 （以及传送自） Metadirectory 引擎的命令和事件。有关 nds.dtd 的详细参照信息，请参见 NDS DTD 参照 (http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/ref/ndsdtd/index.html)。

有关 XSLT 样式表的更多信息，请参见第 5 章 “使用 XSLT 样式表定义策略”在第 347 页 。

可下载的 Identity Manager 策略

Novell 已经提供了样本策略，您可以下载并在您的环境中使用。这些策略可以从 Novell 支持万维网站点 (http://support.novell.com/filefinder/20607/index.html) 下载。下载并解压缩这些文件。 How_To_Install.rtf 文件包含安装说明。

要使用 Designer 导入这些文件，请参见 “从 XML 文件导入策略”在第 54 页 。要使用 iManager 导入这些文件，请参见 “从 XML 文件导入策略”在第 206 页 。

1.3 过滤器

过滤器指定 Metadirectory 引擎为之处理事件的对象类和特性，以及如何处理对这些类和特性所做的更改。

过滤器只能传递发生在以下对象上的事件：这些对象的基础类与过滤器指定的其中一个类相匹配。过滤器不能传递发生在以下对象上的事件：这些对象是在过滤器中指定的类的从属类，除非过滤器也指定了该从属类。每个通道都有独立的过滤器设置，从而可以控制每一个类和特性的同步方向和授权数据源。

注释：在 eDirectory 中，基础类是指用于创建项的对象类。必须在过滤器中指定该类，而不是指定基础类继承自的超类或附加特性可能来自的辅助类。

例如，如果在过滤器中将用户类的 Surname （姓氏）和 Given Name （名）特性设置为同步，则 Metadirectory 引擎将传递对这些特性进行更改的任何事件。但是，如果修改了项的 Telephone Number （电话号码）特性，则 Metadirectory 引擎会放弃此事件，因为 Telephone Number （电话号码）特性不在过滤器中。

必须配置过滤器以包含以下特性：

- 要同步的特性
- 不同步，但用于触发策略执行某些操作的特性

有关定义过滤器的信息，请参见第 6 章 “管理过滤器”在第 361 页。

2 通过使用带有 Designer 的策略构建器定义策略

策略构建器是一个完整的图形界面，可以用来创建和管理定义已连接系统之间进行的数据交换的策略。

以下章节提供了有关策略以及如何使用策略构建器的信息：

- “策略”在第 33 页
- “Designer 中的策略构建器任务”在第 34 页

以下章节还包括如下详细的参照部分：

- “正则表达式”在第 112 页
- “XPath 1.0 表达式”在第 113 页
- “条件”在第 114 页
- “操作”在第 130 页
- “名词标记”在第 173 页
- “动词标记”在第 186 页

2.1 策略

作为了解策略的工作方式的一部分，了解策略的组成非常重要。

- 策略由规则构成。
- 规则是在已定义的操作 （请参见 “操作”在第 130 页 ）发生之前必须事先满足的一组条件 （请参见 “条件”在第 114 页 ）。
- 操作可以具有动态自变量，这些自变量由运行时扩展的标记派生而来。
- 标记可分为两类：名词标记 （请参见 “名词标记”在第 173 页 ）和动词标记 （请参见 “动词标记”在第 186 页 ）。
  - 名词标记可扩展为由当前操作、源数据存储区或目标数据存储区，或一些外部源派生而来的值。
  - 动词标记用于修饰从属于它们的其它标记的已连接结果。
- 规则中通常使用正则表达式 （参见 “正则表达式”在第 112 页 ）和 XPath 1.0 表达式 （参见 “XPath 1.0 表达式”在第 113 页 ）为策略创建所需的结果。
- 策略在 XDS 文档中操作，主要用途是检查和修改该文档。
- 操作是 XDS 文档中作为输入要素和输出要素子级的任意要素。这些要素是 Novell 的 nds.dtd 的一部分；有关详细信息，请参见 NDS DTD (http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/ref/ndsdtd/index.html)。
- 操作通常表示事件、命令或状态。
- 策略将单独应用于每项操作。顺序对每项操作应用策略时，该操作将成为当前操作。每项规则依次应用于当前操作。所有规则都将应用于当前操作，除非某优先规则执行了某一操作，从而导致不再应用后续规则。

- 策略还可以从文档外部获得其它上下文，但所产生的副作用并不会反映到结果文档中。

2.2 Designer 中的策略构建器任务

本节介绍在策略构建器中执行的以下常见任务：

- “打开策略构建器”在第 34 页
- “创建策略”在第 38 页
- “创建规则”在第 46 页
- “创建自变量”在第 55 页
- “编辑策略”在第 65 页
- “使用预定义规则”在第 68 页
- “使用策略模拟器测试策略”在第 97 页
- “编辑 DirXML 底稿”在第 105 页

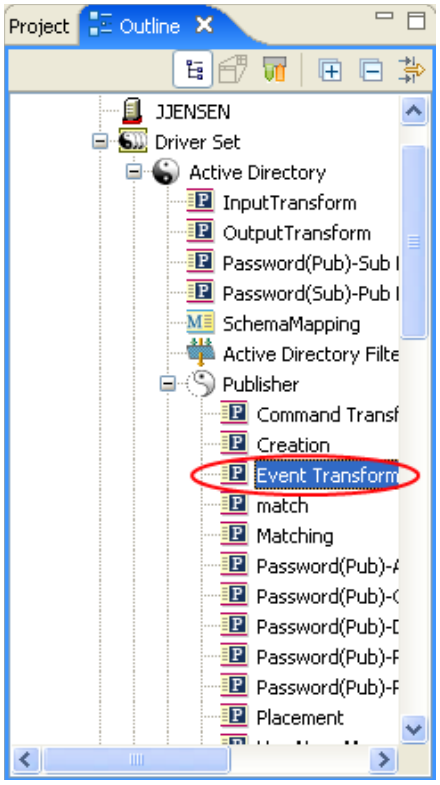
2.2.1 打开策略构建器

可以从 Model Outline （模型大纲）视图、 Policy Flow （策略流程）视图或策略集中打开策略构建器。

模型大纲视图

1. 在 Designer 中打开一个项目。
2. 单击 " 大纲 " 选项卡 > 选择 *Show Model Outline* （显示模型大纲）图标。

3 在 " 模型大纲 " 视图中双击所列出的某一策略，或右击并选择 " 编辑 "。

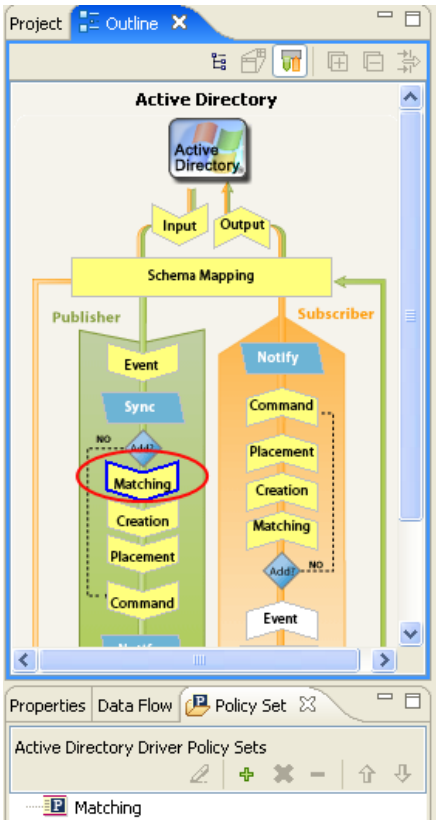


策略流程视图

1 在 Designer 中打开一个项目。

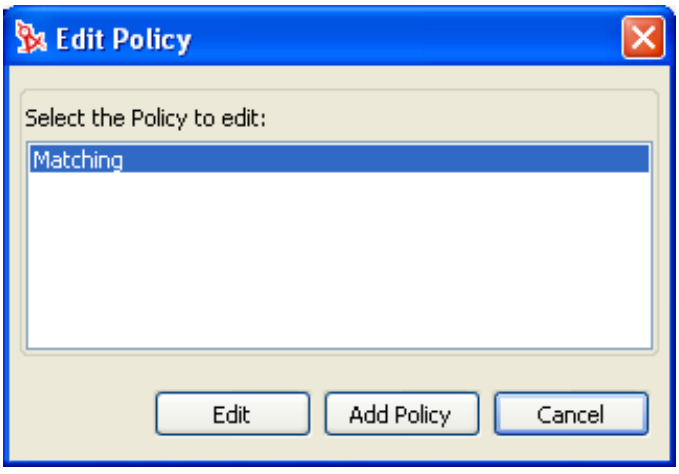
2 选择 " 大纲 " 选项卡 > 选择 *Show Policy Flow* （显示策略流程）图标。

3 在 " 策略流程 " 视图中右击某一策略 （例如匹配策略），然后选择 " 编辑策略 "。



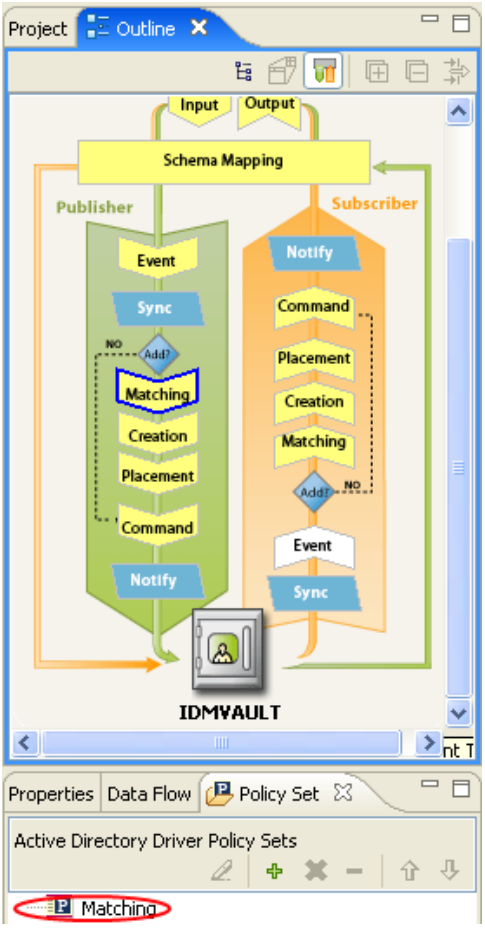
4 还可以在 " 策略流程 " 中双击 " 匹配 " 策略。

5 选择该策略，然后单击 " 编辑 "。



策略集

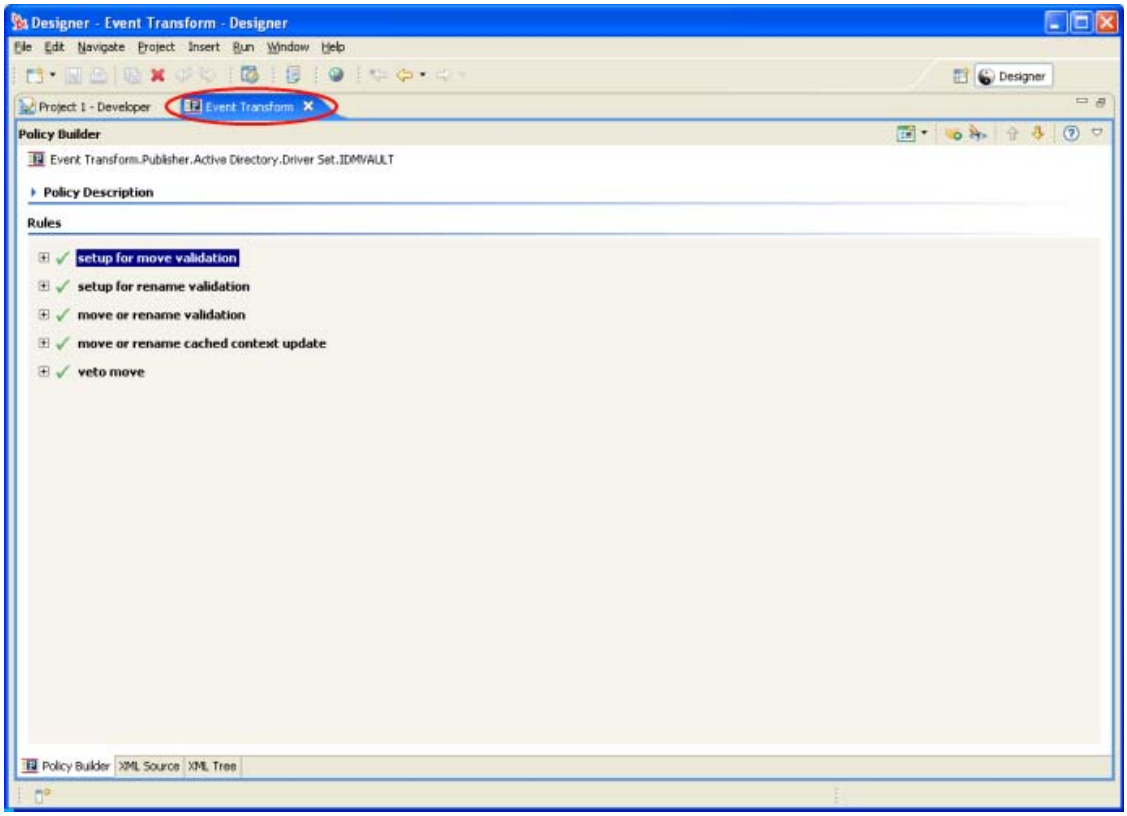
1 在策略集中右击该策略，然后单击 " 编辑 "。



2 还可以在策略集中选择该策略，然后单击 " 编辑策略 " 图标。

要查看 " 策略构建器 " 窗口中的所有信息，请双击 （不要滚动） " 策略 " 选项卡，以使策略构建器充满整个窗口。要最小化该窗口，请双击 " 策略 " 选项卡。

图 ***2-1*** 全屏显示策略构建器



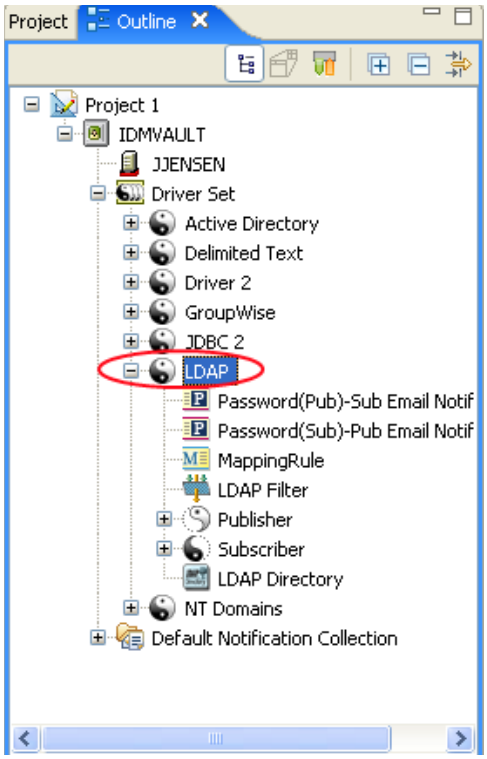
2.2.2 创建策略

策略可将数据发送到已连接系统。通过策略集创建策略。

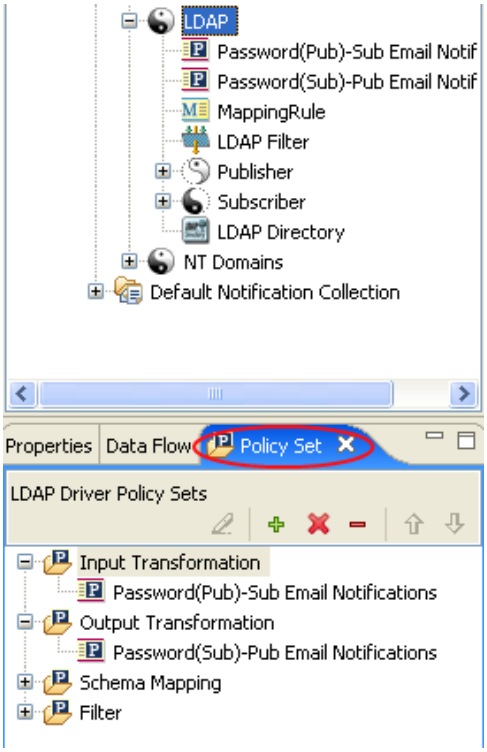
- “访问策略集”在第 39 页
- “使用策略集”在第 40 页
- “使用添加策略向导”在第 42 页

访问策略集

1 从打开项目的 " 大纲 " 视图中选择一个驱动程序对象。

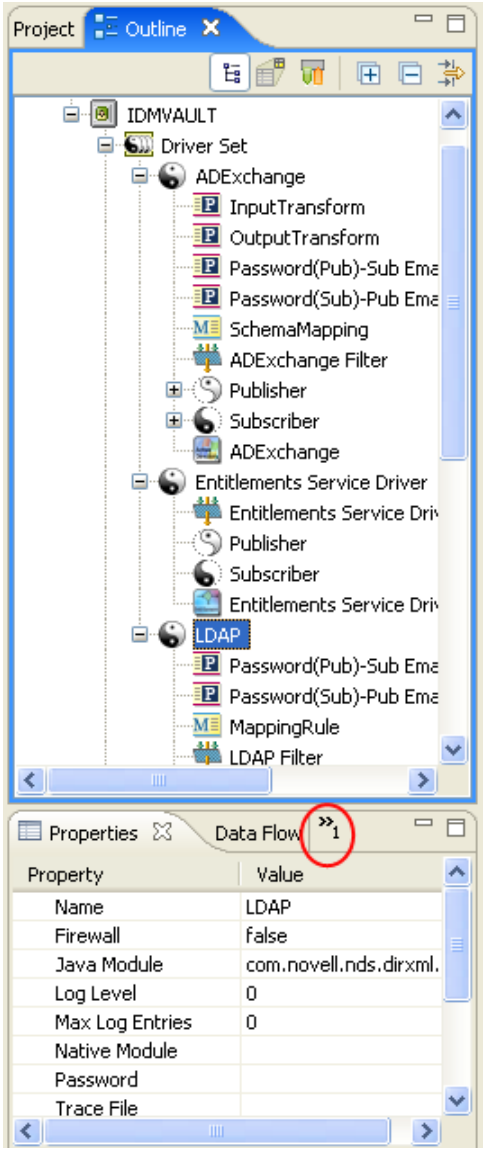


2 选择 " 策略集 " 选项卡。



如果未显示 " 策略集 " 选项卡：

1 单击双箭头。

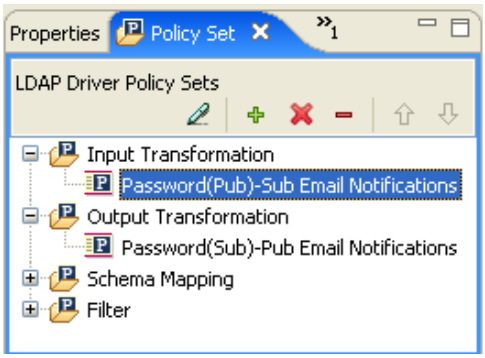


2 选择 " 策略集 "。

使用策略集

策略集包含一个工具栏和一个策略列表。

策略列表显示所选策略集中包含的所有策略。转换时将自上而下执行列表中的策略。工具栏中包含可以用于管理显示在列表中的策略的按钮和下拉菜单，其中包括编辑、添加、删除、重命名和更改策略的处理顺序。



策略集工具栏

策略集将显示策略的副本。根据所选项的不同，可以启用或禁用工具栏上的按钮。下面描述了不同的图标。

表 **2-1** 策略集工具栏

操作	说明
编辑策略	起动策略构建器。
在策略集中创建或添加新策略	起动添加策略向导。
去除和删除所选策略	从项目中删除策略。
从策略集中去除所选策略，但不删除	从所选策略集对象中去除策略，但并不删除策略。
在策略链中向上移动策略	将此策略在处理顺序中上移。
在策略链中向下移动策略	将此策略在处理顺序中下移。

键盘支持

可以通过击键也可以使用鼠标在策略集中移动。下表列出了支持的击键。

表 **2-2** 键盘支持

击键	说明
向上箭头	将所选策略在处理顺序中上移。
向下箭头	将所选策略在处理顺序中下移。
Delete	从项目中删除策略。
减号	将策略从所选策略集中去除，但并不删除。
加号	起动添加策略向导。
Ctrl+Z	复原上次操作。

击键	说明
Ctrl+Y	重做上次操作。

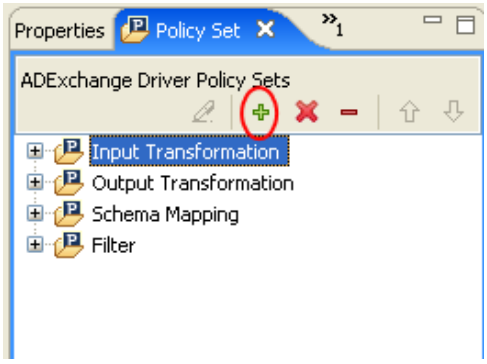
使用添加策略向导

单击工具栏中的 " 在策略集中创建或添加新策略 " 图标时，将起动添加策略向导。可以使用添加策略向导执行以下操作：

- “创建策略”在第 42 页
- “复制策略”在第 44 页
- “链接策略”在第 45 页

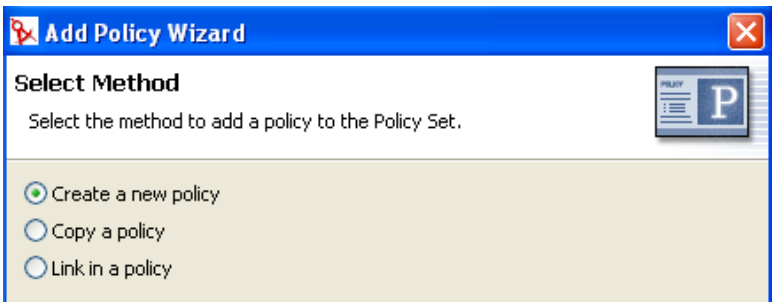
要起动添加策略向导，请执行以下操作：

1. 在 " 大纲 " 视图中选择一个驱动程序。
2. 在策略集中选择一个策略集项，然后单击工具栏中的 " 在策略集中创建或添加新策略 " 图标。

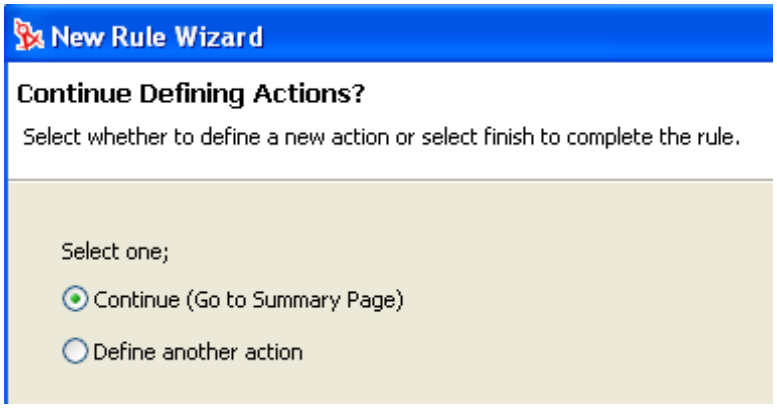


创建策略

1. 在 " 添加策略向导 " 中选择 " 创建新策略 "，然后单击 " 下一步 "。



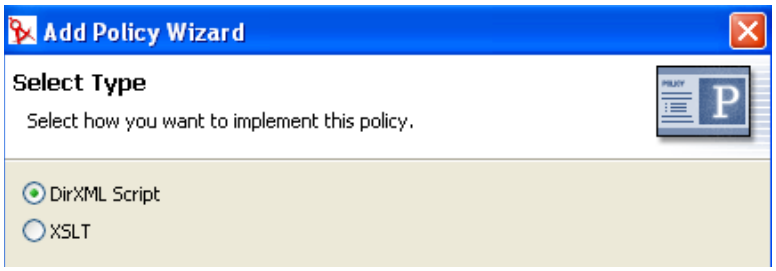
2 提供策略名称。



3 接受默认的树枝，或浏览至要在其中创建策略的驱动程序对象、发布者对象或订购者对象，并选择该对象。

此决定取决于您组织该策略的方式。默认情况下，策略位于起动添加策略向导时在 " 大纲 " 选项卡中的所选树枝对象下。例如，如果您在 " 大纲 " 选项卡中移动到发布者对象，然后向策略集中添加策略，则默认情况下会将该策略添加到发布者树枝。如果您希望在其它树枝中创建策略，则可以更改此设置。例如，可以在一个虚拟驱动程序下设置策略库，并将所有常用策略放置在此驱动程序下，然后只需从其它驱动程序引用这些策略即可。这样，该策略即为常用策略。如果需要更改策略，则只需更改一次。如果某策略不重复用于多个驱动程序，则通常在使用它的驱动程序或通道下创建该策略。

4 选择要实施的策略类型。默认策略类型为 "DirXML 底稿 "。如果您不想使用 DirXML® 底稿，可以选择 *XSLT* 或 " 纲要映射 "。



5 单击 " 完成 "。

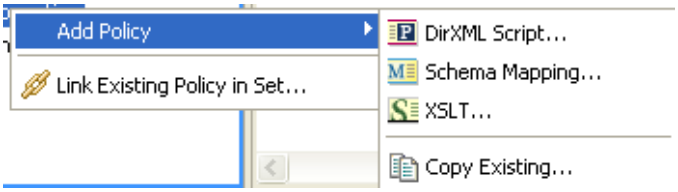
如果选定了纲要映射策略集，则可以使用纲要映射的一个附加选项。新策略将出现在展开的策略集中。

也可以通过右击策略集来添加策略。

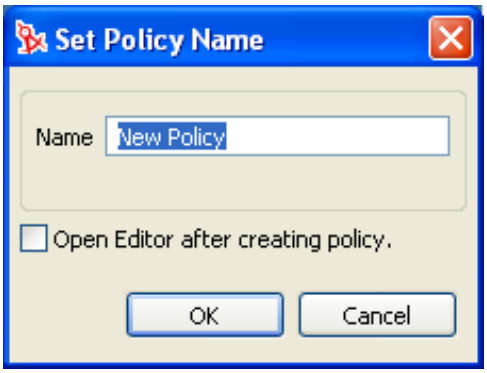
1 右击某策略集 （例如，输入转换集）。

2 选择 " 添加策略 "。

3 选择如何实施该策略："DirXML 底稿 "、 " 纲要映射 "、 *XSLT* 或 *Copy Existing*（复制现有）。



4 为该策略命名。

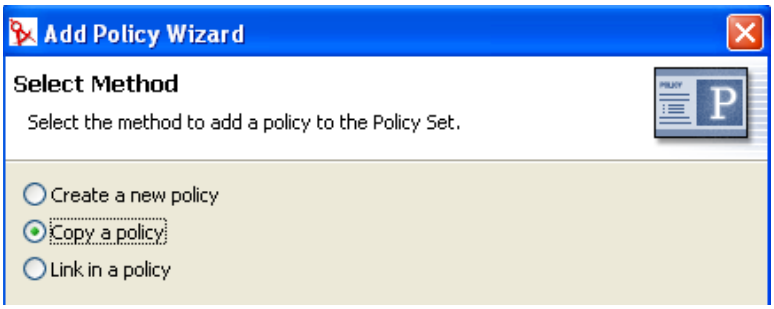


5 单击 *Open Editor after creating policy*（创建策略后打开编辑器）。

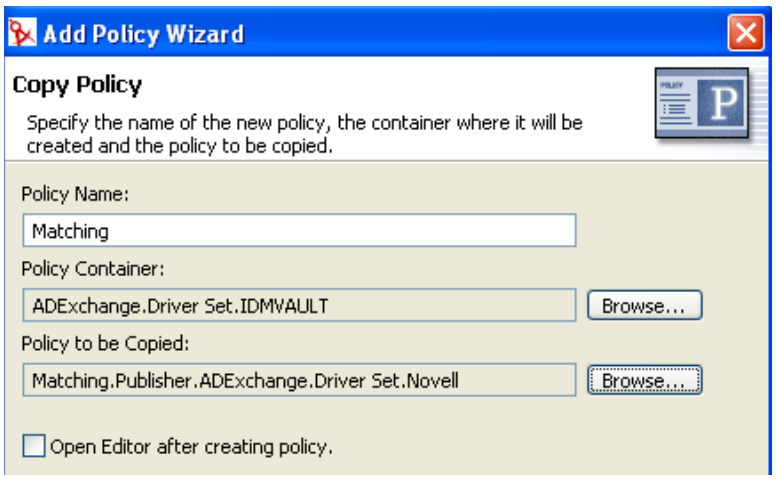
6 单击 " 确定 "。

复制策略

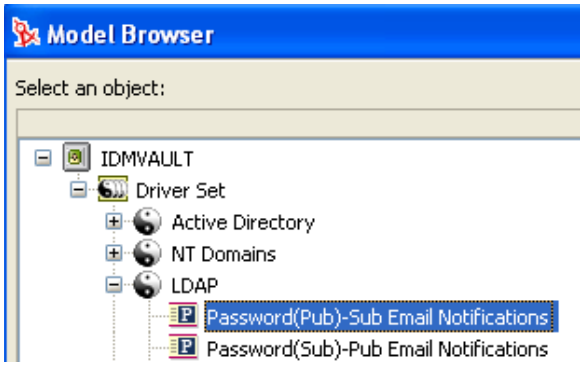
1 在 " 添加策略向导 " 中，选择 *Copy a policy*（复制策略），然后单击 " 下一步 "。



2 为该策略命名。



3 接受默认的树枝，或浏览至要在其中创建策略的驱动程序对象、发布者对象或订购者对象，并选择该对象。

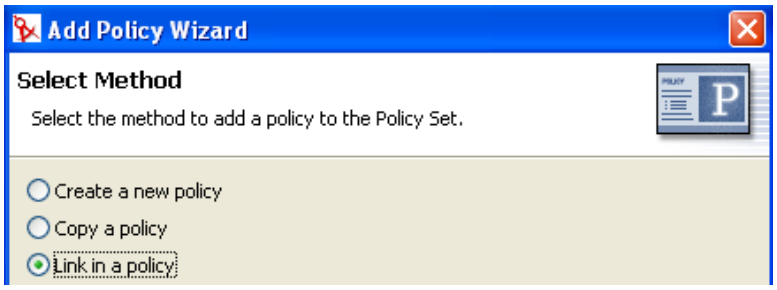


4 浏览至要复制的策略并选择该策略，然后单击 " 确定 "。

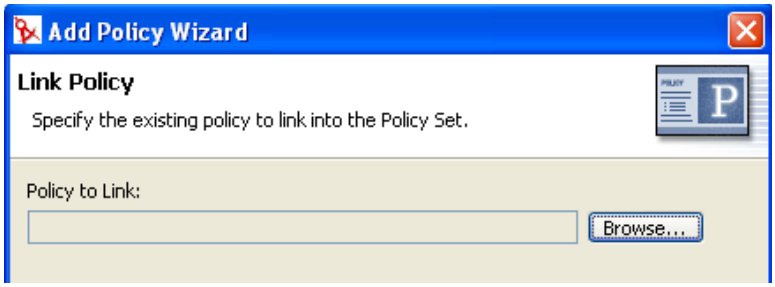
5 单击 " 完成 " 以创建所选策略的拷贝。

链接策略

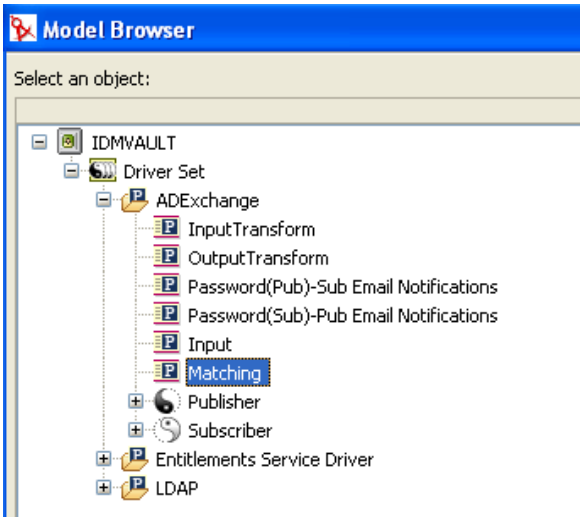
1 在 " 添加策略向导 " 中选择 *Link in a policy*（策略中的链接），然后单击 " 下一步 "。



2 单击 " 浏览 " 起动模型浏览器。



3 浏览至要链接到策略集中的策略对象并选择该对象，然后单击 " 确定 "。



将策略链接到策略集中不会创建新的策略对象。相反，此操作会对现有策略添加一个参照。此参照可以面向当前 Identity Vault 中的任何现有策略。此策略无需包含在当前驱动程序对象中，但对于将与之链接的策略集，策略类型必须是有效的。例如，不能将纲要映射策略链接到输入策略集中。

查看所有策略时不允许将策略链接到策略集中。

4 单击 " 完成 " 以链接到所选策略。

2.2.3 创建规则

规则定义为在某个已定义操作发生之前必须满足的一组条件。可以从条件组、条件和操作中创建规则。

可以以四种不同的方式创建规则：

- “创建新规则”在第 47 页
- “使用预定义规则”在第 51 页
- “包含现有规则”在第 52 页
- “从 XML 文件导入策略”在第 54 页

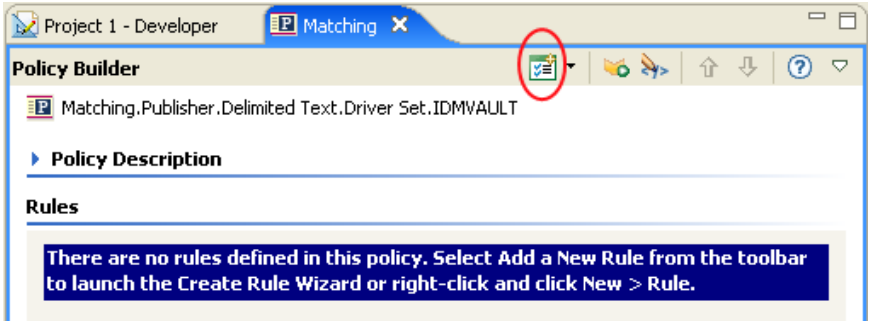
创建新规则

创建规则时也将创建条件组、条件和操作。每个规则都由条件、操作和自变量组成。有关更多信息，请在创建每个项目时单击 " 帮助 " 图标 。帮助文件中包含定义和使用项目的示例。

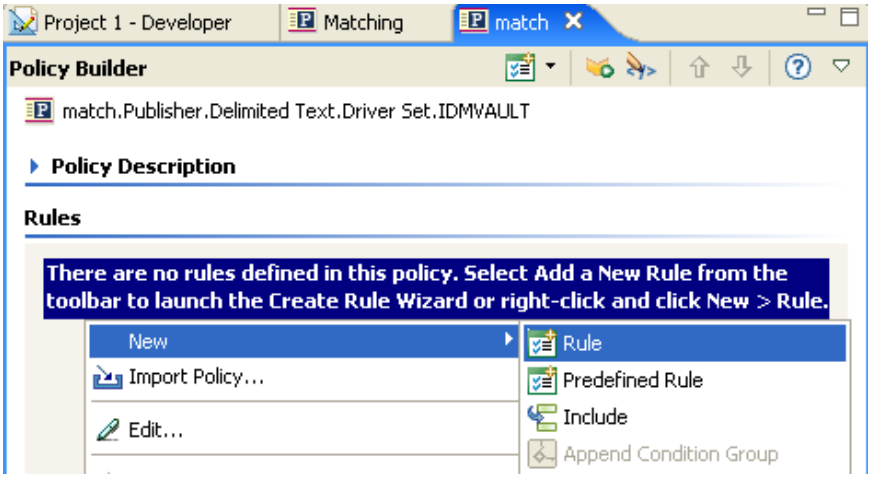
- “创建规则”在第 47 页
- “创建条件组”在第 50 页
- “创建条件”在第 51 页
- “创建操作”在第 51 页

创建规则

1 从 " 策略构建器 " 工具栏中选择 " 规则 "。

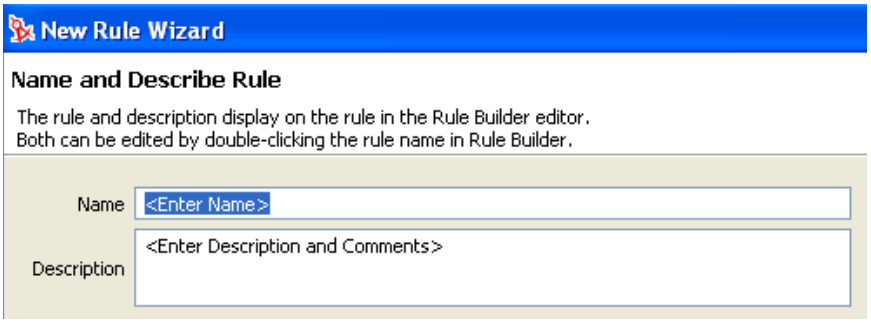


您还可以右击并单击 " 新建 ">" 规则 "。

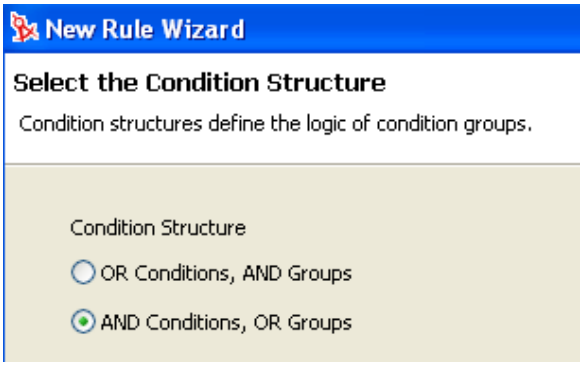


这两种操作都可以起动 Create Rule Wizard （创建规则向导）。

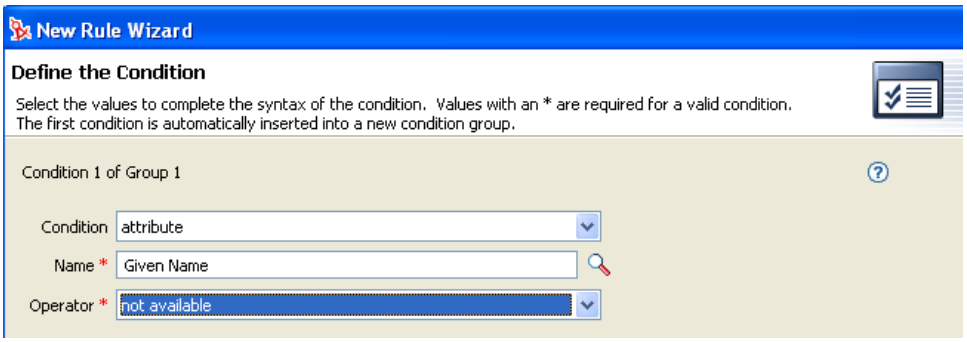
2 指定规则的名称，然后单击 " 下一步 "。



3 选择条件结构 （*OR Conditions, AND Groups* （OR 条件， AND 组）或 *AND Conditions, OR Groups* （AND 条件， OR 组）），然后单击 " 下一步 "。

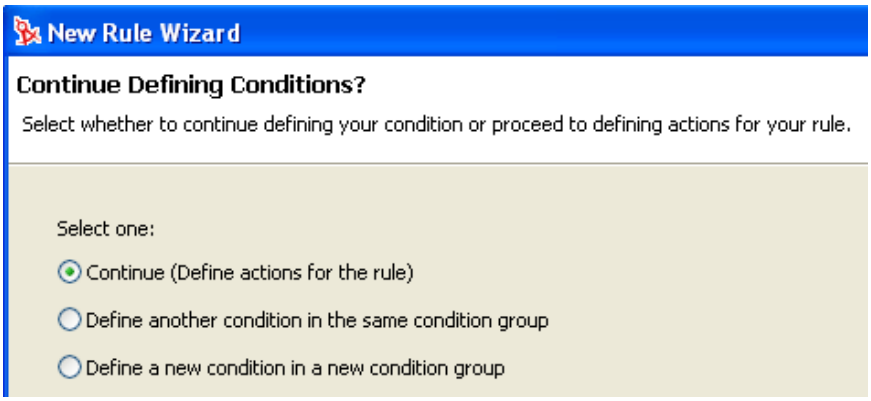


4 选择所需的条件，指定适当的信息，然后单击 " 下一步 "。

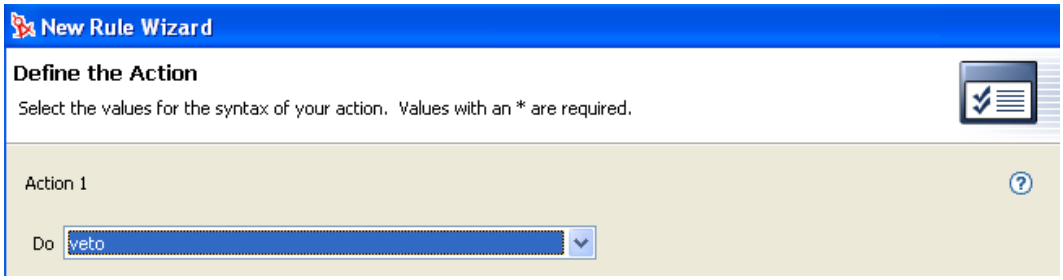


有关可以创建的所有条件的信息，请单击 " 帮助 " 图标 ⓘ。

5 此时可以定义附加条件或条件组。对于本示例，仅存在一个条件。选择 " 继续 "，然后单击 " 下一步 "。

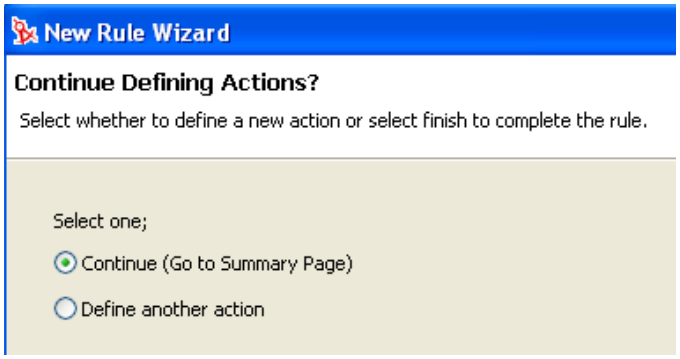


6 选择想要的操作，然后单击 " 下一步 "。



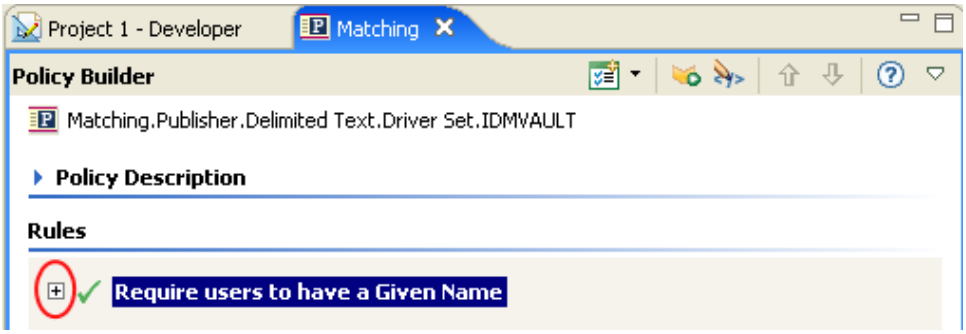
有关可以创建的每个操作的信息，请单击 " 帮助 " 图标 ⓘ。

7 此时可以定义附加操作。对于本示例，仅存在一个操作。选择 " 继续 "，然后单击 " 下一步 "。



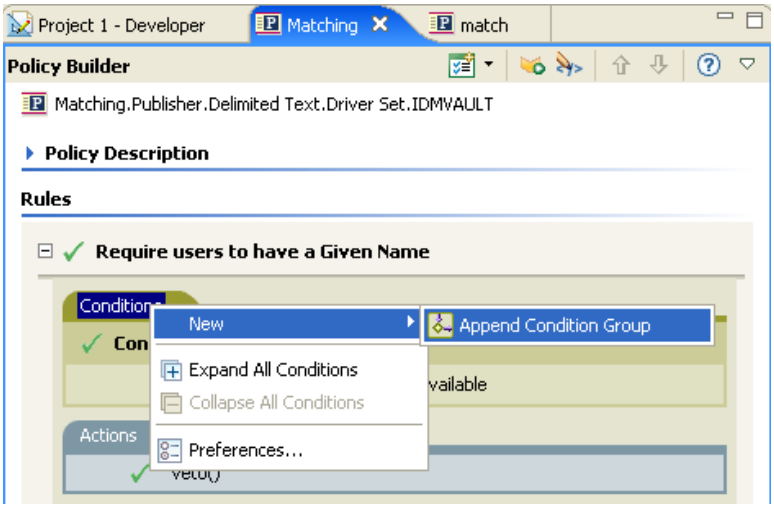
8 摘要页显示已创建的规则。单击 " 完成 " 完成对规则的创建。

可以通过单击加号或减号展开或折叠规则视图。



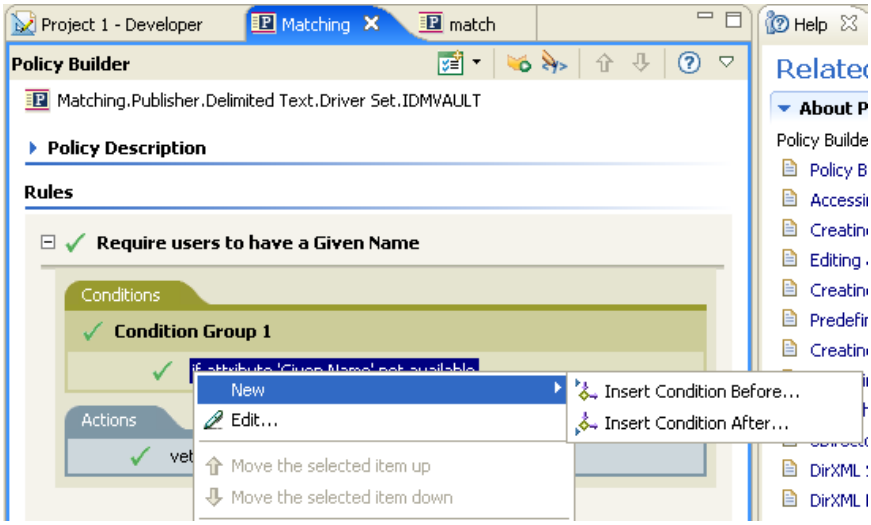
创建条件组

1 右击 " 条件 " 选项卡或右击 " 条件组 " 的名称，然后单击 " 新建 "> *Append Condition Group*（追加条件组）。



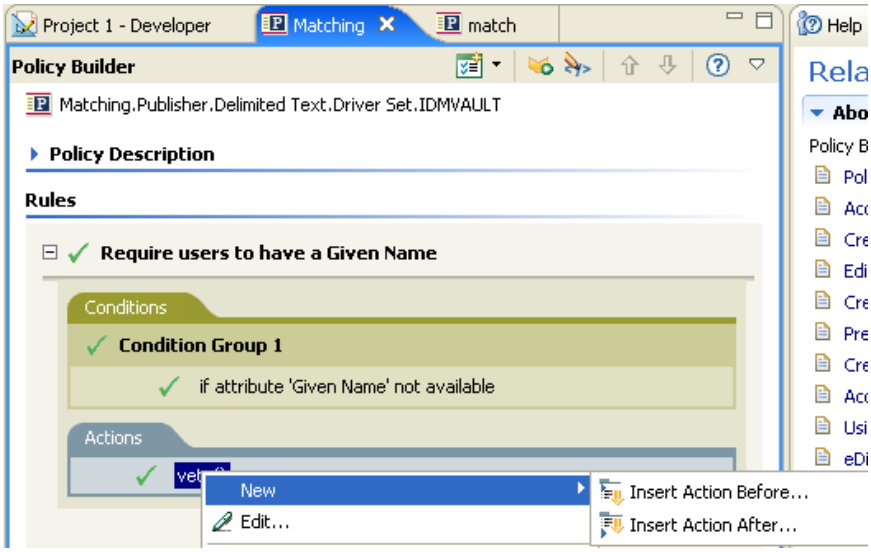
创建条件

1 右击该条件，然后单击 " 新建 "> *Insert Condition Before* （在之前插入条件）或者 *Insert Condition After* （在之后插入条件）。



创建操作

1 右击该操作，然后单击 " 新建 "> *Insert Action Before* （在之前插入操作）或 *Insert Action After* （在之后插入操作）。

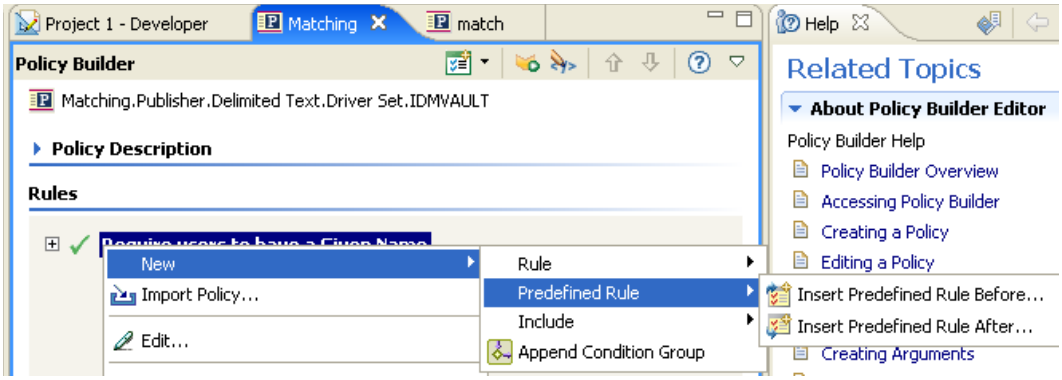


使用预定义规则

Designer 中包含预定义规则的列表。可以导入并使用这些规则，也可以创建自己的规则。

1 在 " 策略构建器 " 中右击，然后选择 " 新建 "> *Predefine Rules* （预定义规则） > *Insert Predefined Rule Before* （在之前插入预定义规则）或 *Insert Predefined Rule After* （在之后插入预定义规则）。

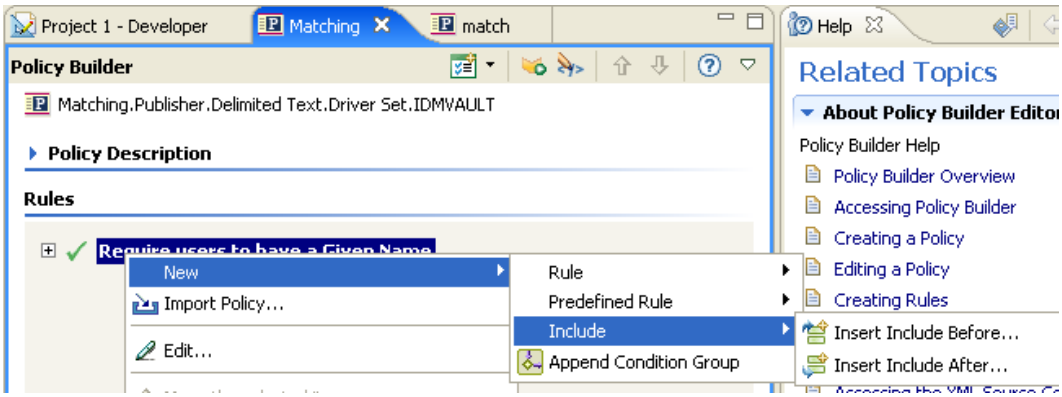
有关详细信息，请参见 “使用预定义规则”在第 68 页 。



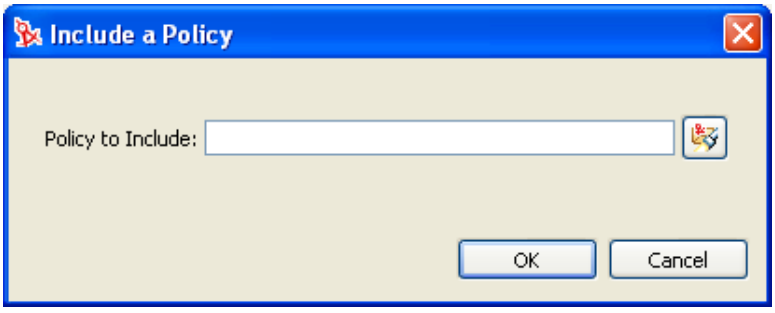
包含现有规则

Designer 允许包含其它策略的规则。

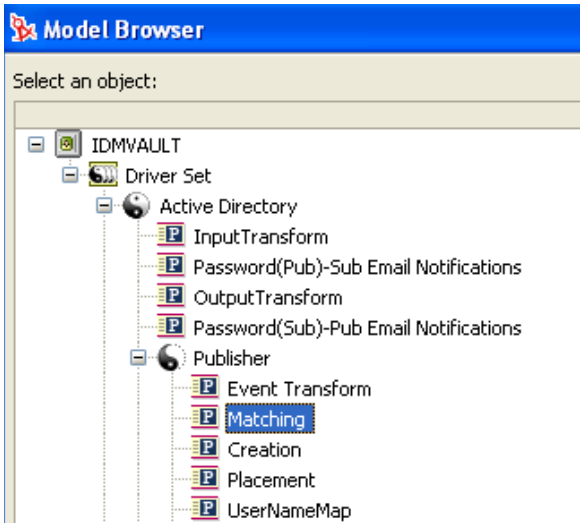
1 在 " 策略构建器 " 中右击，然后单击 " 新建 "> *Include* （包含） > *Insert Include Before* （在之前插入包含）或 *Insert Include After* （在之后插入包含）。



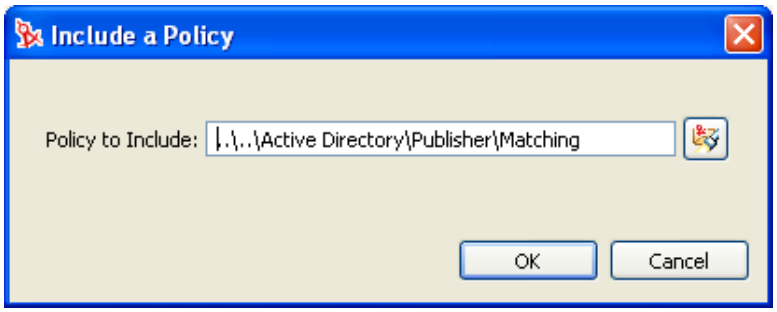
2 单击 " 浏览 " 图标。



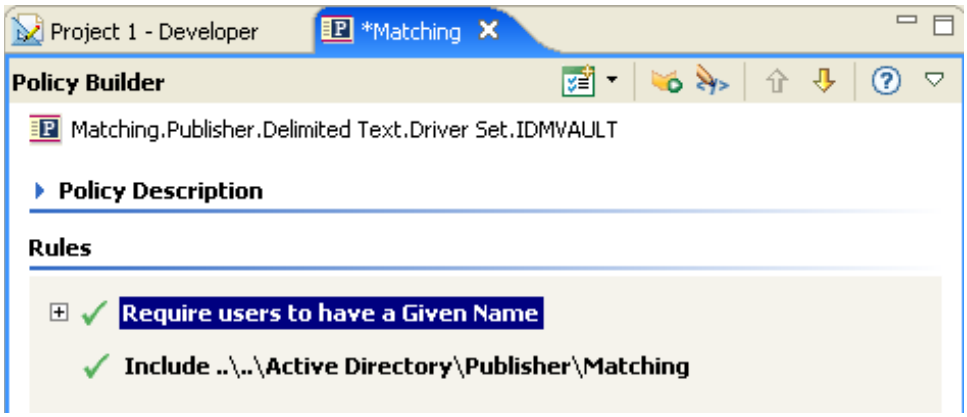
3 浏览至要包含的策略，然后单击 " 确定 "。



4 字段中随即填充该策略的路径。单击 " 确定 "。



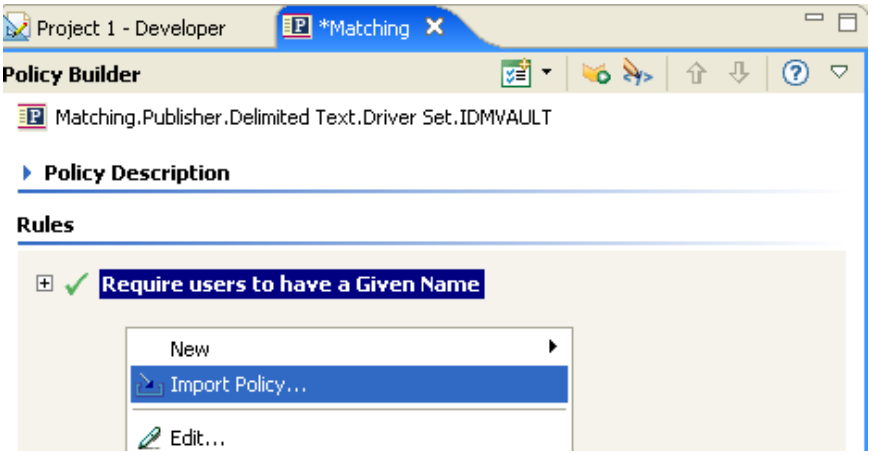
此规则是到原始规则的链接。因此不能在此位置中编辑该规则。可以访问原始规则进行更改。



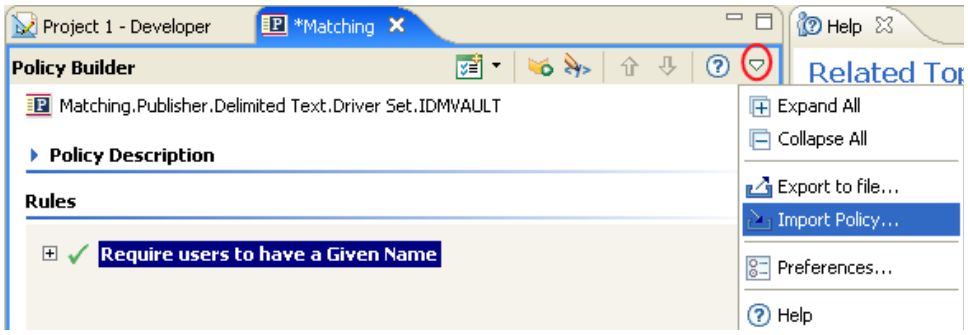
从 **XML** 文件导入策略

可以将规则和策略另存为 XML 文件。如果一个文件中包含希望使用的规则或策略，使用策略构建器可以导入此文件。

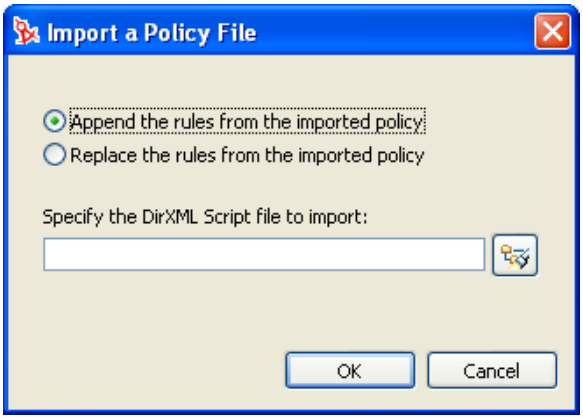
1 在 " 策略构建器 " 中，右击并选择 " 导入策略 "。



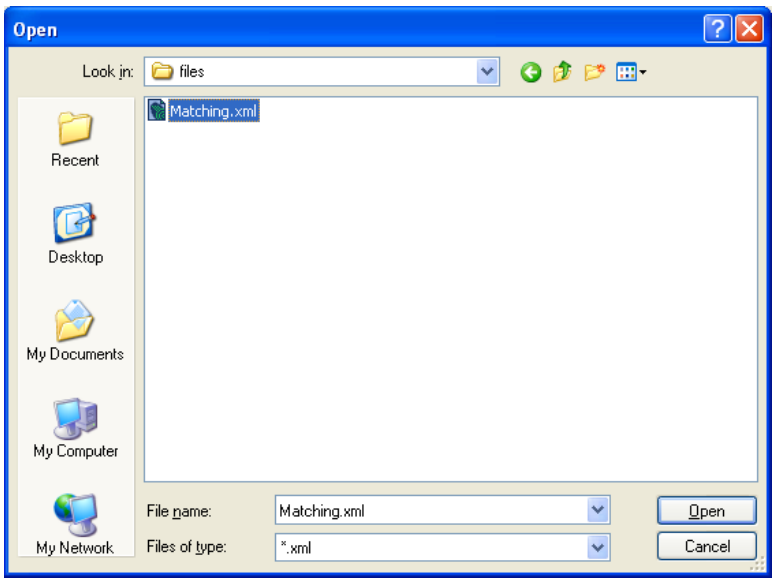
您还可以从工具栏的下拉列表中选择 " 导入策略 " 图标。



2 选择以下两个选项中的一个选项：*Append the rules from the imported policy*（从已导入的策略追加规则）或 *Replace the rules from the imported policy*（从已导入的策略替换规则）。



3 单击 " 浏览 " 图标并选择包含 DirXML 底稿的文件，然后单击 " 打开 "。



4 单击 " 确定 "。

2.2.4 创建自变量

自变量构建器提供了一个动态图形界面，使用此界面可以构造复杂的自变量表达式，以供在策略构建器中使用。要访问自变量构建器，请参见 “自变量构建器”在第 58 页 。

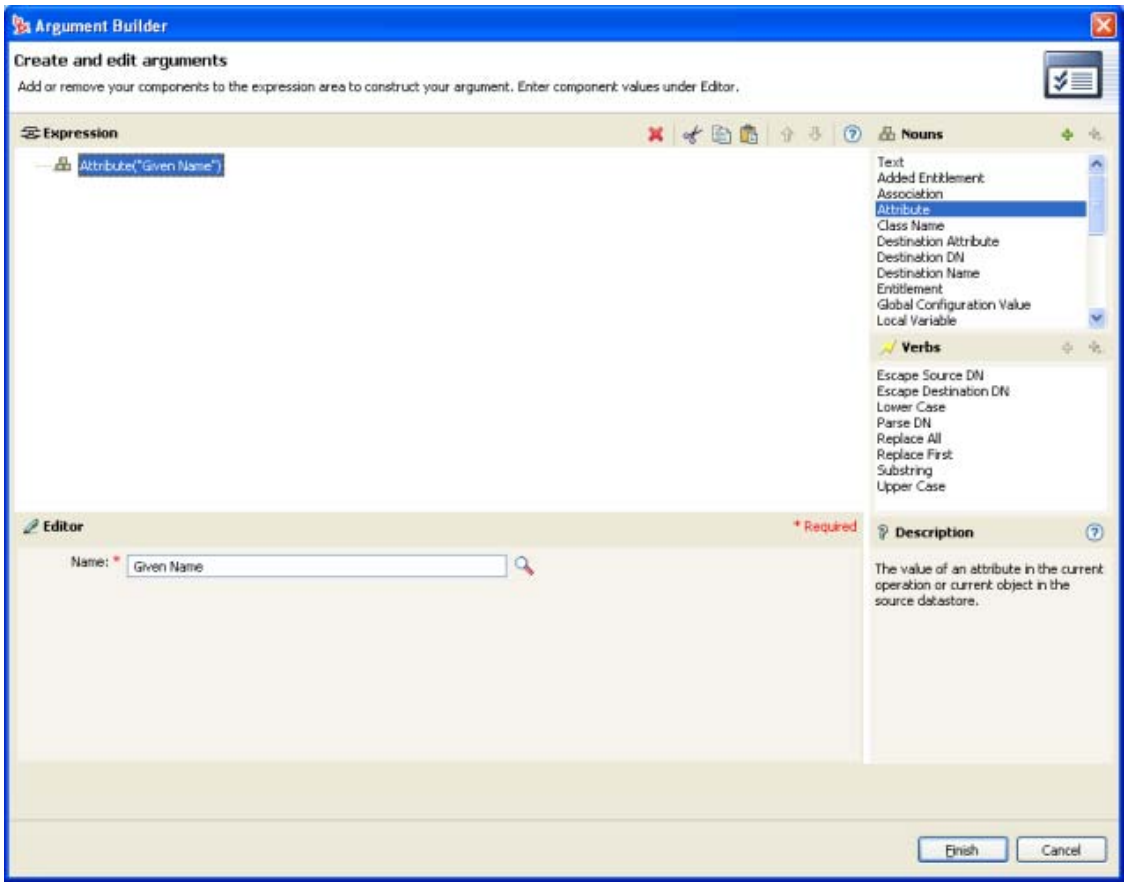
自变量由操作动态使用，并由运行时扩展的标记派生而来。

标记可分为两类：名词标记和动词标记。名词标记可扩展为由当前操作、源数据存储区或目标数据存储区，或一些外部源派生而来的值。动词标记用于修饰从属于它们的其它标记的已连接结果。

要定义表达式，请选择一个或多个名词标记 （值、对象、变量等），并将它们与动词标记（子字符串、转义符、大写和小写）组合，以构造自变量。组合多个标记可以构造复杂自变量。

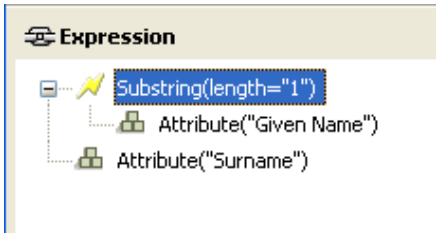
例如，如果要将自变量设置为某一特性值，可以先选择特性名词，再选择特性名称：

图 **2-2** 自变量构建器



如果只需要某特性的一部分，则可以将特性名词与子字符串动词进行组合：

图 **2-3** 表达式



在添加一个名词或动词后，可以在编辑器中提供值，然后即可紧随其后添加另一个名词或动词。无需刷新 " 表达式 " 窗格来应用更改；在执行下一项操作时将自动出现这些更改。

有关自变量构建器中可用标记的详细参照信息，请参见 “名词标记”在第 173 页 和 “动词标记”在第 186 页 。

虽然大部分自变量都是使用自变量构建器定义的，但在策略构建器中还存在若干其它构建器，可供条件编辑器和操作编辑器使用。每个构建器都可以递归地调用以下列表中的任意构建器：

- “操作构建器”在第 57 页
- “自变量构建器”在第 58 页
- “匹配特性构建器”在第 59 页
- “操作自变量组件构建器”在第 60 页
- “自变量值列表构建器”在第 61 页
- “命名字符串构建器”在第 62 页
- “条件自变量组件构建器”在第 63 页
- “模式字符串构建器”在第 64 页

下面的信息描述如何访问各个构建器。

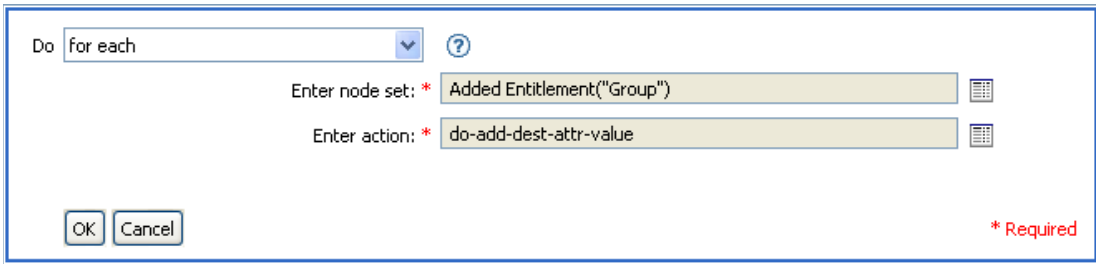
操作构建器

若要起动操作构建器，请选择以下两种操作之一，然后单击 " 编辑自变量 " 图标 。

- 对于每个
- 实施权利

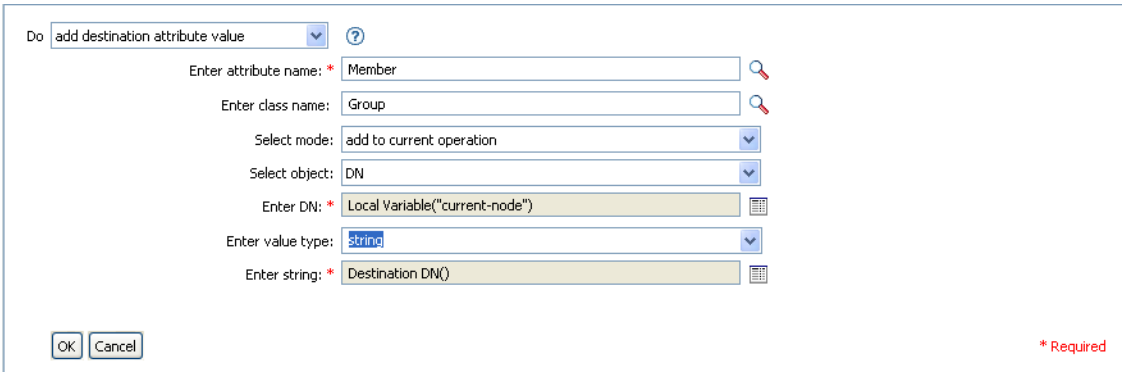
在下面的示例中，对于每一个要添加到当前操作的组权利，都将执行 " 添加目标特性值 " 操作。

图 ***2-4*** 对于每个操作



若要定义 " 添加目标特性值 " 操作，请单击起动 " 操作构建器 " 的图标。在操作构建器中定义所需的操作。在下面的示例中，将 member （成员）特性添加到每个已添加的 " 组 " 权利的目标对象中。

图 **2-5** 自变量操作构建器



自变量构建器

若要起动自变量构建器，请选择以下操作之一，然后单击 " 编辑自变量 " 图标 。

- “添加关联”在第 131 页
- “添加目标特性值”在第 132 页
- “添加目标对象”在第 133 页
- “添加源特性值”在第 135 页
- “追加 XML 文本”在第 137 页
- “清除目标特性值”在第 138 页 （所选对象为 DN 或 Association 时）
- “清除源特性值”在第 139 页 （所选对象为 DN 或 Association 时）
- “删除目标对象”在第 142 页 （所选对象为 DN 或 Association 时）
- “删除源对象”在第 143 页 （所选对象为 DN 或 Association 时）
- “查找匹配对象”在第 143 页
- “对于每个”在第 145 页
- “移动目标对象”在第 148 页
- “移动源对象”在第 149 页
- “重新设置操作特性的格式”在第 150 页
- “去除关联”在第 151 页
- “去除目标特性值”在第 152 页
- “去除源特性值”在第 152 页
- “重命名目标对象”在第 153 页 （所选对象为 DN、 Association 或 Enter String 时）
- “重命名源对象”在第 154 页 （所选对象为 DN、 Association 或 Enter String 时）
- “设置目标特性值”在第 158 页 （所选对象为 DN 或 Association，输入值类型不为 structured 时）
- “设置目标口令”在第 159 页
- “设置局部变量”在第 160 页

- “设置操作关联”在第 161 页
- “设置操作的类名称”在第 162 页
- “设置操作目标 DN”在第 162 页
- “设置操作属性”在第 163 页
- “设置操作源 DN”在第 164 页
- “设置操作模板 DN”在第 164 页
- “设置源特性值”在第 165 页
- “设置源口令”在第 166 页
- “设置 XML 特性”在第 168 页
- “状态”在第 168 页
- “跟踪讯息”在第 170 页

1 使用名词或动词创建自变量。

可以结合使用名词和动词来创建所需的自变量。

2 单击 " 完成 "。

匹配特性构建器

使用匹配特性构建器可以选择 “查找匹配对象”在第 143 页 操作所使用的特性和值，以确定某个匹配对象是否存在于数据存储区中。

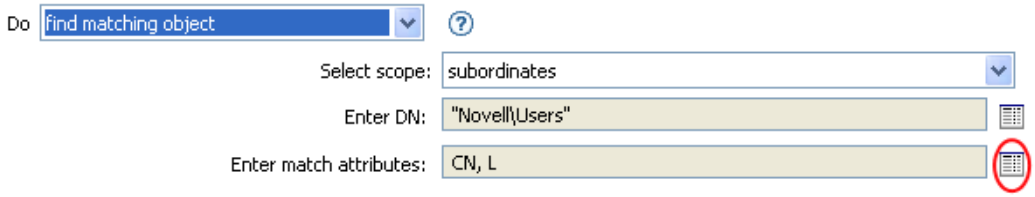
例如，如果要根据常用名和位置匹配用户，则可以：

1 选择 " 查找匹配对象 " 操作。

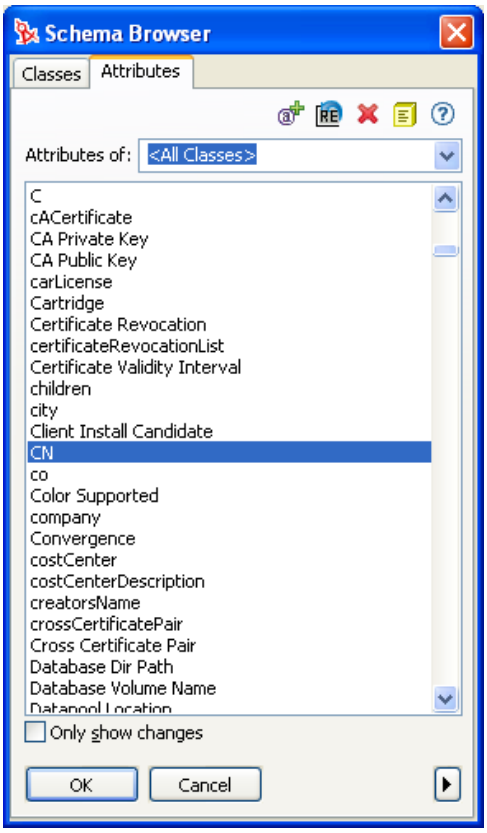
2 选择匹配对象的搜索范围。从 *entry* （项）、 *subordinates* （从属项）或 *subtree* （子树）中选择。

3 指定搜索起点的 DN。

4 单击 *Edit match attributes* （编辑匹配特性）图标 以起动匹配特性构建器。



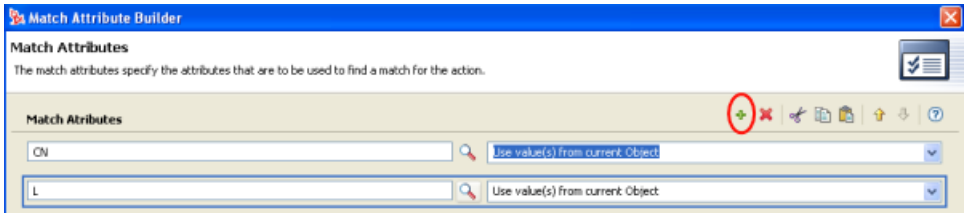
5 单击 " 浏览特性 " 图标以起动 Schema Browser （纲要浏览器）。



6 单击 " 特性 " 选项卡，然后浏览至所需的特性并选择该特性。

7 单击 " 确定 "。

如果要添加多个特性，请单击 " 追加新项目 " 图标 以添加其它行。



8 单击 " 完成 "。

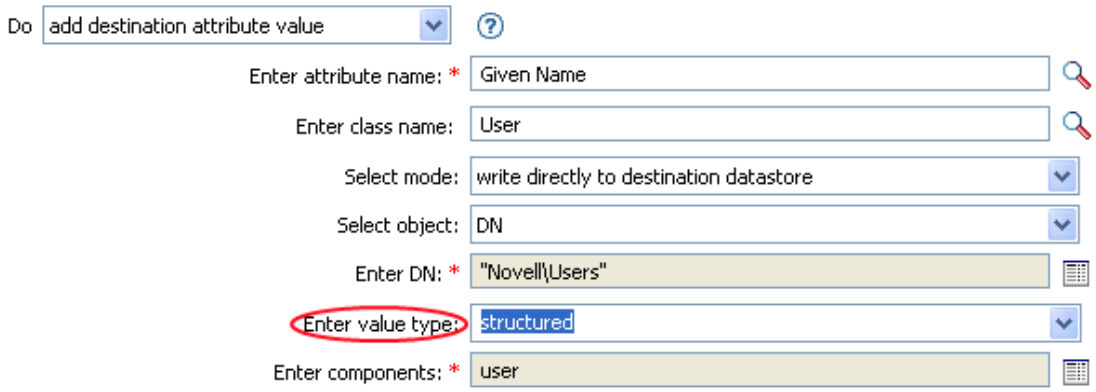
操作自变量组件构建器

若要起动操作自变量组件构建器，请在 " 输入值类型 " 选择为 *structured* 时选择以下操作之一，然后单击 *Edits components* （编辑组件）图标 。

- “添加目标特性值”在第 132 页
- “添加源特性值”在第 135 页
- “重新设置操作特性的格式”在第 150 页
- “去除目标特性值”在第 152 页

- “去除源特性值”在第 152 页
- “设置目标特性值”在第 158 页
- “设置源特性值”在第 165 页

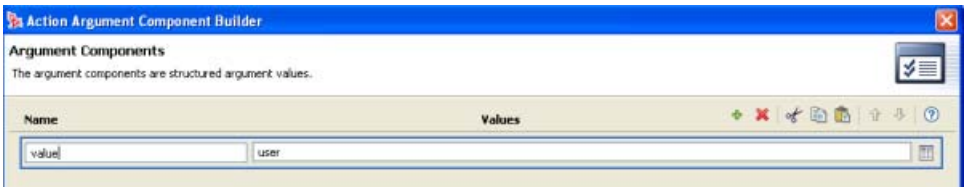
图 **2-6** 添加目标特性值操作



1 将值类型设置为 structured 后单击 " 编辑组件 " 图标 。

2 创建操作组件的值。

可以输入值，或单击 " 编辑自变量 " 图标，从而在自变量构建器中创建值。



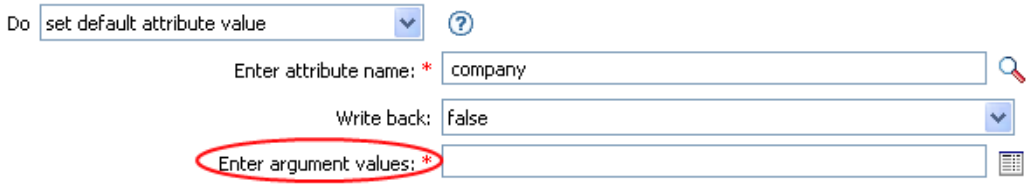
3 单击 " 完成 "。

自变量值列表构建器

若要起动自变量值列表构建器，请选择以下操作，然后单击 " 编辑自变量 " 图标 。

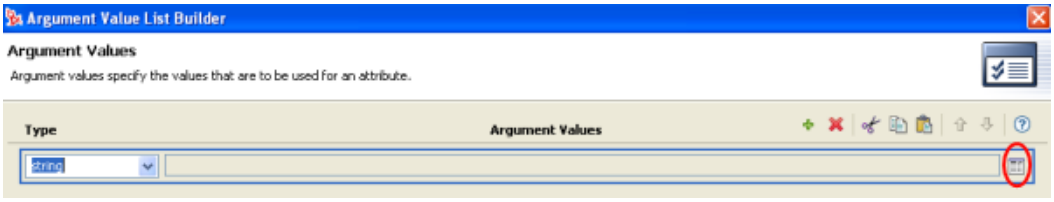
- 设置默认特性值

图 **2-7** 设置默认特性值



1 选择值的类型：*counter*、 *dn*、 *int*、 *interval*、 *octet*、 *state*、 *string*、 *structured*、 *teleNumber*、 *time*。

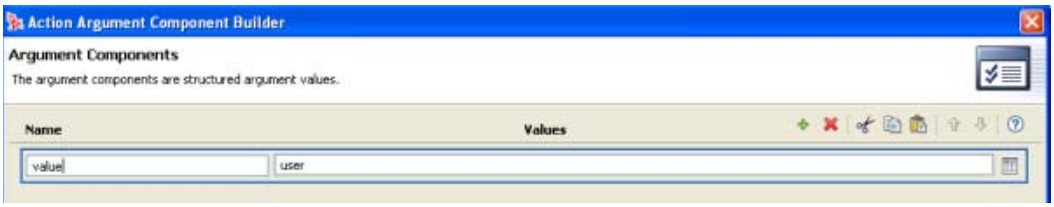
2 单击 *Edit the value lists*（编辑值列表）图标。



3 单击 " 编辑自变量 " 图标。

4 创建操作组件的值。

可以输入值，或单击 " 编辑自变量 " 图标，从而在自变量构建器中创建值。



5 单击 " 完成 "。

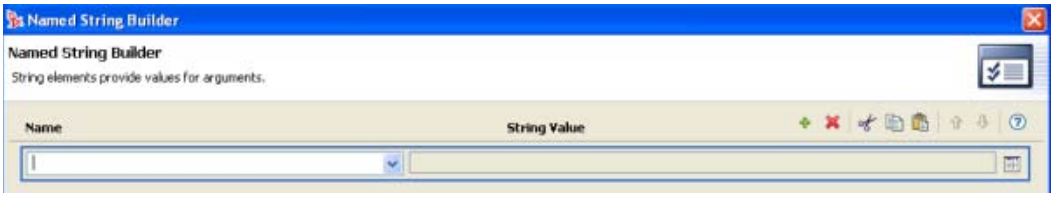
命名字符串构建器

若要起动命名字符串构建器，请选择以下操作之一，然后单击 " 编辑字符串 " 图标。

- 生成事件
- 发送电子邮件
- 通过模板发送电子邮件

1 从下拉列表中选择字符串的名称。

2 通过单击 " 编辑自变量 " 图标 以起动自变量构建器来创建字符串的值。



3 单击 " 完成 "。

对于 " 发送电子邮件 " 操作，命名字符串与电子邮件的要素相对应：

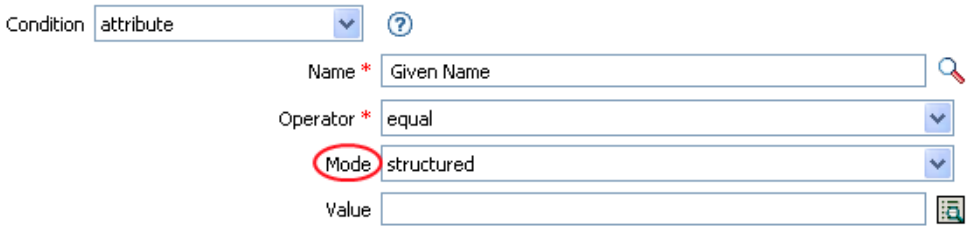
图 *2-8* 发送邮件操作中的电子邮件要素

与起动命名字符串构建器的操作对应的帮助文件中，包含可能出现的值的完整列表。

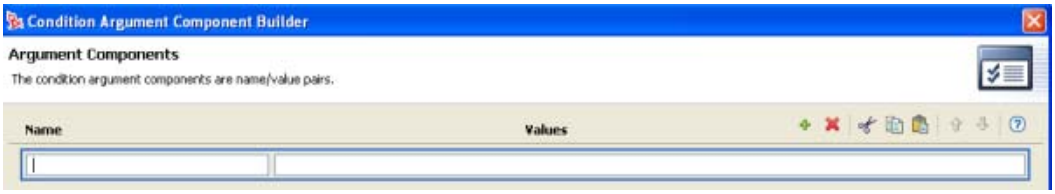
条件自变量组件构建器

若要起动条件自变量组件构建器，请选择以下条件之一，这些条件的 " 方式 " 必须选择 structured，这样就可以看到 " 起动自变量组件构建器 " 图标 。

- If Attribute
- If Destination Attribute
- If Source Attribute
- If Operation Attribute



1 指定条件组件的名称和值。

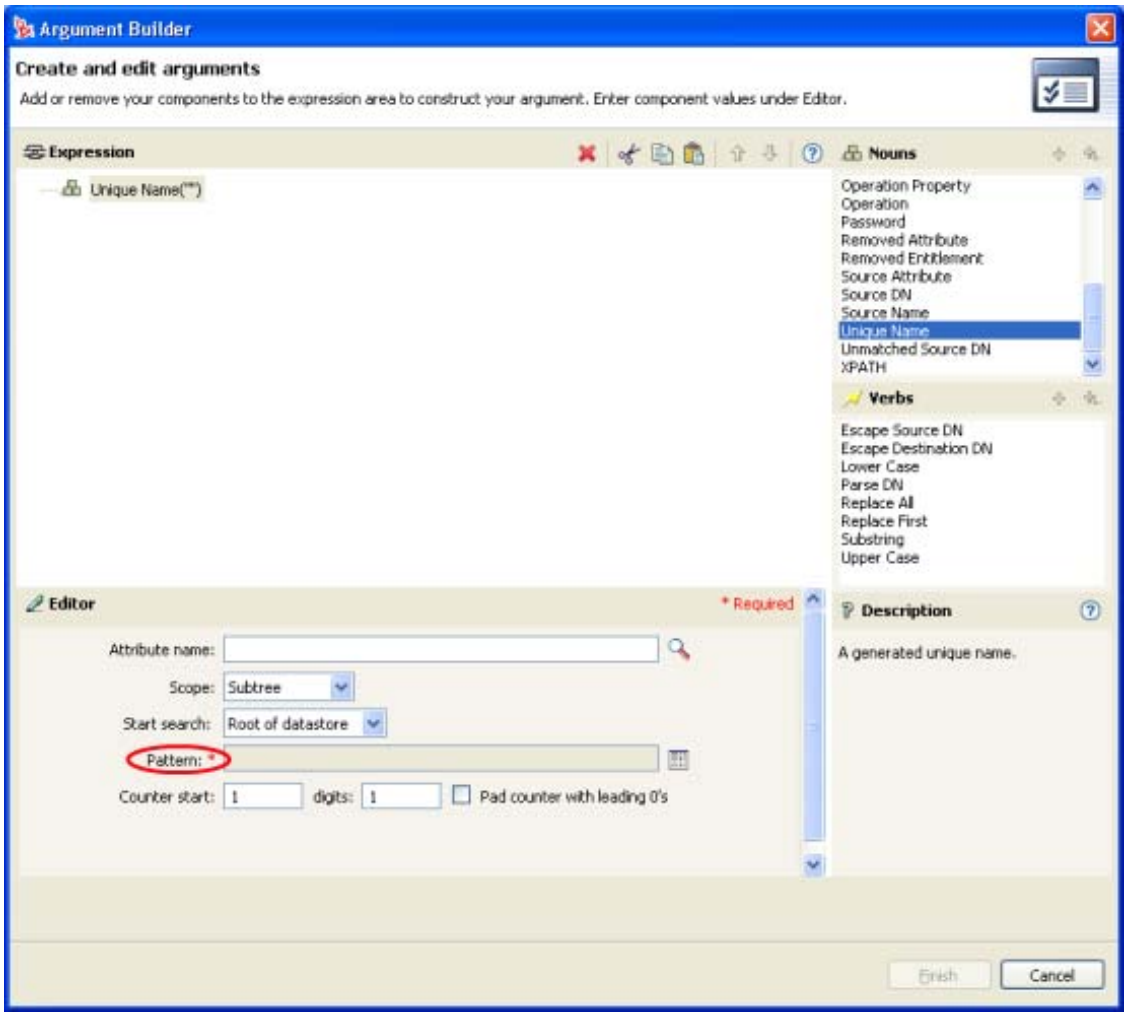


2 单击 " 完成 "。

模式字符串构建器

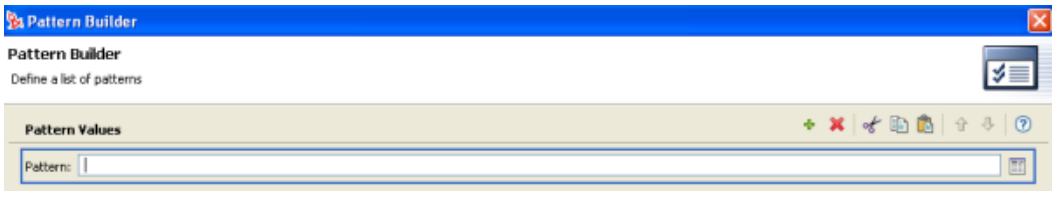
可以在选择唯一名称标记后从自变量构建器的编辑器中起动模式字符串构建器。自变量构建器的 " 编辑器 " 窗格中将显示 " 模式 " 字段，单击此字段可以起动模式字符串构建器。

图 **2-9** 自变量构建器中的唯一名称标记



1 单击 *Edit patterns* （编辑模式）图标 以起动模式构建器。

2 指定模式或单击 " 编辑自变量 " 图标 以使用自变量构建器创建模式。

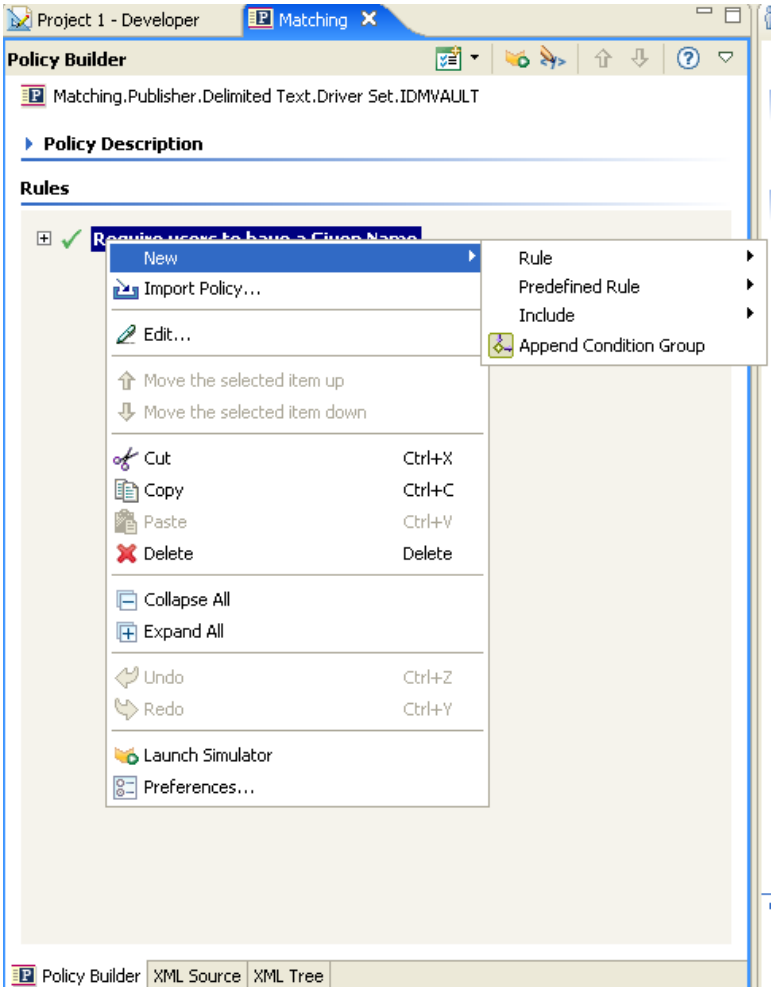


3 单击 " 完成 "。

2.2.5 编辑策略

策略构建器允许您创建和编辑策略。您可以拖放规则、条件和操作。有关其它操作，请访问 " 策略构建器 " 工具栏。要显示上下文菜单，请右击某项目。

图 ***2-10*** 策略构建器上下文菜单和工具栏



策略构建器中的操作和菜单项

下表包含策略构建器中的不同操作和菜单项的列表。

表 ***2-3*** 策略构建器操作和菜单项

操作	说明
Collapse All （全部折叠）	折叠所有展开的规则。
复制	将所选项目复制到剪贴板。
Copy and drop （复制并拖动）	选择项目，按 Ctrl，然后拖动该项目。
剪切	剪切所选项目并将其复制到剪贴板。

操作	说明
删除	删除所选项目。
禁用	禁用规则、条件或操作。单击 ✓ 图标。
拖放	允许您选择某一项目，然后将其重新定位。选择项目，然后将其拖动至新位置。
编辑	允许您编辑所选项目。若要打开规则构建器，请选择某一规则，然后单击 " 编辑 "。
启用	启用规则、条件或操作。单击 ⛔ 图标。
全部展开	展开所有规则以便可以查看每个规则的条件和操作。
导入策略	从文件系统导入一个策略并将其追加至策略中，或替换策略的所有规则。
Launch Simulator （起动模拟器）	起动策略模拟器。
Move and drop （移动并拖动）	允许您选择并移动项目。选择项目，然后拖动它。
Move the selected item down （将选定的项目下移）	在策略列表中下移该项目。
Move the selected item up （将选定的项目上移）	在策略列表中上移该项目。
新建 > 条件组	在某个选定的项目之后创建新的条件组。
新建 > 包含	在某个选定的项目之后创建新的包含。
新建 > 预定义规则	插入预定义规则。
新建 > 规则	在某个选定的项目后创建新规则。
粘贴	在所选项目之后粘贴剪贴板的内容。
自选设置	允许您更改显示信息的方式。
选择	单击任意项目即可将其选定。

键盘支持

可以通过击键或使用鼠标在策略构建器中移动。下表列出了支持的击键。

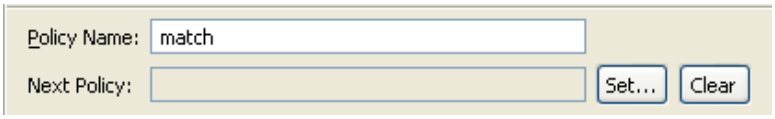
表 ***2-4*** 策略构建器中的键盘支持

击键	说明
Ctrl+C	将所选项目复制到剪贴板。
Ctrl+X	剪切所选项目并将其添加到剪贴板。
Ctrl+V	在所选项目之后粘贴剪贴板的内容。
Delete	删除所选项目。
向左箭头	折叠规则节点。
向右箭头	展开规则节点。

击键	说明
向上箭头	向上导航。
向下箭头	向下导航。
Ctrl+Z	复原
Ctrl+Y	重做

重命名策略

1. 在 " 大纲 " 视图中，选择想要重命名的策略。
2. 右击并选择 " 属性 "。
3. 在 " 策略名称 " 字段中更改策略的名称。



4. 单击 " 确定 "。

保存您的工作结果

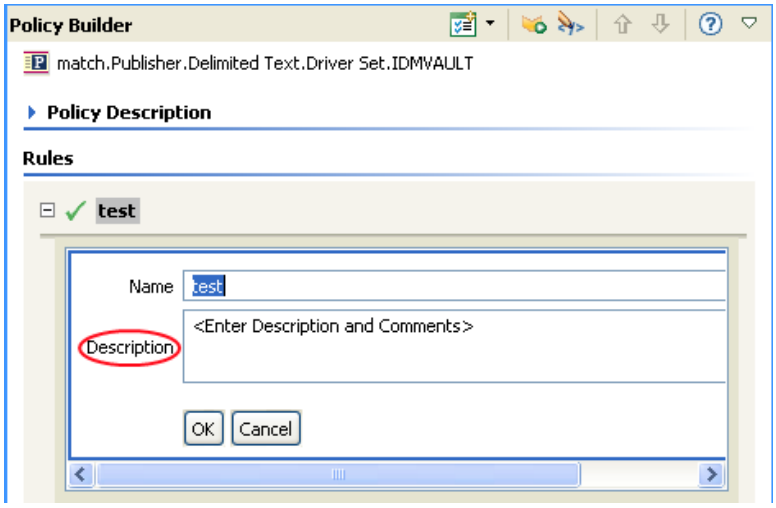
执行下列操作之一：

- 单击主菜单中的 " 文件 ">" 保存 " （或 " 保存全部 "）。
- 通过单击 " 编辑器 " 选项卡中的 *X* 来关闭编辑器。
- 从主菜单的 " 文件 " 菜单中选择 " 关闭 "。
- 按 Ctrl+S。

策略说明

使用 " 说明 " 字段可以添加有关策略功能的注释。

图 ***2-11*** 策略说明

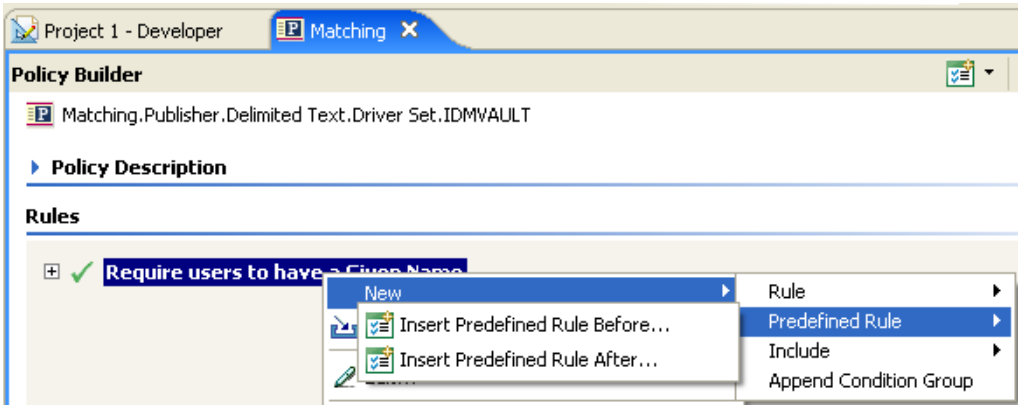


2.2.6 使用预定义规则

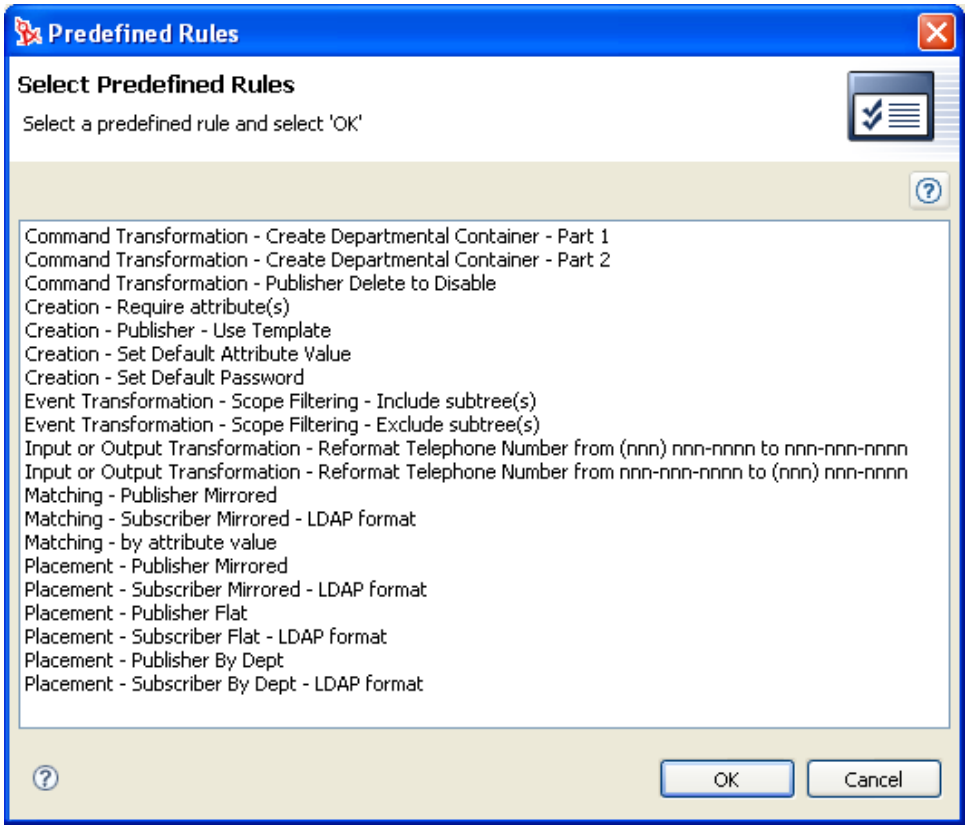
Designer 中包含二十个预定义规则。可以导入并使用这些规则，也可以创建自己的规则。这些规则包含管理员使用的常见任务。需要提供特定于个人环境的信息才可以自定义这些规则。

要访问这些预定义规则，请执行以下操作：

1 在 " 策略构建器 " 中，右击并选择 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。



" 预定义规则 " 对话框显示可用规则的列表。



命令转换 **-** 创建部门树枝 **-** 第 **1** 部分和第 **2** 部分

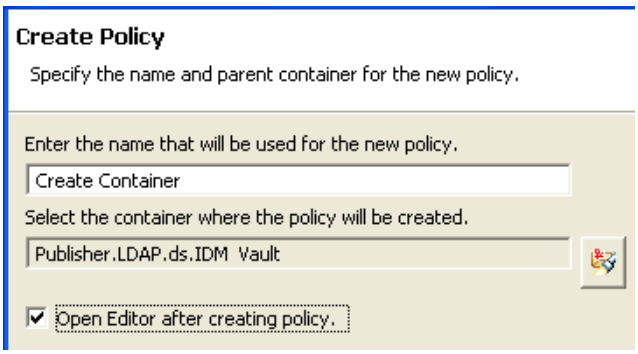
如果目标数据存储区中没有部门树枝，请进行创建。对驱动程序中的命令转换策略实施此规则。可以对订购者通道或发布者通道单独实施此规则，也可以对这两个通道同时实施此规则。

使用预定义规则包含两个步骤：在命令转换策略集中创建一个策略，然后导入预定义规则。如果要为已有的命令转换策略添加此规则，请转到 “导入预定义规则”在第 70 页 。

创建策略

1. 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道或订购者通道。
2. 在 " 策略集 " 视图中选择 " 命令转换 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 ✚ 以创建新策略。
3. 单击 " 创建新策略 "，然后单击 " 下一步 "。
4. 为该策略命名。

5 根据所填充的位置在驱动程序中放置此策略。



6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。

7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。

8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?"（编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的命令转换策略。

导入预定义规则

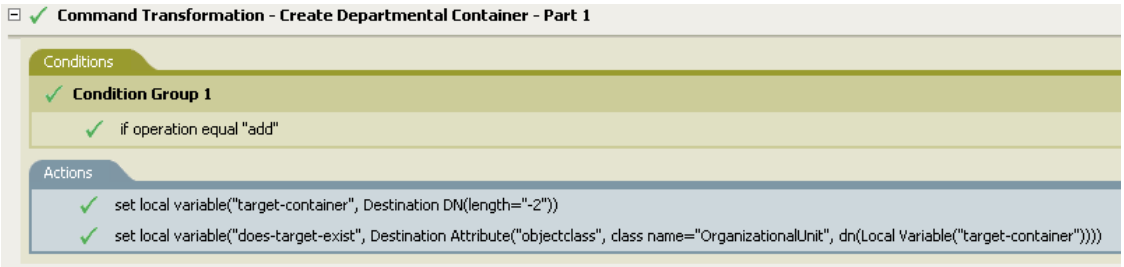
1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。

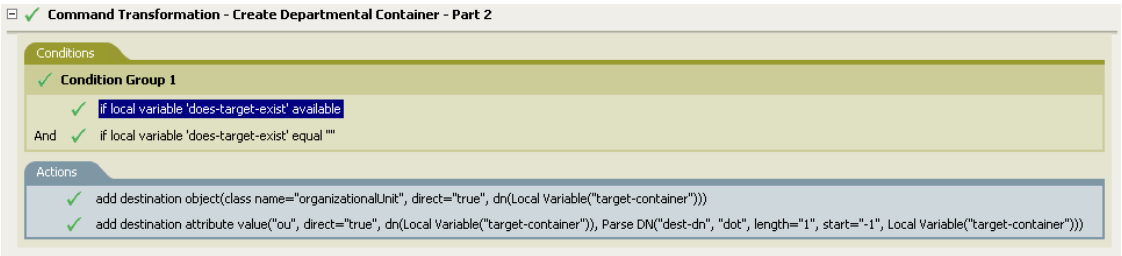
2 选择 *Command Transformation - Create Department Container - Part 1*（命令转换 - 创建部门树枝 - 第 1 部分），然后单击 " 确定 "。

3 右击 " 策略构建器 "，然后单击 " 新建 ">" 预定义规则 "> " 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。

4 选择 *Command Transformation - Create Department Container - Part 2*（命令转换 - 创建部门树枝 - 第 2 部分），然后单击 " 确定 "。

5 单击 " 文件 ">" 保存 "，保存该规则。





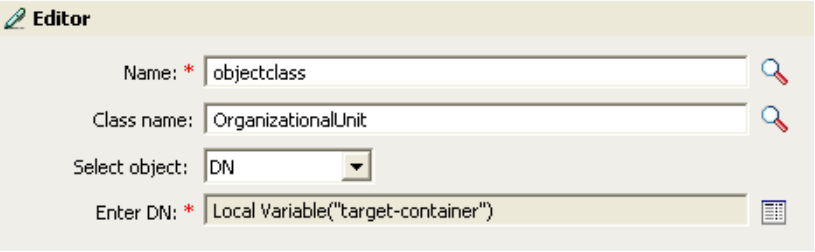
在特定于您的环境的规则中，没有要更改的信息。

重要：确保按顺序列出规则。必须先执行第 1 部分，然后再执行第 2 部分。

该规则的工作方式

如果对象的目标位置不存在，则使用此规则。此规则将创建树枝并将对象放置在树枝中，这样就不会因为无法放置对象而被禁止。

第 1 部分：查找 Add 事件。当发生 Add 事件时，将设置两个局部变量。第一个局部变量名为 target-container。将 target-container 的值设为目标 DN。第二个局部变量名为 does-target-exist。将 does-target-exist 的值设为 objectclass 的目标特性值。将类设置为 OrganizationalUnit。而 OrganizationalUnit 的 DN 设置为局部变量 target-container。



第 2 部分：检查局部变量 does-target-exist 是否可用。同时还检查局部变量 does-target-exist 的值是否设为空值。如果该值为空，则创建一个组织单元对象。将组织单元的 DN 设为局部变量 target-container 的值。同时还将添加 OU 特性的值。将 OU 特性的值设为局部变量 target-container。它使用源格式作为目标 DN 且目标格式为点格式。

命令转换 - 发布者删除 - 禁用

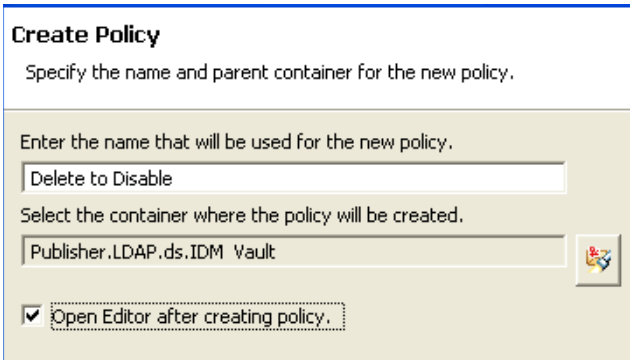
将用户对象的删除事件转换至禁用用户对象。对驱动程序中的命令转换策略实施此规则。需要对发布者通道实施此规则。

使用预定义规则包含两个步骤：在命令转换策略集中创建一个策略，然后导入预定义规则。如果要为已有的命令转换策略添加此规则，请转到 “导入预定义规则”在第 72 页 。

创建策略

1. 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道。
2. 在 " 策略集 " 视图中选择 " 命令转换 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 ✚ 以创建新策略。
3. 单击 " 创建新策略 "，然后单击 " 下一步 "。
4. 为该策略命名。

5 根据所填充的位置在驱动程序中放置此策略。



6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。

7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。

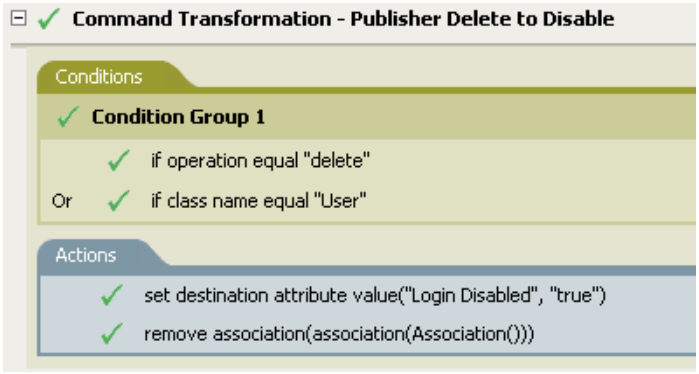
8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?"（编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的命令转换策略。

导入预定义规则

1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。

2 选择 " 命令转换 - 发布者删除 - 禁用 "，然后单击 " 确定 "。

3 单击 " 文件 ">" 保存 "，保存该规则。



在特定于个人环境的规则中没有要更改的信息。

该规则的工作方式

当已连接的数据存储区中发生删除事件时，将使用该规则。在 Identity Vault 中禁用该用户对象，而不是将其删除。任何时候发生用户对象删除事件时， Login Disabled （禁止登录）的目标特性值都会设为 True，同时会去除该用户对象的关联。该用户对象不能再登录到 Novell eDirectory 树，但并未被删除。

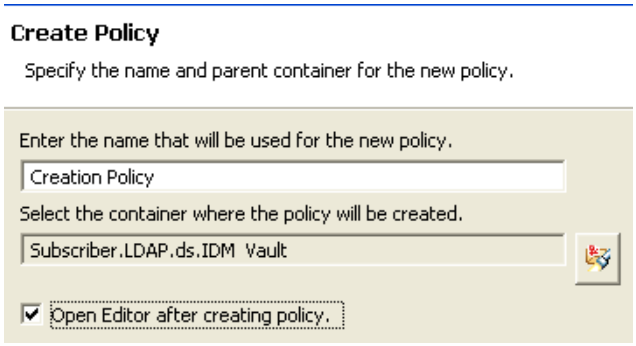
创建 - 必需特性

除非已填充必需特性，否则该规则不允许创建用户对象。在驱动程序的创建策略中实施该规则。可以对订购者通道或发布者通道单独实施此规则，也可以对这两个通道同时实施此规则。

使用预定义规则包含两个步骤：在创建策略集中创建一个策略，然后导入预定义规则。如果要为已有的创建策略添加此规则，请转到 “导入预定义规则”在第 73 页 。

创建策略

1. 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道或订购者通道。
2. 在 " 策略集 " 视图中选择 " 创建 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 以创建新策略。
3. 单击 " 创建新策略 "，然后单击 " 下一步 "。
4. 为该策略命名。
5. 根据所填充的位置在驱动程序中放置此策略。

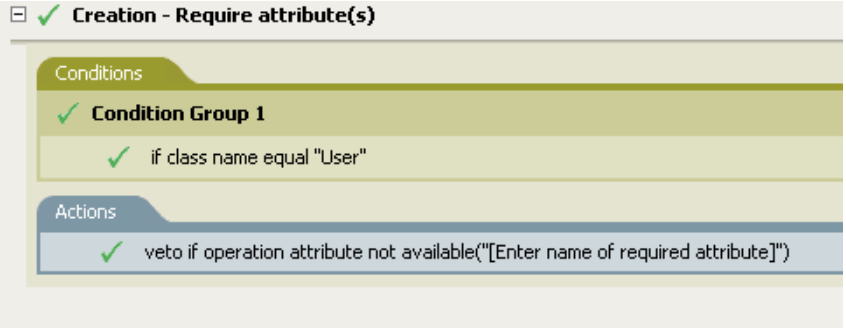


6. 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。
7. 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。
8. 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的创建策略。

导入预定义规则

1. 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。
2. 选择 " 创建 - 必需特性 "，然后单击 " 确定 "。
3. 双击 " 操作 " 选项卡，编辑该操作。
4. 从 " 输入名称 " 字段中删除 [输入所需特性的名称]。
5. 浏览至要创建的用户对象所需的特性，然后单击 " 确定 "。
6. 单击 " 确定 "。

7 选择 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

当业务流程要求用户在创建用户对象时填充特定特性时，将使用此规则。创建用户对象时，除非提供必需特性，否则该规则将禁止创建对象。可以有一个或多个必需特性。

如果需要多个必需特性，则右击该操作并选择 " 新建 "> *Append Action* （追加操作）。选择 " 操作特性不可用时禁止 "，然后浏览至所需的特性。

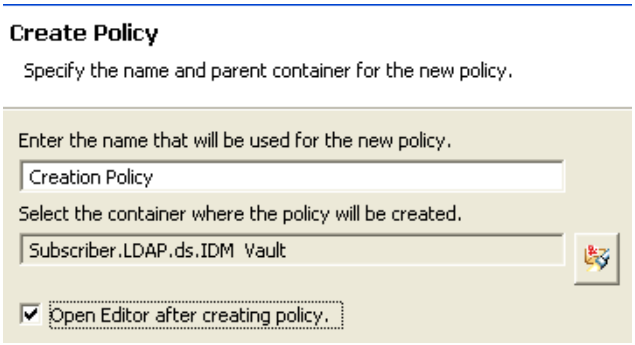
创建 - 发布者 - 使用模板

创建用户对象时允许使用 Novell eDirectory 模板对象。对驱动程序中的发布者创建策略实施该规则。只能在发布者通道上实施该规则。

使用预定义规则包含两个步骤：在创建策略集中创建一个策略，然后导入预定义规则。如果要为已有的创建策略添加此规则，请转到 “导入预定义规则”在第 75 页 。

创建策略

1 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道。
2 在 " 策略集 " 视图中选择 " 创建 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 ✚ 以创建新策略。
3 单击 " 创建新策略 "，然后单击 " 下一步 "。
4 为该策略命名。
5 根据所填充的位置在驱动程序中放置此策略。

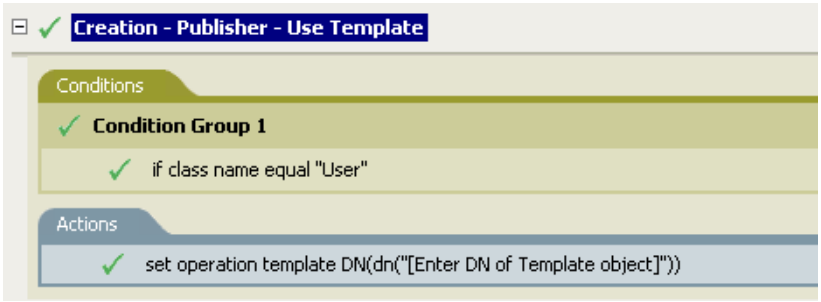


6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。
7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。

8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 謂 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的创建策略。

导入预定义规则

1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 "> " 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。
2 选择 " 创建 - 发布者 - 使用模板 "，然后单击 " 确定 "。
3 双击 " 操作 " 选项卡，编辑该操作。
4 从 " 输入 DN" 字段删除 [输入模板对象的 DN]。
5 单击 " 编辑自变量 " 图标 以起动自变量构建器。
6 选择 *Noun* （名词）列表中的 *Text* （文本）。
7 双击 " 文本 "，将其添加给自变量。
8 在编辑器中，单击 " 浏览 " 图标，浏览至模板对象并选择该对象，然后单击 " 确定 "。
9 单击 " 确定 "。
10 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

当希望使用模板对象在 Identity Vault 中创建用户时，将使用该规则。如果要使不同的用户具有相同的特性，使用模板能够节省时间。将信息填写在模板对象中，这样在创建用户对象时， Identity Manager 将调用该模板并使用此模板来创建用户对象。

创建用户对象时，该规则执行 " 设置操作模板 DN" 操作。该操作调用模板对象并使用模板中的信息创建用户对象。

创建 - 设置默认特性值

您可以为创建用户对象时所指派的特性设置默认值。对驱动程序中的订购者创建策略或发布者创建策略实施此规则。

使用预定义规则包含两个步骤：在创建策略集中创建一个策略，然后导入预定义规则。如果要为已有的创建策略添加此规则，请转到 “导入预定义规则”在第 76 页 。

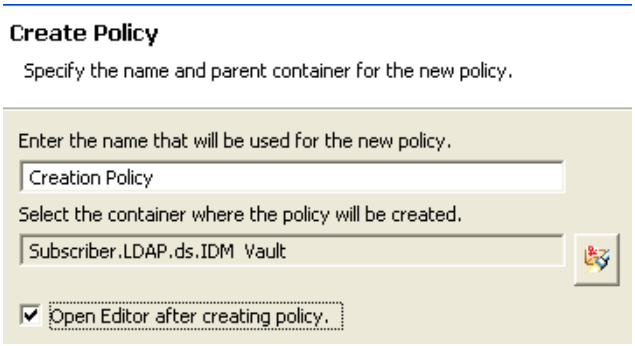
创建策略

1 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道或订购者通道。
2 在 " 策略集 " 视图中选择 " 创建 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 以创建新策略。

3 单击 " 创建新策略 "，然后单击 " 下一步 "。

4 为该策略命名。

5 根据所填充的位置在驱动程序中放置此策略。



6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。

7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。

8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的创建策略。

导入预定义规则

1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。

2 选择 " 创建 - 设置默认特性值 "，然后单击 " 确定 "。

3 双击 " 操作 " 选项卡，编辑该操作。

4 从 *Enter attribute name* （输入特性名称）字段中删除 [输入特性名]。

5 单击 " 浏览 " 图标，然后浏览至要创建的特性并选择该特性。

6 从 *Enter arguments values* （输入自变量值）字段中删除 [输入默认特性值]。

7 单击 " 编辑自变量 " 图标 以起动自变量值列表构建器。

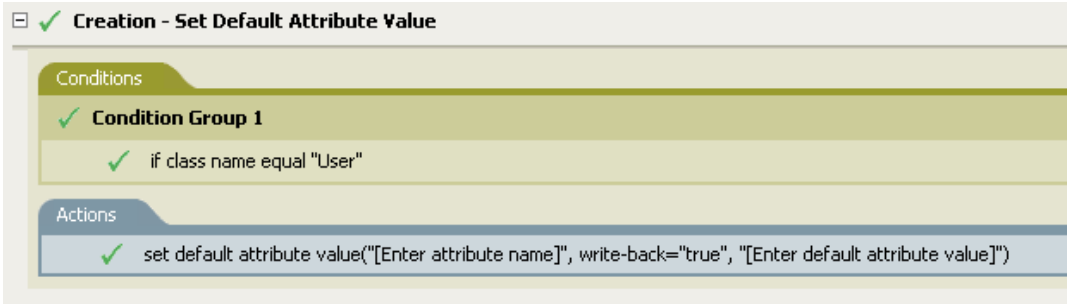
8 选择值的数据类型。

9 单击 " 编辑自变量 " 图标 以起动自变量构建器。

10 在自变量构建器中为特性创建值，然后单击 " 确定 "。

11 单击 " 确定 "。

12 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

当希望使用默认特性和值创建用户对象时，将使用该规则。创建用户对象时，该规则将设置特性和该特性的值。

如果希望定义多个特性值，则右击该操作，然后单击 " 新建 ">" 追加操作 "。选择操作，设置默认特性值并按照步骤 1 在第 76 页 至步骤 12 在第 76 页 为特性指派值。

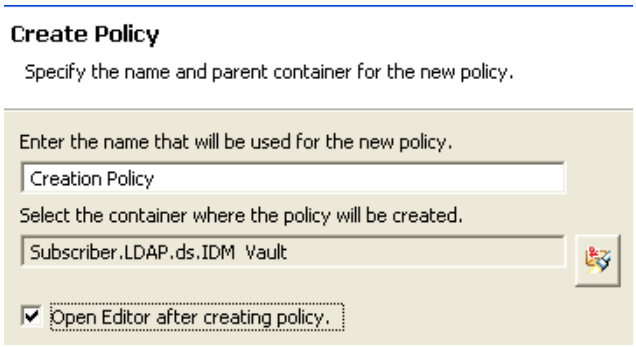
创建 - 设置默认口令

创建用户对象时，该规则为用户对象设置了一个默认口令。在驱动程序的创建策略中实施该规则。可以对订购者通道或发布者通道单独实施此规则，也可以对这两个通道同时实施此规则。

使用预定义规则包含两个步骤：在创建策略集中创建一个策略，然后导入预定义规则。如果要为已有的创建策略添加此规则，请转到 “导入预定义规则”在第 77 页 。

创建策略

1. 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道或订购者通道。
2. 在 " 策略集 " 视图中选择 " 创建 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 以创建新策略。
3. 单击 " 创建新策略 "，然后单击 " 下一步 "。
4. 为该策略命名。
5. 根据所填充的位置在驱动程序中放置此策略。

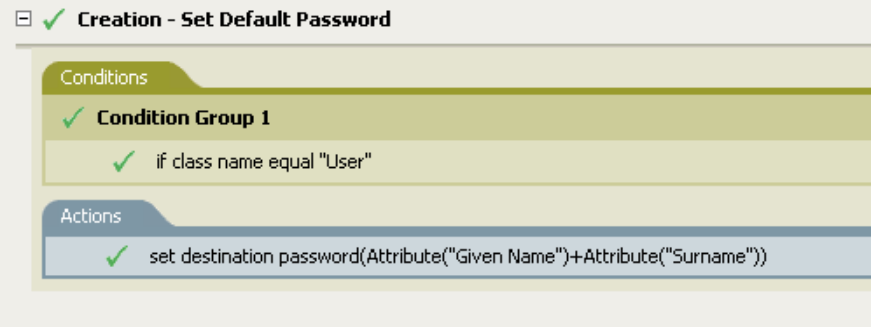


6. 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。
7. 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。
8. 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 謂 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的创建策略。

导入预定义规则

1. 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。
2. 选择 " 创建 - 设置默认口令 "，然后单击 " 确定 "。

3 单击 " 文件 ">" 保存 "，保存该规则。



在特定于个人环境的规则中没有要更改的信息。

该规则的工作方式

如果要使创建的用户对象具有默认口令，则使用此规则。创建用户对象时，为用户对象设置的口令是用户对象的 Given Name （名）特性加上 Surname （姓氏）特性。

可以通过编辑自变量更改默认口令的值。使用自变量构建器可以将口令设置为所需的任意其它值。

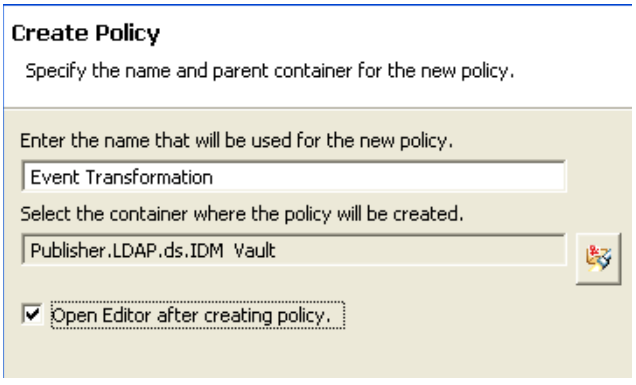
事件转换 - 范围过滤 - 包括子树

除了特定子树的事件外，排除发生的所有其它事件。对驱动程序中的事件转换策略实施该规则。可以对订购者通道或发布者通道单独实施此规则，也可以对这两个通道同时实施此规则。

使用预定义规则包含两个步骤：在事件转换策略集中创建一个策略，然后导入预定义规则。如果要为已有的事件转换策略添加此规则，请转到导入预定义规则 (在第 79 页)。

创建策略

1 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道或订购者通道。
2 在 " 策略集 " 视图中选择 " 事件转换 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 ✚ 以创建新策略。
3 单击 " 创建新策略 "，然后单击 " 下一步 "。
4 为该策略命名。
5 根据所填充的位置在驱动程序中放置此策略。



6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。

7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。

8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新事件转换策略。

导入预定义规则

1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。

2 选择 " 事件转换 - 范围过滤 - 包括子树 "，然后单击 " 确定 "。

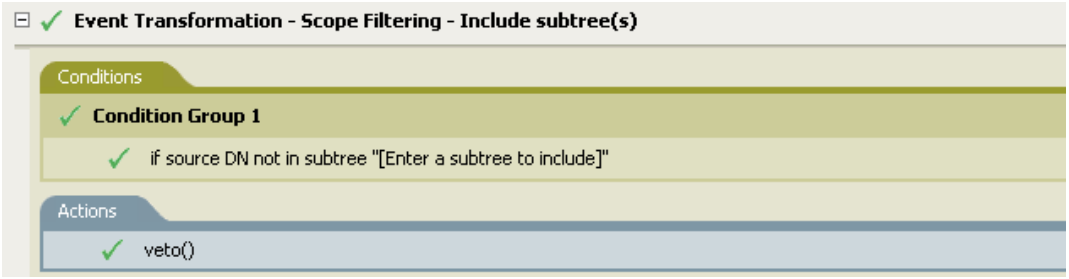
3 双击 " 条件 " 选项卡，编辑该条件。

4 删除 " 值 " 字段中的 [输入要包括的子树]。

5 单击 " 浏览 " 按钮，在 Identity Vault 中浏览至树中要同步事件的部分，然后单击 " 确定 "。

6 单击 " 确定 "。

7 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

当希望从同步中排除部分 Identity Vault 时，将使用该规则。它允许不使用过滤器就能实现同步某些对象而不同步其它对象。事件可以在任意位置发生，但禁止在 Identity Vault 的特定部分发生。

事件转换 - 范围过滤 - 排除子树

排除发生在特定子树中的所有事件。对驱动程序中的事件转换策略实施该规则。可以对订购者通道或发布者通道单独实施此规则，也可以对这两个通道同时实施此规则。

使用预定义规则包含两个步骤：在事件转换策略集中创建一个策略，然后导入预定义规则。如果要为已有的事件转换策略添加此规则，请转到 “导入预定义规则”在第 80 页 。

创建策略

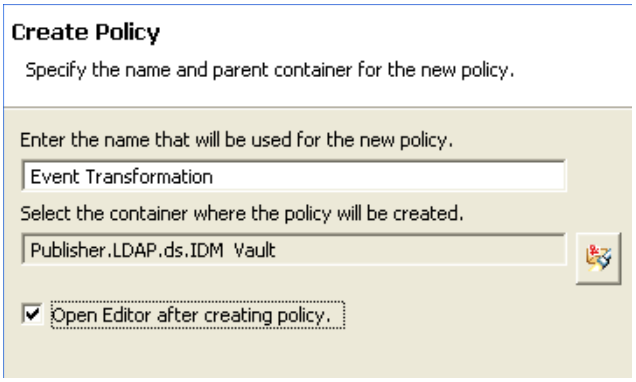
1 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道或订购者通道。

2 选择 " 策略集 " 视图中的 " 事件转换策略集 "，然后单击 " 在策略集中创建或添加新策略 " 图标 ✚ 以创建新策略。

3 单击 " 创建新策略 "，然后单击 " 下一步 "。

4 为该策略命名。

5 根据所填充的位置在驱动程序中放置此策略。



6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。

7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。

8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新事件转换策略。

导入预定义规则

1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 "。

2 选择 " 事件转换 - 范围过滤 - 排除子树 "，然后单击 " 确定 "。

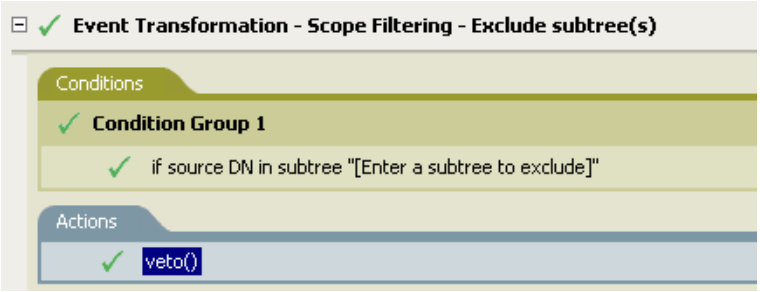
3 双击 " 条件 " 选项卡，编辑该条件。

4 删除 " 值 " 字段中的 [输入要排除的子树]。

5 单击 " 浏览 " 按钮，浏览 Identity Vault 以找到希望从同步中排除的事件所在的树部分，然后单击 " 确定 "。

6 单击 " 确定 "。

7 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

当希望从同步中排除部分 Identity Vault 时，将使用该规则。它允许不使用过滤器就能实现同步某些对象而不同步其它对象。每当事件在 Identity Vault 的该特定部分时，事件都将被禁止。

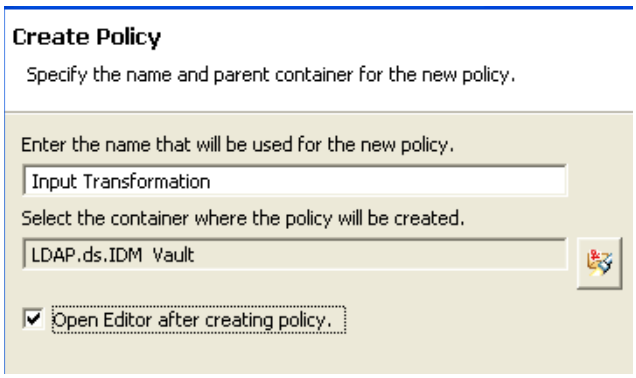
输入或输出转换 - 将电话号码格式重新从 (nnn) nnn-nnnn 设置为 nnn-nnn-nnnn

满足所需条件时，将重设电话号码的格式。对驱动程序中的输入或输出转换策略实施该规则。可以对订购者通道或发布者通道单独实施此规则，也可以对这两个通道同时实施此规则。

使用预定义规则包含两个步骤：在输入或输出转换策略集中创建一个策略，然后导入预定义规则。如果要为已有的输入或输出转换策略添加此规则，请转到 “导入预定义规则”在第 81 页 。

创建策略

1. 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道或订购者通道。
2. 选择 " 策略集 " 视图中的 " 输入或输出转换 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 ✚ 以创建新策略。
3. 单击 " 创建新策略 "，然后单击 " 下一步 "。
4. 为该策略命名。
5. 根据所填充的位置在驱动程序中放置此策略。

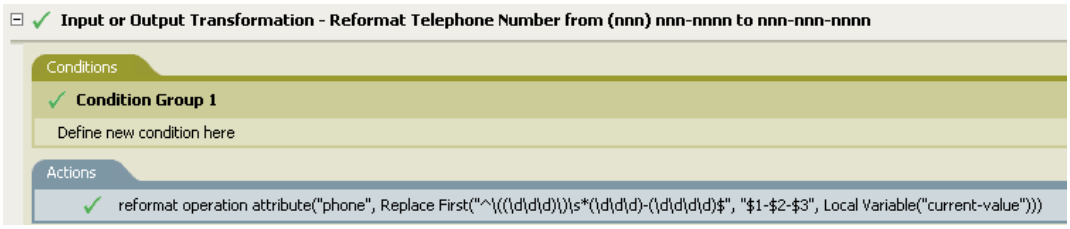


6. 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。
7. 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。
8. 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的输入或输出转换策略。

导入预定义规则

1. 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。
2. 选择 " 输入或输出转换 - 将电话号码格式重新从 (nnn) nnn-nnnn 设置为 nnn-nnn-nnnn"，然后单击 " 确定 "。
3. 双击 " 条件 " 选项卡，编辑该条件。
4. 定义重设电话号码格式时希望出现的条件。
5. 单击 " 确定 "。

6 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

希望重设电话号码的格式时将使用该规则。定义重设电话号码的格式时需要满足的条件。

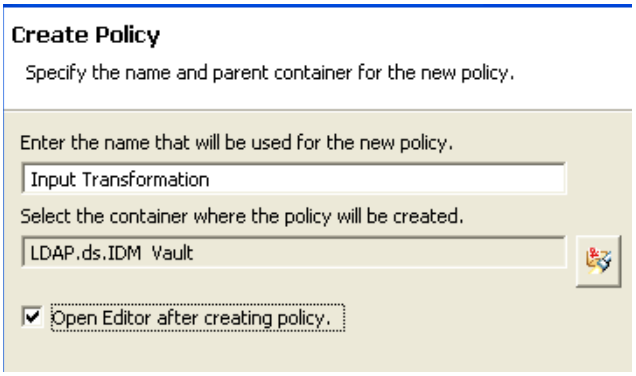
输入或输出转换 - 将电话号码格式重新从 **nnn-nnn-nnnn** 设置为 **(nnn) nnn-nnnn**

满足所需条件时，将重设电话号码的格式。对输入或输出转换策略实施该规则。可以对订购者通道或发布者通道单独实施此规则，也可以对这两个通道同时实施此规则。

使用预定义规则包含两个步骤：在输入或输出转换策略集中创建一个策略，然后导入预定义规则。如果要为已有的输入或输出转换策略添加此规则，请转到 “导入预定义规则”在第 83 页 。

创建策略

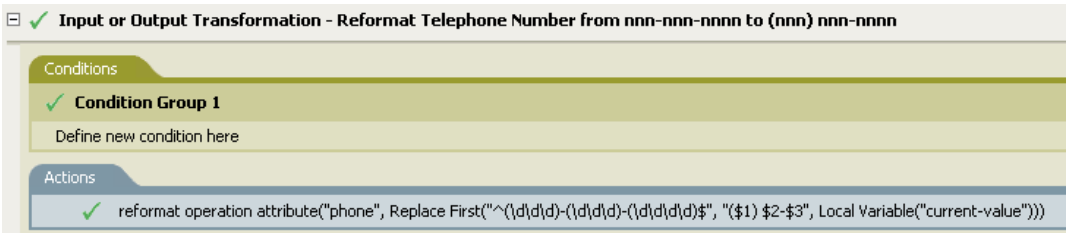
1 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道或订购者通道。
2 选择 " 策略集 " 视图中的 " 输入或输出转换 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 ✚ 以创建新策略。
3 单击 " 创建新策略 "，然后单击 " 下一步 "。
4 为该策略命名。
5 根据所填充的位置在驱动程序中放置此策略。



6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。
7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。
8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的输入或输出转换策略。

导入预定义规则

1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。
2 单击 " 输入或输出转换 - 将电话号码格式重新从 nnn-nnn-nnnn 设置为 (nnn) nnn-nnnn"，然后单击 " 确定 "。
3 双击 " 条件 " 选项卡，编辑该条件。
4 定义重设电话号码格式时希望出现的条件。
5 单击 " 确定 "。
6 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

希望重设电话号码的格式时将使用该规则。定义重设电话号码的格式时需要满足的条件。

匹配 - 已镜像的发布者

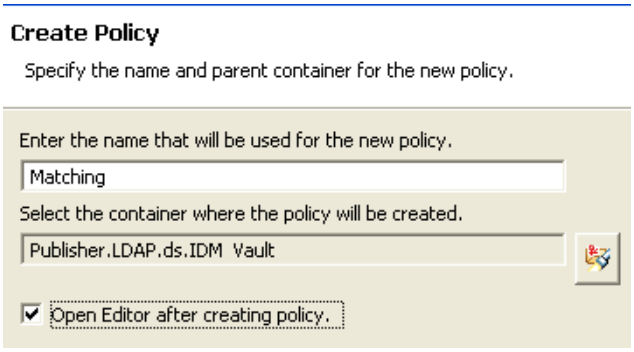
通过使用数据存储区中特定点处的镜像结构匹配 Identity Vault 中的对象。对驱动程序中的匹配策略实施该规则。只能在发布者通道上实施该规则。

使用预定义规则包含两个步骤：在匹配策略集中创建一个策略，然后导入预定义规则。如果要为已有的匹配策略添加此规则，请转到导入预定义规则 (在第 84 页)。

创建策略

1 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道。
2 选择 " 策略集 " 视图中的 " 匹配 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 ✚ 以创建新策略。
3 单击 " 创建新策略 "，然后单击 " 下一步 "。
4 为该策略命名。

5 根据所填充的位置在驱动程序中放置此策略。



6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。

7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。

8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的匹配策略。

导入预定义规则

1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。

2 选择 " 匹配 - 已镜像的发布者 "，然后单击 " 确定 "。

3 双击 " 条件 " 选项卡，编辑该条件。

4 从 " 值 " 字段删除 [输入源层次的基础]。

5 浏览至在源层次中希望开始匹配的树枝，并选择该树枝，然后单击 " 确定 "。

6 单击 " 确定 "。

7 双击 " 操作 " 选项卡，编辑该操作。

8 从 " 输入字符串 " 字段删除 [输入目标层次的基础]。

9 单击 " 编辑自变量 " 图标 以起动自变量构建器。

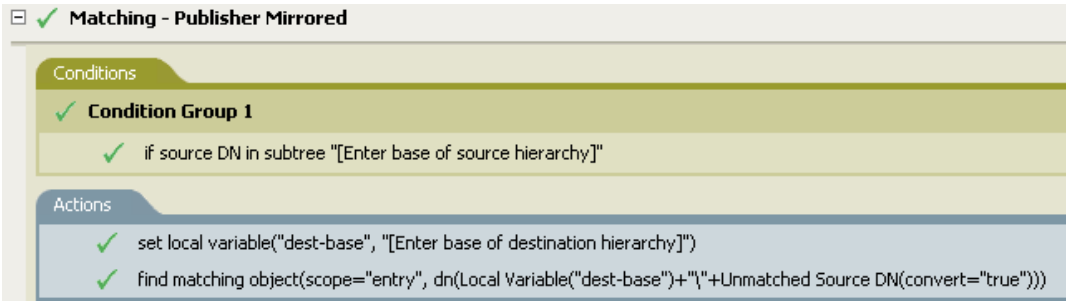
10 选择 " 名词 " 列表中的 " 文本 "。

11 双击 " 文本 "，将其添加给自变量。

12 在编辑器中，单击 " 浏览 " 图标并浏览至在目标层次中要匹配的源结构所在的树枝，然后单击 " 确定 "。

13 单击 " 确定 "。

14 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

通过使用数据存储区中特定点处的镜像结构匹配 Identity Vault 中的对象。发生 Add 事件时，驱动程序将检查是否存在对象，它从数据存储区的特定 DN 处开始检查。驱动程序随后将一个局部变量 dest-base 设置为 Identity Vault 中的起始点，该 Identity Vault 中的结构将镜像到数据存储区中。然后，驱动程序连接该局部变量 dest-base、一个 \ 以及对象的源 DN 来创建要搜索的环境。它以斜线格式创建它要查找的路径。

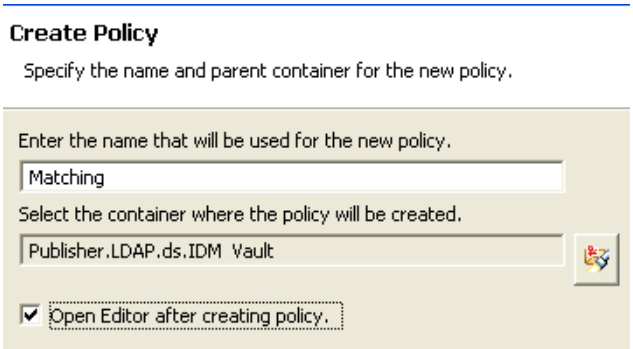
匹配 **-** 已镜像的订购者 **- LDAP** 格式

通过使用 Identity Vault 中特定点处的镜像结构匹配数据存储区中的对象。对驱动程序中的匹配策略实施该规则。只能在订购者通道上实施该规则。

使用预定义规则包含两个步骤：在匹配策略集中创建一个策略，然后导入预定义规则。如果要为已有的匹配策略添加此规则，请转到 “导入预定义规则”在第 86 页 。

创建策略

1 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道。
2 选择 " 策略集 " 视图中的 " 匹配 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 ✚ 以创建新策略。
3 单击 " 创建新策略 "，然后单击 " 下一步 "。
4 为该策略命名。
5 根据所填充的位置在驱动程序中放置此策略。

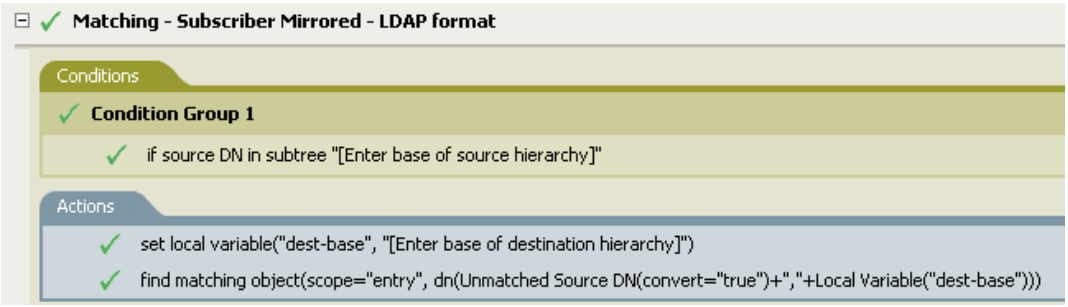


6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。
7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。

8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的匹配策略。

导入预定义规则

1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。
2 选择 " 匹配 - 已镜像的订购者 - LDAP 格式 "，然后单击 " 确定 "。
3 双击 " 条件 " 选项卡，编辑该条件。
4 从 " 值 " 字段删除 [输入源层次的基础]。
5 浏览至在源层次中希望开始匹配的树枝，并选择该树枝，然后单击 " 确定 "。
6 单击 " 确定 "。
7 双击 " 操作 " 选项卡，编辑该操作。
8 从 " 输入字符串 " 字段删除 [输入目标层次的基础]。
9 单击 " 编辑自变量 " 图标 以起动自变量构建器。
10 选择 " 名词 " 列表中的 " 文本 "。
11 双击 " 文本 "，将其添加给自变量。
12 在编辑器中，单击 " 浏览 " 图标，在目标层次中浏览至要匹配的源结构所在的树枝，并选择该树枝，然后单击 " 确定 "。
13 单击 " 确定 "。
14 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

通过使用 Identity Vault 中特定点处的镜像结构匹配数据存储区中的对象。发生 Add 事件时，驱动程序将检查是否存在对象，它从 Identity Vault 中的特定 DN 处开始检查。驱动程序随后将一个局部变量 dest-base 设置为数据存储区的起始点，数据存储区中的结构将镜像到 Identity Vault 中。然后，驱动程序连接对象的源 DN、一个 "," 以及该局部变量 dest-base 来创建要搜索的环境。它以 LDAP 格式创建它要查找的路径。

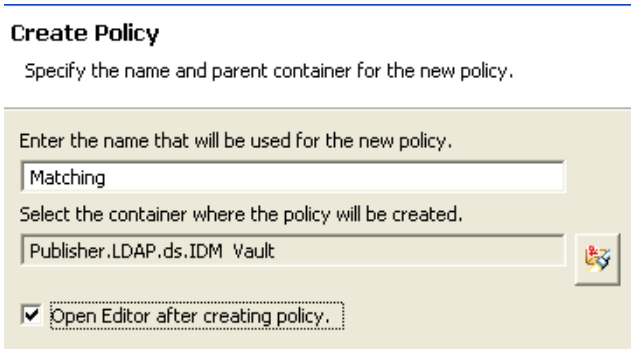
匹配 - 按特性值

按特定特性值匹配对象。对驱动程序中的匹配策略实施该规则。可以对订购者通道或发布者通道单独实施此规则，也可以对这两个通道同时实施此规则。

使用预定义规则包含两个步骤：在匹配策略集中创建一个策略，然后导入预定义规则。如果要为已有的匹配策略添加此规则，请转到 “导入预定义规则”在第 87 页 。

创建策略

1. 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道。
2. 选择 " 策略集 " 视图中的 " 匹配 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 以创建新策略。
3. 单击 " 创建新策略 "，然后单击 " 下一步 "。
4. 为该策略命名。
5. 根据所填充的位置在驱动程序中放置此策略。



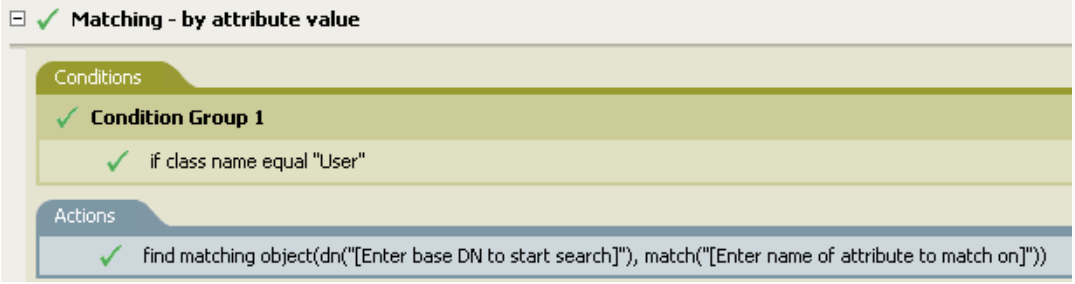
6. 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。
7. 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。
8. 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的匹配策略。

导入预定义规则

1. 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。
2. 选择 " 匹配 - 按特性值 "，然后单击 " 确定 "。
3. 双击 " 操作 " 选项卡，编辑该操作。
4. 从 " 输入 DN" 字段中删除 [输入要开始搜索的基本 DN]。
5. 单击 " 编辑自变量 " 图标 以起动自变量构建器。
6. 选择 " 名词 " 列表中的 " 文本 "。
7. 双击 " 文本 "，将其添加给自变量。
8. 在编辑器中，单击 " 浏览 " 图标，然后浏览至希望开始搜索的树枝并选择此树枝，然后单击 " 确定 "。
9. 从 " 输入匹配特性 " 字段删除 *[Enter name of attribute to match on]* （输入特性名称以进行匹配）。
10. 单击 " 编辑自变量 " 图标 以起动匹配属性构建器。
11. 单击 " 浏览 " 图标并选择希望匹配的特性。可以选择一个或多个要匹配的特性，然后单击 " 确定 "。

12 单击 " 确定 "。

13 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

通过特性匹配用户对象。同步用户对象时，驱动程序将使用该规则检查是否存在指定的特性。如果不存在指定的特性，则创建新用户对象。

布局 - 已镜像的发布者

通过使用数据存储区特定点处的镜像结构来放置 Identity Vault 中的对象。对驱动程序中的布局策略实施该规则。只能在发布者通道上实施该规则。

使用预定义规则包含两个步骤：在布局策略集中创建一个策略，然后导入预定义规则。如果要为已有的布局策略添加此规则，请转到 “导入预定义规则”在第 89 页 。

创建策略

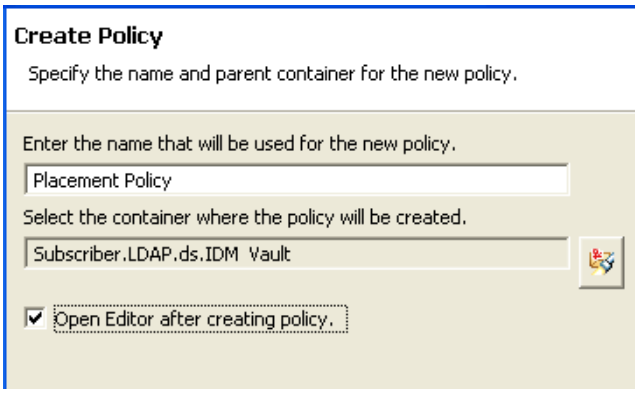
1 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道。

2 选择 " 策略集 " 中的 " 布局 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 以创建新策略。

3 单击 " 创建新策略 "，然后单击 " 下一步 "。

4 为该策略命名。

5 根据所填充的位置在驱动程序中放置此策略。



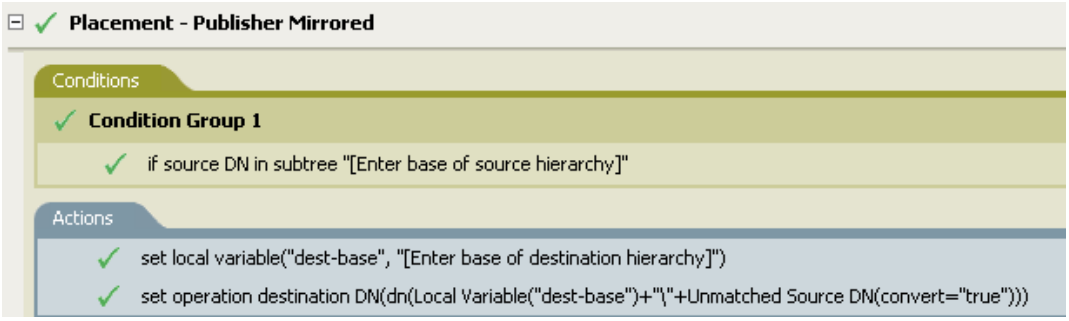
6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。

7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。

8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 謂 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的布局策略。

导入预定义规则

1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。
2 选择 " 布局 - 已镜像的发布者 "，然后单击 " 确定 "。
3 双击 " 条件 " 选项卡，编辑该条件。
4 从 " 值 " 字段删除 [输入源层次的基础]。
5 浏览至源层次中希望处理对象的树枝，并选择该树枝，然后单击 " 确定 "。
6 双击 " 操作 " 选项卡，编辑该操作。
7 从 " 输入字符串 " 字段删除 [输入目标层次的基础]。
8 单击 " 编辑自变量 " 图标 以起动自变量构建器。
9 选择 " 名词 " 列表中的 " 文本 "。
10 双击 " 文本 "，将其添加给自变量。
11 在编辑器中，单击 " 浏览 " 图标，在目标层次中浏览至希望放置对象的树枝，并选择此树枝，然后单击 " 确定 "。
12 单击 " 确定 "。
13 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

如果用户对象驻留在源层次中，则对象放置在数据存储区中已镜像的结构中。布局从局部变量 dest-base 定义的点处开始。它将用户对象放置到 dest-base\unmatched source DN 所在的位置。该规则使用斜线格式。

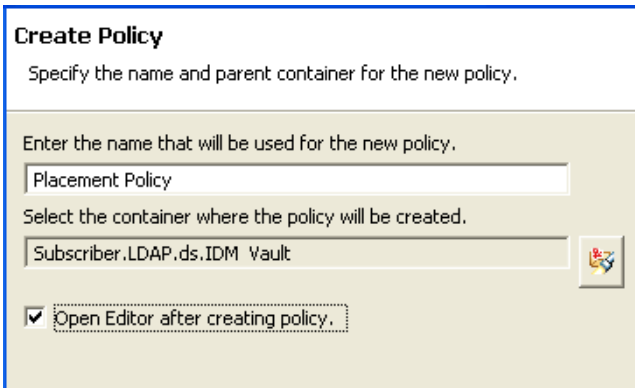
布局 **-** 已镜像的订购者 **- LDAP** 格式

使用 Identity Vault 中特定点处的已镜像结构将对象放置在数据存储区中。对驱动程序中的布局策略实施该规则。只能在订购者通道上实施该规则。

使用预定义规则包含两个步骤：在布局策略集中创建一个策略，然后导入预定义规则。如果要为已有的布局策略添加此规则，请转到 “导入预定义规则”在第 90 页 。

创建策略

1. 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道。
2. 选择 " 策略集 " 视图中的 " 布局 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 以创建新策略。
3. 单击 " 创建新策略 "，然后单击 " 下一步 "。
4. 为该策略命名。
5. 根据所填充的位置在驱动程序中放置此策略。

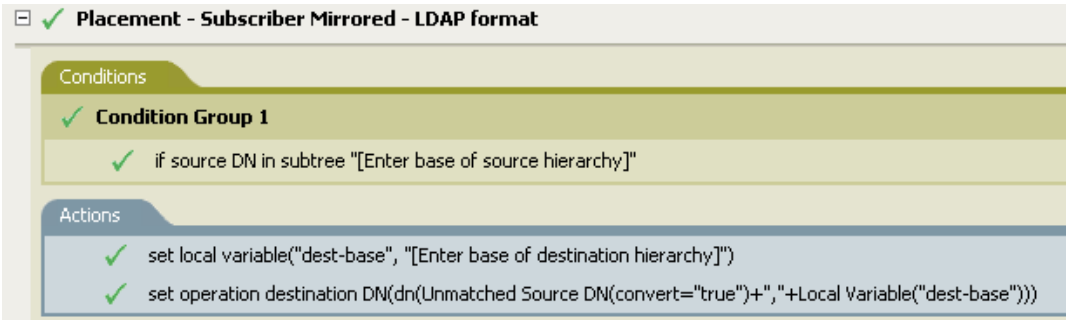


6. 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。
7. 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。
8. 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的布局策略。

导入预定义规则

1. 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。
2. 选择 *Placement - Subscriber Mirrored - LDAP format* （布局 - 已镜像的订购者 - LDAP 格式），然后单击 " 确定 "。
3. 双击 " 条件 " 选项卡，编辑该条件。
4. 从 " 值 " 字段删除 [输入源层次的基础]。
5. 浏览至源层次中希望对象开始行动的树枝，然后单击 " 确定 "。
6. 双击 " 操作 " 选项卡，编辑该操作。
7. 从 " 输入字符串 " 字段删除 [输入目标层次的基础]。
8. 单击 " 编辑自变量 " 图标 以起动自变量构建器。
9. 选择 " 名词 " 列表中的 " 文本 "。
10. 双击 " 文本 "，将其添加给自变量。
11. 在编辑器中，单击 " 浏览 " 图标，浏览至目标层次中希望放置对象的树枝，然后单击 " 确定 "。
12. 单击 " 确定 "。

13 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

如果用户对象驻留在源层次中，则对象放置在 Identity Vault 中已镜像的结构中。布局从局部变量 dest-base 定义的点处开始。它将用户对象放置在源 DN 不匹配 (dest-base) 所在的位置。该规则使用 LDAP 格式。

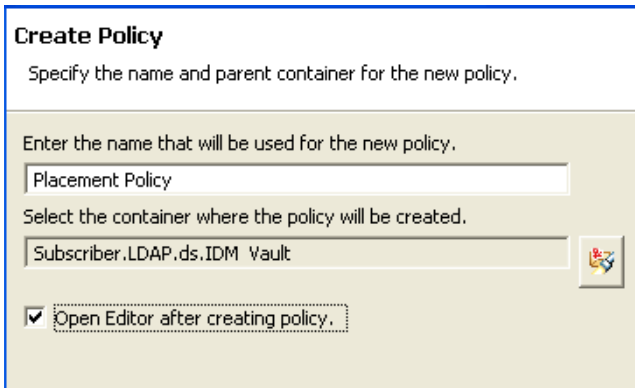
布局 - 发布者平面文件

将对象从数据存储区放置到 Identity Vault 中的某个树枝中。对驱动程序中的布局策略实施该规则。只能在发布者通道上实施该规则。

使用预定义规则包含两个步骤：在布局策略集中创建一个策略，然后导入预定义规则。如果要为已有的布局策略添加此规则，请转到 “导入预定义规则”在第 92 页 。

创建策略

1 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道。
2 选择 " 策略集 " 视图中的 " 布局 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 ✚ 以创建新策略。
3 单击 " 创建新策略 "，然后单击 " 下一步 "。
4 为该策略命名。
5 根据所填充的位置在驱动程序中放置此策略。

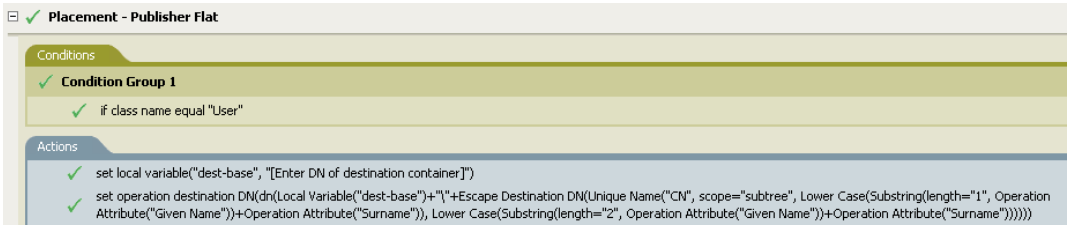


6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。
7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。

8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的布局策略。

导入预定义规则

1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。
2 选择 " 布局 - 发布者平面文件 "，然后单击 " 确定 "。
3 双击 " 操作 " 选项卡，编辑该操作。
4 从 " 输入字符串 " 字段删除 [输入目标树枝的 DN]。
5 单击 " 编辑自变量 " 图标 以起动自变量构建器。
6 选择 " 名词 " 列表中的 " 文本 "。
7 双击 " 文本 "，将其添加给自变量。
8 在编辑器中，单击 " 浏览 " 图标，然后浏览至希望放置所有用户对象的目标树枝，并选择此树枝，然后单击 " 确定 "。
9 单击 " 确定 "。
10 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

该规则将所有用户对象放置在目标 DN 中。该规则将目标树枝的 DN 设置为局部变量 dest-base。然后该规则将目标 DN 设置为 dest-base 的 CN 特性。用户对象的 CN 特性是 Given Name （名）特性的前两个字母加上 Surname （姓氏）特性，均为小写。该规则使用斜线格式。

布局 - 订购者平面文件 - LDAP 格式

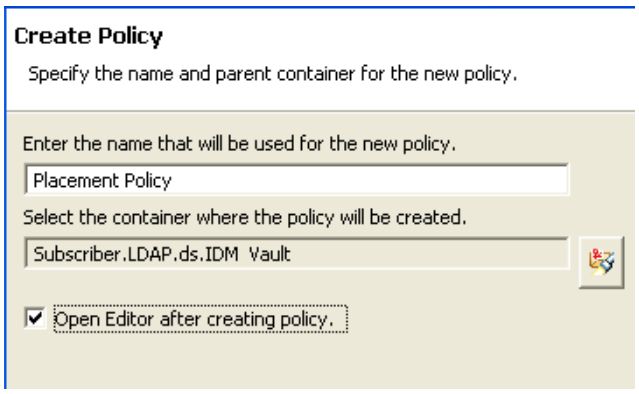
将 Identity Vault 中的对象放置到数据存储区的某个树枝中。对驱动程序中的订购者布局策略实施该规则。

使用预定义规则包含两个步骤：在布局策略集中创建一个策略，然后导入预定义规则。如果要为已有的布局策略添加此规则，请转到 “导入预定义规则”在第 93 页 。

创建策略

1 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道。
2 选择 " 策略集 " 视图中的 " 布局 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 以创建新策略。
3 单击 " 创建新策略 "，然后单击 " 下一步 "。
4 为该策略命名。

5 根据所填充的位置在驱动程序中放置此策略。



6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。

7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。

8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的布局策略。

导入预定义规则

1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。

2 选择 " 布局 - 订购者平面文件 - LDAP 格式 "，然后单击 " 确定 "。

3 双击 " 操作 " 选项卡，编辑该操作。

4 从 " 输入字符串 " 字段删除 [输入目标树枝的 DN]。

5 单击 " 编辑自变量 " 图标 以起动自变量构建器。

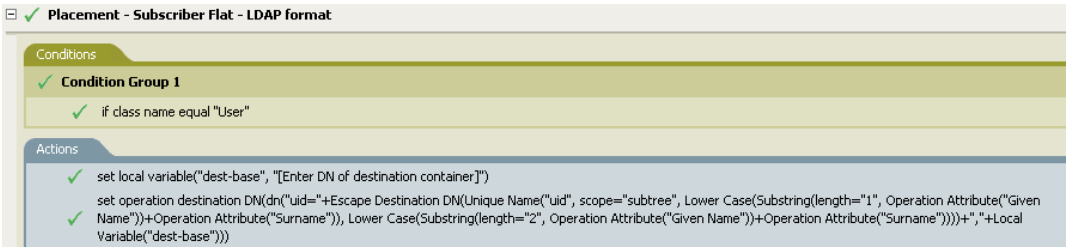
6 选择 " 名词 " 列表中的 " 文本 "。

7 双击 " 文本 "，将其添加给自变量。

8 在编辑器中，添加希望放置所有用户对象的目标树枝。确保将该树枝指定为 LDAP 格式，然后单击 " 确定 "。

9 单击 " 确定 "。

10 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

该规则将所有用户对象放置在目标 DN 中。该规则将目标树枝的 DN 设置为局部变量 dest-base。然后该规则将目标 DN 设置为 uid= 唯一标识符 (dest-base)。用户对象的 uid 特性是 Given Name 特性的前两个字母加上 Surname 特性 （小写）。该规则使用 LDAP 格式。

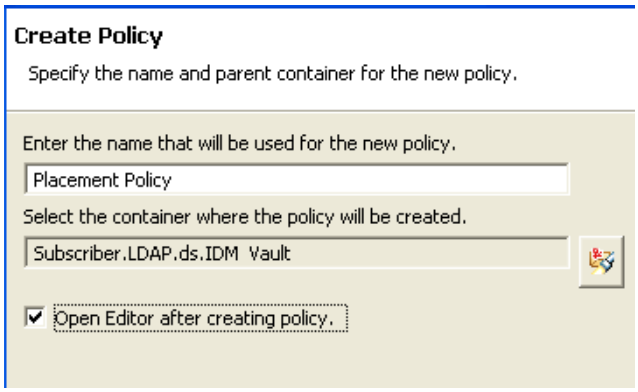
布局 - 部门发布者

将对象从数据存储区的某个树枝放置到 Identity Vault 中的多个树枝中。对驱动程序中的布局策略实施该规则。只能在发布者通道上实施该规则。

使用预定义规则包含两个步骤：在布局策略集中创建一个策略，然后导入预定义规则。如果要为已有的布局策略添加此规则，请转到 “导入预定义规则”在第 94 页 。

创建策略

1. 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道。
2. 选择 " 策略集 " 视图中的 " 布局 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 以创建新策略。
3. 单击 " 创建新策略 "，然后单击 " 下一步 "。
4. 为该策略命名。
5. 根据所填充的位置在驱动程序中放置此策略。



6. 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。
7. 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。
8. 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?" （编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的布局策略。

导入预定义规则

1. 右击策略构建器，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。
2. 选择 " 布局 - 部门发布者 "，然后单击 " 确定 "。
3. 双击 " 操作 " 选项卡，编辑该操作。
4. 从 " 输入字符串 " 字段删除 [输入目标组织的 DN]。
5. 单击 " 编辑自变量 " 图标 以起动自变量构建器。

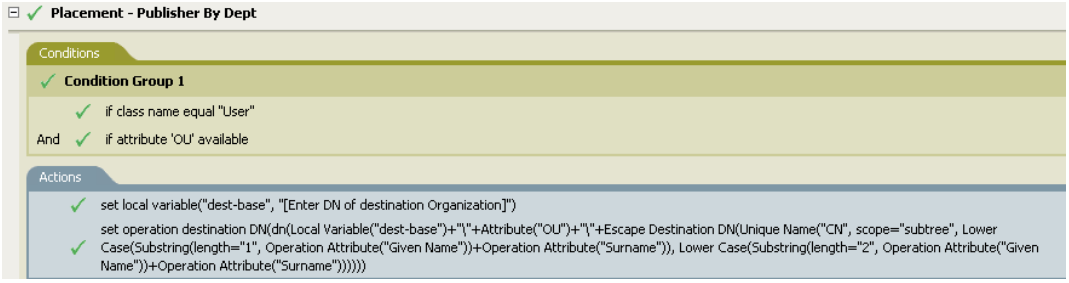
6 选择 " 名词 " 列表中的 " 文本 "。

7 双击 " 文本 "，将其添加给自变量。

8 在编辑器中，单击 " 浏览 " 图标，然后浏览至 Identity Vault 中的父树枝并选择此树枝。确保所有部门树枝都是该 DN 的子树枝，然后单击 " 确定 "。

9 单击 " 确定 "。

10 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

该规则将用户对象放置在适当的部门树枝中，具体取决于储存在 OU 特性中的值。如果需要放置用户对象，该对象也具有可用的 OU 特性，则将用户对象放置在 dest-base\OU 特性的值 \CN 特性中。

dest-base 是一个局部变量。该 DN 必须是部门树枝的相对根路径。它可以是组织或组织单元。储存在 OU 特性中的值必须是局部变量 dest-base 的子树枝名称。

该子树枝必须与要放置的用户对象相关联。 OU 特性的值必须是子树枝的名称。如果该 OU 特性不存在，则不执行该规则。

用户对象的 CN 特性是 Given Name 特性的前两个字母加上 Surname 特性 （小写）。该规则使用斜线格式。

布局 **-** 部门订购者 **- LDAP** 格式

根据 OU 特性将对象从 Identity Vault 中的某个树枝放置到数据存储区中的多个树枝中。对驱动程序中的布局策略实施该规则。只能在订购者通道上实施该规则。

使用预定义规则包含两个步骤：在布局策略集中创建一个策略，然后导入预定义规则。如果要为已有的布局策略添加此规则，请转到 “导入预定义规则”在第 96 页 。

创建策略

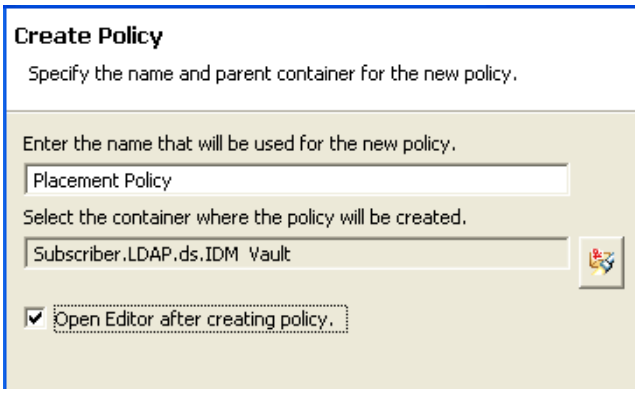
1 在 " 大纲 " 视图或 " 策略流程 " 视图中，选择发布者通道。

2 选择 " 策略集 " 视图中的 " 布局 " 策略集，然后单击 " 在策略集中创建或添加新策略 " 图标 ✚ 以创建新策略。

3 单击 " 创建新策略 "，然后单击 " 下一步 "。

4 为该策略命名。

5 根据所填充的位置在驱动程序中放置此策略。



6 选择 " 创建策略后打开编辑器 "，然后单击 " 下一步 "。

7 选择 "DirXML 底稿 " 作为策略的类型，然后单击 " 完成 "。

8 出现文件冲突窗口，显示讯息 "Before editing this item you need to save.Do you wish to save the editor 調 changes and continue?"（编辑此项目前需要保存。希望保存编辑器的更改并继续吗？）单击 " 是 "。将起动策略构建器并保存新的布局策略。

导入预定义规则

1 在 " 策略构建器 " 中右击，然后单击 " 新建 ">" 预定义规则 ">" 在之前插入预定义规则 " 或 " 在之后插入预定义规则 "。

2 选择 " 布局 - 部门订购者 - LDAP 格式 "，然后单击 " 确定 "。

3 双击 " 操作 " 选项卡，编辑该操作。

4 从 " 输入字符串 " 字段删除 [输入目标组织的 DN]。

5 单击 " 编辑自变量 " 图标 以起动自变量构建器。

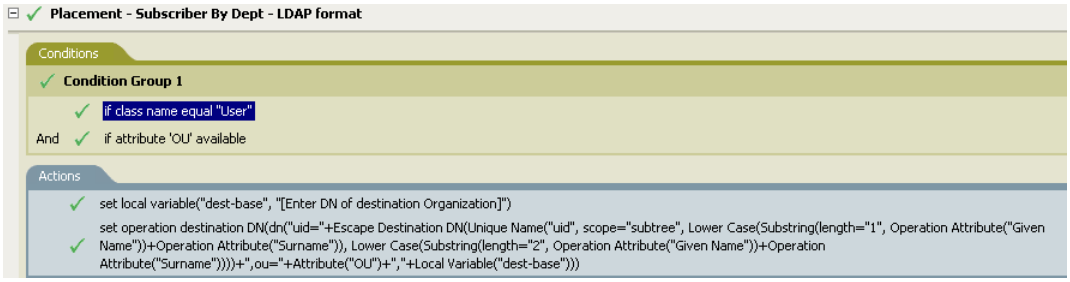
6 选择 " 名词 " 列表中的 " 文本 "。

7 双击 " 文本 "，将其添加给自变量。

8 在编辑器中，将父树枝添加到数据存储区中。必须将该父树枝指定为 LDAP 格式。确保所有部门树枝都是该 DN 的子树枝，然后单击 " 确定 "。

9 单击 " 确定 "。

10 单击 " 文件 ">" 保存 "，保存该规则。



该规则的工作方式

该规则将用户对象放置在适当的部门树枝中，具体取决于储存在 OU 特性中的值。如果需要放置用户对象，该对象也具有可用的 OU 特性，则用户对象将放置在 uid （唯一标识符）、ou （OU 特性的值）和 dest-base 中。

dest-base 是一个局部变量。该 DN 必须是部门树枝的相对根路径。它可以是组织或组织单元。储存在 OU 特性中的值必须是局部变量 dest-base 的子树枝名称。

该子树枝必须与要放置的用户对象相关联。 OU 特性的值必须是子树枝的名称。如果该 OU 特性不存在，将不执行该规则。

用户对象的 uid 特性是 Given Name （名）特性的前两个字母加上 Surname （姓氏）特性，均为小写。该规则使用 LDAP 格式。

2.2.7 使用策略模拟器测试策略

策略模拟器允许在驱动程序流程的任意点处执行策略并查看结果，而无需在 Identity Vault 中实施策略。可以在不影响生产环境或已连接系统的情况下测试策略。

有关使用策略模拟器进行的普通任务的更多信息，请参见以下章节：

- “访问策略模拟器”在第 97 页
- “使用策略模拟器”在第 99 页

策略模拟器使用 XML。 eDirectory 文档类型定义文件 (nds.dtd) 定义 XML 文档的纲要，Metadirectory 引擎可以处理该纲要。不符合该纲要的 XML 文档将生成错误。要验证该文档是否符合 nds.dtd 以及查找有关错误原因的信息，请参见 eDirectory DTD Commands and Events （eDirectory DTD 命令和事件）(http://developer.novell.com/ndk/doc/dirxml/index.html?page=/ndk/doc/dirxml/dirxmlbk/data/a36pjzu.html)。

策略模拟器不能模拟应用程序驱动程序 （如 SOAP 和定界文本）的初始策略集。这些驱动程序使用逗号分隔文件或文本文件作为输入， XML 或 XDS 由策略链中的策略派生。目前，策略模拟器仅能接受有效的 XML 或 XDS 作为输入。以后的版本将考虑其它功能。

访问策略模拟器

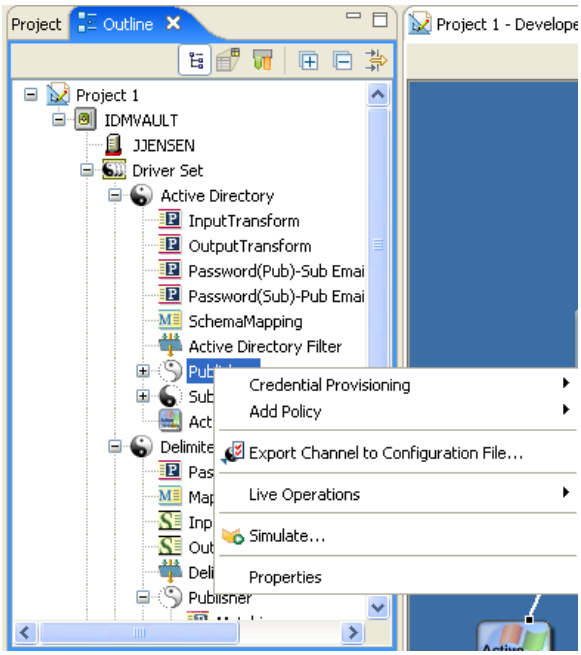
可以以三种不同的方式访问策略模拟器：

- “" 大纲 " 视图”在第 97 页
- “策略流程”在第 98 页
- “编辑器”在第 98 页

" 大纲 " 视图

1 单击 " 显示模型大纲 " 图标 。

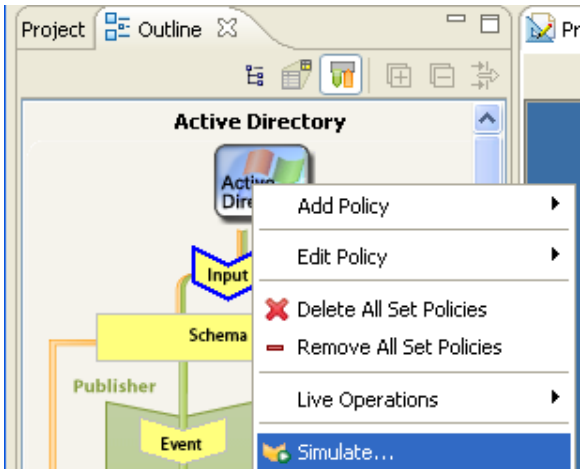
2 右击驱动程序、发布者、订购者、映射规则、过滤器或要模拟的任何策略，然后单击 *Simulate*（模拟）。



策略流程

1 单击 " 显示策略流程 " 图标 。

2 右击输入、输出、纲要映射、过滤器或要模拟的任何策略集图标，然后单击 " 模拟 "。



编辑器

可以通过策略构建器、纲要映射编辑器或过滤器编辑器，选择 " 策略模拟器 " 图标 来访问策略模拟器。

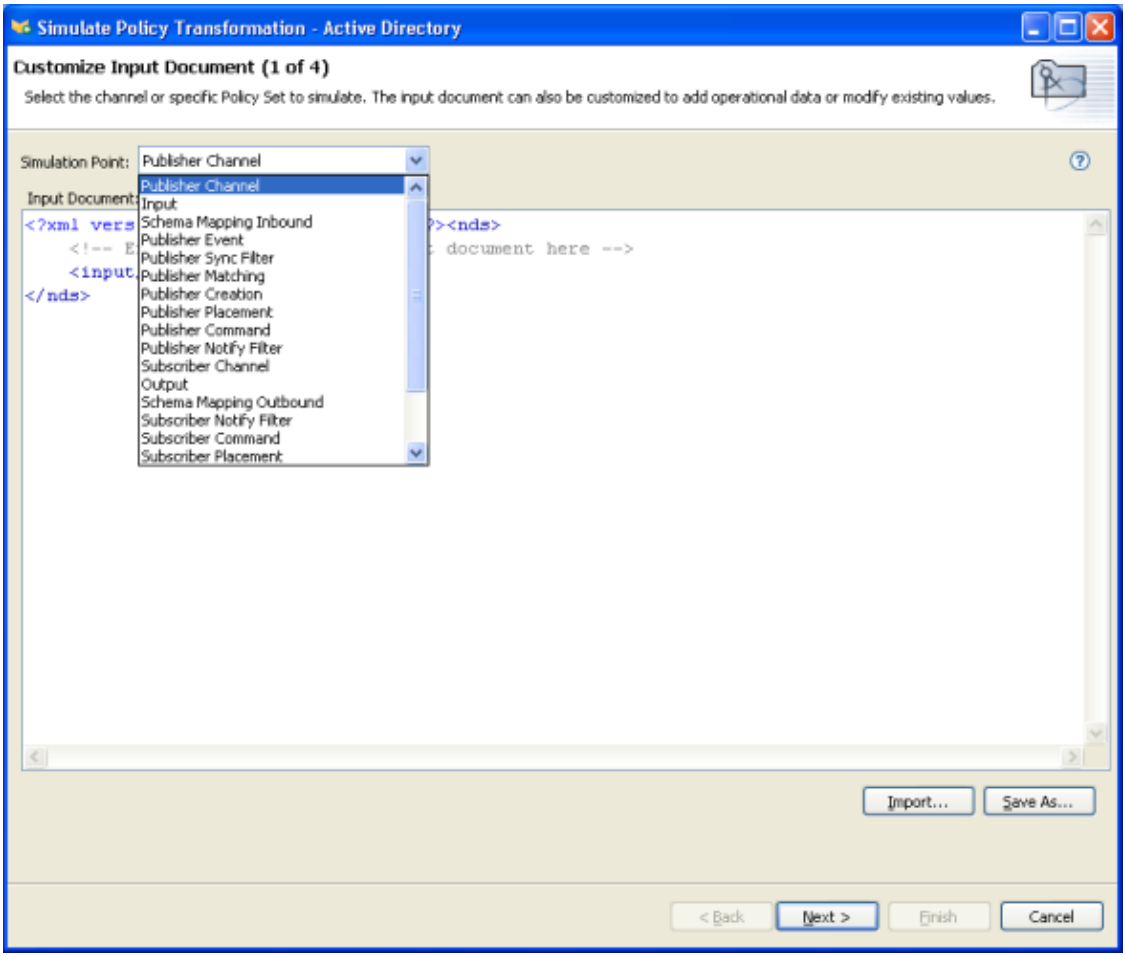
使用策略模拟器

策略模拟器允许选择驱动程序流程中的一点，通过特定操作来测试策略。它允许在测试过程中编辑输入和输出文档。如果要保留这些更改，选择 " 另存为 " 图标，将文档另存为 XML 文件。

要使用策略模拟器，请执行以下操作：

1 从 Simulation Point （模拟点）下拉列表中，选择在驱动程序流程中希望测试策略的位置。您可以选择以下项中的任意项：Publisher Channel （发布者通道）、 Subscriber Channel （订购者通道）、 Input （输入）、 Schema Mapping （纲要映射）、 Event （事件）、 Sync Filter （同步过滤器）、 Matching （匹配）、 Creation （创建）、 Placement （布局）、 Command （命令）和 Notify Filter （通知过滤器）。

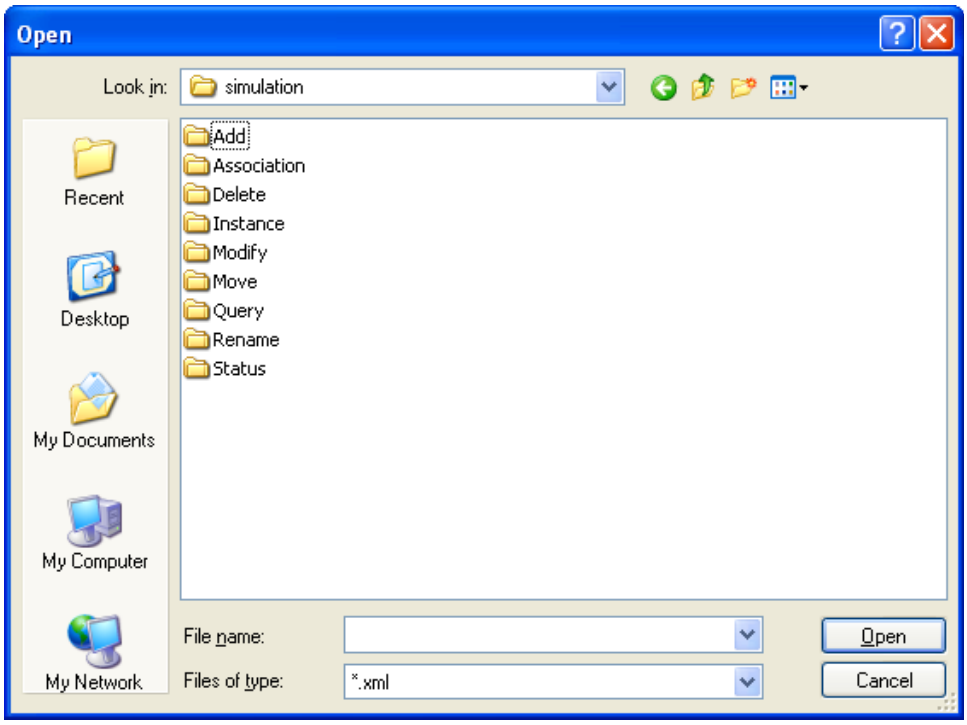
如果选择特定的策略或规则进行测试，则 " 模拟点 " 选项将仅显示 *To NDS* （至 NDS）或 *From NDS* （从 NDS）。



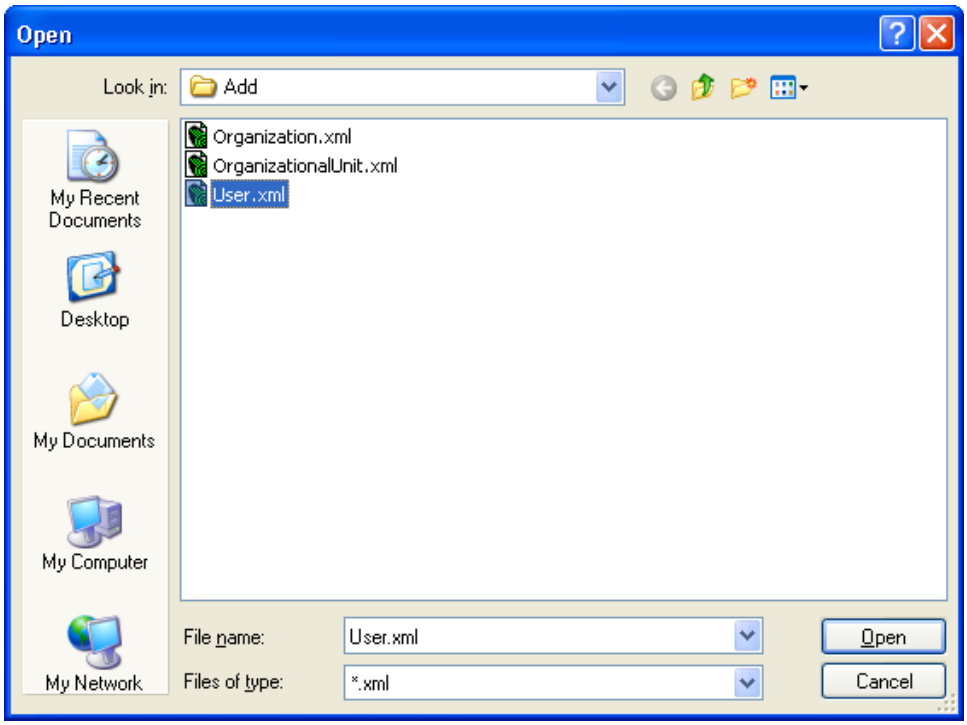
2 选择 " 导入 "，然后浏览至要测试的文件并选择此文件。

Designer 带有样本事件文件，您可以使用这些文件。这些文件位于插件 com.novell.designer.idm.policy\simulation 中。事件为 Add （添加）、 Association （关

联）、 Delete Instance （删除实例）、 Modify （修改）、 Move （移动）、 Query （查询）、 Rename （重命名）和 Status （状态）。

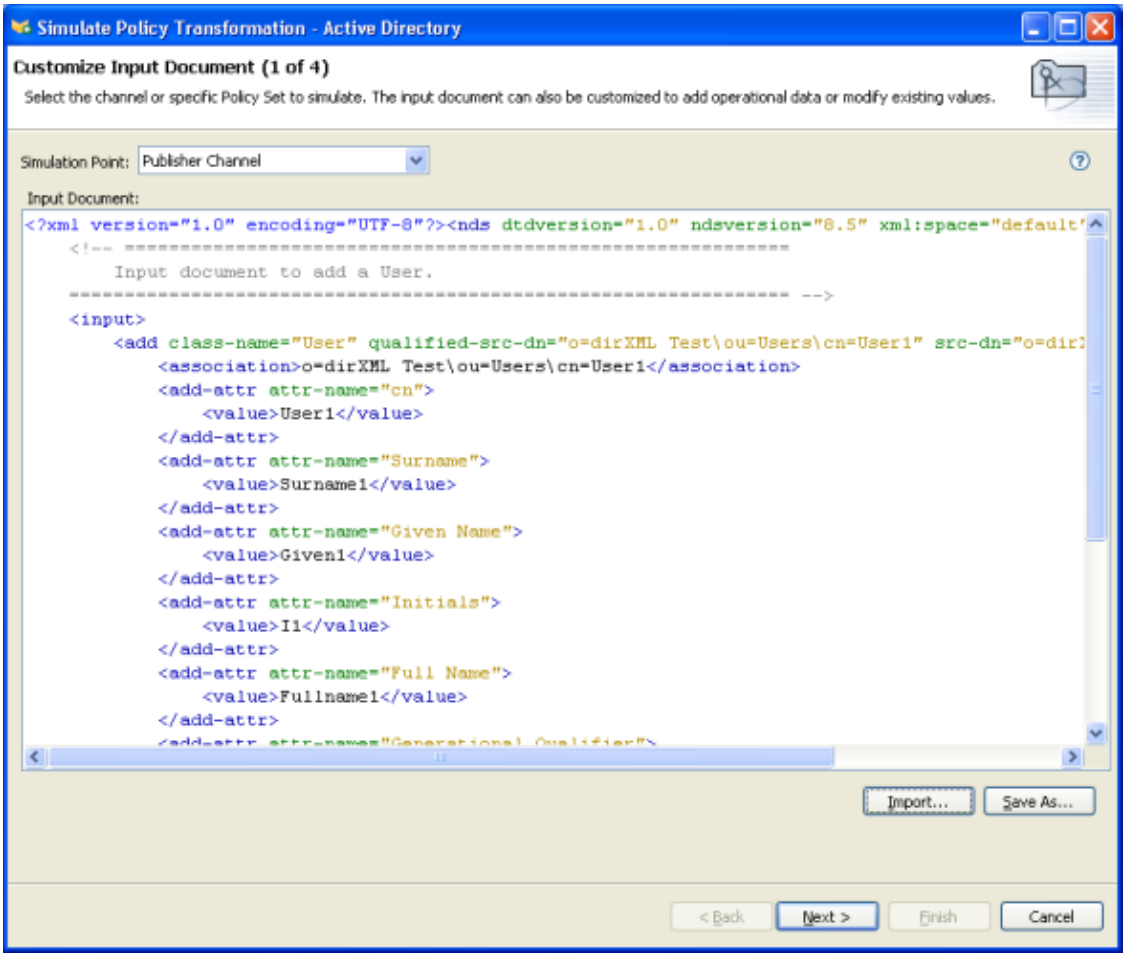


3 双击一个文件夹，显示可用事件。每个事件都有不同文件可供选择。例如，如果选择 *Add*，将有以下三个选项：Organization.xml、 OrganizationalUnit.xml 和 User.xml。文件指示事件。可以选择 User.xml，它是用户对象的一个 Add 事件。

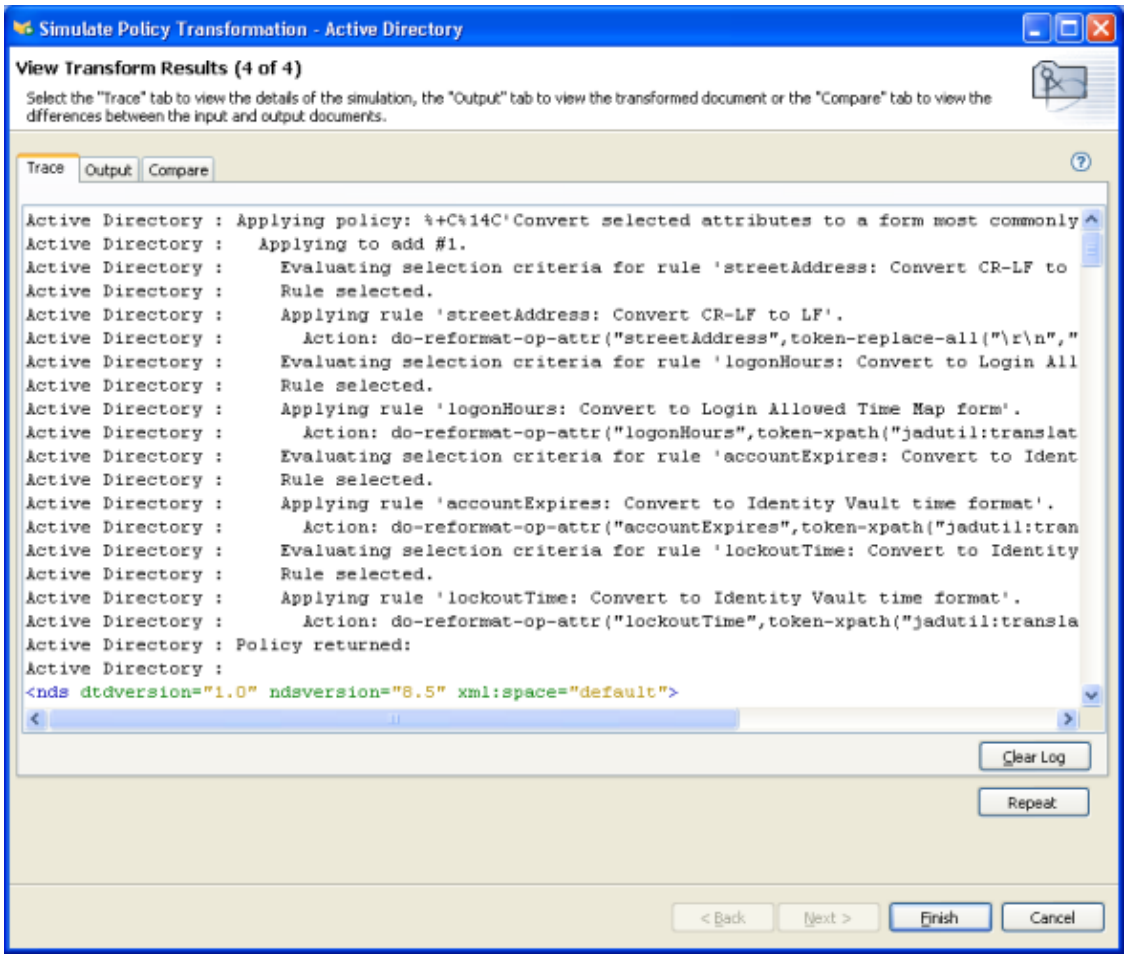


4 选择一个文件，然后单击 " 打开 " 以在窗口中显示输入文档。

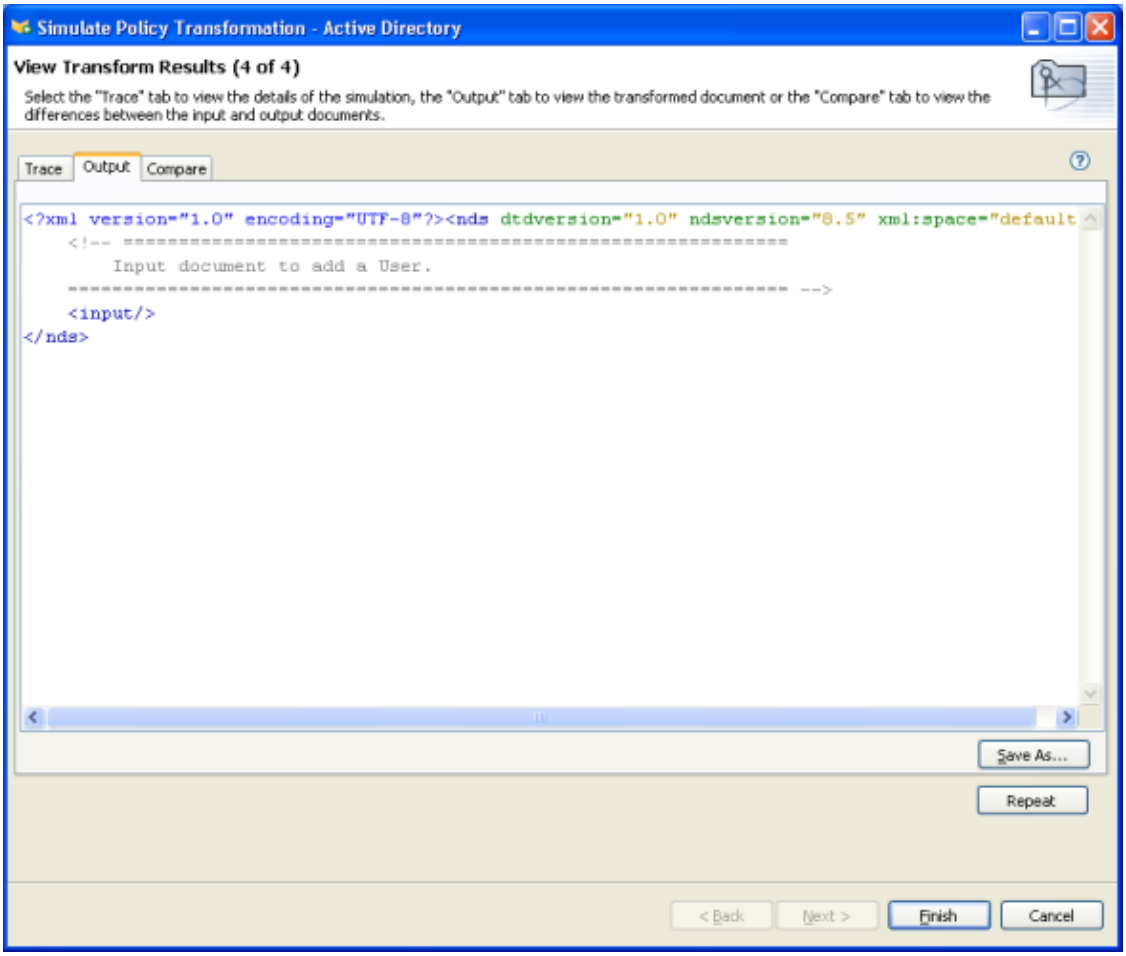
5 单击 " 下一步 "。



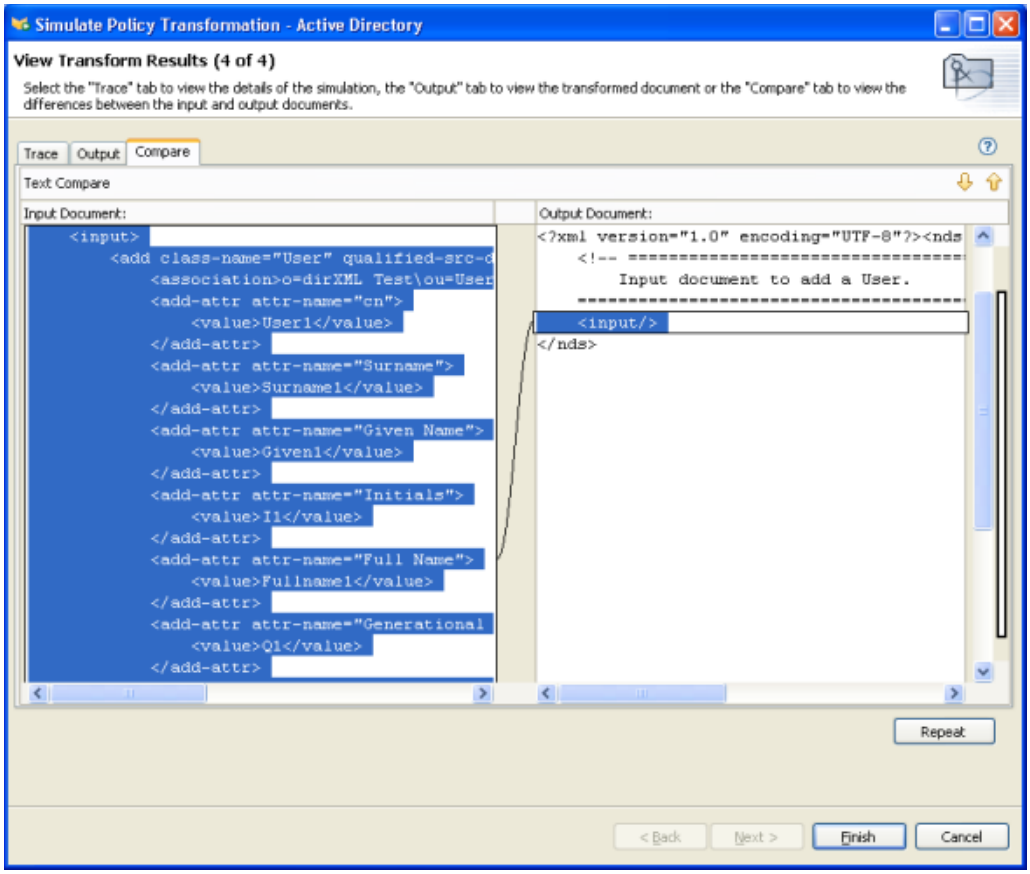
6 选择 " 跟踪 " 选项卡以查看事件结果 （即事件的处理结果）。该窗口中的信息与您在 DSTRACE 中看到的信息相同。



7 选择 " 输出 " 选项卡，查看生成的输出文档。



8 选择 " 比较 " 选项卡，比较输出文档和输入文档。



9 查看完结果后，单击 " 重复 "，以针对该策略测试另一事件。

10 完成测试后，单击 " 完成 " 关闭策略模拟器。

模拟带有 **Java** 扩展的策略

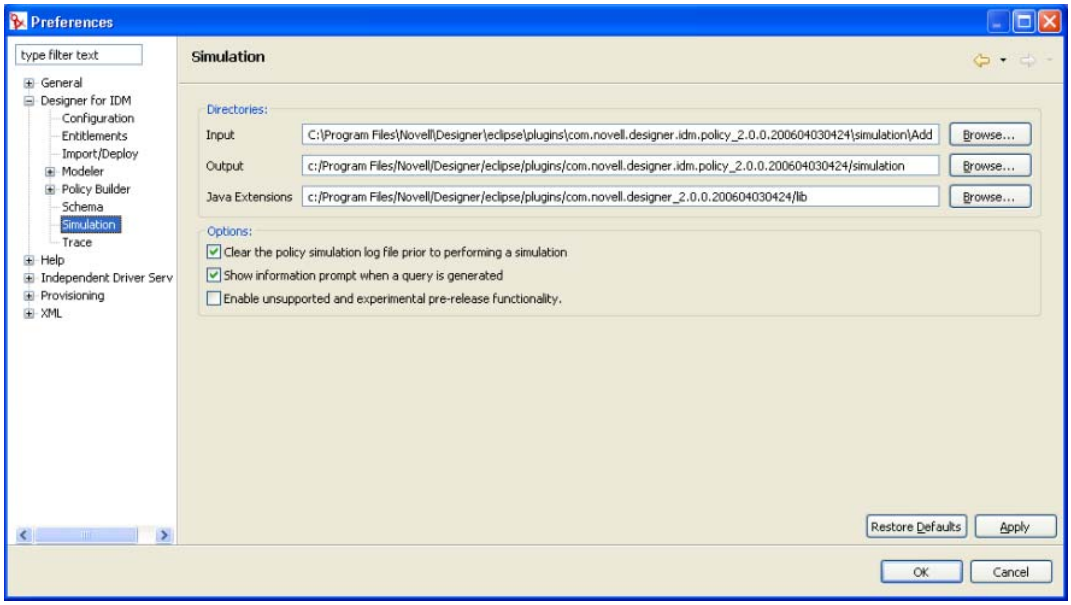
现在可以模拟引用外部 Java 扩展的策略，方法是指定 jar 文件所在的目录。

要确定或更改扩展目录，请执行以下操作：

1 从工具栏中选择 *Windows* >" 自选设置 "。

2 导航到 *Designer for IDM* > *Simulation* （" 用于 IDM 的 Designer">" 模拟 "）。

3 将包含 Java 类的 jar 文件复制到指定目录，然后模拟策略。



注释：*Enable unsupported and experimental pre-release functionality*（启用不受支持和实验性的预发布功能）选项允许策略模拟器测试在线 Identity Vault 或已连接系统的策略。在 Designer 1.2 中该选项不受支持，且没有相关文档。

2.2.8 编辑 DirXML 底稿

在 Designer 中，您可以使用 XML 编辑器或文本编辑器来查看、编辑和验证 XML。

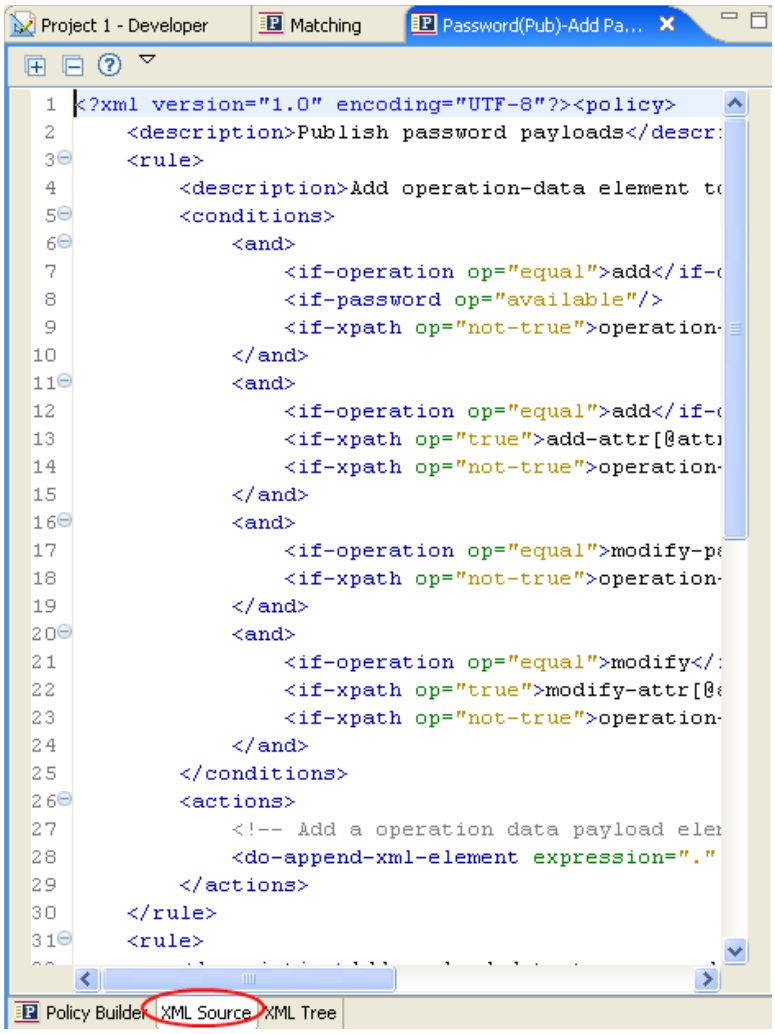
- “查看 XML 源”在第 105 页
- “编辑 XML 源”在第 109 页
- “验证 XML 源”在第 112 页

查看 XML 源

可以查看 XML 格式或 XML 树格式的 XML 源。

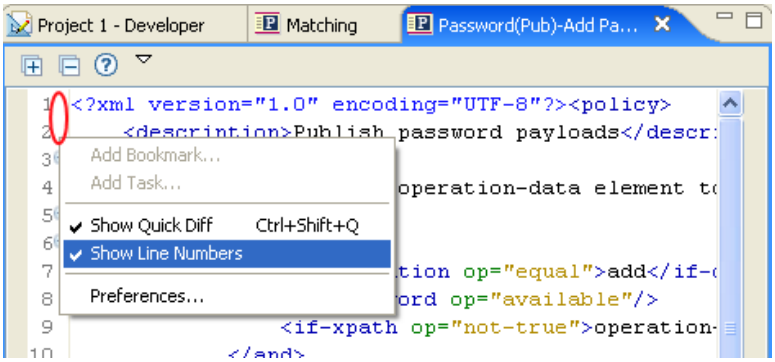
要打开 XML 源视图，请执行以下操作：

1 单击策略构建器工作区底部的 *XML Source* （XML 源）。



XML 编辑器可以显示行号。

2 要查看行号，可以右击左边距，然后选择 " 显示行号 "。



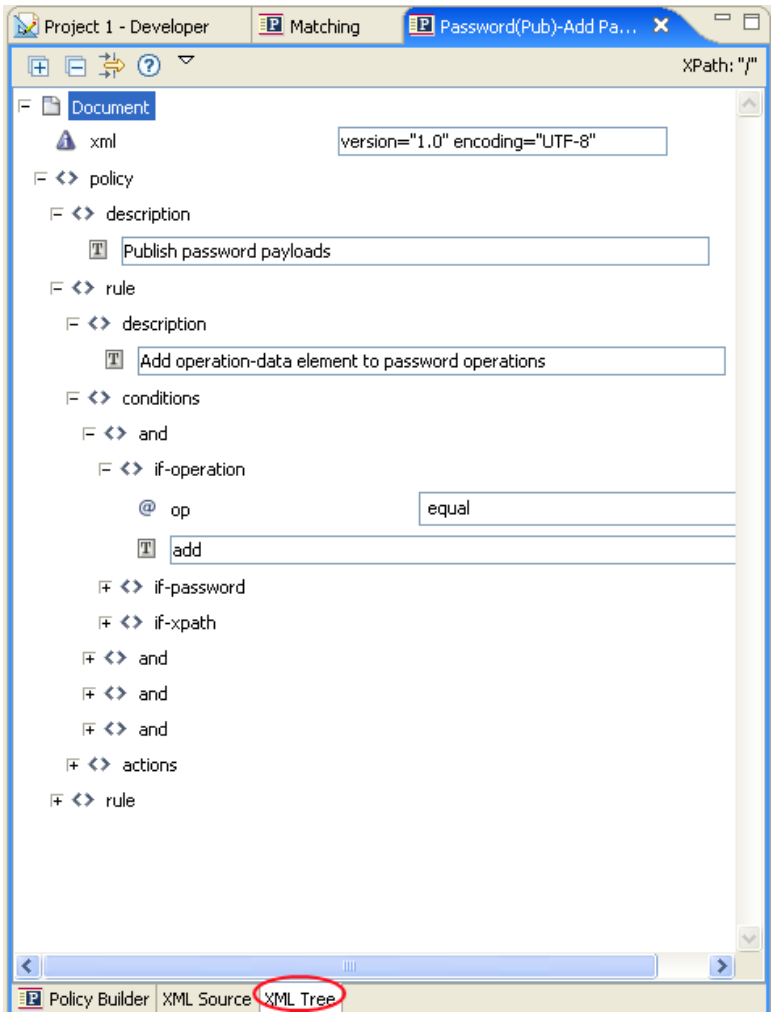
XML 编辑器可以按函数展开或折叠 XML。如果函数中包含大量 XML，可以通过单击左上角的减号图标折叠 XML。

3 要展开所有 XML 函数，则单击左侧的加号图标。

在左边距中，每个要素都有其各自的加号或减号图标。

要以树格式查看 XML，请执行以下操作：

1 单击策略构建器工作区底部的 *XML Tree* （XML 树）。

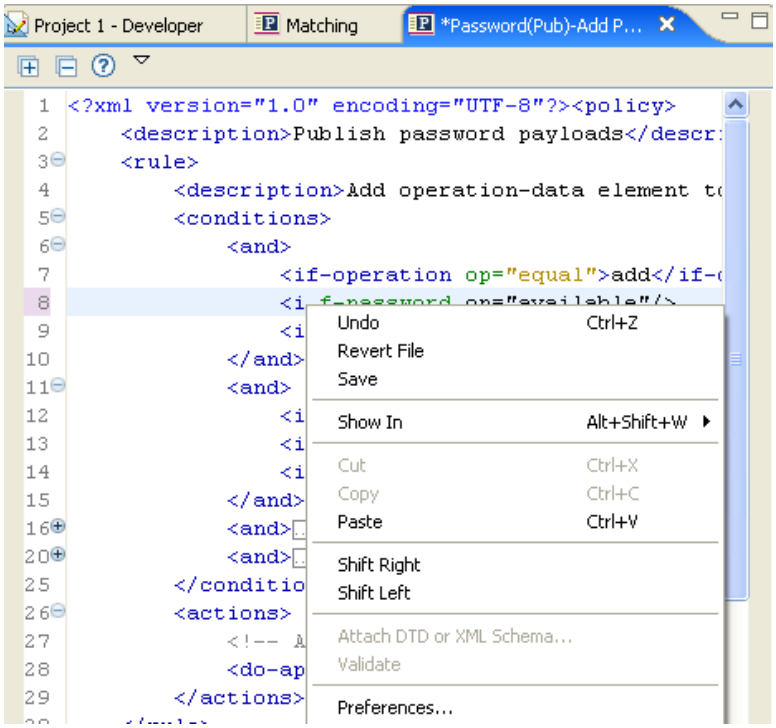


要查看整个树视图，请展开列出的每一项。

编辑 **XML** 源

可以通过 XML 编辑器来编辑 XML。可以在此处进行更改，也可以通过 GUI 界面来进行更改。

图 ***2-12*** *编辑 XML 源*



加载的默认编辑器与 .xml 文件类型相关联。如果找不到默认编辑器，则装载系统文本编辑器。 XML 源视图的功能取决于所加载的编辑器。

右击以显示 XML 编辑器所包含的功能列表。

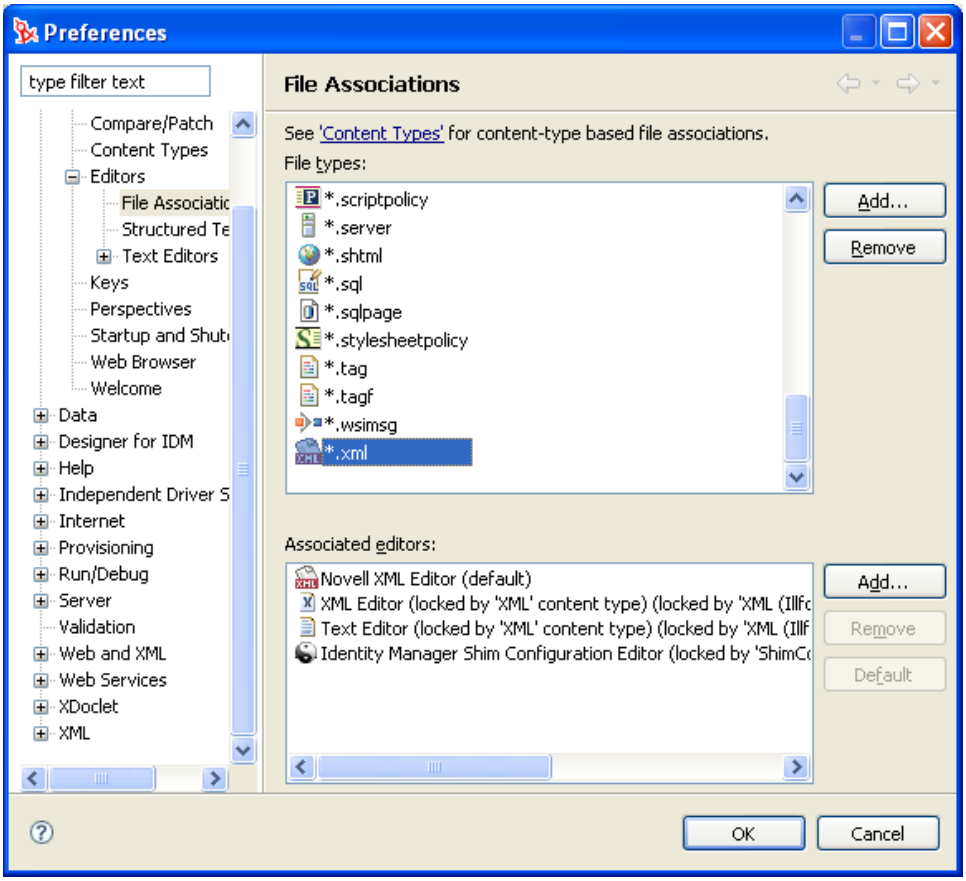
表 ***2-5*** *XML 编辑器选项*

功能	说明
Undo （复原）	复原上一操作。
Revert File （还原文件）	将文件还原为所保存的上一版本。
Saves （保存）	保存文件。
Cut （剪切）	剪切所选信息。
Paste （粘贴）	将信息粘贴到文档中。
Shift Right （右移）	将本行向右缩排。
Shift Left （左移）	将本行向左缩排。
Attach DTD or XML Schema （挂接 DTD 或 XML 纲要）	挂接 DTD 或 XML 纲要文件以验证策略。

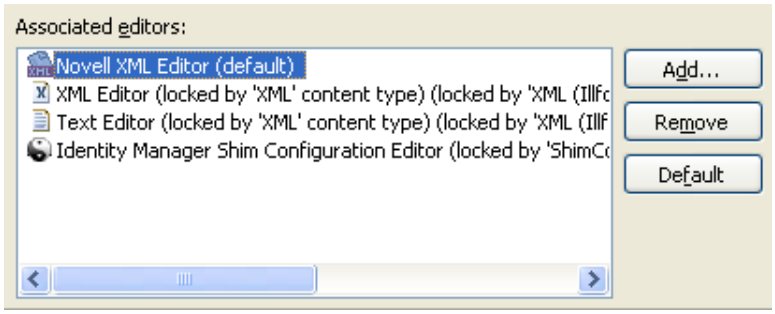
功能	说明
Validate （验证）	验证 XML 代码。
Preferences （自选设置）	设置 XML 编辑器的自选设置。

要选择源视图的其它 XML 编辑器，请执行以下操作：

1. 在 " 主 " 菜单中，选择 " 窗口 ">" 自选设置 "。
2. 选择 " 常规 ">" 编辑器 "> *File Associations* （文件关联）。
3. 从 " 文件类型 " 下面的列表中选择 **.xml*。



4 在 Associated editors（关联的编辑器）窗格中，选择希望使用的编辑器（例如， Novell XML 编辑器）。（如果列表中没有您需要的编辑器，可以单击 " 添加 "，然后将其添加到列表中。）



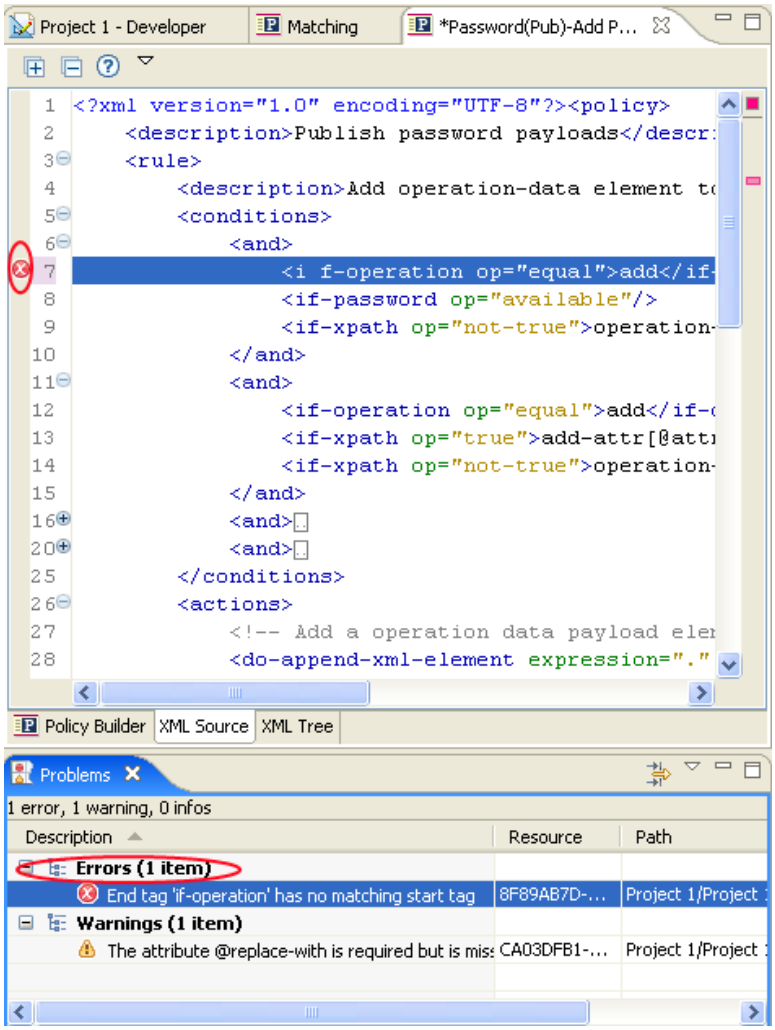
5 单击 " 确定 "。

6 关闭并重新打开策略构建器。

验证 **XML** 源

XML 编辑器可以验证 XML 代码。右击并选择 " 验证 "。如果存在错误，则出现错误的行将显示一个红色的 x。窗口底部的解释将提供有关该问题的详细信息。

图 ***2-13*** 验证 *XML* 源



在以上示例中， if-operation 的结束标签与开始标签不匹配。

2.3 正则表达式

正则表达式是匹配遵循某种模式的文本字符串的公式。正则表达式由普通字符和元字符组成。普通字符包括大写字母、小写字母和数字。元字符则具有特殊的含义。下表包含一些最常用的元字符及其含义。

表 **2-6** 常用正则表达式

元字符	说明
.	与任意单个字符相匹配。
$	匹配行尾。
^	匹配行首。
*	匹配之前出现了零次或多次的某一字符。
\	文字转义符。允许搜索任意元字符。例如，\$ 会查找 $1000，而不是查找与行的结尾相匹配的字符。
[ ]	与方括号中的任一字符相匹配。
[0-9]	匹配用连字符连接的一组字符。此示例将匹配所有数字。
[A-Za-z]	同时匹配多个范围。此示例将匹配所有大写和小写字母。

自变量构建器设计为按照 Java 中的定义使用正则表达式。有关详细信息，请访问 Java 万维网站点 (http://java.sun.com/j2se/1.4.2/docs/api/java/util/regex/Pattern.html)。

2.4 XPath 1.0 表达式

某些条件、操作和标记的自变量使用 XPath 1.0 表达式。 XPath 是一种语言，可以为 XSLT 和 XPointer 之间共享的功能提供通用的语法和语义。它主要用于 XML 文档的寻址部分，但也提供了处理字符串、数字和布尔值的基本工具。

XPath 规范要求嵌入的应用程序提供的环境应包括由应用程序定义的若干信息。在 DirXML 底稿 （请参见 “DirXML 底稿”在第 11 页 ）中， XPath 在以下环境中求值：

- 环境节点为当前操作。
- 环境位置和大小均为 1。
- 有若干可用变量：
  - 以下变量可用作 Identity Manager 中样式表的参数 （当前为 fromNDS、srcQueryProcessor、 destQueryProcessor、 srcCommandProcessor、destCommandProcessor 和 dnConverter）。
  - 全局配置变量。
  - 局部策略变量。
  - 如果不同变量源间出现名称冲突，则优先权的顺序将为局部变量、样式表参数、全局变量。
- 在策略要素上声明名称空间。
- 有若干可用功能：
  - 所有的内置 XPath 1.0 功能。
  - 由 NXSL 提供的 Java 扩展功能。

必须在策略要素上声明将 Java 类与前缀相关联的名称空间声明。

有关详细信息，请访问 W3 万维网站点 (http://www.w3.org/TR/1999/REC-xpath-19991116)。

2.5 条件

本节包含有关可以通过策略构建器界面使用的所有条件的详细信息。

2.5.1 If Association

对当前操作或当前对象的关联值进行测试。

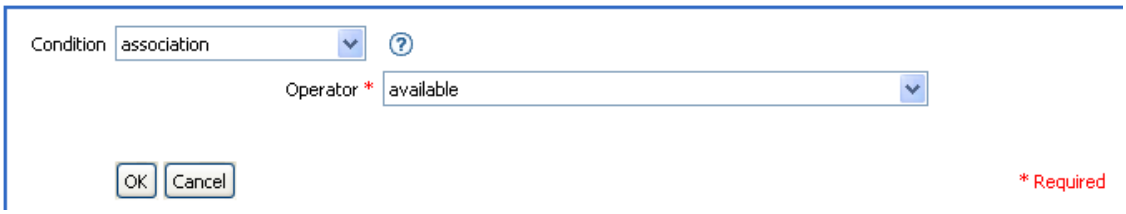
字段

操作符与满足条件的情形

操作符	满足条件的情形
associated	当前对象已建立一个关联。
available	存在由当前操作指定的非空关联值。
equal	当前操作指定的关联值与 If Association 的内容完全相同。
not-associated	当前对象未建立关联。
not available	与当前对象间的关联不可用。
not-equal	当前操作指定的关联值不等于 If Association 的内容。

示例

本示例测试关联是否可用。如果满足条件，将执行所定义的操作。



2.5.2 If Attribute

对当前操作或源数据存储区中的当前对象的特性值进行测试。此条件从逻辑上可以视为 If Operation Attribute 或 If Source Attribute，因为只有在源数据存储区或操作中满足条件此测试才能成功。

字段

名称

指定要测试的特性的名称。

操作符

选择此条件的测试类型。

比较方式

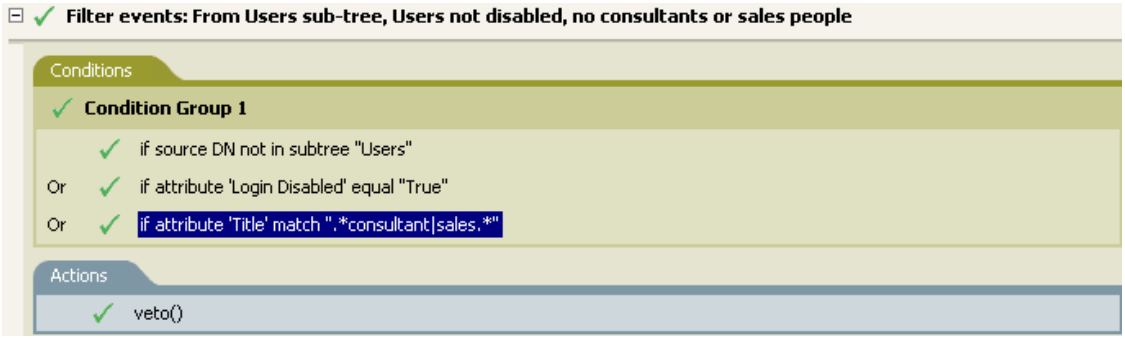
选择比较方式。请参见 “比较方式”在第 193 页 。

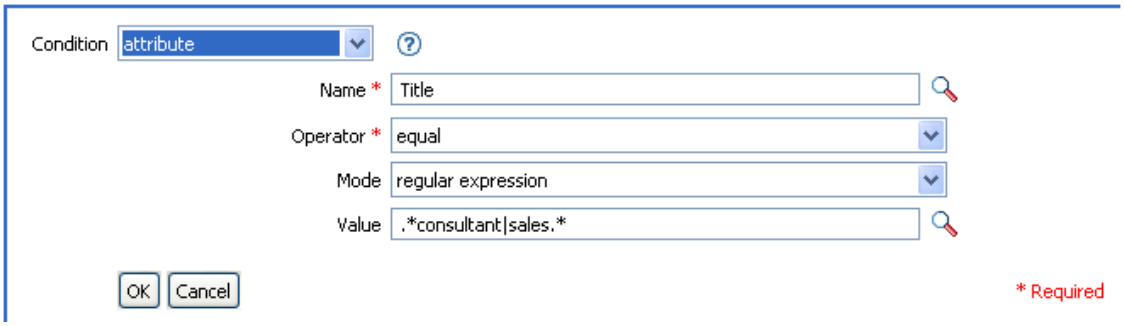
操作符与满足条件的情形

操作符	满足条件的情形
available	在当前操作或数据存储区中，存在指定特性的可用值。
equal	在当前操作或源数据存储区中，存在指定特性的可用值，它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

示例

下面的示例在过滤禁用的或具有特定职务的用户对象时使用 If Attribute 条件。此策略是 Policy to Filter Events （过滤事件策略），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





此条件查找 Title （职务）特性的值为 consultant 或 sales 的所有用户对象。

2.5.3 If Class Name

对当前操作中的对象类名称进行测试。

字段

操作符

选择此条件的测试类型。

比较方式

选择比较方式。请参见 “比较方式”在第 193 页 。

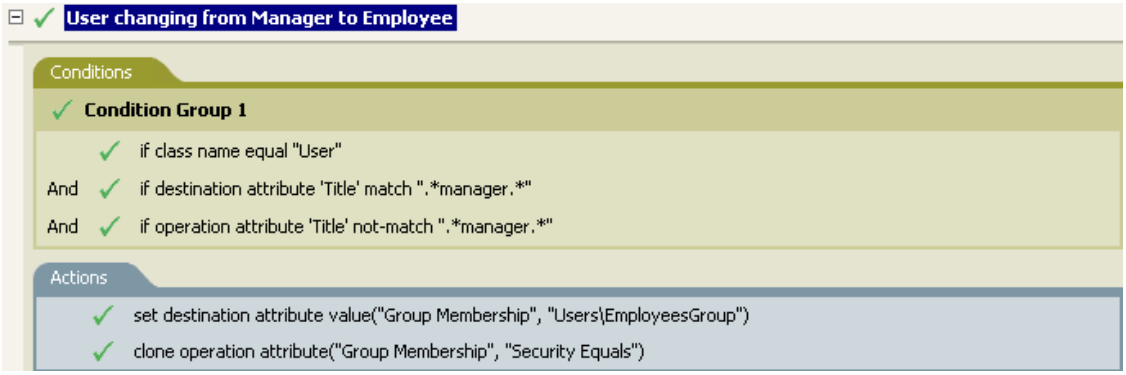
操作符与满足条件的情形

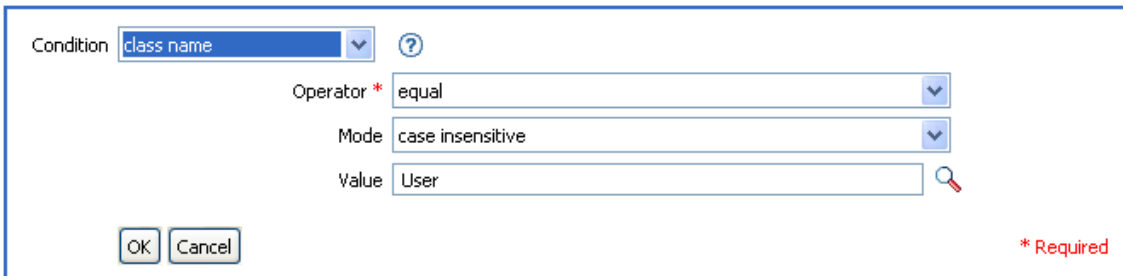
操作符	满足条件的情形
available	当前操作中有一个可用的对象类名称。
equal	当前操作中有一个可用的对象类名称，它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。

操作符	满足条件的情形
not-equal	如果等于，则返回 False。

示例

本示例将根据用户对象的职务使用 If Class Name 条件管理他们的组成员资格。所用的策略是 Govern Groups for User Based on Title Attribute（根据职务特性管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





检查当前对象的类名称是否为 User。

2.5.4 If Destination Attribute

对目标数据存储区中的当前对象的特性值进行测试。

字段

名称

指定要测试的特性的名称。

操作符

选择此条件的测试类型。

比较方式

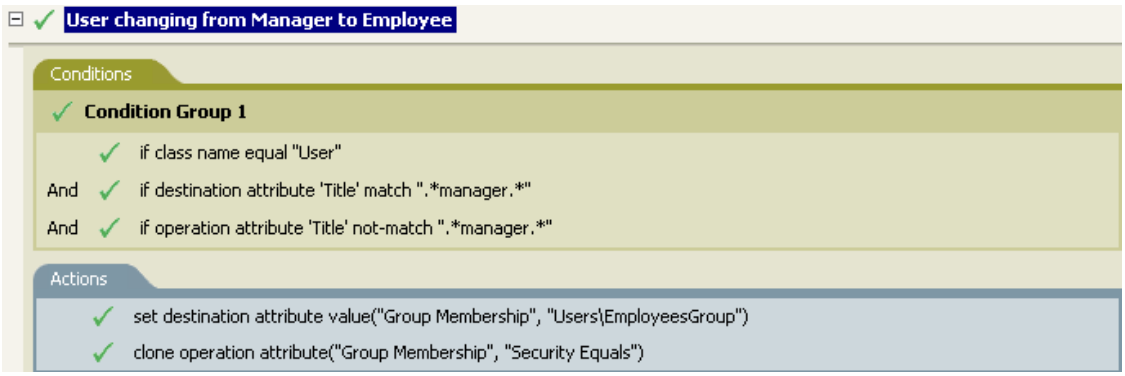
选择比较方式。请参见 “比较方式”在第 193 页 。

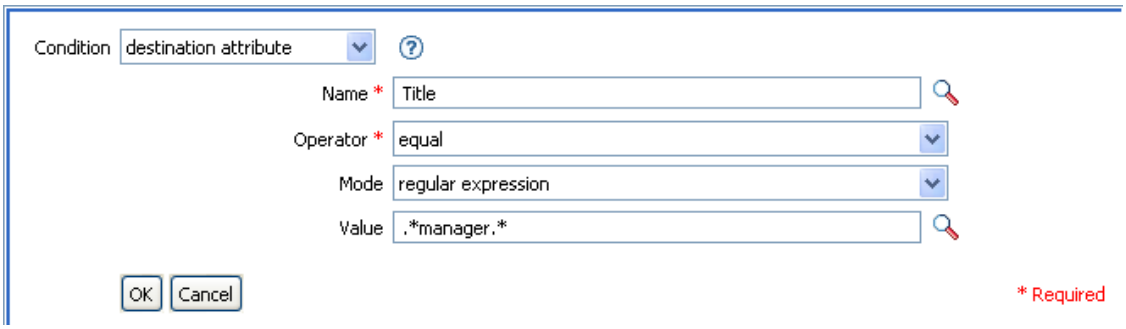
操作符与满足条件的情形

操作符	满足条件的情形
available	目标数据存储区中存在一个指定特性的可用值。
equal	在目标数据存储区中存在一个指定特性的可用值，它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

示例

本示例将根据用户对象的职务使用 If Attribute 条件管理他们的组成员资格。所用的策略是 Govern Groups for User Based on Title Attribute （根据职务特性管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见“可下载的 Identity Manager 策略”在第 31 页。





此策略检查 Title 特性的值是否包含 manager。

2.5.5 If Destination DN

对当前操作中的目标 DN 进行测试。

字段

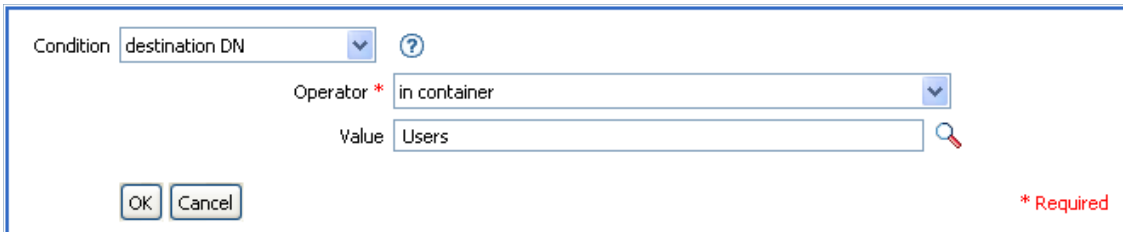
操作符

选择此条件的测试类型。

操作符与满足条件的情形

操作符	满足条件的情形
available	存在可用的目标 DN。
equal	存在可用的目标 DN，它等于使用适用于目标数据存储区的 DN 格式的语义进行比较时得到的值。
in-container	存在可用的目标 DN，当使用适用于目标数据存储区的 DN 格式的语义与该目标 DN 进行比较时，它表示由值指定的树枝中的某个对象。
in-subtree	存在可用的目标 DN，当使用适用于目标数据存储区的 DN 格式的语义与该目标 DN 进行比较时，它表示由值指定的子树中的某个对象。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。
not-in-container	若为 In-container，则将返回 False。
not-in-subtree	若为 In-subtree，则返回 False。

示例



2.5.6 If Entitlement

对当前操作或 Identity Vault 中当前对象的权利进行测试。

字段

名称

指定所选条件下要测试的权利的名称。

操作符

选择此条件的测试类型。

比较方式

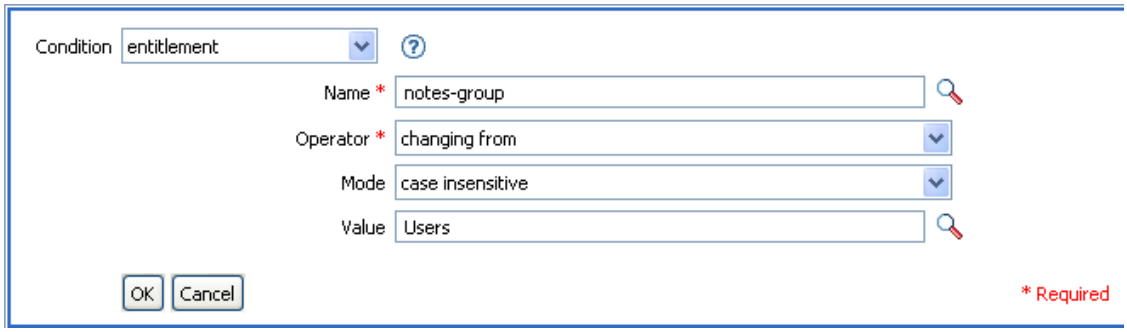
选择比较方式。请参见 “比较方式”在第 193 页 。

操作符与满足条件的情形

操作符	满足条件的情形
available	命名权利在当前操作或 Identity Vault 中均可用。
changing	当前操作包含对命名权利的更改 （修改特性或添加特性）。

操作符	满足条件的情形
changing-from	当前操作包含的更改将去除命名权利的某个值 （去除值），该权利的一个值等于使用指定的比较方式进行比较时得到的值。
changing-to	当前操作包含向命名权利添加值 （添加值或添加特性）的更改。它的一个值等于使用指定的比较方式进行比较时得到的值。
equal	在目标数据存储区中存在一个指定特性的可用值，它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-changing	若为 Changing，则返回 False。
not-changing-from	若为 Changing-from，则返回 False。
not-changing-to	若为 Changing-to，则返回 False。
not-equal	如果等于，则返回 False。

示例



2.5.7 If Global Configuration Value

对全局配置变量进行测试。

字段

名称

指定所选条件下要测试的全局变量的名称。

操作符

选择此条件的测试类型。

比较方式

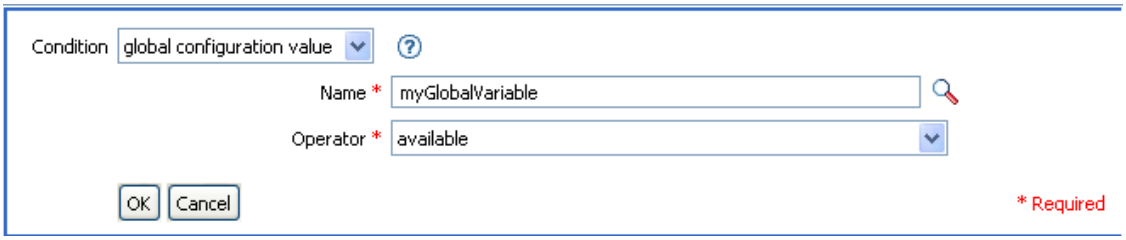
选择比较方式。请参见 “比较方式”在第 193 页 。

操作符与满足条件的情形

操作符	满足条件的情形
available	存在具有指定名称的全局配置变量。

操作符	满足条件的情形
equal	存在具有指定名称的全局配置变量，其值等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

示例



2.5.8 If Local Variable

对局部变量进行测试。

字段

名称

指定所选条件下要测试的局部变量的名称。

操作符

选择此条件的测试类型。

比较方式

选择比较方式。请参见 “比较方式”在第 193 页 。

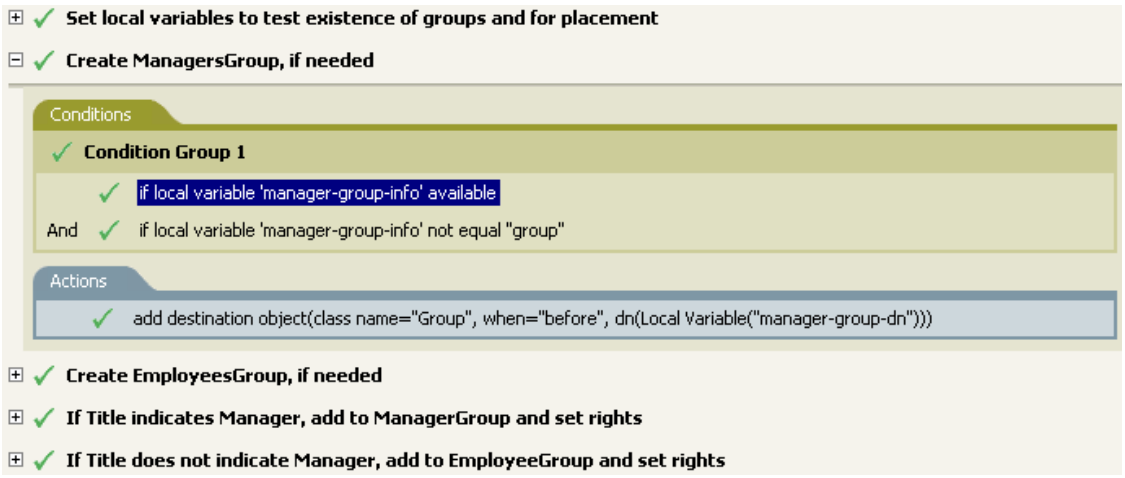
操作符与满足条件的情形

操作符	满足条件的情形
available	存在具有指定名称的局部变量，它由策略中早期规则的操作定义。
equal	存在具有指定名称的局部变量，其值等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

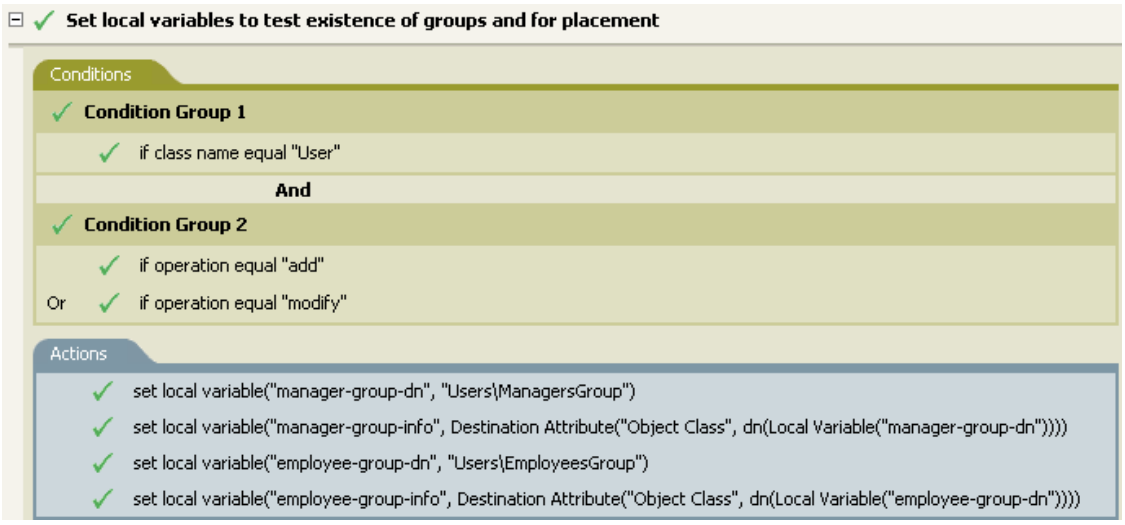
示例

下面的示例根据职务，向相应的 Employee 或 Manager 组中添加用户对象。在需要时，它也能创建组并设置与该组相对应的安全性。所用的策略是 Govern Groups for User Based on Title

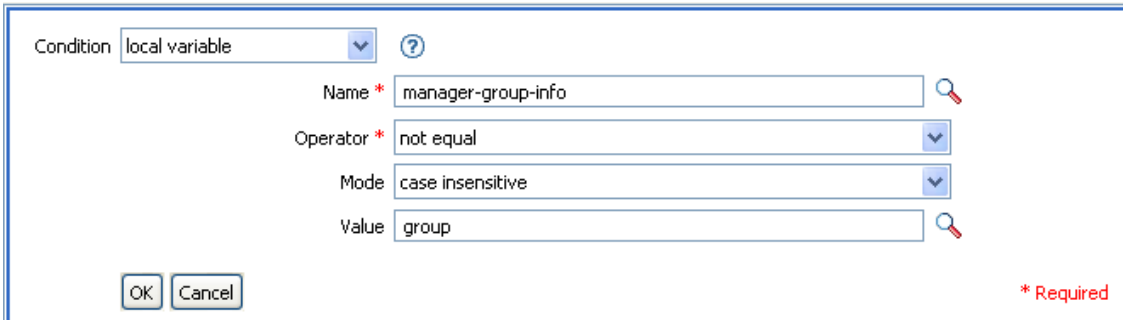
Attribute （根据职务特性管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。



此策略包含互相依存的五条规则。



为了使局部变量条件生效，第一条规则将设置四个不同的局部变量，用于测试组及组的位置。



规则所查找的条件用于检查局部变量 manager-group-info 是否可用以及 manager-group-info 是否不等于 group。如果这些条件都满足，将添加组的目标对象。

2.5.9 If Named Password

对当前操作中具有指定名称的口令进行测试。

字段

名称

指定所选条件下要测试的命名口令的名称。

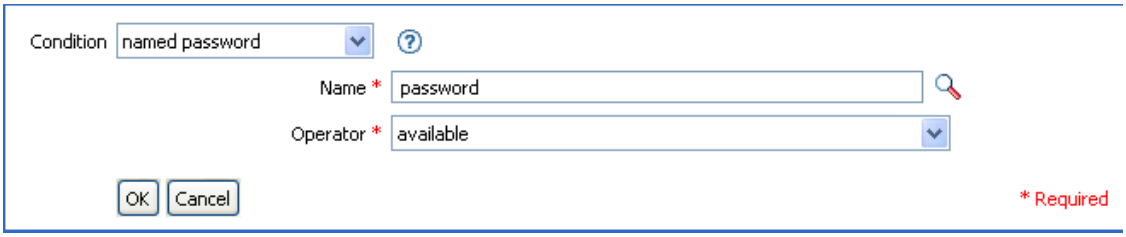
操作符

选择此条件的测试类型。

操作符与满足条件的情形

操作符	满足条件的情形
available	存在具有指定名称的口令。
not available	如果可用，则返回 False。

示例



2.5.10 If Operation

对当前操作的名称进行测试。

字段

操作符

选择此条件的测试类型。

操作符与满足条件的情形

操作符	满足条件的情形
equal	当前操作的名称与 If Operation 的内容完全相等。
not-equal	如果等于，则返回 False。

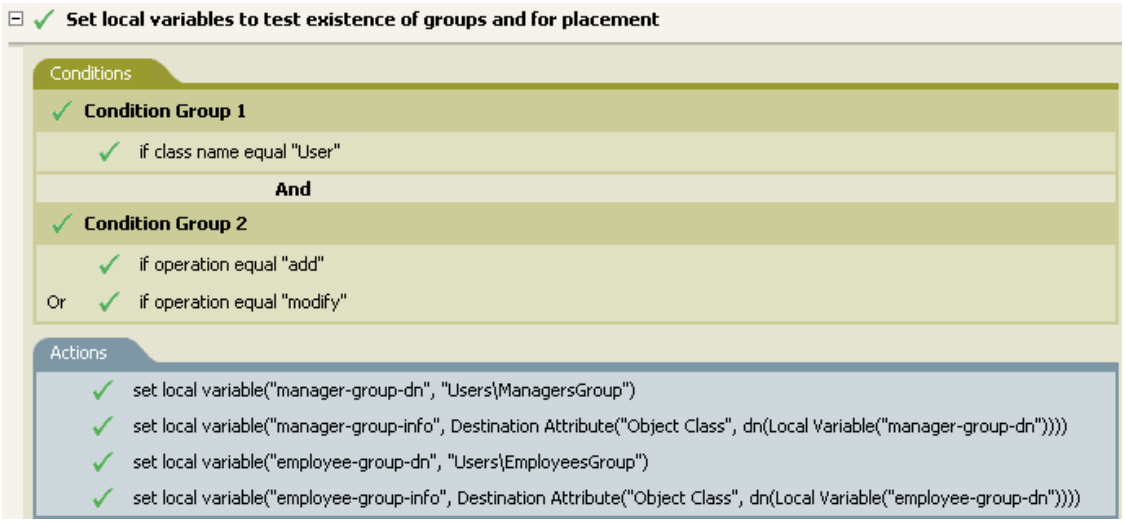
值

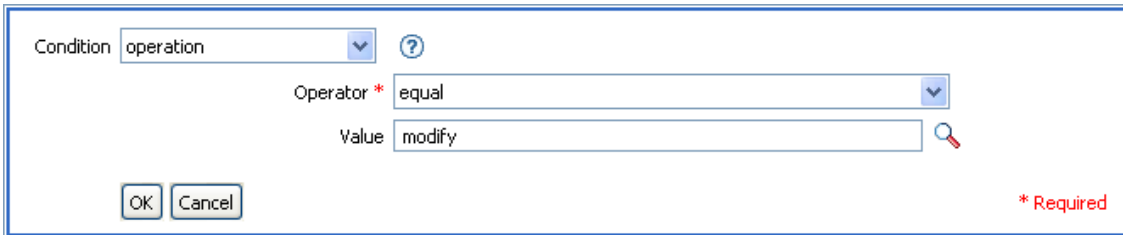
下列值是在此条件下 Metadirectory 引擎所搜索的操作：

- add
- add-association
- check-object-password
- delete
- get-named-password
- modify
- modify-association
- modify-password
- move
- init-params
- instance

示例

下面的示例根据职务，向相应的 Employee 或 Manager 组中添加用户对象。在需要时，它也能创建组并设置与该组相对应的安全性。此策略名为 Govern Groups for User Based on Title Attribute （根据职务特性管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





此条件检查是否已进行添加或修改操作。只要出现其中一项操作，就会设置局部变量。

2.5.11 If Operation Attribute

对当前操作中的特性值进行测试。

字段

名称

指定要测试的特性的名称。

操作符

选择此条件的测试类型。

比较方式

选择比较方式。请参见 “比较方式”在第 193 页 。

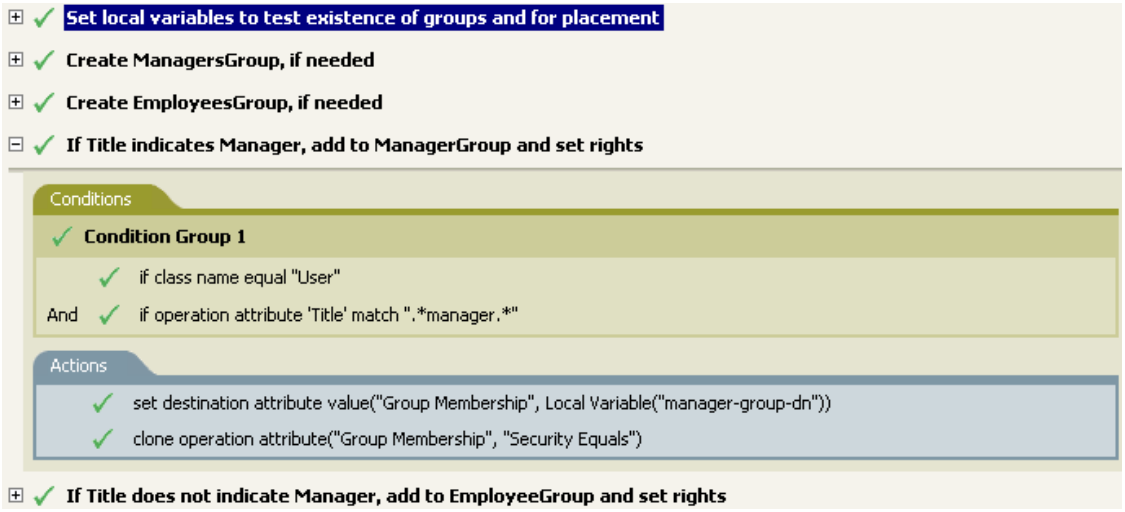
操作符与满足条件的情形

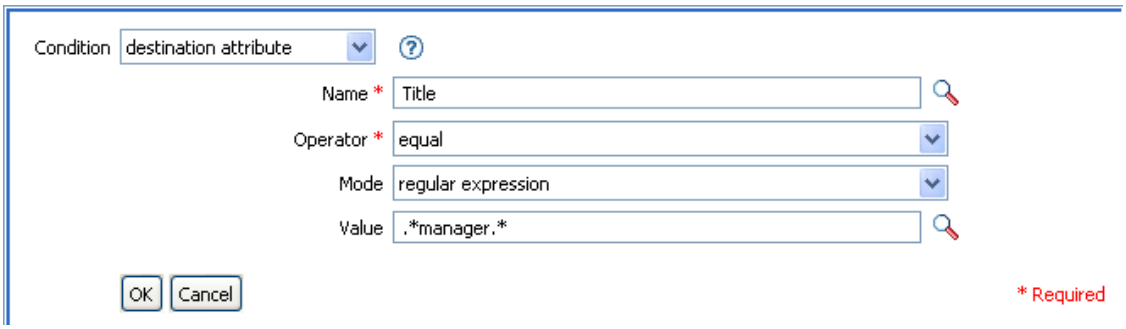
操作符	满足条件的情形
available	当前操作 （添加特性、添加值、特性）中存在可用于指定特性的值。
changing	当前操作包含对指定特性的更改 （修改特性或添加特性）。
changing-from	当前操作包含去除指定特性的某个值 （去除值）的更改。它等于使用指定的比较方式进行比较时得到的值。
changing-to	当前操作包含的更改将向指定的特性添加一个值 （添加值或添加特性）。它等于使用指定的比较方式进行比较时得到的值。
equal	当前操作中存在可用于指定特性的值 （非去除值）。它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-changing	若为 Changing，则返回 False。
not-changing-from	若为 Changing-from，则返回 False。
not-changing-to	若为 Changing-to，则返回 False。
not-equal	如果等于，则返回 False。

示例

下面的示例根据职务，向相应的 Employee 或 Manager 组中添加用户对象。在需要时，它也能创建组并设置与该组相对应的安全性。此策略名为 Govern Groups for User Based on Title

Attribute （根据职务特性管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





此条件检查 Title 特性是否等于正则表达式 .*manager*。它查找在 manager 之前没有或有多个字符以及在 manager 之后有一个字符的职务。如果用户对象的职务为 sales managers，则它将找到一个匹配。

2.5.12 If Operation Property

对当前操作的一个操作属性进行测试。

字段

名称

指定所选条件下要测试的操作属性的名称。

操作符

选择此条件的测试类型。

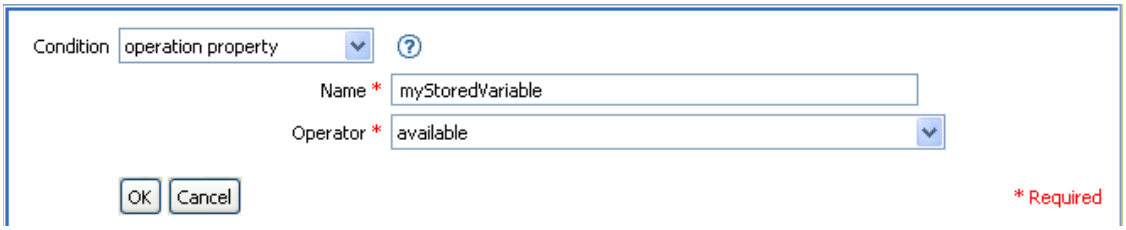
比较方式

选择比较方式。请参见 “比较方式”在第 193 页 。

操作符与满足条件的情形

操作符	满足条件的情形
available	当前操作中有一个具有指定名称的操作属性。
equal	当前操作中存在具有指定名称的操作属性，其值等于使用指定的比较方式进行比较时提供的内容。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

示例



2.5.13 If Password

对当前操作中的口令进行测试。

字段

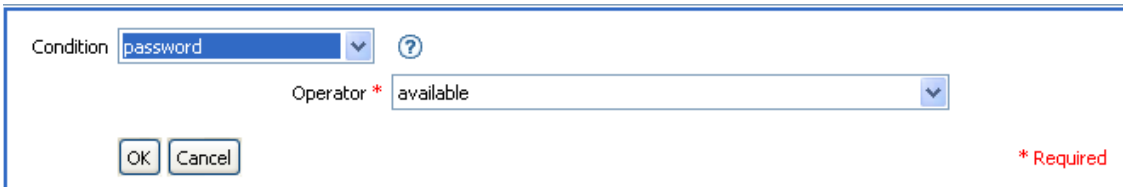
操作符

选择此条件的测试类型。

操作符与满足条件的情形

操作符	满足条件的情形
available	当前操作中存在一个可用的口令。
not available	如果可用，则返回 False。

示例



2.5.14 If Source Attribute

对源数据存储区中当前对象的特性值进行测试。

字段

名称

指定所选条件下要测试的源特性的名称。

操作符

选择此条件的测试类型。

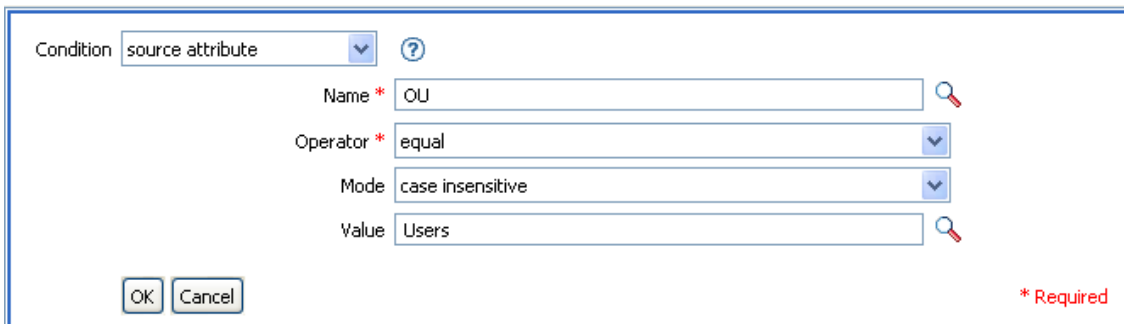
比较方式

选择比较方式。请参见 “比较方式”在第 193 页 。

操作符与满足条件的情形

操作符	满足条件的情形
available	源数据存储区中存在一个指定特性的可用值。
equal	源数据存储区中存在一个指定特性的可用值。它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

示例



2.5.15 If Source DN

对当前操作中的源 DN 进行测试。

字段

操作符

选择此条件的测试类型。

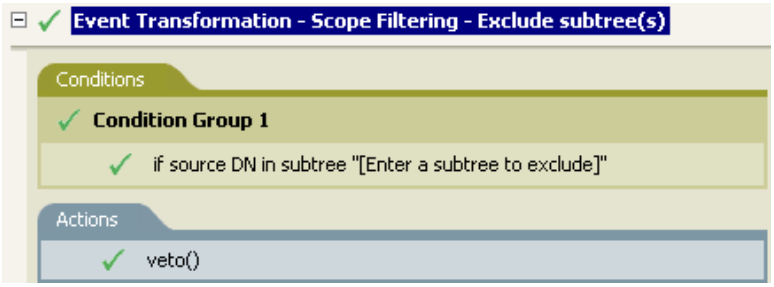
操作符与满足条件的情形

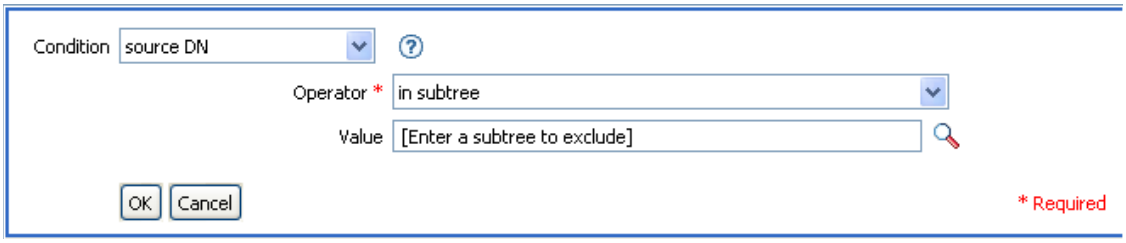
操作符	满足条件的情形
available	DN 可用。

操作符	满足条件的情形
equal	存在可用的源 DN，与指定值 in-container 的内容相等。存在可用的源 DN，表示由指定值标识的树枝中的对象。
in-subtree	存在可用的源 DN，表示由指定值标识的子树中的对象。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。
not-in-container	若为 In-container，则将返回 False。
not-in-subtree	若为 In-subtree，则返回 False。

示例

此示例使用 If Source DN 条件检查用户对象是否在源 DN 中。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “事件转换 - 范围过滤 - 排除子树”在第 79 页 。





此条件检查源 DN 是否位于用户树枝中。如果此对象属于该树枝，则将禁止此对象。

2.5.16 If XPath Expression

对 XPath 1.0 表达式的求值结果进行测试。

字段

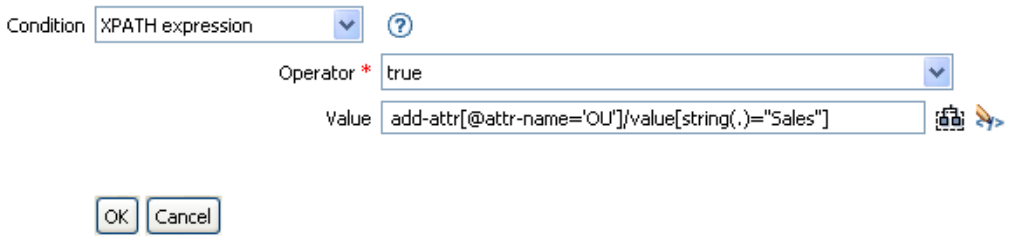
操作符

选择此条件的测试类型。

操作符与满足条件的情形

操作符	满足条件的情形
true	XPath 表达式的求值结果为 True。
false	如果为 True，则返回 False。

示例



2.6 操作

本节包含可通过策略构建器界面执行的所有操作的详细参考信息。

2.6.1 添加关联

向具有指定关联的 Identity Vault 发送添加关联命令。

字段

方式

选择是将该操作添加至当前操作，还是将其直接写入 Identity Vault。

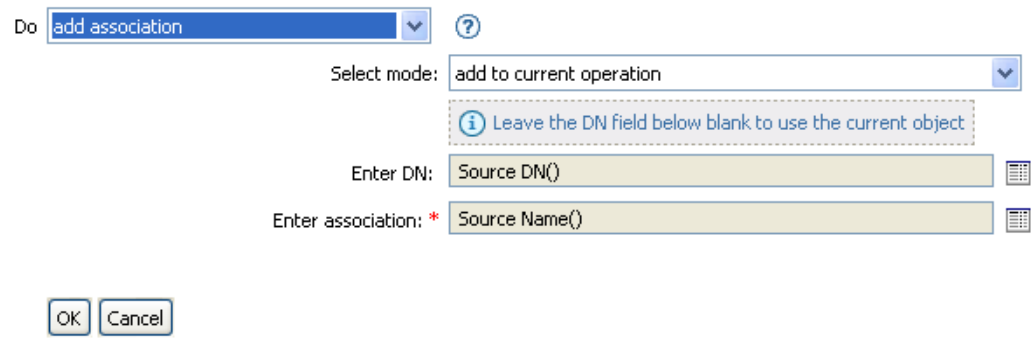
DN

指定目标对象的 DN，若保留空白则使用当前对象的 DN。

关联

指定要添加的关联值。

示例



2.6.2 添加目标特性值

向目标数据存储区中的某对象的特性添加一个值。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

值类型

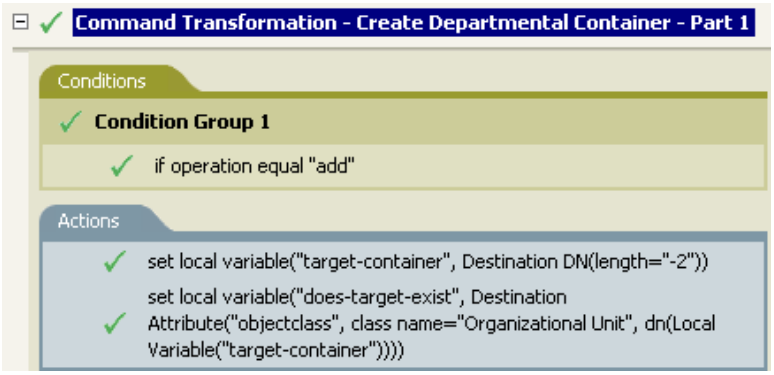
选择要添加的特性值的语法。

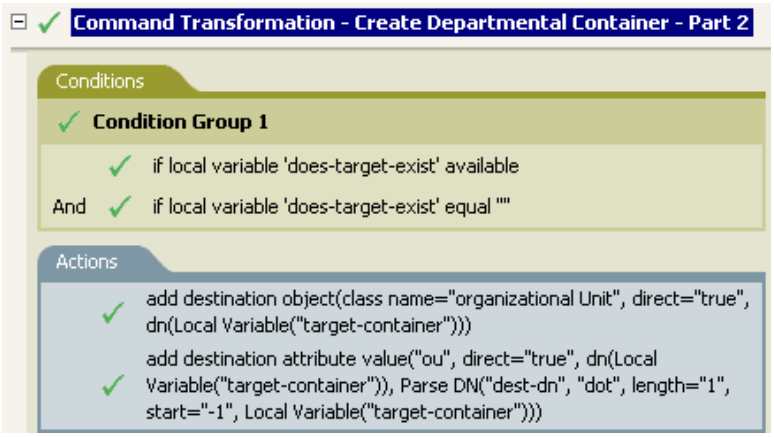
值

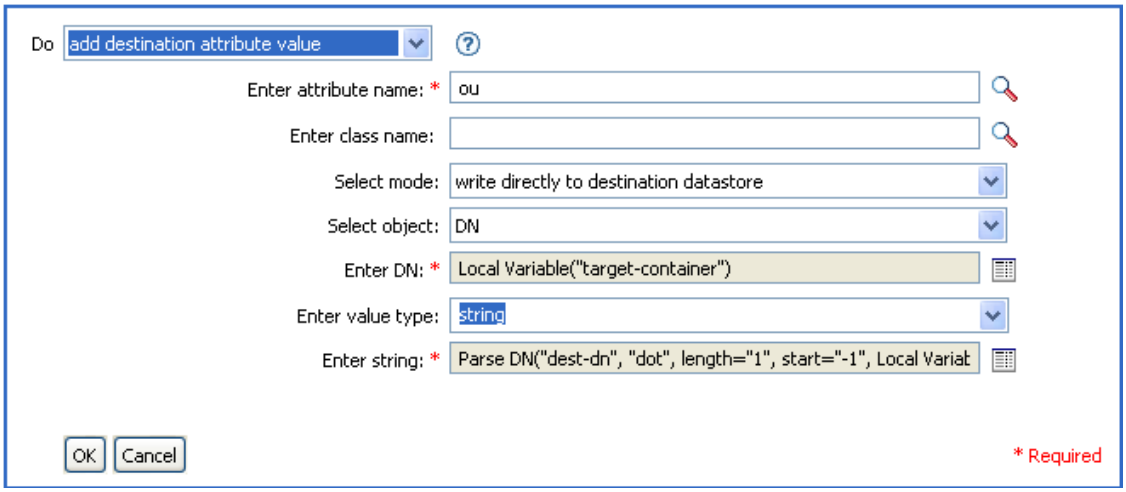
指定要添加的特性值。

示例

本示例在 OU 特性中添加目标特性值，该值由所创建的局部变量创建。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “命令转换 - 创建部门树枝 - 第 1 部分和第 2 部分”在第 69 页 。







2.6.3 添加目标对象

在目标数据存储区中新建对象。

字段

类名称

指定要创建对象的类名称。

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

DN

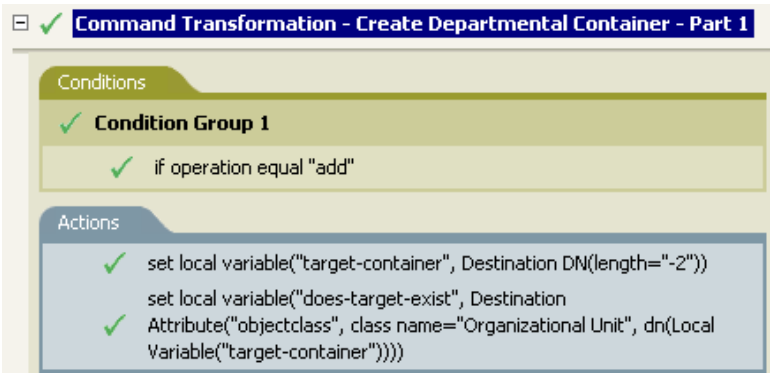
指定要创建的对象的 DN。

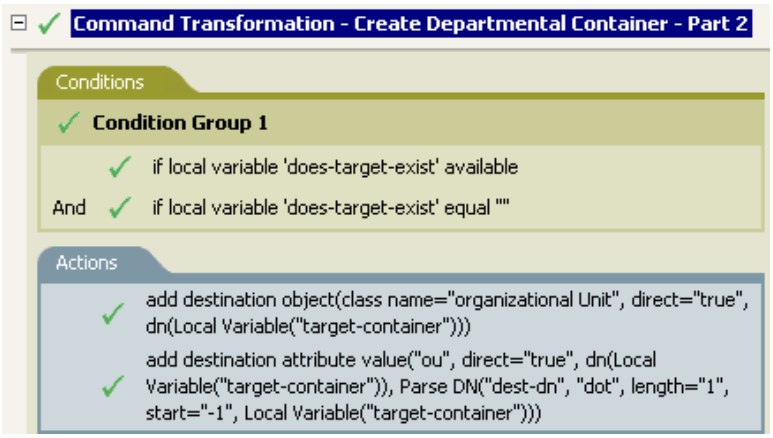
注释

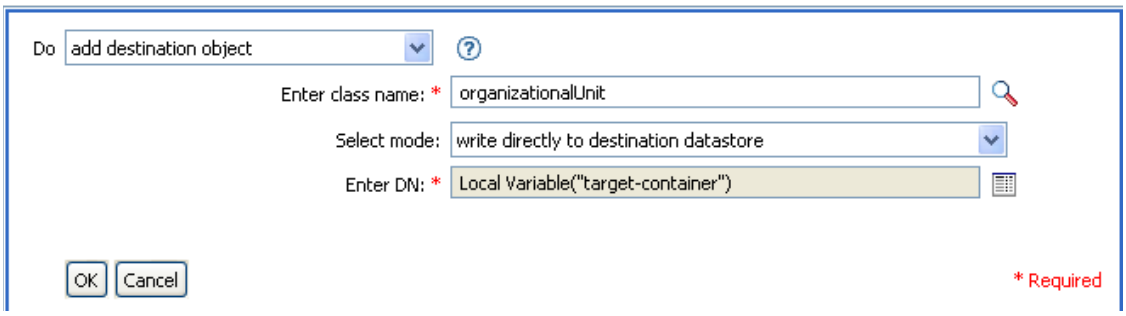
在创建对象的过程中添加的特性值，必须使用同一 DN 在后续操作 （“添加目标特性值”在第 132 页）中执行。

示例

本示例创建所需的部门树枝。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “命令转换 - 创建部门树枝 - 第 1 部分和第 2 部分”在第 69 页。







组织单元对象创建完毕。在此操作之后，通过目标特性值操作创建此 OU 特性的值。

2.6.4 添加源特性值

向源数据存储区中的某对象的特性添加一个值。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

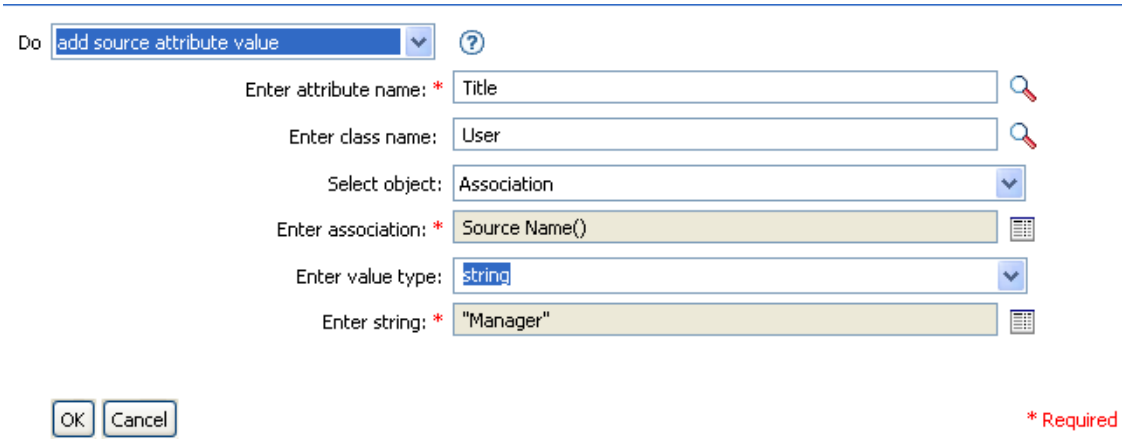
值类型

选择要添加的特性值的语法。

值

指定要添加的特性值。

示例



2.6.5 添加源对象

在源数据存储区中创建指定类型的对象。在创建对象的过程中添加的特性值，必须使用同一 DN 在后续操作 （添加源特性值 (在第 135 页)）中执行。

字段

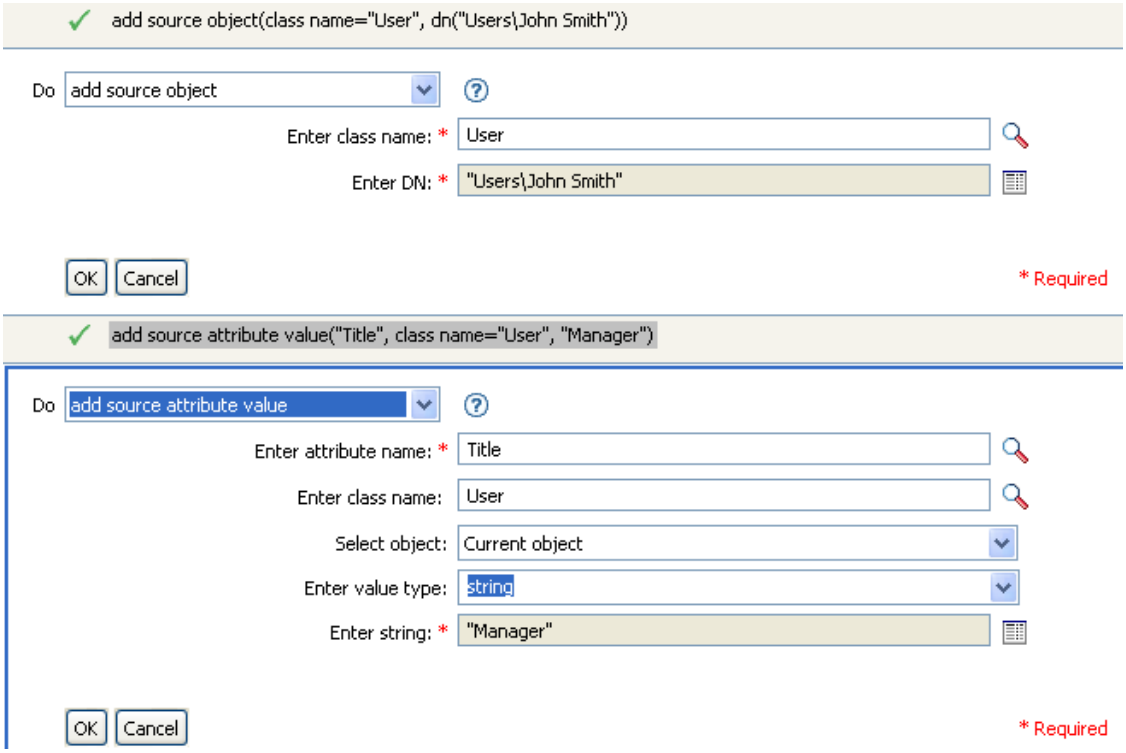
类名称

指定要添加对象的类名称。

DN

指定要添加对象的 DN。

示例



2.6.6 追加 XML 要素

在由 XPath 表达式选定的一组要素中，追加一个要素。

字段

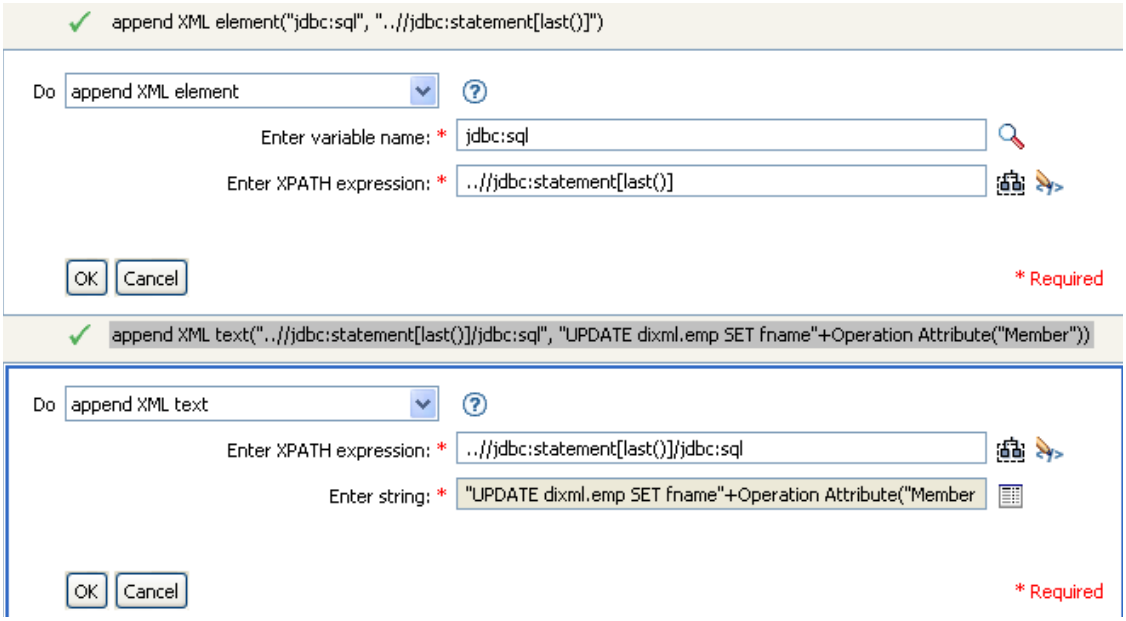
变量名

指定 XML 要素的标签名。如果先前已在此策略中定义了名称空间前缀，则标签名中可以包含该前缀。

XPath 表达式

指定一个 XPath 1.0 表达式，该表达式会返回一个节点集，其中包含要追加新要素的要素。

示例



2.6.7 追加 XML 文本

在由 XPath 表达式选定的一组要素中追加文本。

字段

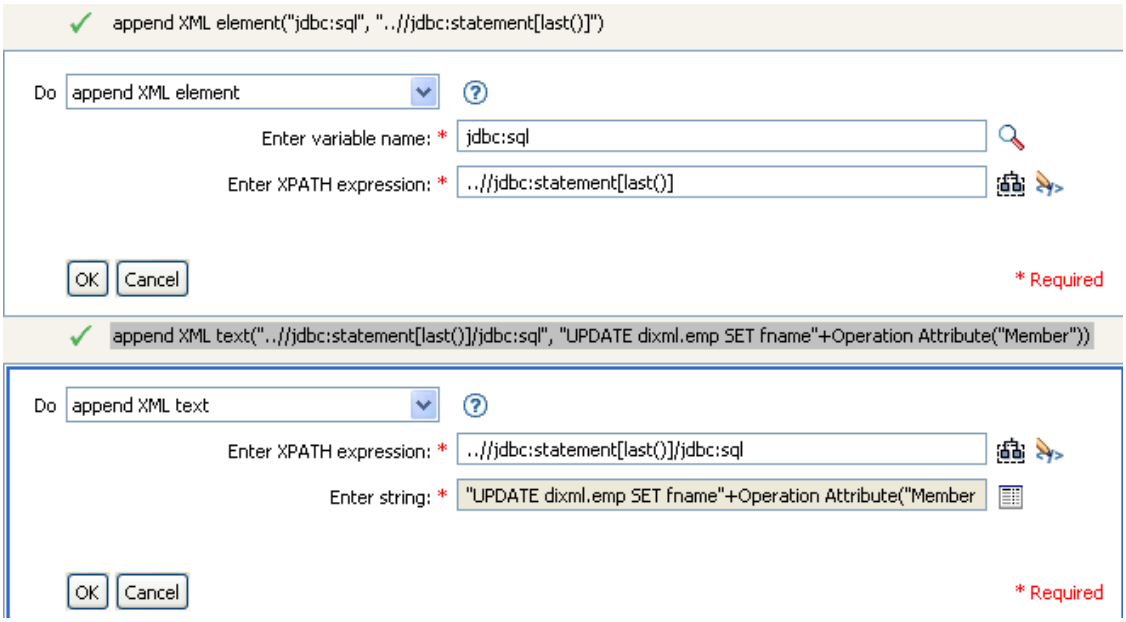
XPath 表达式

XPath 1.0 表达式，该表达式将返回一个节点集，其中包含要追加文本的要素。

字符串

指定要追加的文本。

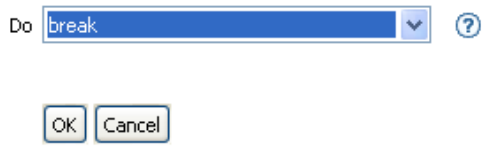
示例



2.6.8 中断

使用当前策略结束对当前操作的处理。

示例



2.6.9 清除目标特性值

在目标数据存储区中，将某对象一个特性的所有值去除。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

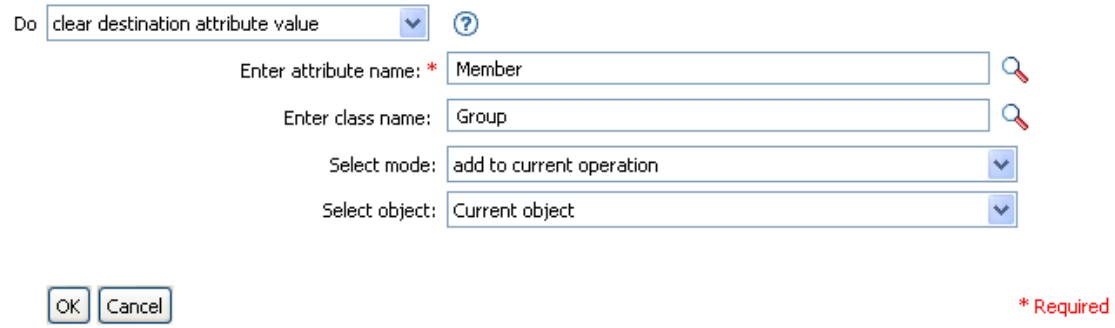
方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

示例



2.6.10 清除操作属性

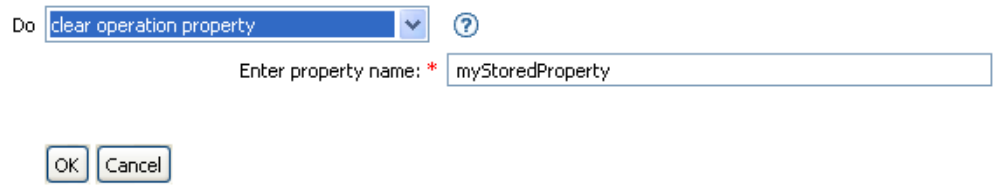
清除当前操作的所有操作属性。

字段

属性名

指定要清除的操作属性的名称。

示例



2.6.11 清除源特性值

在源数据存储区中，将某对象一个特性的所有值去除。

字段

特性名称

指定该特性的名称。

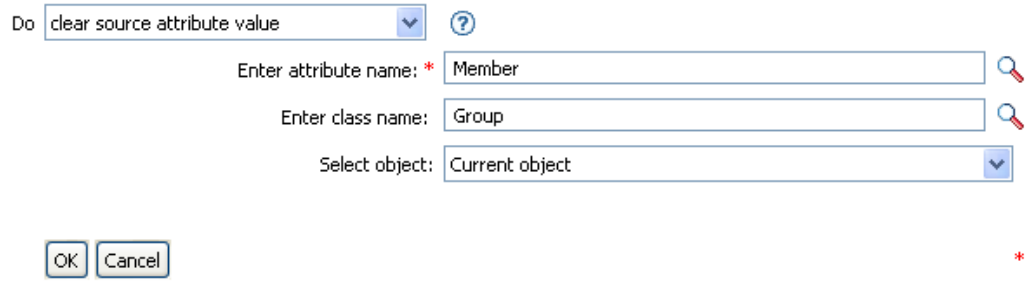
类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

示例



2.6.12 清除 SSO 身份凭证

清除一次签到身份凭证，以便取消对象供应。此操作是身份凭证供应策略的一部分。有关详细信息，请参见第 4 章 “Novell 身份凭证供应策略”在第 305 页。

字段

身份凭证储存对象 **DN**

指定储存库对象的 DN。

目标用户 **DN**

指定目标用户的 DN。

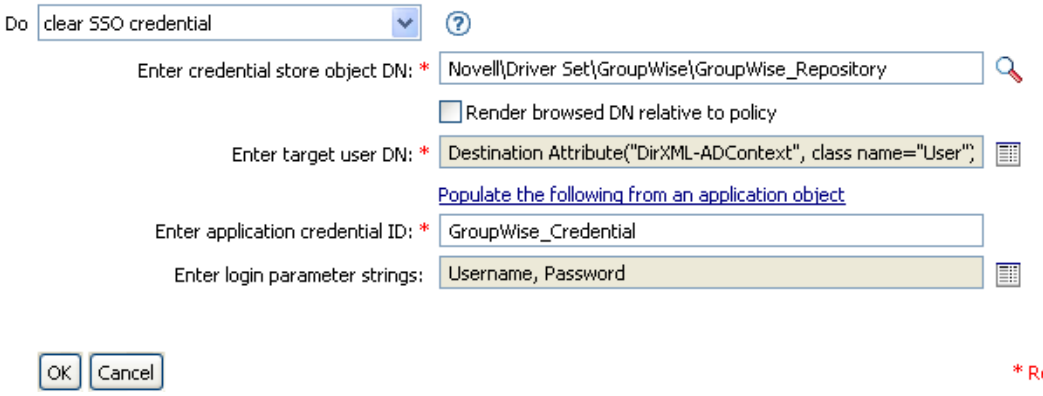
应用程序身份凭证 **ID**

指定储存在应用程序对象中的应用程序身份凭证。

登录参数字符串

指定应用程序的所有登录参数。登录参数是储存在应用程序对象中的鉴定密钥。

示例



2.6.13 通过 XPath 表达式克隆

将一个 XPath 表达式选定的 XML 节点集的深层拷贝追加到由另一个 XPath 表达式选定的一组要素中。

字段

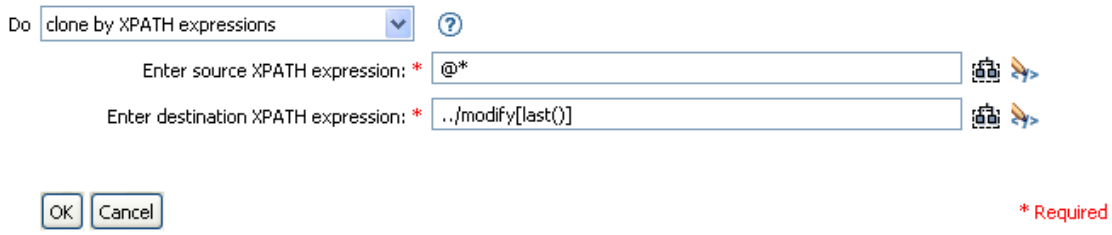
源 **XPath** 表达式

指定 XPath 1.0 表达式，该表达式返回的节点集中包含要复制的节点。

目标 **XPath** 表达式

指定 XPath 1.0 表达式，该表达式返回的节点集中包含追加已复制节点的要素。

示例



2.6.14 克隆操作特性

将当前操作中所有具体出现的某一特性复制到当前操作中的另一特性中。

字段

源名称

指定被复制的特性的名称。

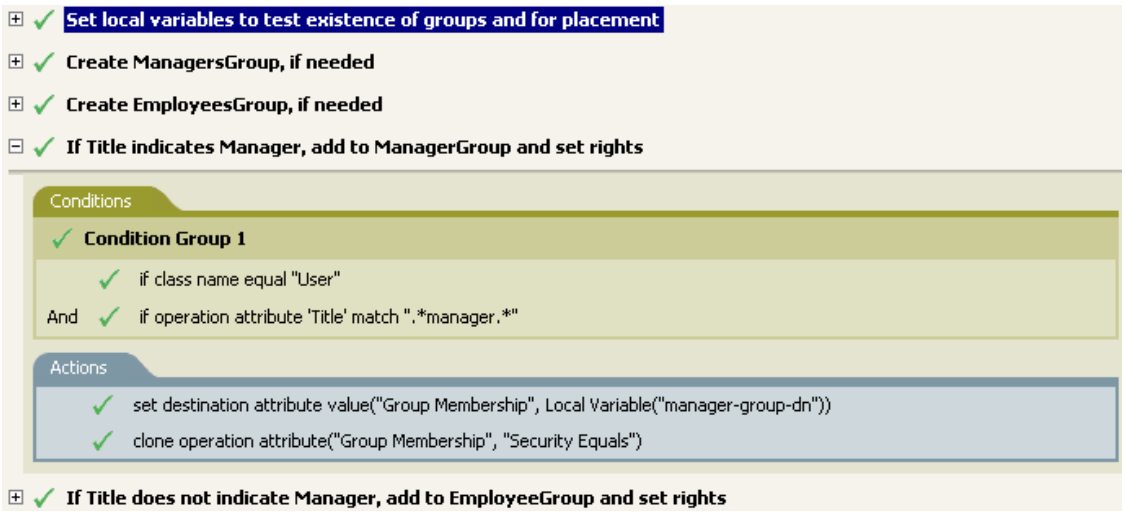
目标名称

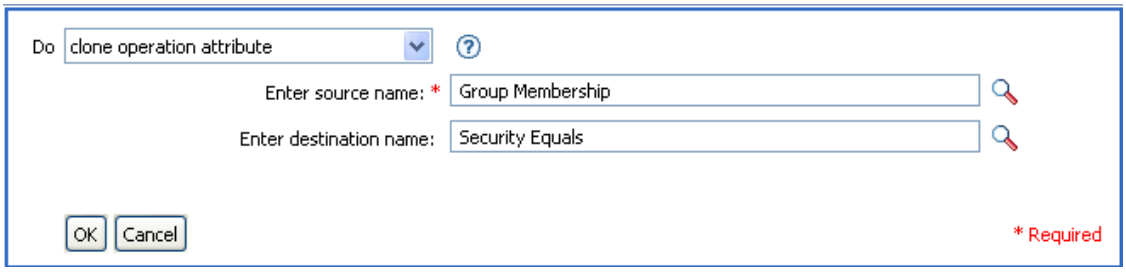
指定要复制到的特性的名称。

示例

下面的示例根据职务，向相应的 Employee 或 Manager 组中添加用户对象。在需要时，它也能创建组并设置与该组相对应的安全性。所用的策略是 Govern Groups for User Based on Title

Attribute （根据职务特性管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





克隆操作特性从 Group Membership （组成员资格）特性获取信息，并将该信息添加到 Security Equals （安全性等效）特性，使两特性的值相同。

2.6.15 删除目标对象

删除目标数据存储区中的一个对象。

字段

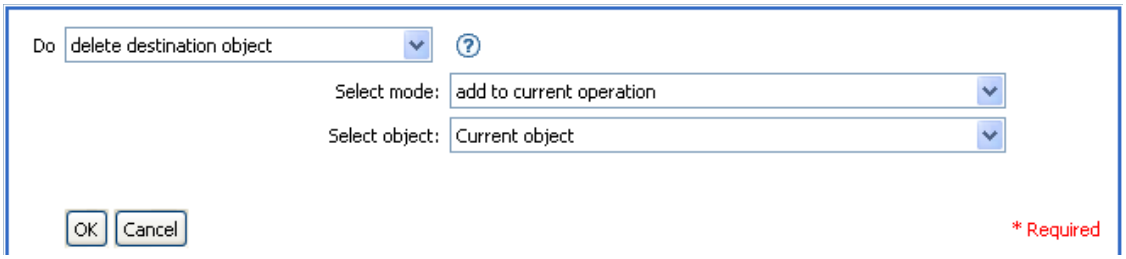
方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

示例



2.6.16 删除源对象

删除源数据存储区中的一个对象。

字段

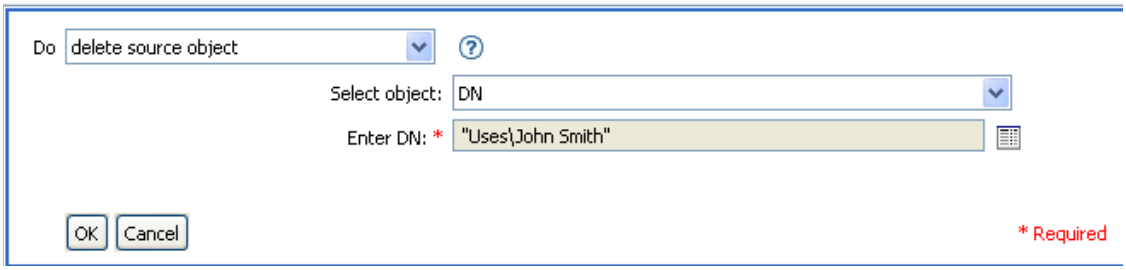
方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

选择要从源数据存储区中删除的目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

示例



2.6.17 查找匹配对象

在目标数据存储区中查找当前对象的匹配项。

字段

范围

选择搜索范围。范围可能是 entry （项）、 subordinates （从属项）或 subtree （子树）。

DN

指定作为搜索基础的 DN。

匹配特性

指定要搜索的特性值。

注释

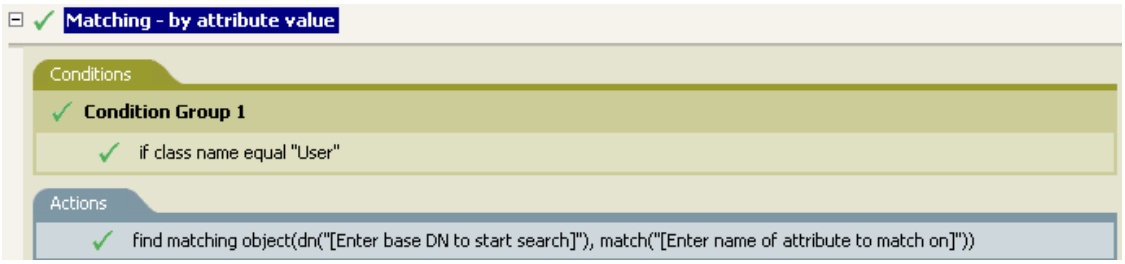
只有当前操是添加操作时，＂查找匹配对象＂才有效。

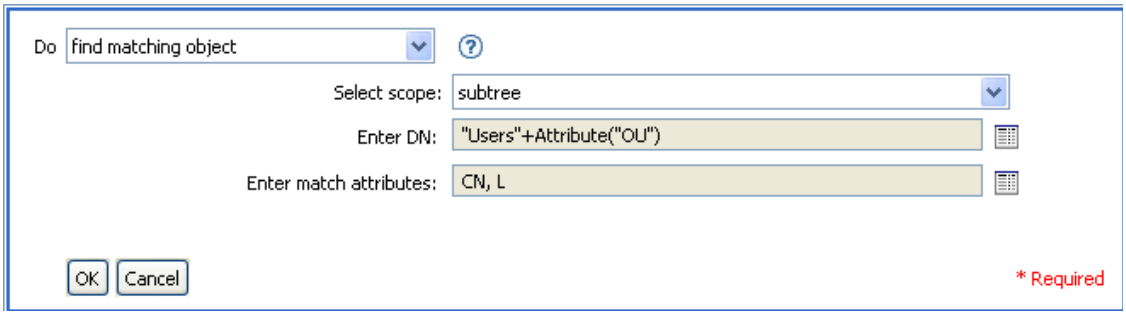
范围为 "entry"（项）时， DN 自变量是必需的，在其它情况下该自变量是可选的。范围为 "subtree"（子树）或 "subordinates"（从属项）时，至少需要一个匹配特性。如果范围为 entry 且存在多个指定的匹配特性，则结果为未定义。如果目标数据存储区为已连接的应用程序，会在返回的每个成功匹配的当前操作中添加关联。如果当前操作已存在非空关联，则不执行查询，因此可以根据同一规则将多个＂查找匹配对象＂操作连接在一起。

如果目标数据存储区是 Identity Vault，则应设置当前操作的目标 DN 特性。如果当前操作已存在非空目标 DN 特性，则不执行查询，因此可以根据同一规则将多个＂查找匹配对象＂操作连接在一起。如果只返回一个结果且该结果尚未关联，则将当前操作的目标 DN 设置为匹配对象的源 DN。如果只返回一个结果且该结果已关联，则将当前操作的目标 DN 设置为单一字符 ￼。如果返回多个结果，则将当前操作的目标 DN 设置为单一字符 &#xFFFD;。

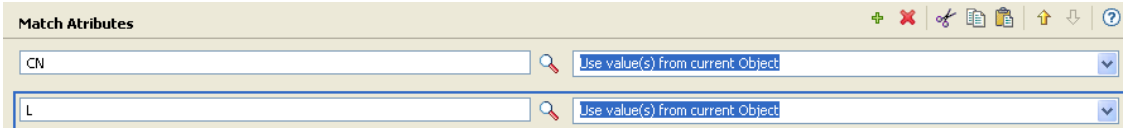
示例

本示例使用 CN 和 L 特性匹配用户对象。此规则搜索的开始位置是 Users 树枝，并将储存在 OU 特性中的信息添加至 DN。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “匹配 - 按特性值”在第 86 页 。





单击＂自变量构建器＂图标时，将出现＂匹配特性构建器＂。请在此构建器中指定要匹配的特性。本示例使用了 CN 和 L 特性。



2.6.18 对于每个

对节点集中的每个节点重复一组操作。

字段

节点集

指定节点集。

操作

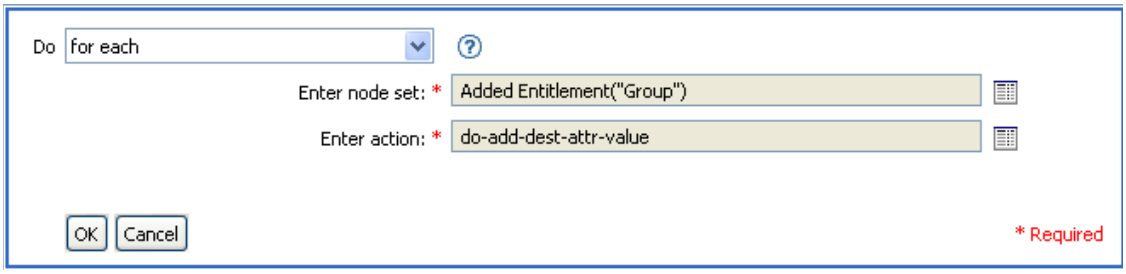
指定要在节点集的各个节点上执行的操作。

注释

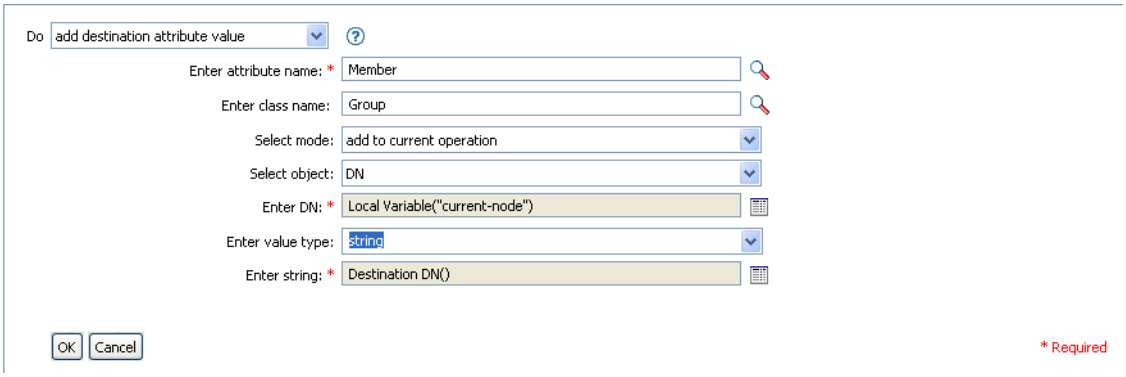
如果使用局部变量，则当前节点对于操作的每个迭代分别为不同的值。

如果节点集中的某个节点是权利，则对其隐式执行 “实施权利”在第 147 页 操作。

示例



在以下示例中，自变量操作构建器用于提供操作自变量：



2.6.19 生成事件

将用户定义的事件发送到 Novell Audit。

字段

ID

指定事件 ID。此 ID 必须是介于 1000 到 1999 之间的整数。

级别

选择事件级别。

级别	说明
记录紧急事件	引起 Metadirectory 引擎或驱动程序关闭的事件。
记录警报	需要立即注意的事件。
记录关键	引起 Metadirectory 引擎或驱动程序部分出现故障的事件。
记录错误	说明可由 Metadirectory 引擎或驱动程序处理的错误的事件。
记录警告	否定事件，并不表示出现问题。
记录通知	可供管理员了解或改进使用和操作的肯定或否定事件。
记录信息	任何重要的肯定事件。
记录调试	用于技术支持或工程师调试 Metadirectory 引擎或驱动程序时使用的相关事件。

字符串

指定事件中包含的由用户定义的字符串、整数和二进制值。这些值由命名字符串构建器提供。

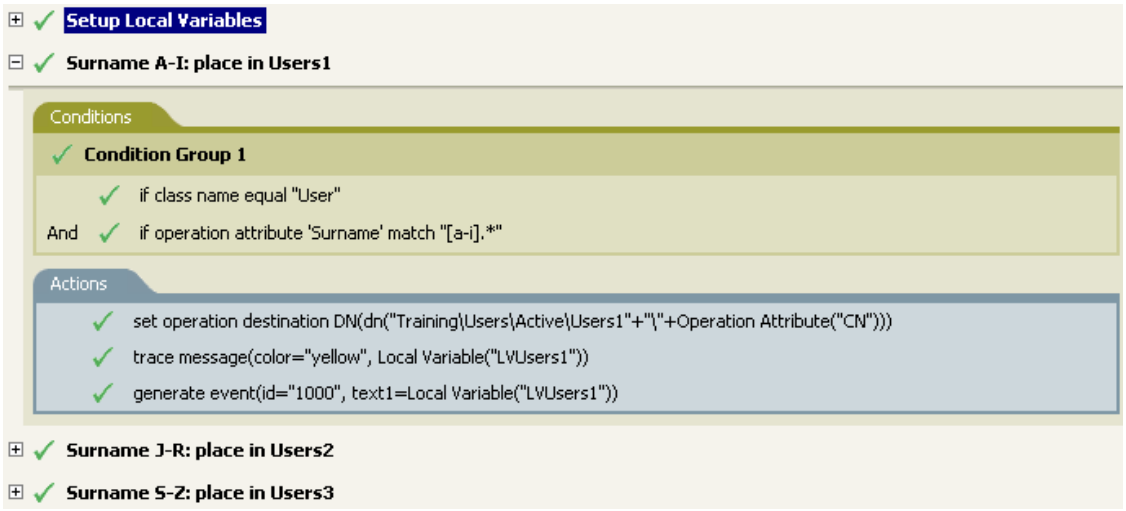
字符串名称	说明
target	要处理的对象。
target-type	指定目标预定义格式的整数。当前 target-type 的预定义值为： ◆ 0 = 无 ◆ 1 = 斜线表示法 ◆ 2 = 点表示法 ◆ 3 = LDAP 表示法
subTarget	要处理的目标的子组件。
text1	此处输入的文本储存在 text1 事件字段中。
text2	此处输入的文本储存在 text2 事件字段中。
text3	此处输入的文本储存在 text3 事件字段中。
value	此处输入的任何数字都将储存在 value 事件字段中。
value3	此处输入的任何数字都将储存在 value3 事件字段中。
data	此处输入的数据将储存在 BLOB 事件字段中。

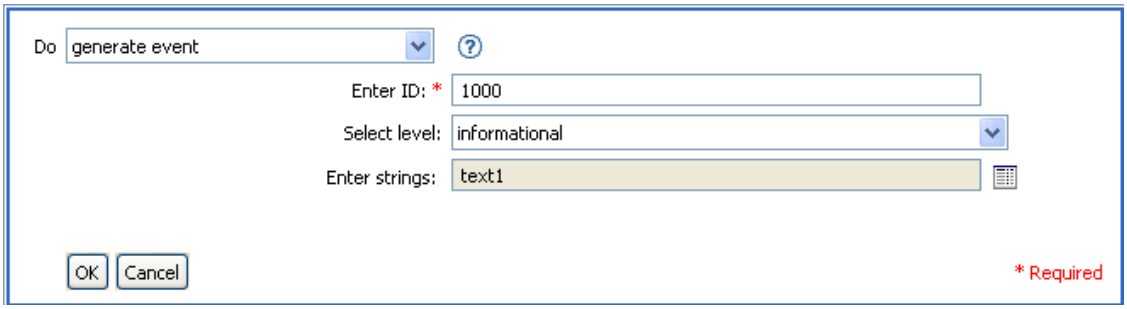
注释

Novell Audit 事件结构包含一个 target、一个 subTarget、三个字符串 （text1、 text2、 text3）、两个整数 （value、 value3）以及一个通用字段 (data)。如果您的环境不支持更大的数据字段，则文本字段最多可以输入 256 个字节，数据字段最多可以包含 3 KB 的信息。

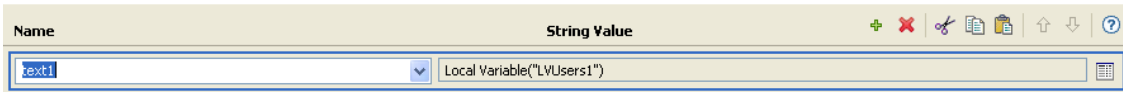
示例

本示例包含四条规则，它们根据 Surname （姓氏）特性的首字符实施对用户对象的布局策略。并生成一条跟踪讯息和一个自定义 Novell Audit 事件。 " 生成事件 " 操作用于发送事件 Novell Audit。策略名称为 Policy to Place by Surname （按姓氏布局策略），可从 Novell 支持万维网站点下载。详情请见 “可下载的 Identity Manager 策略”在第 31 页 。





在以下示例中，命名字符串构建器用于提供字符串自变量。



" 生成事件 " 创建具有 ID 1000 的事件，并显示 LVUser1 局部变量生成的文本。局部变量 LVUser1 就是字符串 User:Operation Attribute"cn"+"added to the"+"Training\Users\Active\Users1"+"container"。该事件读取 User:jsmith added to the Training\Users\Active\Users1 container。

2.6.20 实施权利

指定实施权利的操作，以便向授予或撤销权利的代理报告这些权利的状态。

字段

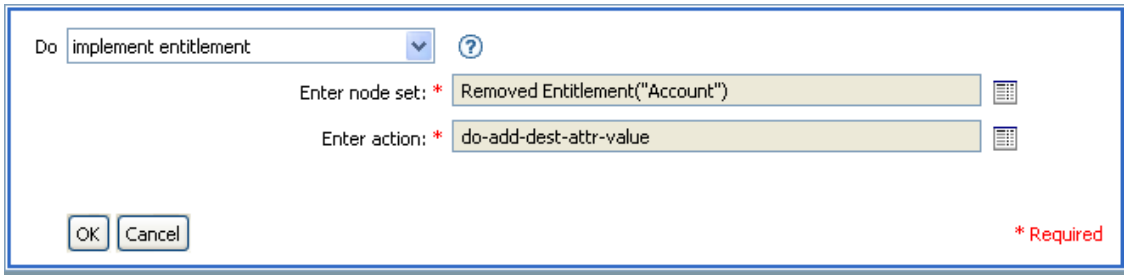
节点集

节点集包含由指定的操作实施的权利。

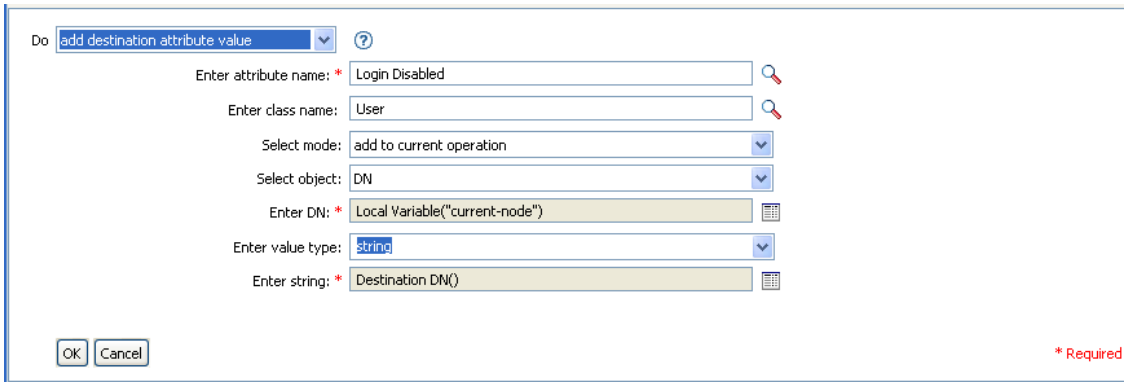
操作

实施指定权利的操作。

示例



在以下示例中，自变量操作构建器用于提供操作自变量：



2.6.21 移动目标对象

移动目标数据存储区中的对象。

字段

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

类名称

（可选）指定要移动的对象的类名称。若保留空白则使用当前对象的类名称。

要移动的对象

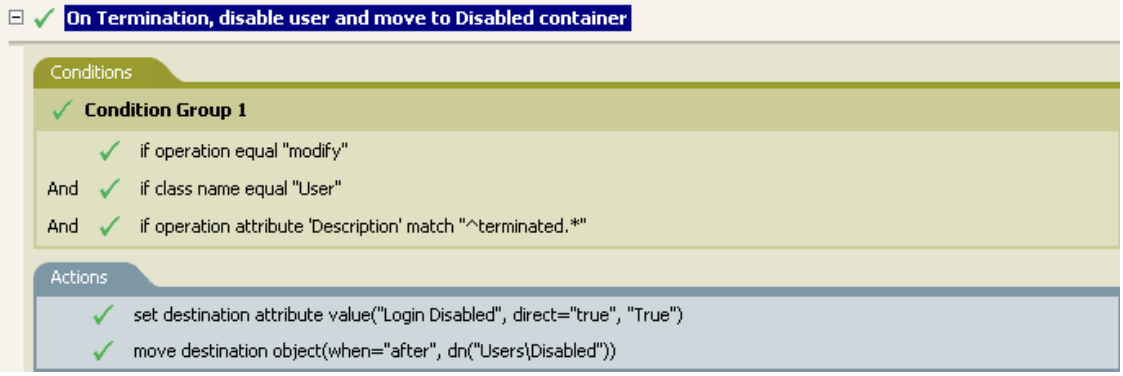
选择要移动的对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

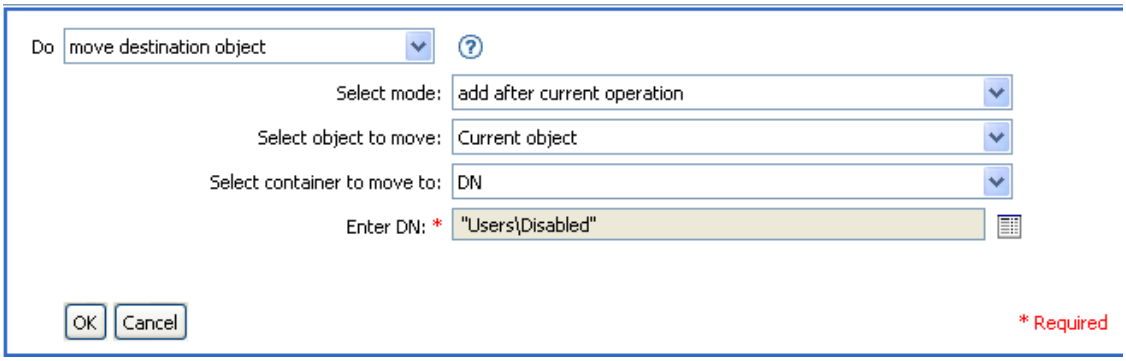
要移动到的树枝

选择目标树枝。此树枝由 DN 或关联指定。

示例

本示例包含一个规则，当 Description （说明）特性指示用户已终止时，该规则将禁用用户帐户并将其移动到禁用的树枝中。此策略名为 Disable User Account and Move When Terminated （终止时禁用用户帐户并移动），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





此策略将检查该事件是否为用户对象上的修改事件，以及 Description （说明）特性中是否包含值 terminated。如果出现这种情况，则该策略将 " 禁止登录 " 特性设置为 "true"，并将该对象移动到用户禁用的树枝中。

2.6.22 移动源对象

移动源数据存储区中的对象。

字段

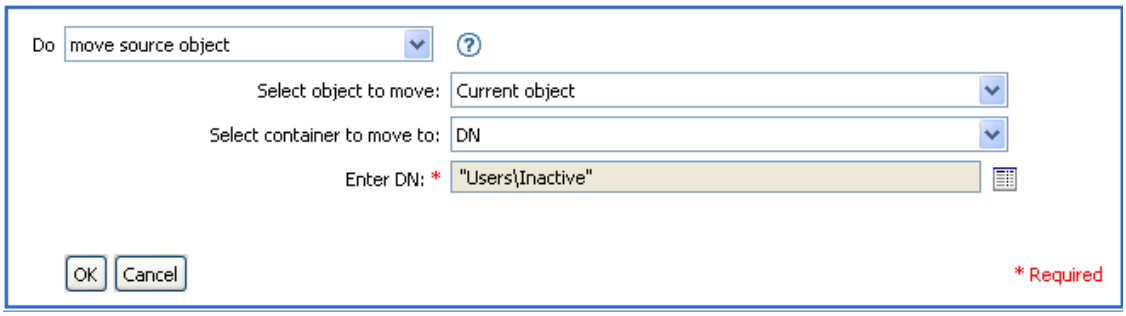
要移动的对象

选择要移动的对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

要移动到的树枝

选择目标树枝。此树枝由 DN 或关联指定。

示例



2.6.23 重新设置操作特性的格式

在当前操作中，使用某种模式重设特性的所有值的格式。

字段

名称

指定该特性的名称。

值类型

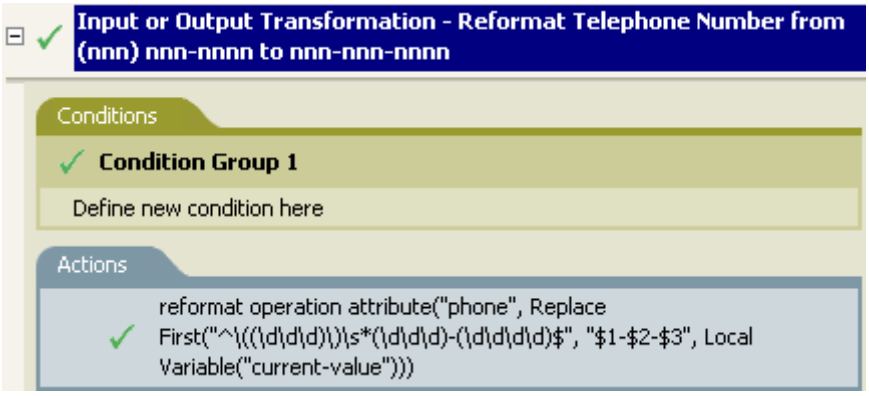
指定新特性值的语法。

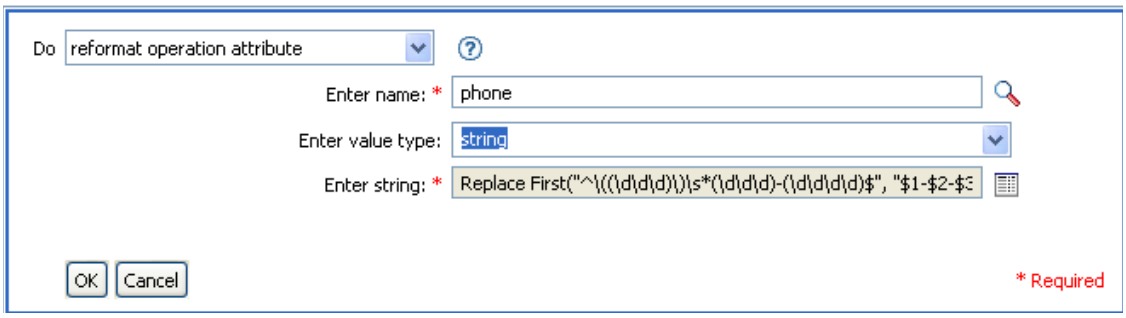
值

指定一个值，作为特性值新格式的模式。如果需要使用原始值构造新值，则必须通过引用局部变量 current-value 获取该原始值。

示例

本示例重设了电话号码的格式，由 (nnn)-nnn-nnnn 更改为 nnn-nnn-nnnn。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “输入或输出转换 - 将电话号码格式重新从 (nnn) nnn-nnnn 设置为 nnn-nnn-nnnn”在第 81 页 。





" 重新设置操作特性的格式 " 操作更改了电话号码的格式。此规则使用自变量构建器和正则表达式更改信息的显示方式。

2.6.24 去除关联

将去除关联命令发送至 Identity Vault。

字段

方式

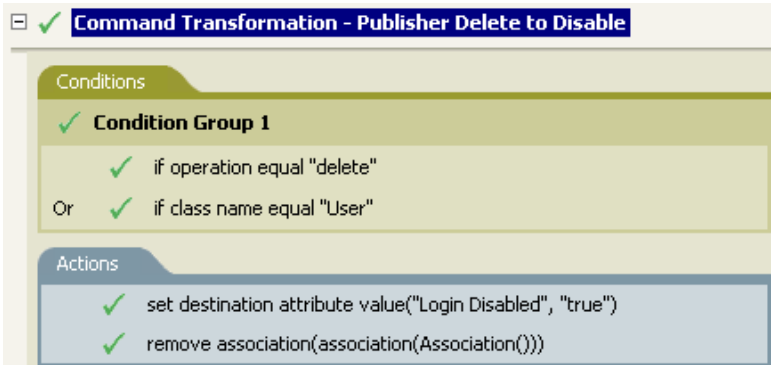
选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

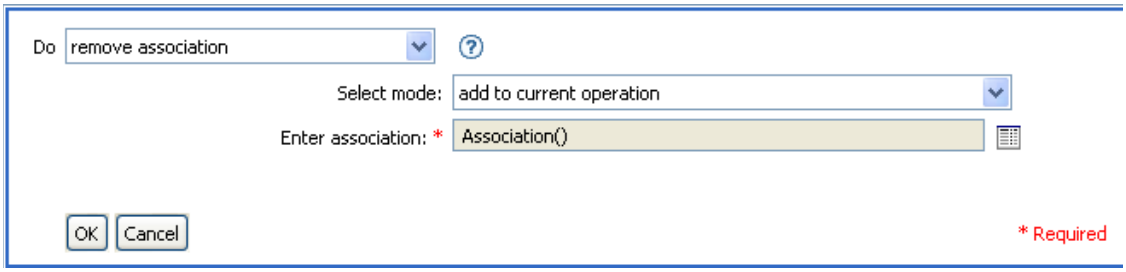
关联

指定要去除的关联值。

示例

本示例执行了删除操作并禁用用户对象。它将转换事件。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “命令转换 - 发布者删除 - 禁用”在第 71 页 。





对用户对象执行删除操作时， Login Disabled （禁止登录）特性值将被设置为 true，并将关联从该对象中去除。由于已连接的应用程序中的关联对象已不存在，因此将去除该关联。

2.6.25 去除目标特性值

去除目标数据存储区中对象的特性值。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

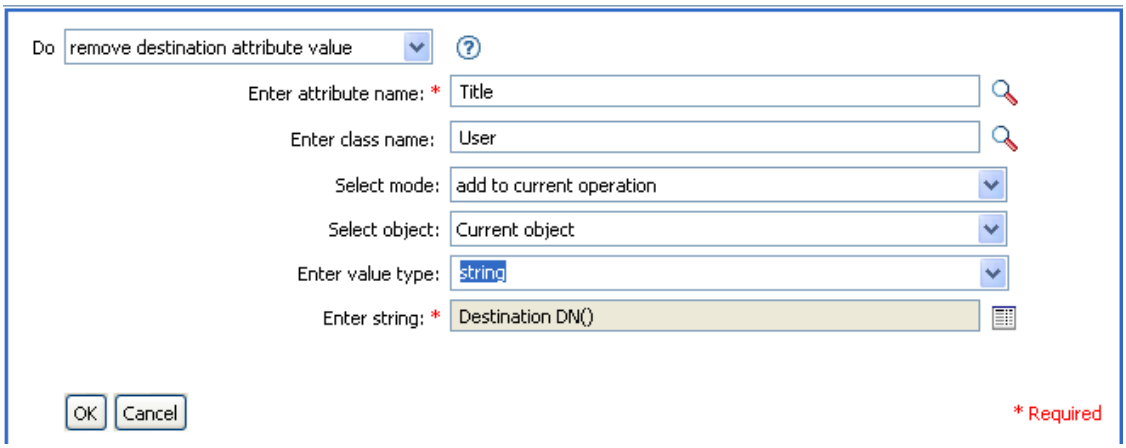
值类型

指定要去除的特性值的语法。

值

指定新特性的值。

示例



2.6.26 去除源特性值

在源数据存储区中，从对象的命名特性中去除指定的值。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

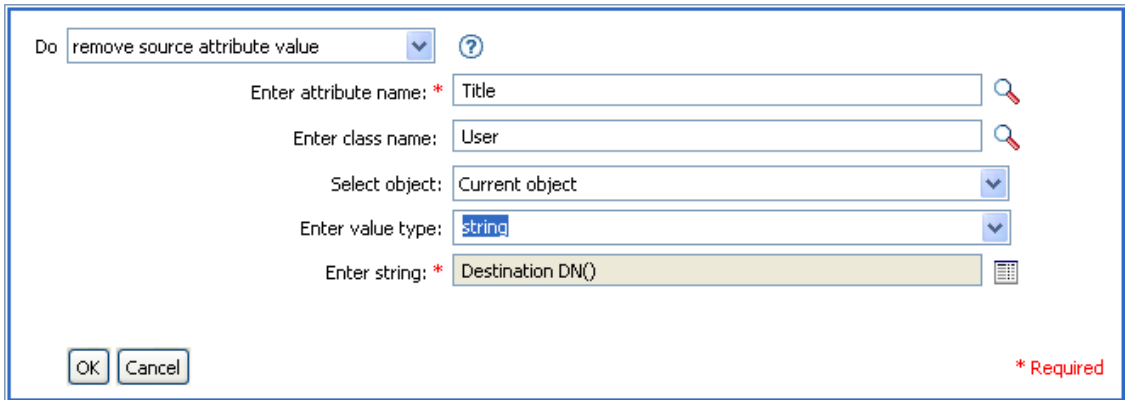
值类型

指定要去除的特性值的语法

值

指定要去除的特性值。

示例



2.6.27 重命名目标对象

重命名目标数据存储区中的对象

字段

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

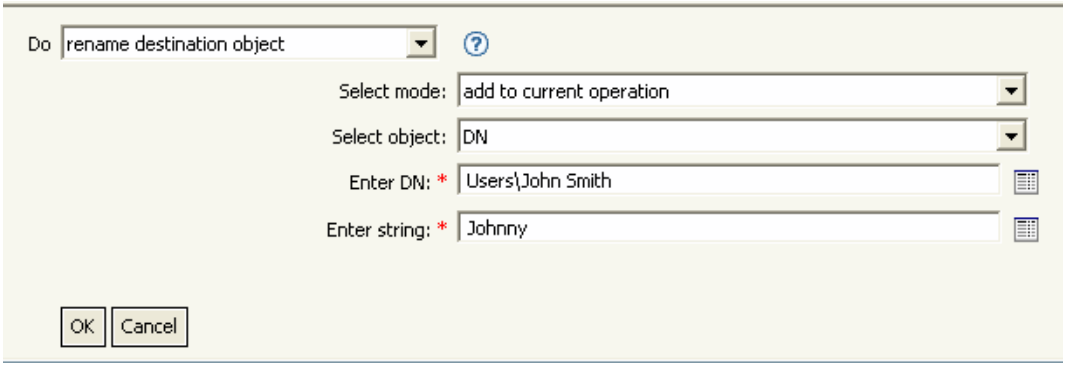
对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

字符串

指定对象的新名称。

示例



2.6.28 重命名操作特性

重命名当前操作中所有具体出现的某一特性。

字段

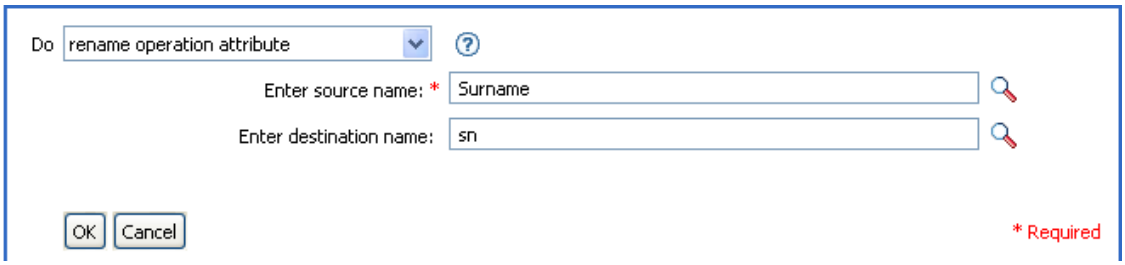
源名称

指定原始特性名称。

目标名称

指定新的特性名称。

示例



2.6.29 重命名源对象

重命名源数据存储区中的某一对象。

字段

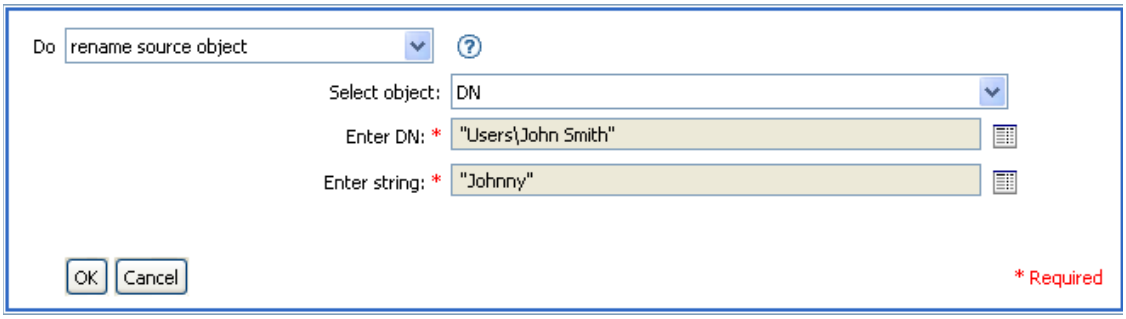
对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

字符串

指定对象的新名称。

示例



2.6.30 发送电子邮件

发送电子邮件通知。

字段

ID

（可选）在发送讯息的 SMTP 系统中指定用户 ID。

服务器

指定 SMTP 服务器的名称。

口令

（可选）指定 SMTP 服务器帐户的口令。

重要：口令特性的值以明文形式储存。

类型

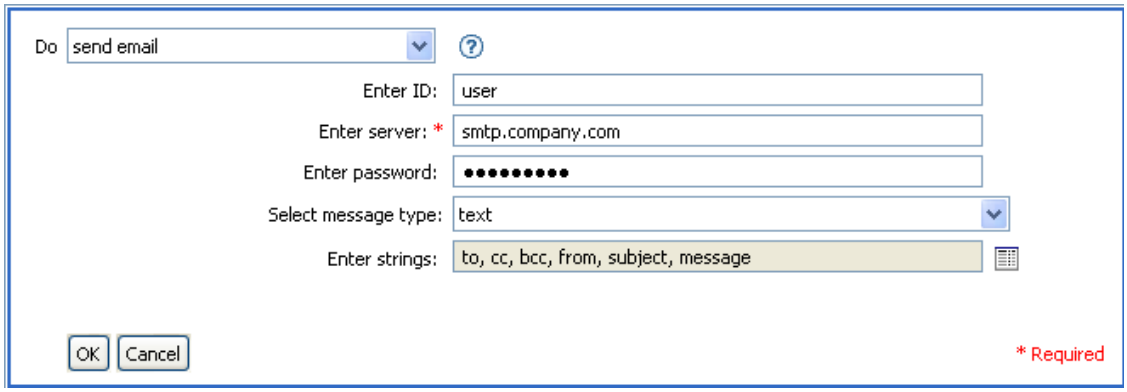
选择电子邮件讯息的类型。

字符串

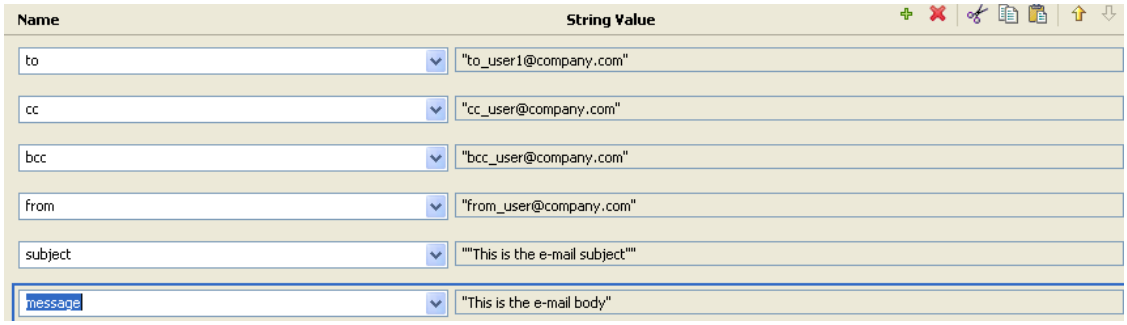
指定包含各种电子邮件地址、主题和内容的值。下表列出了有效的命名字符串自变量：

字符串名称	说明
to	将地址添加到电子邮件收件人列表中；允许添加多个地址。
cc	将地址添加到 " 抄送 " 电子邮件收件人列表中；允许添加多个地址。
bcc	将地址添加到 " 暗送 " 电子邮件收件人列表中；允许添加多个地址。
from	指定电子邮件的发送地址。
reply-to	指定电子邮件讯息的回复地址。
subject	指定电子邮件主题。
message	指定电子邮件讯息的内容。
encoding	指定电子邮件讯息使用的字符编码。

示例



在以下示例中，命名字符串构建器用于提供字符串自变量：



2.6.31 通过模板发送电子邮件

使用模板生成电子邮件通知。

字段

通知 **DN**

指定 SMTP 通知配置对象的斜线格式 DN。

模板 **DN**

指定电子邮件模板对象的斜线格式 DN。

口令

（可选）指定 SMTP 服务器帐户的口令。

重要：口令特性的值以明文形式储存。

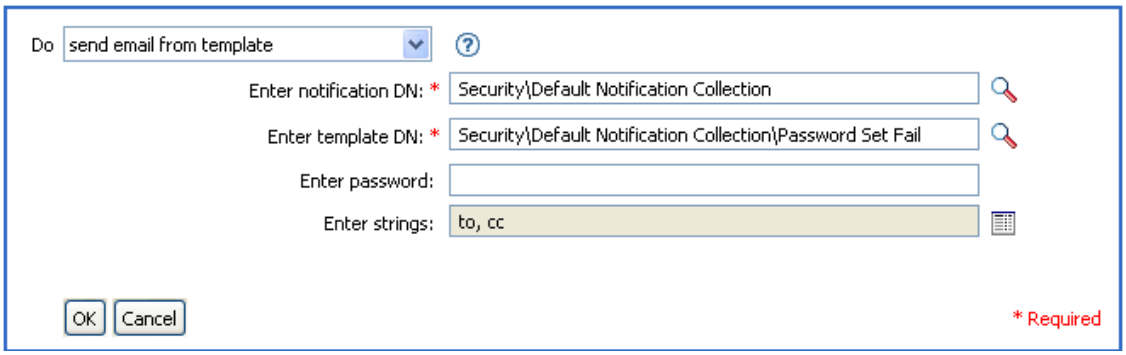
字符串

指定电子邮件讯息的附加字段。下表包含了保留的字段名称，用于指定各种电子邮件地址：

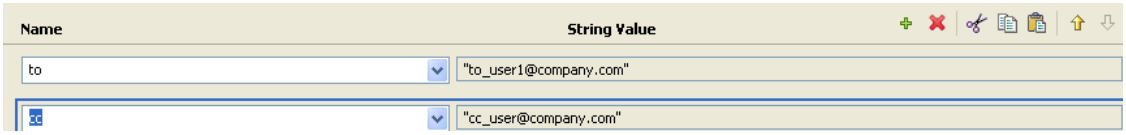
字符串名称	说明
to	将地址添加到电子邮件收件人列表中；允许添加多个地址。
cc	将地址添加到 " 抄送 " 电子邮件收件人列表中；允许添加多个地址。
bcc	将地址添加到 " 暗送 " 电子邮件收件人列表中；允许添加多个地址。
reply-to	指定电子邮件讯息的回复地址。
encoding	指定电子邮件讯息使用的字符编码。

每个模板可能还定义了其它字段，这些字段可在电子邮件讯息的主题和正文中进行替换。

示例



在以下示例中，命名字符串构建器用于提供字符串自变量：



2.6.32 设置默认特性值

如果某特性无值，则在当前操作中为该特性添加默认值 （也可以将默认值添加至源数据存储区中的当前对象）。当前操作是 " 添加 " 时它才有效。

字段

特性名称

指定默认特性的名称。

写回

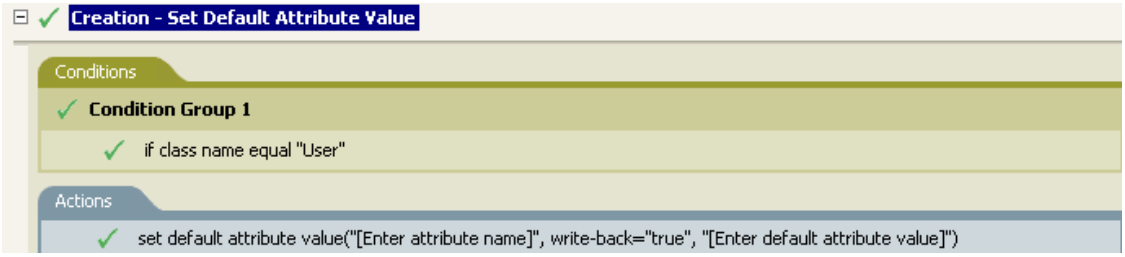
选择是否要将默认值也写回源数据存储区。

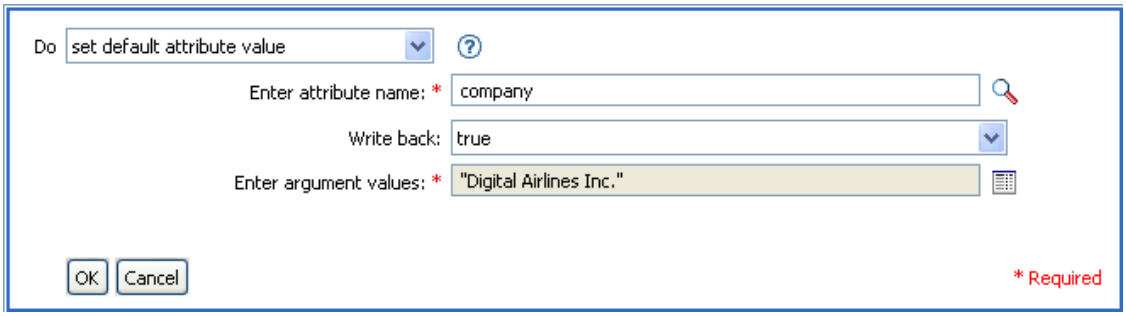
值

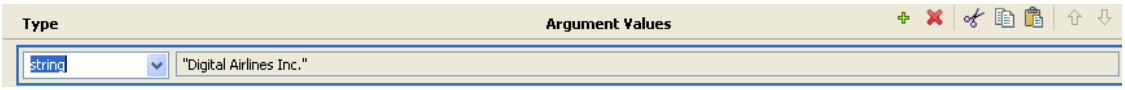
指定特性的默认值。

示例

本示例设置 company （公司）特性的默认值。也可以设置任意特性的值。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “创建 - 设置默认特性值”在第 75 页 。







若要构建该值，请起动自变量值列表构建器。有关此构建器的更多信息，请参见 “自变量值列表构建器”在第 61 页 。可以根据需要设置此值。在这种情况下，将使用自变量构建器并将文本设置为公司名称。

2.6.33 设置目标特性值

向目标数据存储区中的某对象的特性添加一个值，并去除该特性的所有其它值。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定目标数据存储区中目标对象的类名称。若保留空白则使用当前对象的类名称。

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

值类型

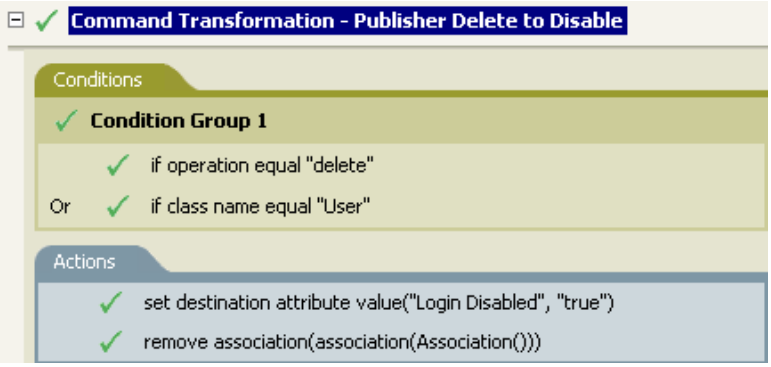
选择要设置的特性值的语法。

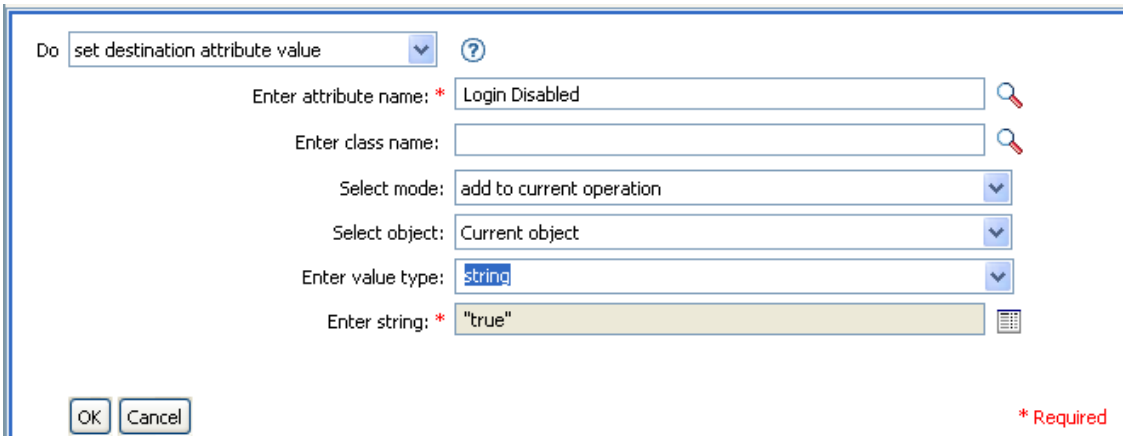
值

指定要设置的特性值。

示例

本示例执行了删除操作并禁用用户对象。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “命令转换 - 发布者删除 - 禁用”在第 71 页。





此规则将 Login Disabled （禁止登录）特性的值设置为 true。它使用自变量构建器为此特性的值添加文本 true。有关此构建器的更多信息，请参见 “自变量构建器”在第 58 页。

2.6.34 设置目标口令

设置目标数据存储区中当前对象的口令。

字段

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

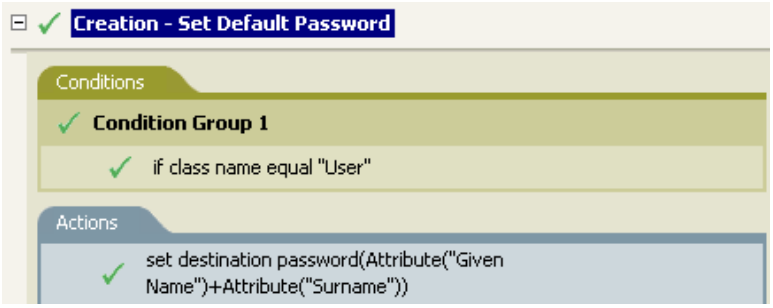
选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

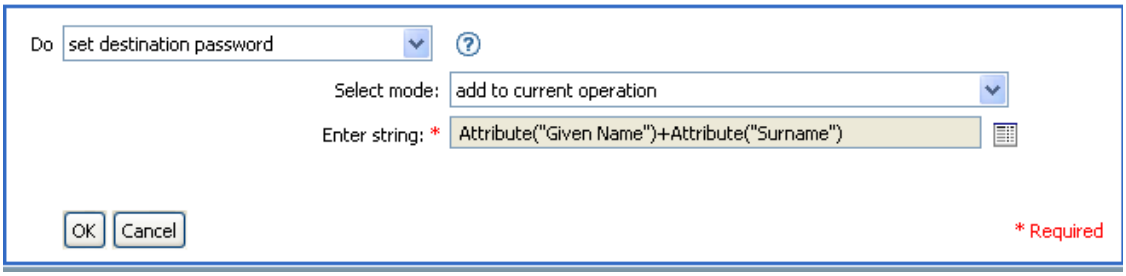
字符串

指定要设置的口令。

示例

此示例为所创建的用户对象设置默认口令。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “创建 - 设置默认口令”在第 77 页 。





创建用户对象时，此口令设置为 Given Name （名）特性加 Surname （姓氏）特性。

2.6.35 设置局部变量

设置局部变量。

字段

变量名

指定该局部变量的名称。

变量类型

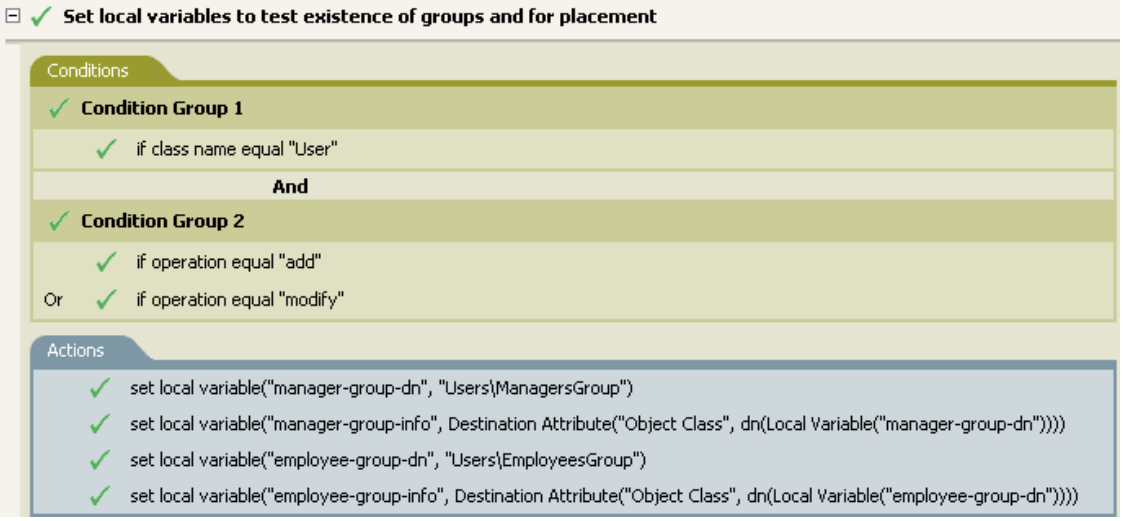
选择局部变量的类型。可以为字符串、 XPath 1.0 节点集或 Java 对象。

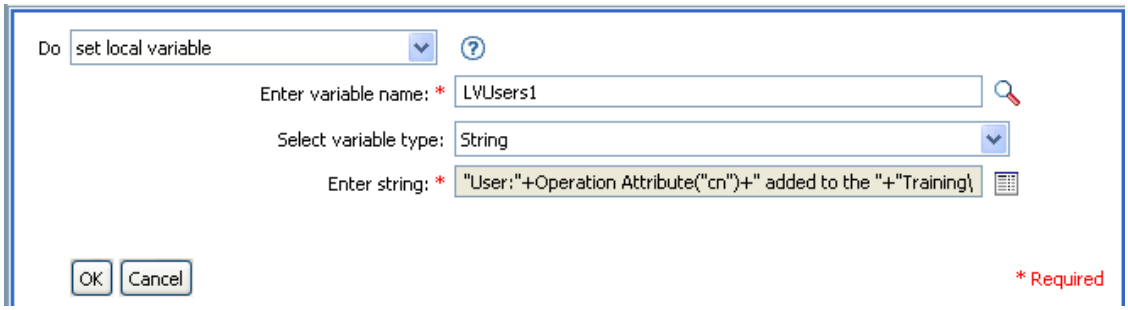
值

指定局部变量的值。

示例

下面的示例根据职务，向相应的 Employee 或 Manager 组中添加用户对象。在需要时，它也能创建组并设置与该组相对应的安全性。此策略名为 Govern Groups for User Based （根据职务管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





将局部变量设置为用户对象的目标特性 Object Class 中的值加上局部变量 manager-group-info。自变量构建器用于构造局部变量。有关详细信息，请参见 “自变量构建器”在第 58 页 。

2.6.36 设置操作关联

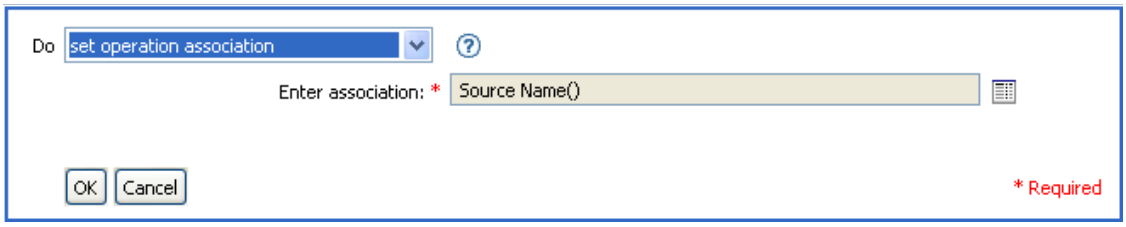
设置当前操作的关联值。

字段

关联

提供新的关联值。

示例



2.6.37 设置操作的类名称

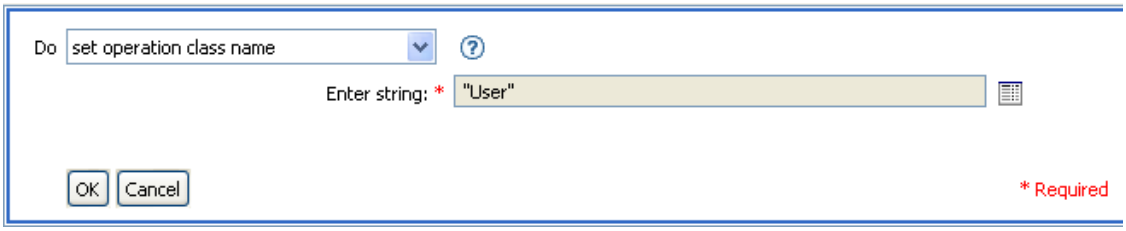
设置当前操作的对象类名称。

字段

字符串

提供新的类名称。

示例



2.6.38 设置操作目标 DN

设置当前操作的目标 DN。

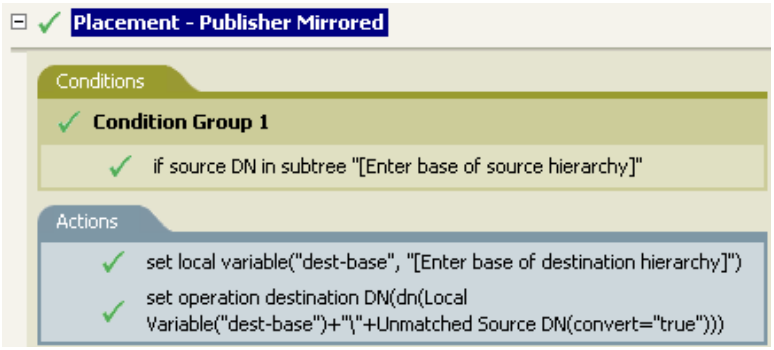
字段

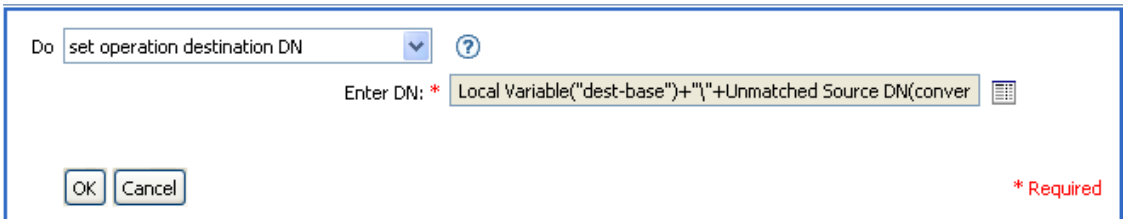
DN

指定新的目标 DN。

示例

本示例使用从已连接系统镜像得到的结构将对象放置到 Identity Vault 中。您需要在源数据存储区和目标数据存储区中定义镜像的起始点。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “创建 - 设置默认特性值”在第 75 页。





该规则将操作目标 DN 设置为目标基位置加上源 DN 组成的局部变量。

2.6.39 设置操作属性

设置操作属性。操作属性是储存在操作中的命名值。它通常用于提供其它环境，处理操作结果的策略可能需要这些环境。

字段

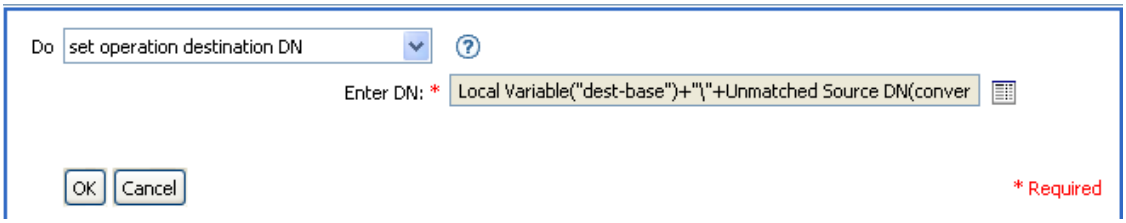
属性名

指定操作属性的名称。

字符串

指定操作属性的名称。

示例



2.6.40 设置操作源 DN

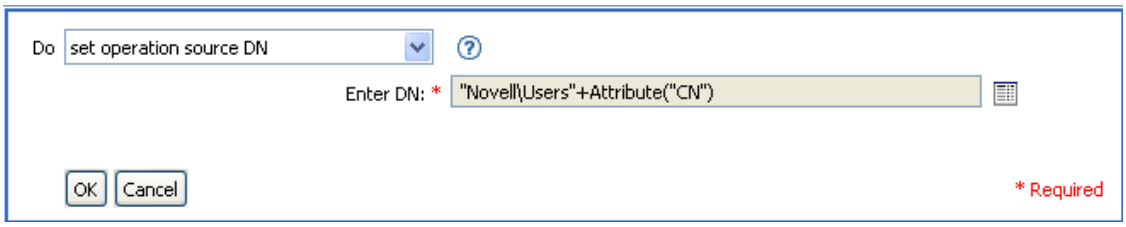
设置当前操作的源 DN。

字段

DN

指定新的源 DN。

示例



2.6.41 设置操作模板 DN

将当前操作的模板 DN 设置为指定值。当前操作是 " 添加 " 时此操作才有效。

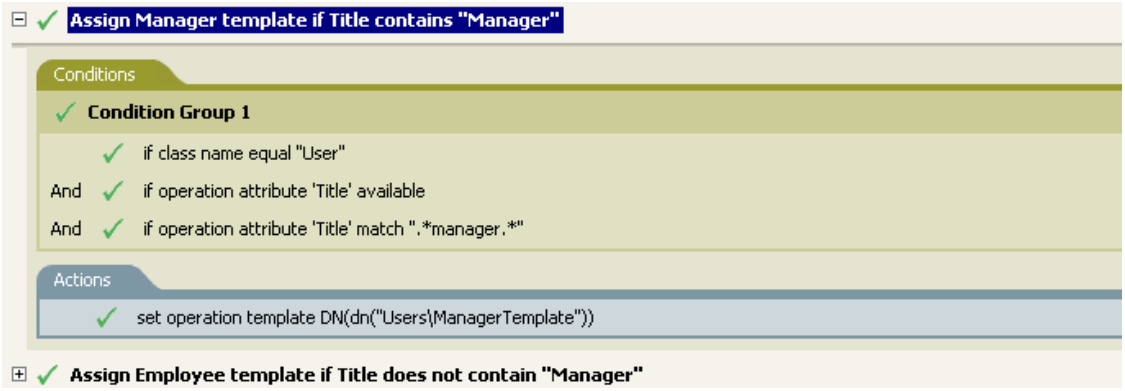
字段

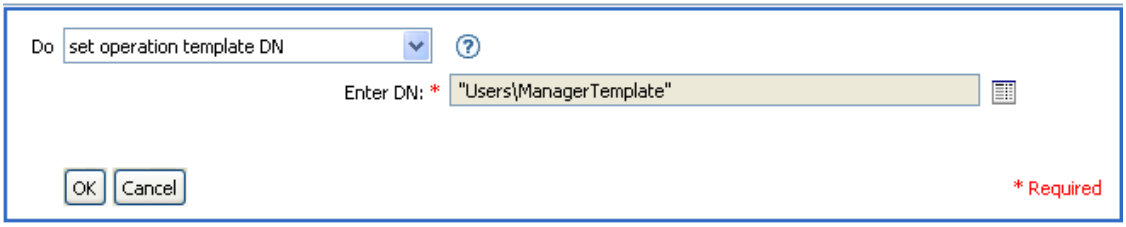
DN

指定模板 DN。

示例

如果 Title （职务）特性中包含文字 Manager，本示例将应用 Manager （经理）模板。策略名称为 Policy:Assign Template to User Based on Title （根据职务为用户指派模板），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页。





模板 Manager Template （经理模板）应用于具有特性 Title （职务），并且在职务中的某一位置包含了文字 manager 的任何用户对象。此策略使用正则表达式查找所有可能的匹配项。

2.6.42 设置源特性值

向源数据存储区中的某对象的特性添加一个值，并去除该特性的所有其它值。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定源数据存储区中的目标对象的类名称。若保留空白则使用当前对象的类名称。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

值类型

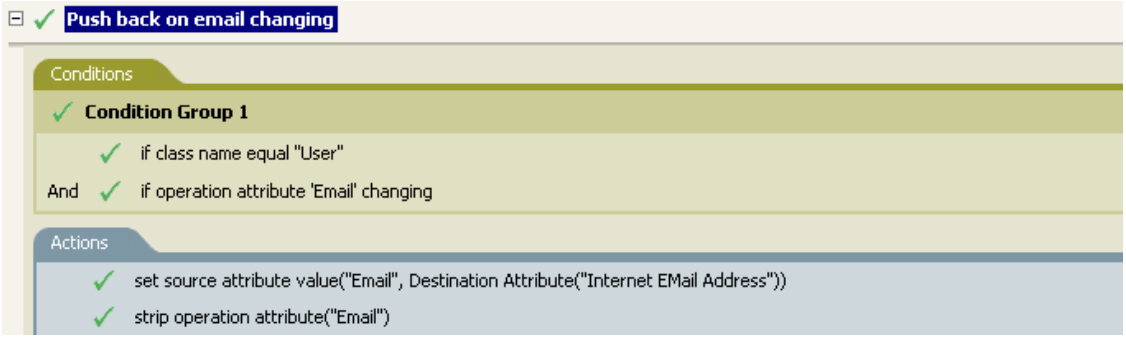
选择特性值的语法。

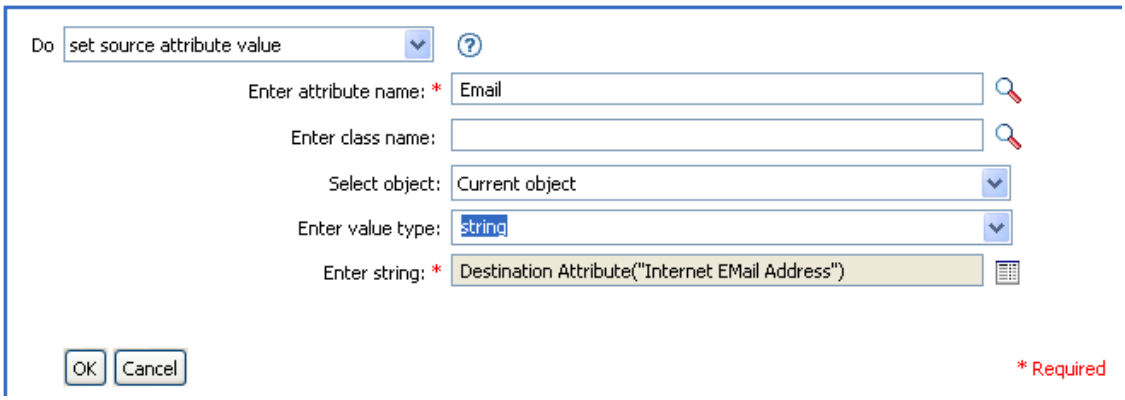
值

指定要设置的特性值。

示例

本示例在检测到电子邮件地址发生更改时，将其设置回原来的地址。策略名称为 Policy:Reset Value of the E-mail Attribute （重设置电子邮件特性的值），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





此操作将采用目标特性 Internet EMail Address （因特网电子邮件地址）的值，并将源特性 Email （电子邮件）设置为此值。

2.6.43 设置源口令

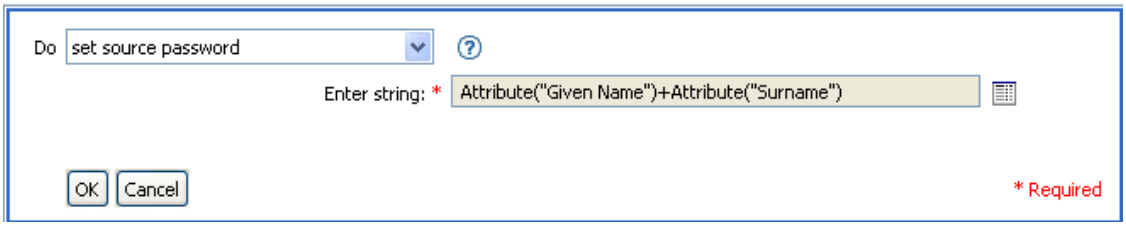
设置源数据存储区中的当前对象的口令。

字段

字符串

指定要设置的口令。

示例



2.6.44 设置 SSO 身份凭证

创建用户对象或修改口令后，设置 SSO 身份凭证。此操作是身份凭证供应策略的一部分。有关详细信息，请参见第 4 章 “Novell 身份凭证供应策略”在第 305 页 。

字段

身份凭证储存对象 **DN**

指定储存库对象的 DN。

目标用户 **DN**

指定目标用户的 DN。

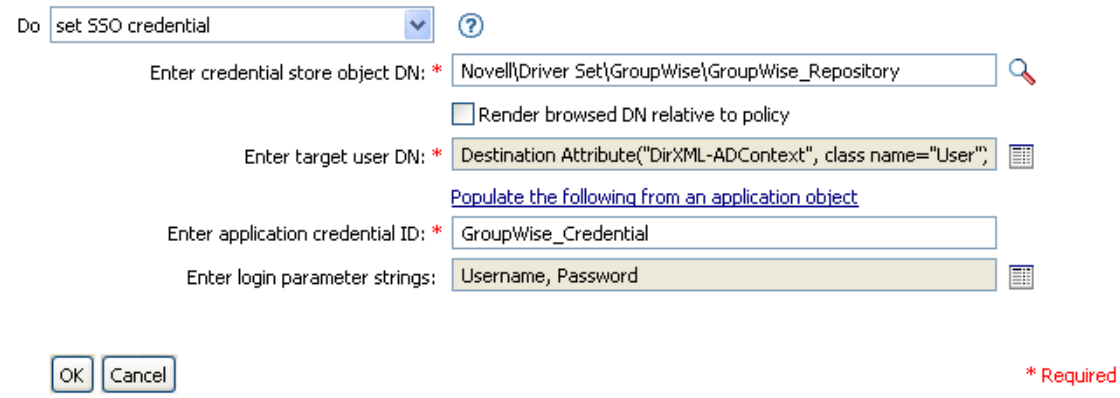
应用程序身份凭证 **ID**

指定储存在应用程序对象中的应用程序身份凭证。

登录参数字符串

指定应用程序的所有登录参数。登录参数是储存在应用程序对象中的鉴定密钥。

示例



2.6.45 设置 SSO 通行口令

设置供应用户对象时的 Novell SecureLogin® 通行口令和答案。此操作是身份凭证供应策略的一部分。有关详细信息，请参见第 4 章 “Novell 身份凭证供应策略”在第 305 页。

字段

身份凭证储存对象 **DN**

指定储存库对象的 DN。

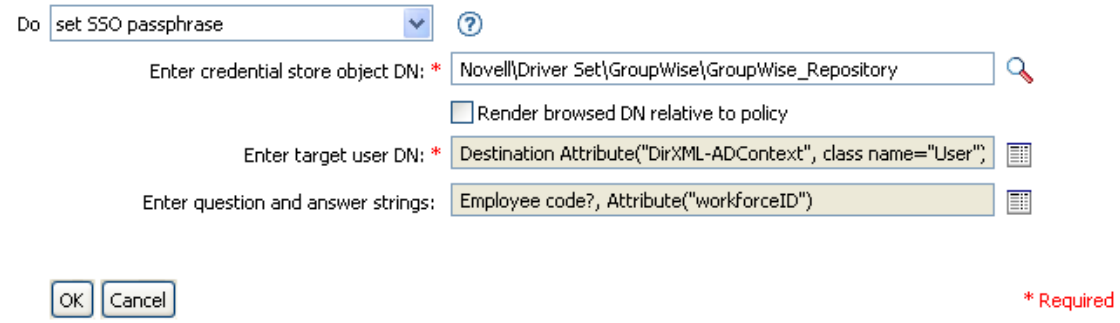
目标用户 **DN**

指定目标用户的 DN。

问题和答案字符串

指定 SecureLogin 通行口令的问题和答案。

示例



SecureLogin 通行口令的问题和答案以字符串的形式储存在策略中。单击 *Edit the strings*（编辑字符串）图标 以起动字符串构建器。指定通行口令的问题和答案。

2.6.46 设置 XML 特性

设置通过 XPath 表达式选择的一组要素的 XML 特性。

字段

名称

指定 XML 特性的名称。如果先前已在此策略中定义了名称空间前缀，则特性名称中可以包含该前缀。

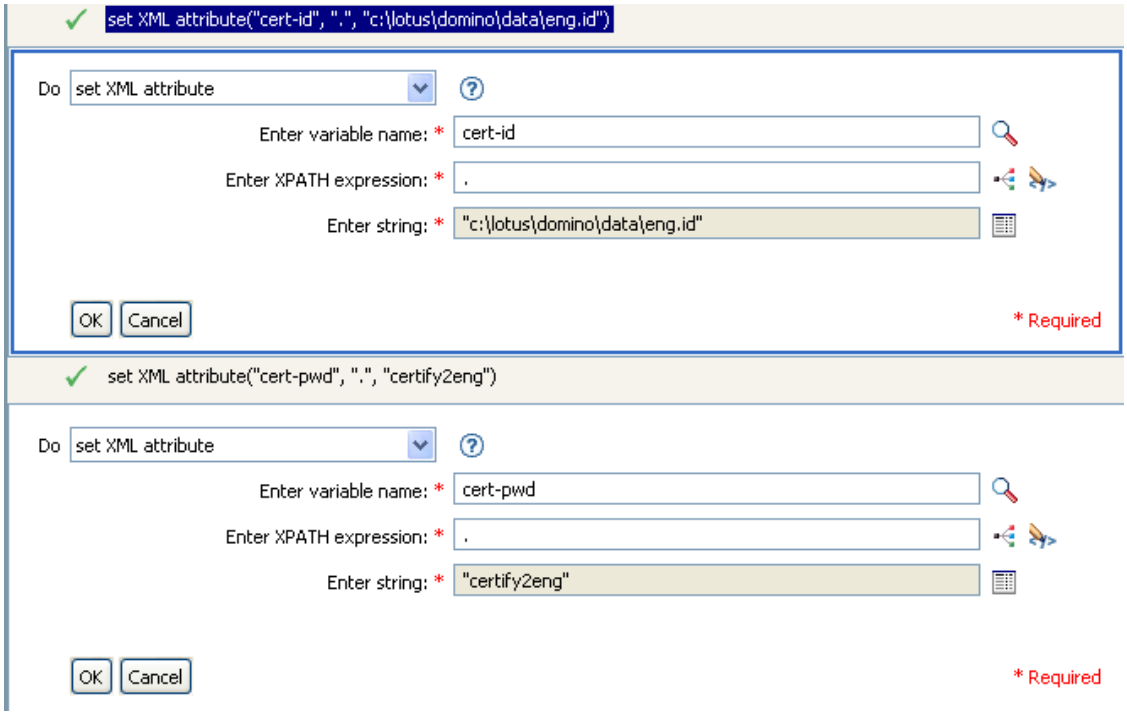
XPath 表达式

XPath 1.0 表达式，该表达式将返回一个节点集，其中包含设置此 XML 特性的要素。

字符串

指定 XML 特性的值。

示例



2.6.47 状态

生成状态通知。

字段

级别

指定通知的状态级别。

讯息

使用自变量构建器提供状态讯息。

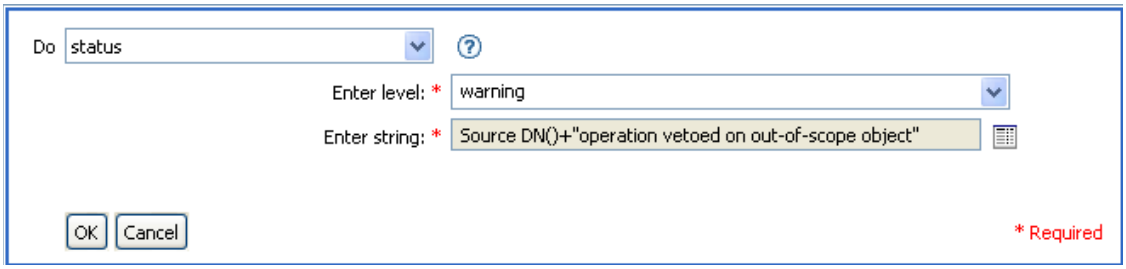
注释

如果级别为重试，则此策略会立即停止处理输入文档并安排重试当前正在处理的事件。

如果级别为致命，则此策略会立即停止处理输入文档并关闭此驱动程序。

如果当前操作具有事件 ID，则状态通知将使用该事件 ID，否则将不报告事件 ID。

示例



2.6.48 去除操作特性

去除当前操作中所有具体出现的某一特性。

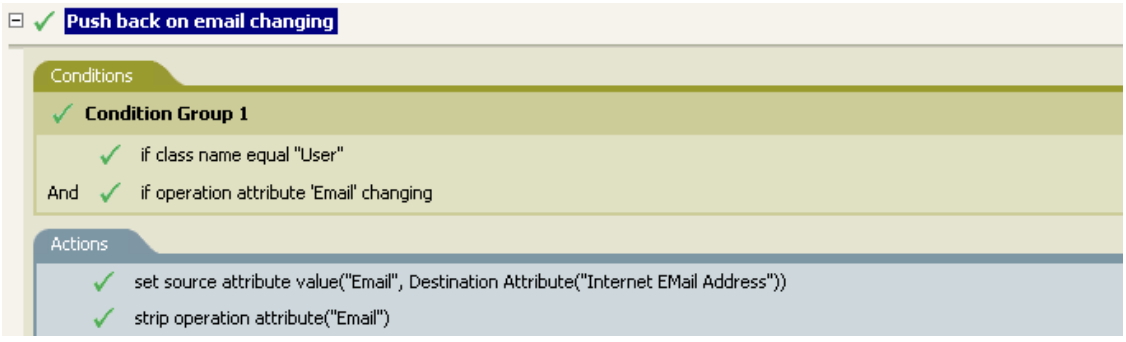
字段

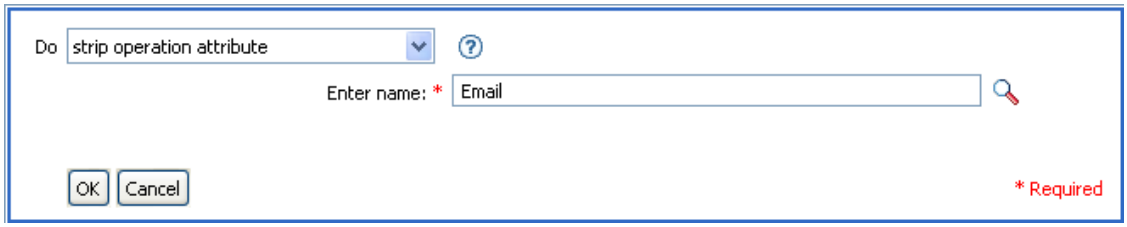
名称

指定要去除的特性的名称。

示例

本示例在检测到电子邮件地址发生更改时，将其设置回原来的地址。策略名称为 Policy:Reset Value of the E-mail Attribute （重设置电子邮件特性的值），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





此操作将去除 Email （电子邮件）特性。所保留的值是目标 Email （电子邮件）特性中的原有值。

2.6.49 去除 XPath

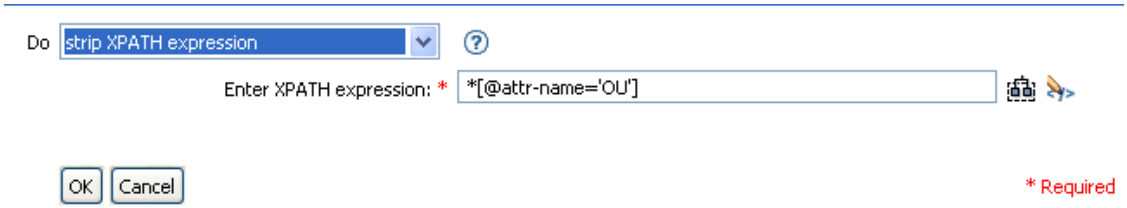
去除由 XPath 表达式选择的节点。

字段

XPath 表达式

指定返回节点集的 XPath 1.0 表达式，此节点集包含要去除的节点。

示例



2.6.50 跟踪讯息

将讯息发送至 DSTRACE。

字段

级别

指定讯息的跟踪级别。默认级别为 0。只有在指定的跟踪级别小于或等于驱动程序中配置的跟踪级别时，此讯息才会出现。

有关如何在驱动程序中设置跟踪级别的信息，请参见 《*Novell Identity Manager 3.0* 管理指南》中的 “Viewing Identity Manager Processes （查看 Identity Manager 进程） ”。

颜色

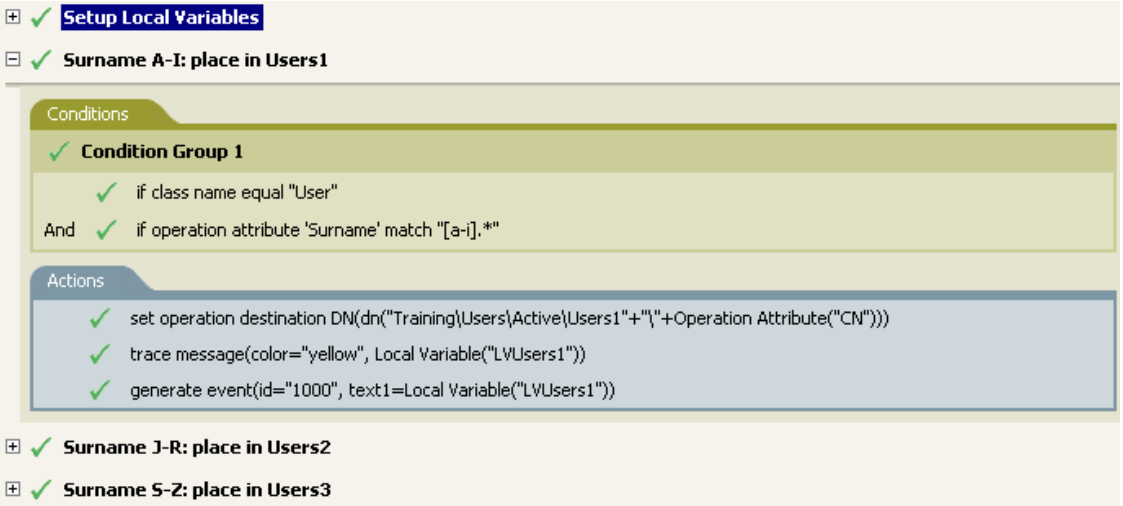
选择跟踪讯息的颜色。

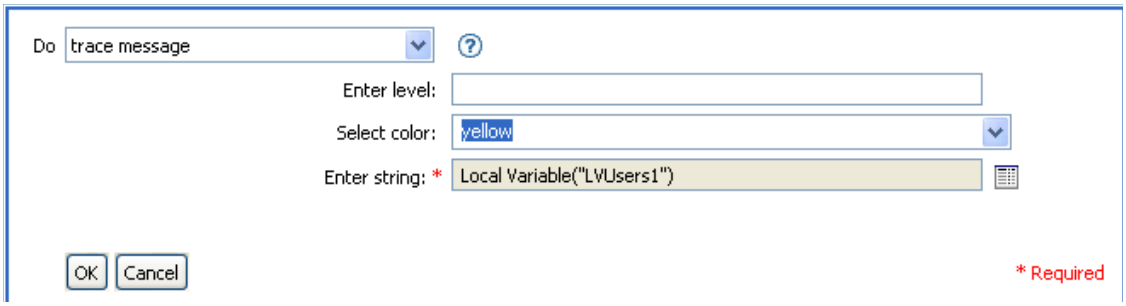
字符串

指定跟踪讯息的值。

示例

本示例包含四条规则，它们根据 Surname （姓氏）特性的首字符实施对用户对象的布局策略。并生成一条跟踪讯息和一个自定义 Novell Audit 事件。 " 跟踪讯息 " 操作用于将跟踪讯息发送至 DSTRACE。策略名称为 Policy to Place by Surname （按姓氏布局策略），可从 Novell 支持万维网站点下载。详情请见 “可下载的 Identity Manager 策略”在第 31 页 。





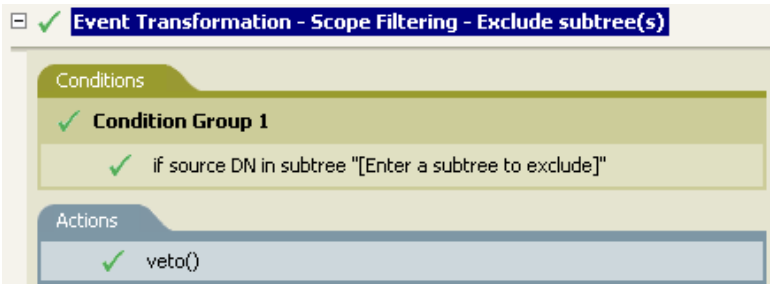
此操作将跟踪讯息发送至 DSTRACE。局部变量的内容为 LVUsers1，在 DSTRACE 中显示为黄色。

2.6.51 禁止

禁止当前操作。

示例

本示例将排除所有来自指定子树的事件。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “事件转换 - 范围过滤 - 排除子树”在第 79 页 。



此操作将禁止所有来自指定子树的事件。

2.6.52 如果操作特性不可用则禁止

根据当前操作中特性的可用性，有条件地取消当前操作并结束对当前策略的处理。

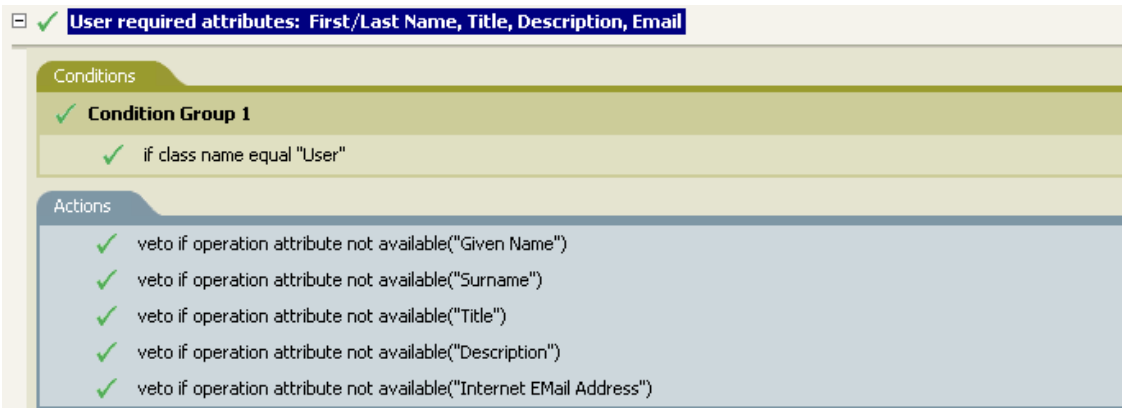
字段

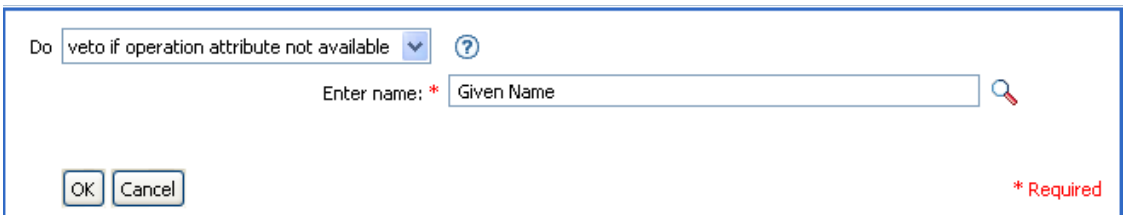
名称

指定该特性的名称。

示例

除非以下特性：Given Name （名）、 Surname （姓氏）、 Title （职务）、 Description （说明）和 Internet EMail Address （因特网电子邮件地址）可用，否则本示例不会允许创建所有用户对象。策略名称为 Policy to Enforce the Presences of Attributes （强制验证特性存在策略），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





如果以下特性：Given Name （名）、 Surname （姓氏）、 Title （职务）、 Description （说明）和 Internet EMail Address （因特网电子邮件地址）不可用，此操作将禁止创建对象。

2.7 名词标记

本节包含可使用 " 自变量构建器 " 界面访问的所有名词标记的详细参考信息。

2.7.1 已添加的权利

扩展到当前操作中授予的权利的值。

字段

名称

权利的名称。

示例

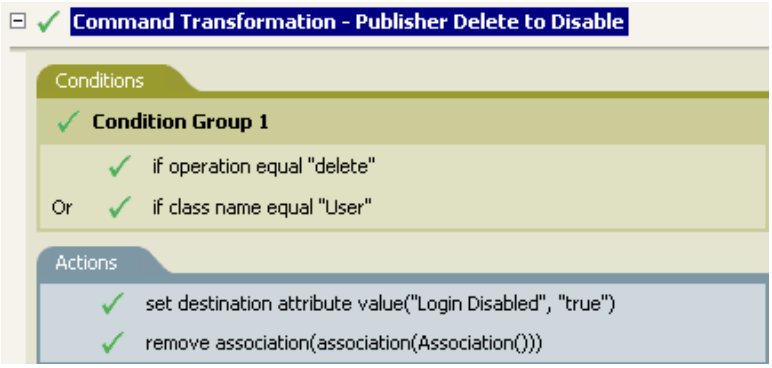
2.7.2 关联

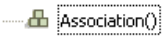
从当前操作扩展到关联值。

示例

本示例运用了 Identity Manager 3.0 附带的预定义规则。有关预定义规则的更多信息，请参见“命令转换 - 发布者删除 - 禁用”在第 71 页。

" 去除关联 " 操作使用 " 关联 " 标记检索当前操作的值。此规则将去除用户对象的关联，因此所有遇到的新事件都不会影响用户对象。





2.7.3 特性

扩展到当前操作和源数据存储区中的当前对象的某一特性的值。逻辑上，可以将它看作 " 操作特性 " 标记和 " 源特性 " 标记的组合。但不包括修改操作去除的值。

字段

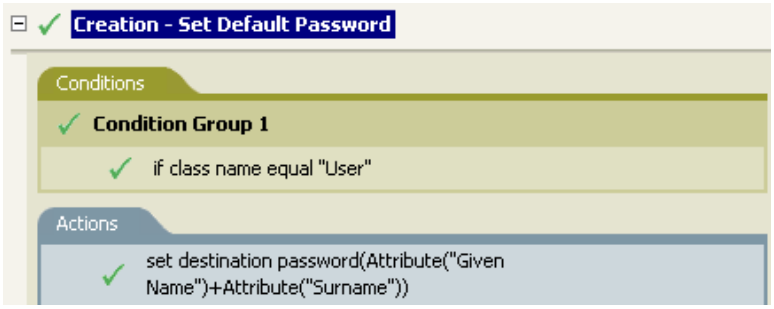
名称

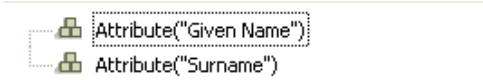
指定该特性的名称。

示例

本示例运用了 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见“创建 - 设置默认口令”在第 77 页。

" 设置目标口令 " 操作使用 " 特性 " 标记创建口令。该口令由 Given Name （名）特性和 Surname （姓）特性构成。在 " 自变量构建器编辑器 " 中可以浏览并选择想要使用的特性。

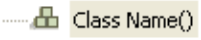




2.7.4 类名称

扩展到当前操作的对象类名称。

示例



2.7.5 目标特性

扩展到目标数据存储区中的当前对象、 DN 或关联的指定特性值。

字段

类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

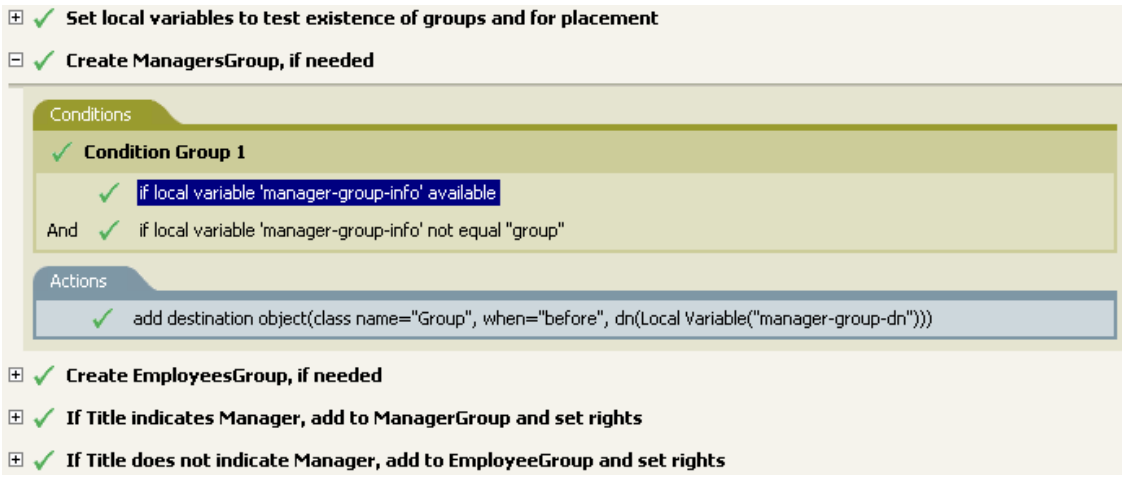
名称

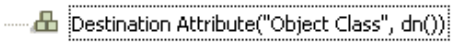
特性的名称。

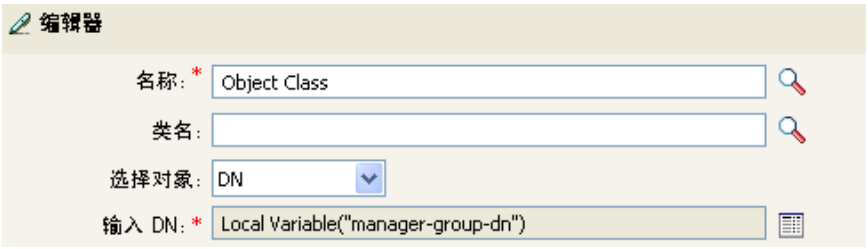
示例

本示例运用了 Govern Groups for User Based on Title （根据职务管理用户组）策略，可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。

此策略使用自变量构建器创建 Destination Attribute （目标特性）。 " 设置局部变量 " 操作包含 " 目标特性 " 标记。







通过编辑器构建 " 目标特性 "。在本示例中，将特性设置为 Object Class。 DN 用于选择目标对象。 DN 的值是局部变量 manager-group-dn。

2.7.6 目标 DN

扩展到当前操作的目标 DN。

字段

转换

选择是否将 DN 转换为源数据存储区使用的格式。

开始

指定起始 RDN 索引：

- 索引 0 是最靠近根的 RDN
- 正索引是从最靠近根的 RDN 计算的偏移量
- 索引 -1 是最靠近叶的段
- 负索引是从最靠近叶的 RDN 到最靠近根的 RDN 的偏移量

长度

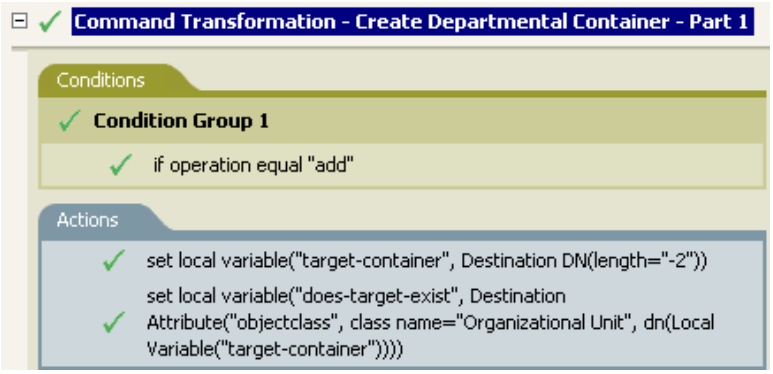
指定要包括的 RDN 数。负数可解释为 （段的总数 + 长度） + 1。例如，对于具有 5 个段的 DN，其长度 -1 = (5 + (-1)) + 1 = 5、 -2 = (5 + (-2)) + 1 = 4，依此类推。

注释

如果起始索引和长度设置为默认值 {0,-1}，则使用整个 DN ；否则仅使用由起始索引和长度指定的 DN 部分。

示例

本示例使用 " 目标 DN" 标记设置局部变量 target-container 的值。如果部门树枝不存在，则该策略将为用户对象创建该树枝。此策略运用了 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “命令转换 - 创建部门树枝 - 第 1 部分和第 2 部分”在第 69 页 。



Destination DN(length="-2")

2.7.7 目标名称

扩展到从当前操作指定的目标 DN 的相对判别名 (RDN)。

示例

Destination Name()

2.7.8 权利

扩展到当前对象中授予的权利的值。

字段

名称

指定权利的名称。

示例

Entitlement("manager")

2.7.9 全局配置值

扩展到全局配置值的值。

字段

名称

全局配置值的名称。

示例

2.7.10 局部变量

扩展到局部变量的值。

字段

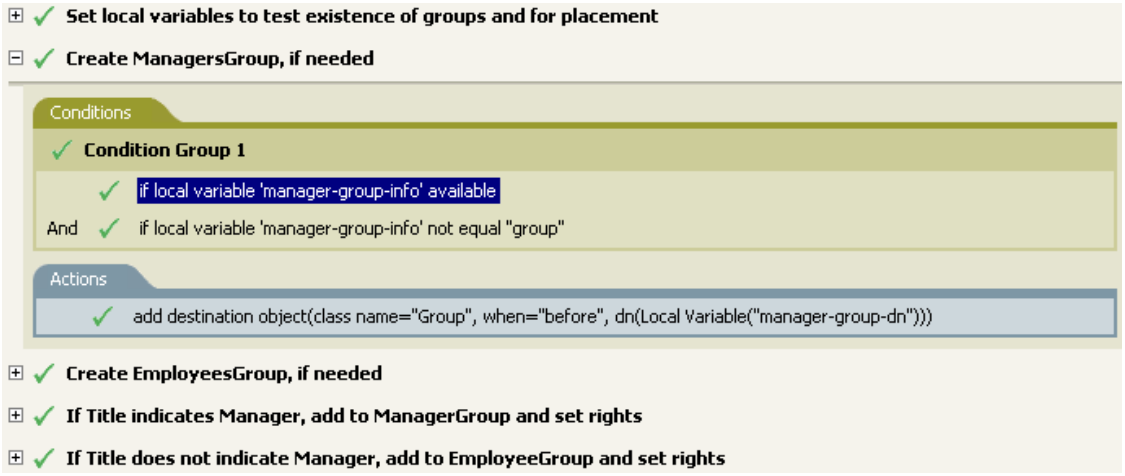
名称

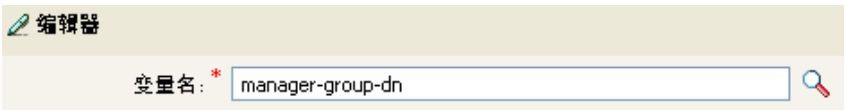
指定该局部变量的名称。

示例

本示例运用了 Govern Groups for User Based on Title （根据职务管理用户组）策略，可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。

" 添加目标对象 " 操作使用 " 局部变量 " 标记。





只有先前已在策略中使用过 " 设置局部变量 " 操作时，才可以使用局部变量。它将设置储存在局部变量中的值。在编辑器中，单击 " 浏览 " 图标，即可列出已定义的所有局部变量。选择正确的局部变量。

局部变量的值是 group-manager-dn。在上一个规则中， " 设置局部变量 " 操作将经理组 Users\ManagersGroup 的 DN 定义为 group-manager-dn。

2.7.11 命名口令

扩展到驱动程序的命名口令。

字段

名称

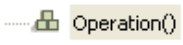
指定口令的名称。

示例

2.7.12 操作

扩展到当前操作的名称。

示例



2.7.13 操作特性

扩展到当前操作的特性的值。但不包括修改操作去除的值。

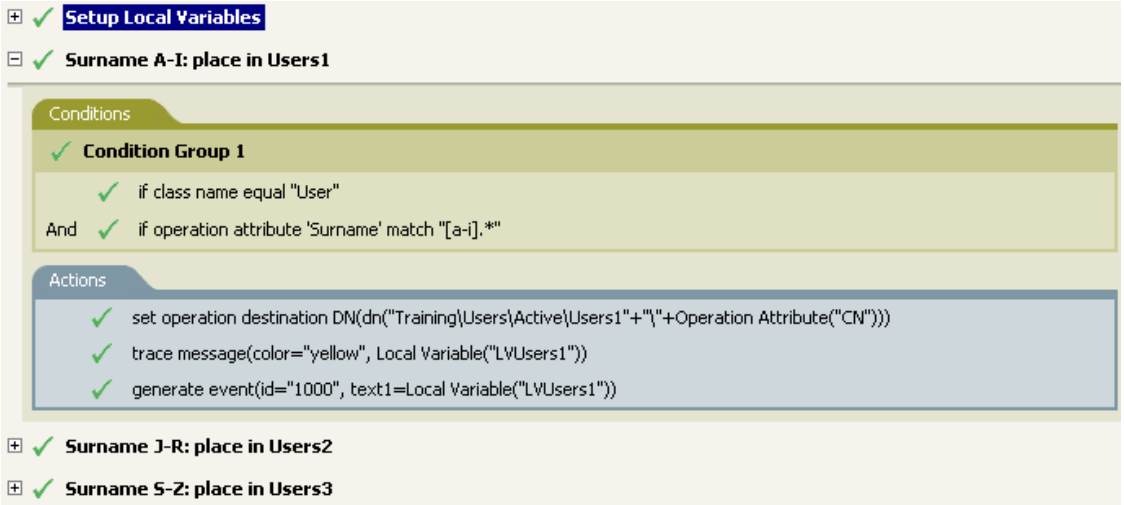
字段

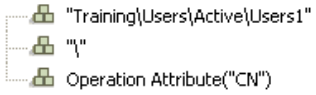
名称

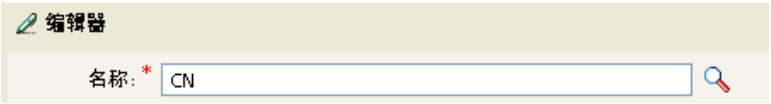
指定该特性的名称。

示例

本示例包含四条规则，它们根据 Surname （姓氏）特性的首字符实施对用户对象的布局策略。并生成一条跟踪讯息和一个自定义 Novell Audit 事件。策略名称为 Policy to Place by Surname （按姓氏布局策略），可从 Novell 支持万维网站点下载。详情请见 “可下载的 Identity Manager 策略”在第 31 页 。







" 设置操作目标 DN" 操作包含 " 操作特性 " 标记。 " 操作特性 " 标记将目标 DN 设置为 CN 特性。该规则采用 Training\Users\Active\Users 的环境，并添加一个 \ 以及 CN 特性的值。

2.7.14 操作属性

扩展到当前操作的操作属性的值。

字段

名称

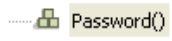
指定操作属性的名称

示例

2.7.15 口令

扩展到当前操作的口令。

示例



2.7.16 已去除的特性

扩展到当前操作已去除的特性的值。它仅适用于修改操作。

字段

名称
　指定特性的名称。

示例

2.7.17 已去除的权利

扩展到当前操作中已取消的权利的值。

字段

名称
　指定权利的名称。

示例

2.7.18 源特性

扩展到源数据存储区中的对象的某一特性的值。

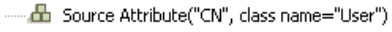
字段

类名称
　（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

名称

特性的名称。

示例



2.7.19 源 DN

扩展到当前操作的源 DN。

字段

转换

选择是否将 DN 转换为目标数据存储区使用的格式。

开始

指定起始 RDN 索引：

- 索引 0 是最靠近根的 RDN
- 正索引是从最靠近根的 RDN 计算的偏移量
- 索引 -1 是最靠近叶的段
- 负索引是从最靠近叶的 RDN 到最靠近根的 RDN 的偏移量

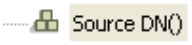
长度

指定要包括的 RDN 段数。负数可解释为 （段的总数 + 长度） + 1。例如，对于具有 5 个段的 DN，其长度 -1 = (5 + (-1)) + 1 = 5、 -2 = (5 + (-2)) + 1 = 4，依此类推。

注释

如果起始索引和长度设置为默认值 {0,-1}，则会使用整个 DN ；否则将仅使用由起始索引和长度指定的 DN 部分。

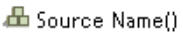
示例



2.7.20 源名称

扩展到当前操作的目标 DN 的无资格的相对判别名 (RDN)。

示例



2.7.21 文本

扩展到文本。

字段

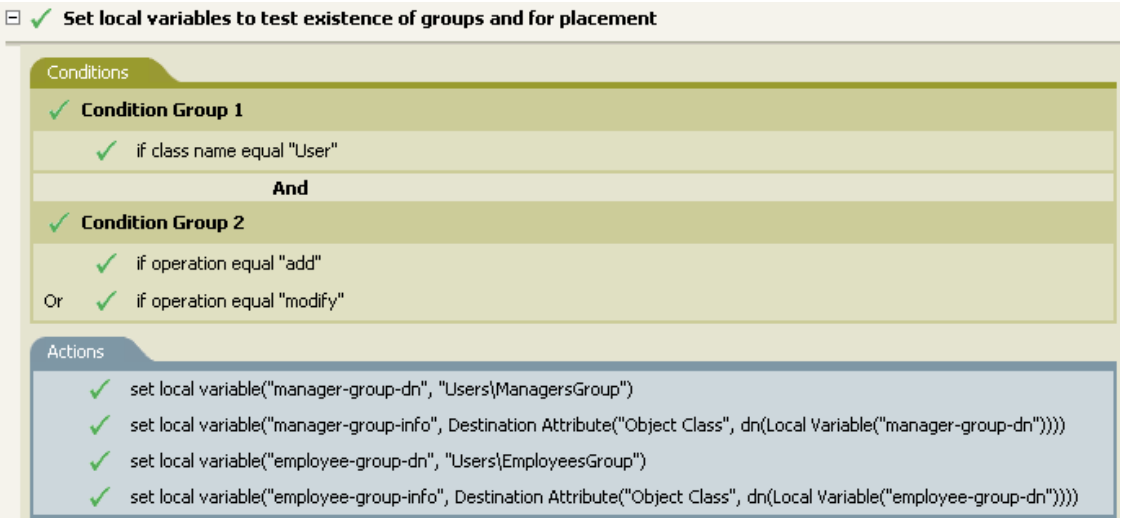
文本

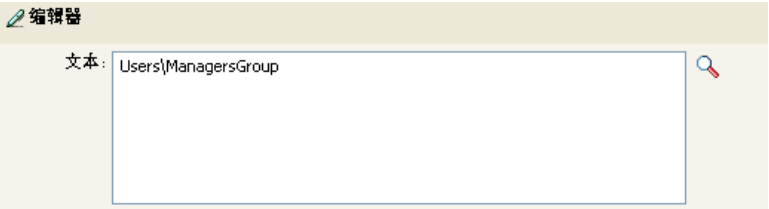
指定文本。

示例

本示例运用了 Govern Groups for User Based on Title （根据职务管理用户组）策略，可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。

" 设置局部变量 " 操作使用 " 文本 " 标记定义经理组的 DN。 " 文本 " 标记可以包含对象或纯文本。





文本名词包含管理员组的 DN。可以浏览至要使用的对象，或在编辑器中键入信息。

2.7.22 唯一名称

根据指定的准则扩展到基于模式的名称，该名称在目标数据存储区中是唯一的。

字段

名称

指定要检查唯一性的特性名称。

范围

指定检查唯一性的范围。

开始搜索

选择搜索的起始点。起始点可以是数据存储区的根，或由 DN 或关联指定的根。

模式

指定使用自变量构建器生成唯一值时所用的模式。

计数器起点

指定查找唯一名称时需要使用的计数器的起始数字。

位数

指定计数器的宽度 （以位数为单位），默认为 1。选中 " 平板计数器 （初始为 0） " 复选框会在前面追加 0，以与位数长度相匹配。例如，如果位数宽度为 3，则将初始唯一值追加为 001，然后为 002，依次类推。

注释

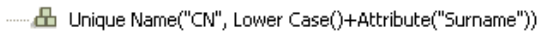
对于每个指定的模式，将使用提供的特性名称、范围和搜索起始点，执行针对目标数据存储区的查询。还将按顺序尝试每个指定模式，直到找到不会返回任何已找到对象的值为止。

如果已用尽所有指定模式，将在最终模式后追加一个计数器，然后重复尝试该模式 （每尝试一次，计数器都加一），直到此查询不返回任何实例。

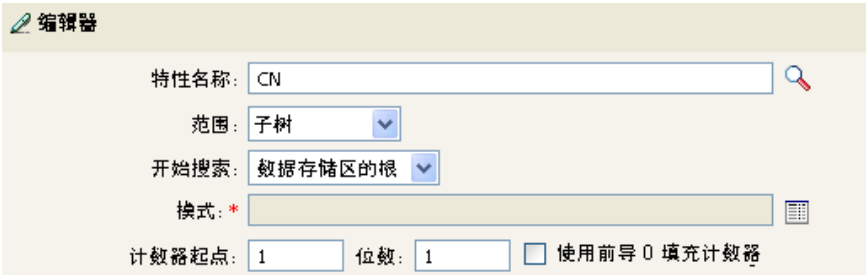
使用 " 计数器起点 " 字段可以为计数器设置不同的起点数字。计数器使用由 " 位数 " 字段指定的位数。如果该位数少于指定的位数，则使用 0 填充该计数器。如果该位数超过指定的位数，则无法生成唯一名称，且附加规则将返回一个错误状态。

如果目标数据存储区为 Identity Vault 且名称字段保留为空，则根据伪特性 "[Entry].rdn" 执行搜索，该特性表示对象的 RDN 而不考虑可能的命名特性。如果目标数据存储区是已连接的应用程序，则需要填写名称字段。

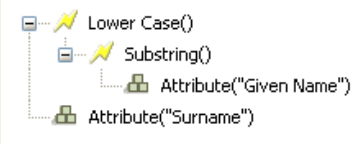
示例



在以下示例中， " 编辑器 " 窗格用于构造唯一名称自变量：



已构造了下面的模式来提供唯一名称：



如果此模式未生成唯一名称，将会追加一位，以增加到指定的位数。本示例中，通过追加位数会生成九个附加的唯一名称 (pattern1 - pattern9)，之后会发生错误。

2.7.23 源 DN 不匹配

扩展到当前操作中的源 DN 部分，它对应于与 If Source DN 条件的最近匹配项不相匹配的 DN 部分。

字段

转换

选择是否将 DN 转换为目标数据存储区使用的格式。

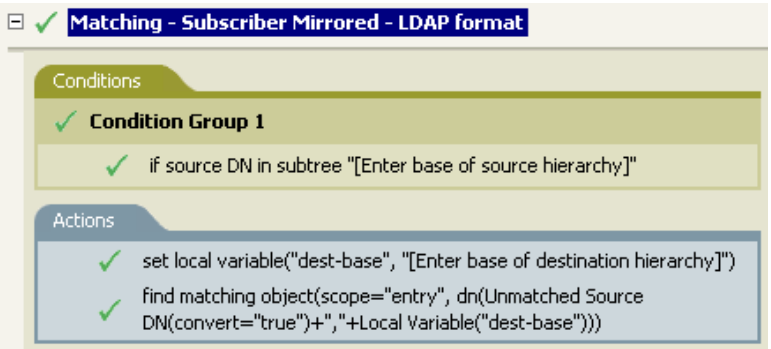
注释

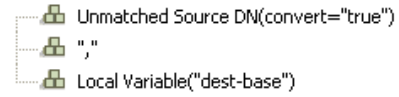
如果没有匹配项，则使用整个 DN。

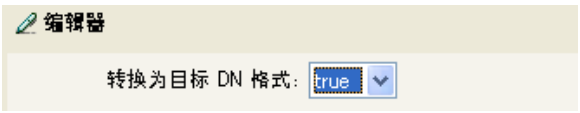
示例

本示例运用了 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “匹配 - 已镜像的订购者 - LDAP 格式”在第 85 页 。

" 查找匹配对象 " 操作使用 " 源 DN 不匹配 " 标记以 LDAP 格式构建匹配信息。它对源 DN 的不匹配部分进行匹配。







2.7.24 XPath

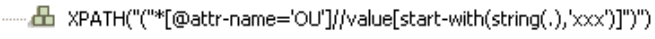
扩展到 XPath 1.0 表达式的求值结果。

字段

表达式

指定要求值的 XPath 1.0 表达式。

示例



2.8 动词标记

本节包含可使用 " 自变量构建器 " 界面访问的所有动词标记的详细参考信息。

- “转义目标 DN”在第 186 页
- “转义源 DN”在第 187 页
- “小写”在第 187 页
- “语法分析 DN”在第 188 页
- “全部替换”在第 190 页
- “替换第一个”在第 190 页
- “子字符串”在第 192 页
- “大写”在第 193 页

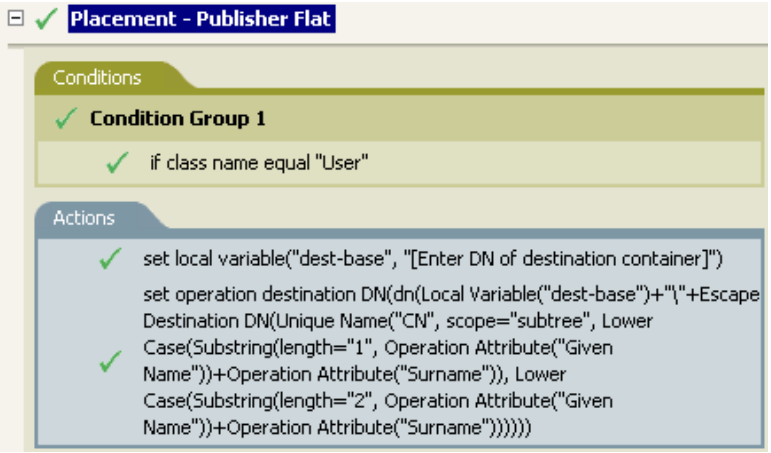
2.8.1 转义目标 DN

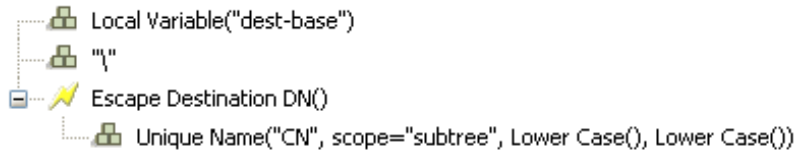
根据目标数据存储区的 DN 格式规则转义字符串。

示例

本示例运用了 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “布局 - 发布者平面文件”在第 91 页 。

" 设置操作目标 DN" 操作使用 " 转义目标 DN" 标记构建用户对象的目标 DN。



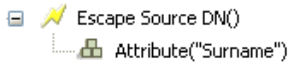


" 转义目标 DN" 标记采用 " 唯一名称 " 中的值，并将其设置为目标 DN 的格式。

2.8.2 转义源 DN

根据源数据存储区的 DN 格式规则转义字符串。

示例



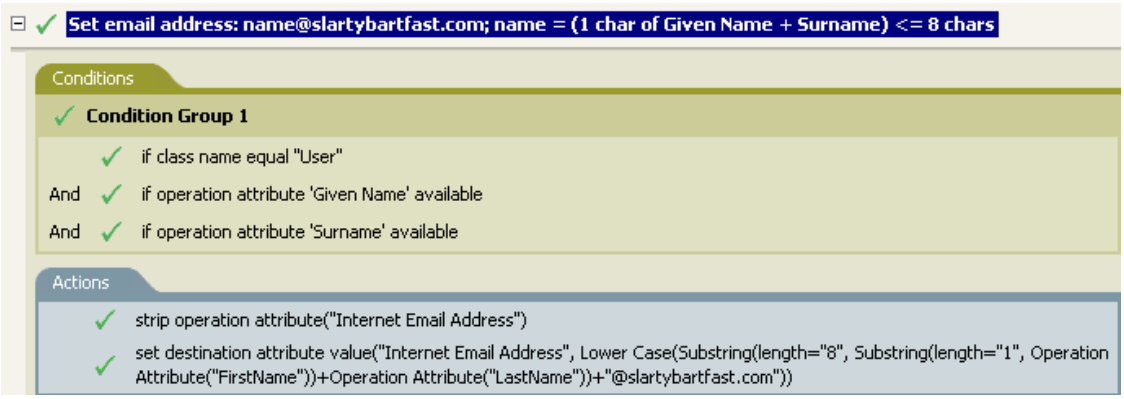
2.8.3 小写

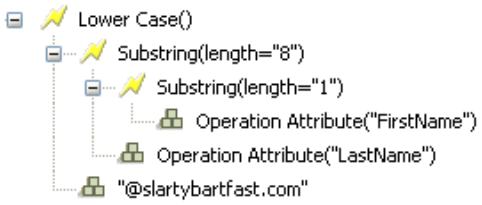
将字符串中的字符转换为小写字母。

示例

此示例将电子邮件地址设置为 name@slartybartfast.com，其中的 name 等于 Given Name （名）的首字母加 Surname （姓氏）。策略名称为 Policy:Create E-mail from Given Name and

Surname （由名和姓氏创建电子邮件），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





" 小写 " 标记将 " 设置目标特性值 " 操作中的所有信息设置为小写字母。

2.8.4 语法分析 DN

将 DN 转换为替换格式。

字段

开始

指定起始 RDN 索引：

- 索引 0 是最靠近根的 RDN
- 正索引是从最靠近根的 RDN 计算的偏移量
- 索引 -1 是最靠近叶的段
- 负索引是从最靠近叶的 RDN 到最靠近根的 RDN 的偏移量

长度

要包括的 RDN 的数量。负数可解释为 （段的总数 + 长度） + 1。例如，对于具有 5 个段的 DN，其长度 -1 = (5 + (-1)) + 1 = 5、 -2 = (5 + (-2)) + 1 = 4，依此类推。

源 **DN** 格式

指定对源 DN 进行语法分析所使用的格式。

目标 **DN** 格式

指定输出已经过语法分析的 DN 所使用的格式。

源 **DN** 分界符

如果源 DN 格式设置为自定义，则指定自定义源 DN 分界符集。

目标 **DN** 分界符

如果目标 DN 格式设置为自定义，则指定自定义目标 DN 分界符集。

注释

如果起始索引和长度设置为默认值 {0,-1}，则使用整个 DN ；否则仅使用由起始索引和长度指定的 DN 部分。

指定自定义 DN 格式时，构成分界符集的八个字符定义如下：

1. 已定义类型名称的布尔标志：0 表示未定义类型的名称，而 1 表示已定义类型的名称

2. Unicode 无映射字符布尔标志：0 表示不输出无法映射的 Unicode 字符，也不会将其解释为转义的十六进制字符串，如 \FEFF。 eDirectory 不接受以下 Unicode 字符：0xfeff、 0xfffe、0xfffd 和 0xffff。

3. 相对 RDN 分界符

4. RDN 分界符

5. 名称分隔符

6. 名称值分界符

7. 通配符字符

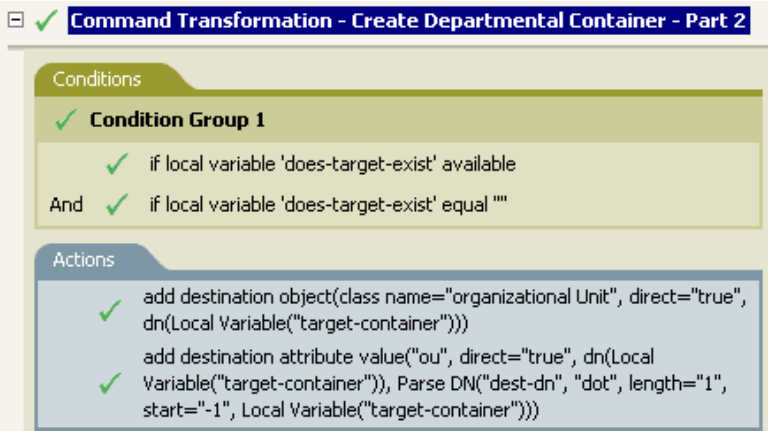
8. 转义字符

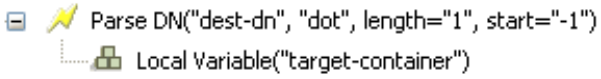
如果 RDN 分界符和相对 RDN 分界符是同一个字符，则名称方向为根右侧，否则其方向为根左侧。

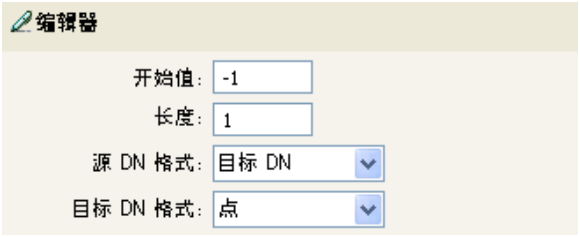
如果分界符集超过八个字符，则将多余字符视为需要转义的字符，但这些字符没有其它特殊的含义。

示例

本示例使用 " 语法分析 DN" 标记来构建 " 添加目标特性值 " 操作的值。本示例运用了 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “命令转换 - 创建部门树枝 - 第 1 部分和第 2 部分”在第 69 页 。







" 语法分析 DN" 标记从源 DN 获取信息，然后将其转换为点表示法。语法分析 DN 的信息储存在 OU 特性值中。

2.8.5 全部替换

替换字符串中所有具体出现的某一正则表达式。

字段

正则表达式

指定与要替换的子字符串匹配的正则表达式。

替换为

指定替换字符串。

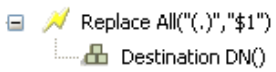
注释

有关创建正则表达式的详细信息，请参见：

- Sun 的 Java 万维网站点 (http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Pattern.html)
- Sun 的 Java 万维网站点 (http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Matcher.html#replaceAll (java.lang.String))

此处使用的模式选项为 CASE_INSENSITIVE、 DOTALL 和 UNICODE_CASE，但是可以通过使用适当的嵌入式转义符进行反转。

示例



2.8.6 替换第一个

替换字符串中第一次出现的某一正则表达式。

字段

正则表达式

指定与要替换的子字符串匹配的正则表达式。

替换为

指定替换字符串。

注释

将匹配实例替换为由 " 替换为 " 字段中指定的值指定的字符串。

有关创建正则表达式的详细信息，请参见：

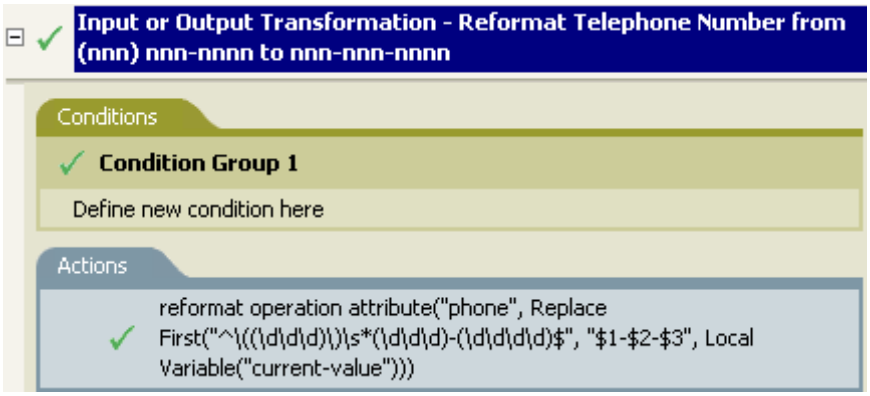
- http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Pattern.html (http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Pattern.html)
- http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Matcher.html#replaceAll(java.lang.String) (http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Matcher.html#replaceAll (java.lang.String))

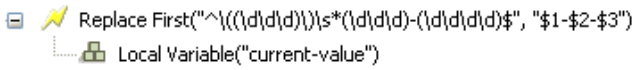
此处使用的模式选项为 CASE_INSENSITIVE、 DOTALL 和 UNICODE_CASE，但是可以通过使用适当的嵌入式转义符进行反转。

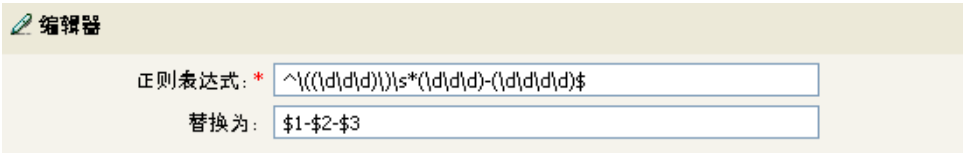
示例

该示例将电话号码的格式由 (nnn)-nnn-nnnn 重设为 nnn-nnn-nnnn。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “输入或输出转换 - 将电话号码格式重新从 (nnn) nnn-nnnn 设置为 nnn-nnn-nnnn”在第 81 页 。

" 重新设置操作特性的格式 " 操作将使用 " 替换第一个 " 标记。







正则表达式 ^\(((\d\d\d)\)\s*(\d\d\d)-(\d\d\d\d)$ 表示 (nnn) nnn-nnnn，而正则表达式 $1-$2-$3 表示 nnn。此规则将电话号码的格式从 (nnn) nnn-nnnn 转换为 nnn-nnn-nnnn。

2.8.7 子字符串

抽取字符串的一部分。

字段

开始

指定起始字符的索引：

- 索引 0 表示第一个字符。
- 正索引表示距离字符串起始位置的偏移量
- 索引 -1 表示最后一个字符
- 负索引表示从字符串中最后一个字符开始的向前偏移量

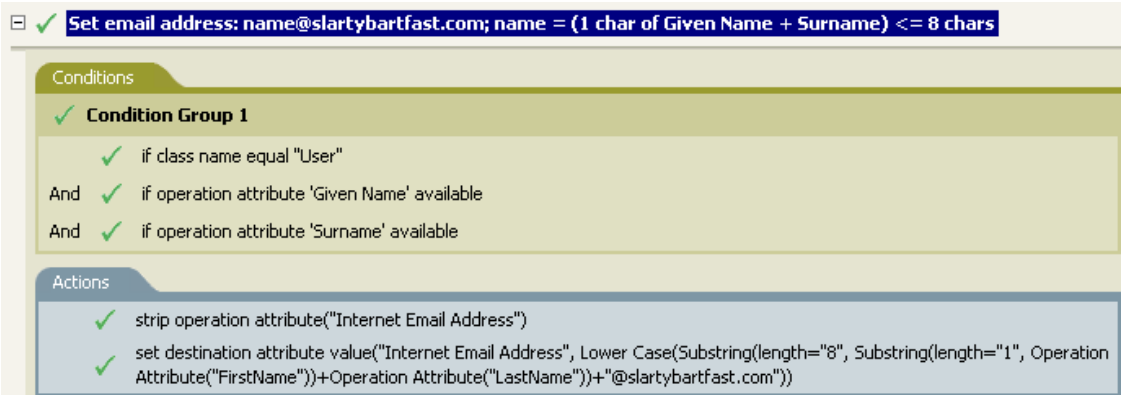
例如，如果起始索引指定为 -2，则从结尾第一个字符开始读取。如果起始索引指定为 -3，则从结尾第 2 个字符开始读取。

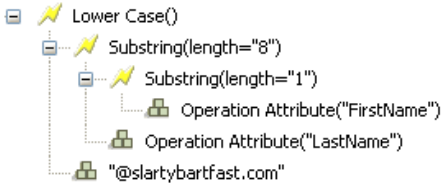
长度

子字符串中从开始位置起包含的字符数。负数可解释为 （字符总数 + 长度） + 1。例如， -1 表示全长或原始字符串长度。如果指定为 -2，则长度为全长 -1。对于具有 5 个字符的字符串，长度 -1 = (5 + (-1)) + 1 = 5 ； 长度 -2 = (5 + (-2)) + 1 = 4，依此类推。

示例

此示例将电子邮件地址设置为 name@slartybartfast.com，其中的 name 等于 Given Name （名）的首字母加 Surname （姓氏）。策略名称为 Policy:Create E-mail from Given Name and Surname （使用名和姓氏创建电子邮件），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





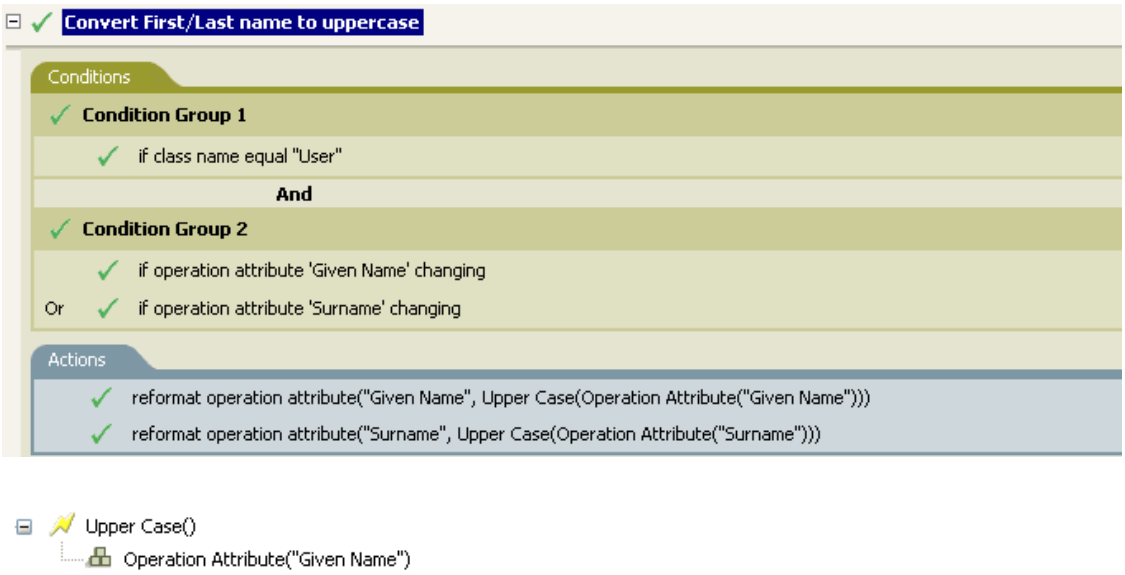
在 " 设置目标特性值 " 操作中，两次使用了 " 子字符串 " 标记。该标记使用 First Name （名）特性的第一个字符加上 Last Name （姓）特性的八个字符来构成一个子字符串。

2.8.8 大写

将字符串中的字符转换为大写字母。

示例

在本示例中，将用户对象的名和姓特性转换成大写字母。策略名称为 Policy:Convert First/Last Name to Upper Case （将名 / 姓转换为大写），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。



2.9 值

本节包含常用策略构建器值的列表。

2.9.1 比较方式

表 **2-7** 比较方式

方式	说明
case	逐个字符进行区分大小写比较。
nocase	逐个字符进行不区分大小写比较。

方式	说明
regex	整个字符串的正则表达式匹配项。默认情况下，区分大小写，但是可以通过表达式中的转义符进行更改。 请参见 Sun 的 Java 万维网站点 (http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Pattern.html) 和 Sun 的 Java 万维网站点 (http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Matcher.html#matches())。 此处使用的模式选项为 CASE_INSENSITIVE、DOTALL 和 UNICODE_CASE，但是使用适当的嵌入式转义符可以进行反转。
src-dn	使用适用于源数据存储区的 DN 格式的语义进行比较。
dest-dn	使用适用于目标数据存储区的 DN 格式的语义进行比较。
numeric	数值比较。
octet	比较八位组 （Base64 编码）值。
structured	根据用于特性的结构化语法的比较规则对结构化特性进行比较。

在 iManager 中使用策略构建器定义策略

3

策略构建器是一个完整的图形界面，可用于创建和管理策略，定义已连接系统之间的数据交换。

以下章节提供了有关策略以及如何使用策略构建器的信息：

- “策略”在第 33 页
- “iManager 中的策略构建器任务”在第 196 页

本节还包含以下详细参考章节：

- “条件”在第 227 页
- “操作”在第 244 页
- “名词标记”在第 283 页
- “动词标记”在第 296 页

3.1 策略

作为了解策略的工作方式的一部分，了解策略的组成非常重要。

- 策略由规则构成。
- 规则是在已定义的操作 （请参见 “操作”在第 244 页 ）发生之前必须事先满足的一组条件 （请参见 “条件”在第 227 页 ）。
- 操作可以具有动态自变量，这些自变量由运行时扩展的标记派生而来。
- 标记可分为两类：名词标记 （参见 “名词标记”在第 283 页 ）和动词标记 （参见 “动词标记”在第 296 页 ）。
  - 名词标记可扩展为由当前操作、源数据存储区或目标数据存储区，或一些外部源派生而来的值。
  - 动词标记用于修饰从属于它们的其它标记的已连接结果。
- 规则中通常使用正则表达式 （参见 “正则表达式”在第 226 页 ）和 XPath 1.0 表达式 （参见 “XPath 1.0 表达式”在第 227 页 ）为策略创建所需的结果。
- 策略在 XDS 文档中操作，主要用途是检查和修改该文档。
- 操作是 XDS 文档中作为输入要素和输出要素子级的任意要素。这些要素是 Novell 的 nds.dtd 的一部分；有关详细信息，请参见 NDS DTD (http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/ref/ndsdtd/index.html)。
- 操作通常表示事件、命令或状态。
- 策略将单独应用于每项操作。顺序对每项操作应用策略时，该操作将成为当前操作。每项规则依次应用于当前操作。所有规则都将应用于当前操作，除非某优先规则执行了某一操作，从而导致不再应用后续规则。
- 策略还可以从文档外部获得其它上下文，但所产生的副作用并不会反映到结果文档中。

3.2 iManager 中的策略构建器任务

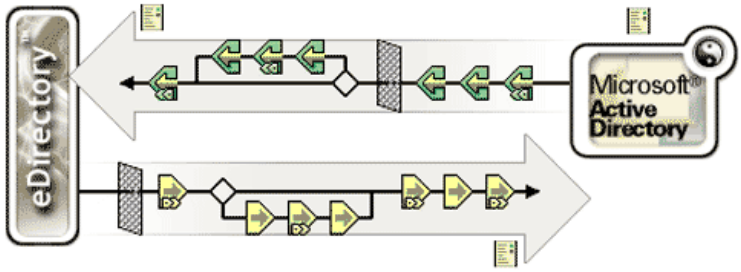
本节介绍在策略构建器中执行的以下常见任务：

- “打开策略构建器”在第 196 页
- “创建策略”在第 196 页
- “修改策略”在第 205 页
- “定义策略中的单个规则”在第 197 页
- “在规则中定义单个自变量”在第 198 页
- “使用预定义规则”在第 207 页

3.2.1 打开策略构建器

1. 在 iManager 中，展开 *Identity Manager* 职能，然后单击 "Identity Manager 概述 "。
2. 指定驱动程序集。
3. 单击要管理其策略的驱动程序。将打开 "Identity Manager 驱动程序概述 "：

图 ***3-1*** *Identity Manager* 驱动程序概述



策略是在 "Identity Manager 驱动程序概述 " 中进行管理的。

3.2.2 创建策略

1. 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2. 单击表示要定义的策略的图标。

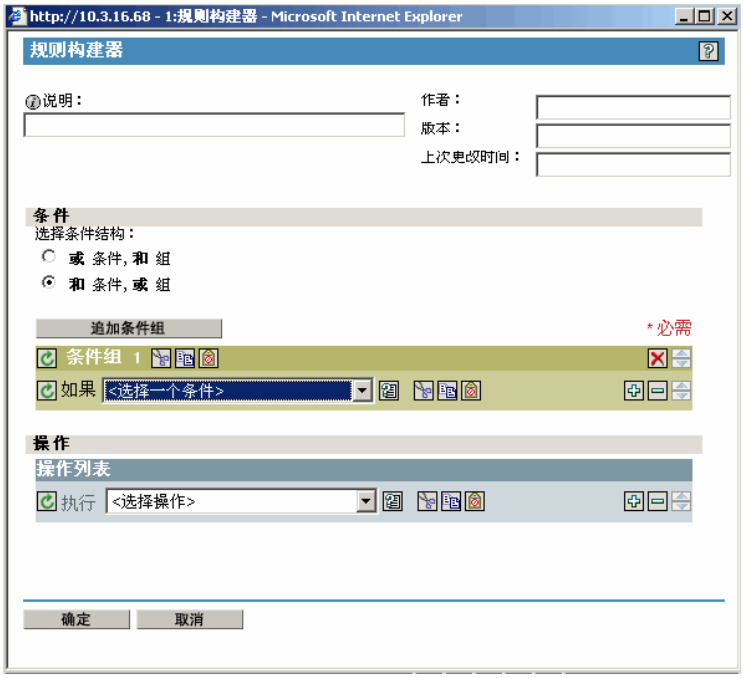
   表示未定义的策略。

   表示已定义的策略。
3. 单击 " 插入 "。
4. 输入新策略的名称，然后选择策略构建器。
5. 将显示该策略。要为此策略定义一个或多个规则，请单击 " 追加新规则 "，然后按照“定义策略中的单个规则”在第 197 页中的说明进行操作。

3.2.3 定义策略中的单个规则

将在策略构建器的 " 规则构建器 " 窗口中定义规则：

图 *3-2* 策略构建器的 " 规则构建器 " 窗口



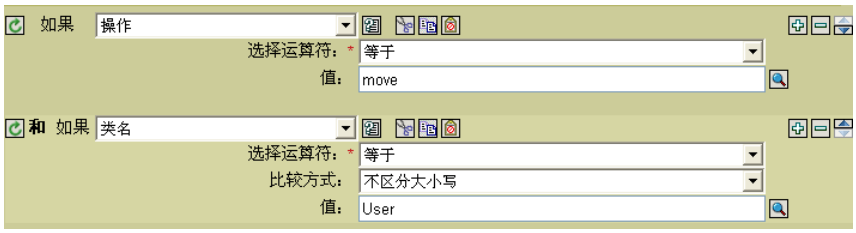
" 规则构建器 " 界面允许您使用智能下拉菜单快速创建和修改规则。

在规则构建器中，可以定义发生已定义的操作必须事先满足的一组条件。

例如，如果需要创建不允许在您的环境中添加任何新对象的规则，则需要定义与以下规则类似的规则：当添加操作出现时，禁止该操作。

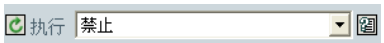
要在规则构建器中实现此逻辑，则可以选择以下条件：

图 *3-3* 在 " 规则构建器 " 界面中移动用户条件



以及以下操作：

图 *3-4* 在 " 规则构建器 " 界面中禁止操作

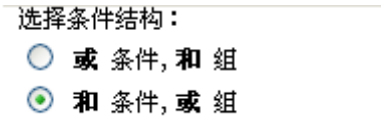


请参见 “条件”在第 227 页 和 “操作”在第 244 页，或规则构建器中提供的条件和操作的详细参照。

提示

要创建更复杂的条件，您可以使用 and/or 语句将条件或条件组连接在一起。可以通过选择条件结构修改这些条件连接的方式：

图 *3-5* 条件结构单项选择按钮



- 单击 图标查看字段的值列表。在以上示例中，单击此图标将打开有效类名称的列表。
- 单击 图标以使用自变量构建器界面构造自变量。
- 单击 图标禁用策略、规则、条件或操作。单击 图标则可以重新启用它。
- 单击 图标可以对策略或规则添加注释。注释直接储存在策略或规则中，并且没有储存时间限制。
- 使用"剪切"/"复制"/"粘贴"图标， 可以使用策略构建器剪贴板。如果剪贴板上的当前内容在此位置处无效，则 " 粘贴 " 图标将被禁用。
- 使用 图标可以添加、去除和定位条件。
- 使用 追加条件组 按钮可以添加条件组。
- 使用 图标可以去除和定位条件组。

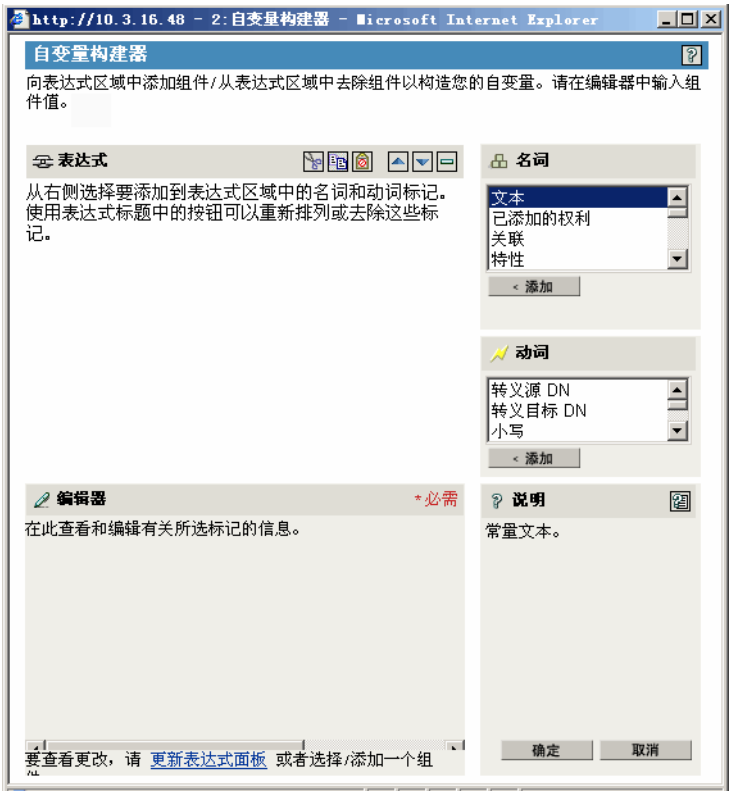
3.2.4 在规则中定义单个自变量

自变量构建器提供了动态图形界面，允许您构造规则构建器中使用的复杂自变量表达式。要访问自变量构建器，请参见 “自变量构建器”在第 201 页。

自变量由操作动态使用，并由运行时扩展的标记派生而来。

标记可分为两类：名词标记和动词标记。名词标记可扩展为由当前操作、源数据存储区或目标数据存储区，或一些外部源派生而来的值。动词标记用于修饰从属于它们的其它标记的已连接结果。

图 **3-6** 默认 " 自变量构建器 " 界面



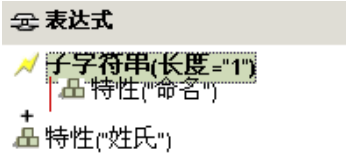
要定义表达式，请选择一个或多个名词标记 （值、对象、变量等），并将它们与动词标记（子字符串、转义符、大写和小写）组合，以构造自变量。组合多个标记可以构造复杂自变量。

例如，如果要将自变量设置为某一特性值，可以先选择 " 特性 " 标记，再选择特性名称：

图 **3-7** 编辑器将 *ds.novell* 显示为文本自变量

如果只需要该特性的一部分，则可以将 " 特性 " 标记和 " 子字符串 " 标记组合：

图 **3-8** 显示 *Give Name* （名）特性的一个长度为 *1* 的子字符串，并显示 *Surname* （姓氏）特性的表达式。



添加标记后，可以在编辑器中编辑它的字段。

有关自变量构建器中可用的名词和动词的详细参考信息，请参见 “名词标记”在第 283 页 和 “动词标记”在第 296 页 。

提示

- 要创建更多复杂条件，可以使用 and/or 语句将条件或条件组连接在一起。
- 使用  图标可以移动或删除名词标记和动词标记。
- 单击  图标查看字段的值列表。
- 添加名词标记或动词标记后，您可以在编辑器中提供值，然后立即添加其它名词标记或动词标记。无需刷新 " 表达式 " 窗格来应用更改；在执行下一项操作时将自动出现这些更改。

虽然大部分自变量都是使用自变量构建器定义的，但在策略构建器中还存在若干其它构建器，可供条件编辑器和操作编辑器使用。每个构建器都可以递归地调用以下列表中的任意构建器：

- “自变量操作构建器”在第 200 页
- “自变量构建器”在第 201 页
- “匹配特性构建器”在第 202 页
- “操作自变量组件构建器”在第 203 页
- “自变量值列表构建器”在第 203 页
- “命名字符串构建器”在第 204 页
- “条件自变量组件构建器”在第 204 页

自变量操作构建器

自变量操作构建器允许您设置对于每个 (在第 257 页) 操作和实施权利 (在第 260 页) 操作所需的操作。

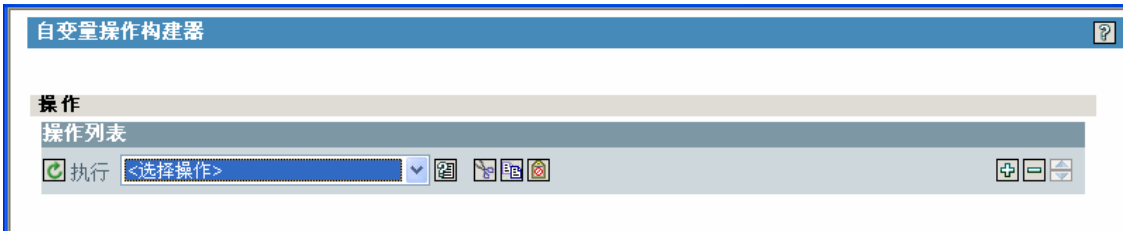
在以下示例中，将为当前操作中添加的每个组权利执行 " 添加目标特性值 " 操作。

图 ***3-9*** 自变量操作构建器

要定义 " 添加目标特性值 " 操作，请单击起动自变量操作构建器的图标。在自变量操作构建器中，可以定义所需的操作。在下面的示例中，将 member （成员）特性添加到每个已添加的 " 组 " 权利的目标对象中。

图 ***3-10*** 自变量操作构建器

图 ***3-11*** 自变量操作构建器



自变量构建器

通过在以下操作中单击 " 编辑自变量 " 图标，起动自变量构建器。

- 添加关联 (在第 246 页)
- 添加目标特性值 (在第 246 页)
- 添加目标对象 (在第 247 页)
- 添加源特性值 (在第 248 页)
- 追加 XML 文本 (在第 251 页)
- 清除目标特性值 (在第 252 页) 所选对象为 DN 或 Association。
- 清除源特性值 (在第 254 页) 所选对象为 DN 或 Association。
- 删除目标对象 (在第 255 页) 所选对象为 DN 或 Association。
- 删除源对象 (在第 256 页) 所选对象为 DN 或 Association。
- 查找匹配对象 (在第 256 页)
- 对于每个 (在第 257 页)
- 移动目标对象 (在第 261 页)
- 移动源对象 (在第 262 页)
- 重新设置操作特性的格式 (在第 262 页)
- 去除关联 (在第 263 页)

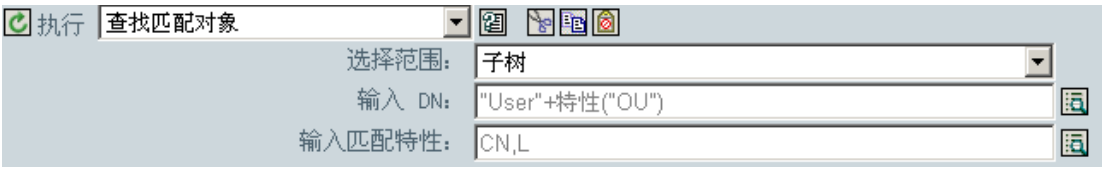
- 去除目标特性值 (在第 264 页)
- 去除源特性值 (在第 265 页)
- 重命名目标对象 (在第 265 页) 所选对象为 DN、 Association 或 Enter String。
- 重命名源对象 (在第 266 页) 所选对象为 DN、 Association 或 Enter String。
- 设置目标特性值 (在第 270 页) 所选对象为 DN 或 Association，输入值类型不为 structured。
- 设置目标口令 (在第 271 页)
- 设置局部变量 (在第 272 页)
- 设置操作关联 (在第 273 页)
- 设置操作类名 (在第 273 页)
- 设置操作目标 DN (在第 273 页)
- 设置操作属性 (在第 274 页)
- 设置操作源 DN (在第 274 页)
- 设置操作模板 DN (在第 275 页)
- 设置源特性值 (在第 275 页)
- 设置源口令 (在第 276 页)
- 设置 XML 特性 (在第 278 页)
- 状态 (在第 279 页)
- 跟踪讯息 (在第 281 页)

匹配特性构建器

使用匹配特性构建器可以选择 “查找匹配对象”在第 256 页 操作所使用的特性和值，以确定某个匹配对象是否存在于数据存储区中。

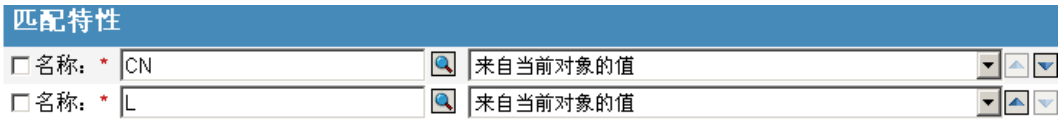
例如，如果您想根据常用名和位置匹配用户，则需要选择以下条件：

图 *3-12* 查找匹配对象



然后单击 " 输入匹配特性 " 字段旁的 " 编辑自变量 " 图标起动 " 匹配特性构建器 " 界面：

图 *3-13* 匹配特性构建器



选择 " 浏览特性 " 图标以浏览至要匹配的特性并选择该特性。在本示例中，它们为 L 和 CN。

第二列允许您通过选择 *Use value(s) from current Object*（使用当前对象的值）来匹配储存在特性中的当前值。通过选择 " 其它值 " 还可以与其它值进行匹配。可以创建要匹配的任何值。选择值类型，则相应的构建器在 " 输入状态 " 字段中可用。

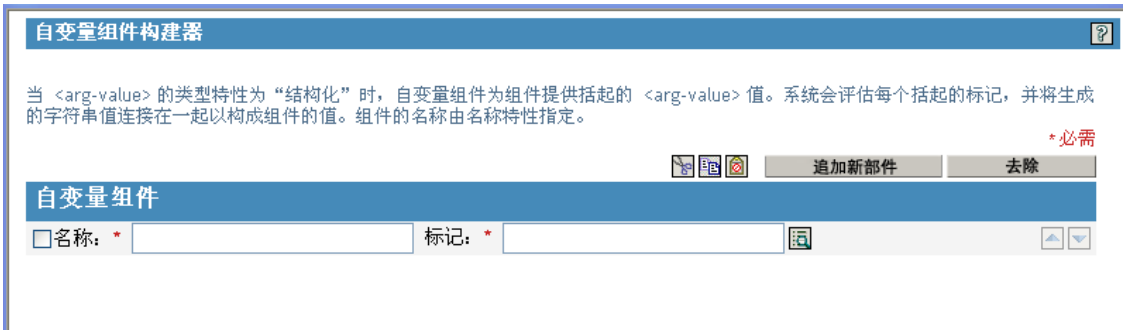
操作自变量组件构建器

如果 " 输入值类型 " 选择的是 Structured，则通过选择以下操作可以起动操作自变量组件构建器。

- 添加目标特性值 (在第 246 页)
- 添加源特性值 (在第 248 页)
- 重新设置操作特性的格式 (在第 262 页)
- 去除目标特性值 (在第 264 页)
- 去除源特性值 (在第 265 页)
- 设置默认特性值 (在第 269 页)
- 设置源特性值 (在第 275 页)

图 ***3-14*** 操作自变量组件构建器

图 ***3-15*** 操作自变量组件构建器



自变量值列表构建器

使用自变量值列表构建器可以构造设置默认特性值 (在第 269 页) 操作的默认自变量值。

例如，如果要将默认位置设置为 Unknown （未知），请选择以下操作：

图 *3-16* 自变量值列表构建器

然后单击 " 输入值 " 字段旁的图标起动 " 自变量值列表构建器 " 界面，再构造类似于以下自变量的自变量：

图 *3-17* 自变量值列表构建器

命名字符串构建器

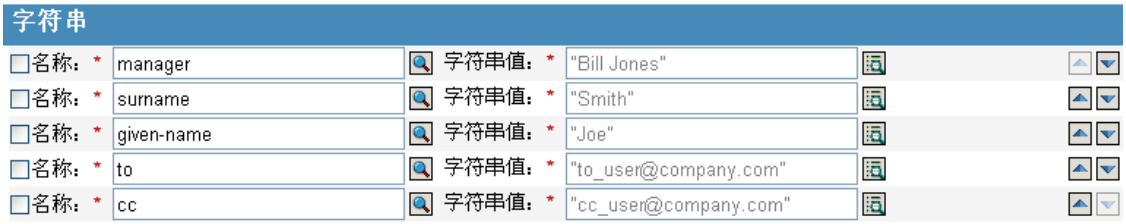
命名字符串构建器允许您构造名称 / 值对，用于如生成事件 (在第 258 页)、发送电子邮件 (在第 267 页) 和通过模板发送电子邮件 (在第 268 页) 等操作中。

对于 " 生成事件 " 操作，命名字符串对应于随事件提供的自定义值字段：

图 *3-18* 命名字符串构建器

对于 " 发送邮件 " 操作，命名字符串相当于电子邮件要素：

图 *3-19* 发送邮件操作



与起动命名字符串构建器的操作对应的帮助文件中，包含可能出现的值的完整列表。

条件自变量组件构建器

单击 " 编辑自变量 " 图标起动条件自变量组件构建器。

要想看到该图标，您必须将以下条件的 " 方式 " 选为 Structured：

- If Attribute (在第 228 页)
- If Destination Attribute (在第 230 页)
- If Source Attribute (在第 241 页)

图 *3-20* 结构化选项

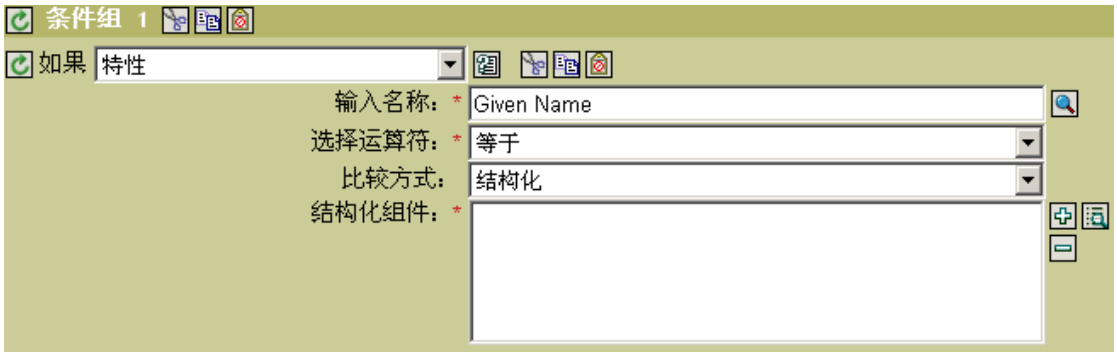
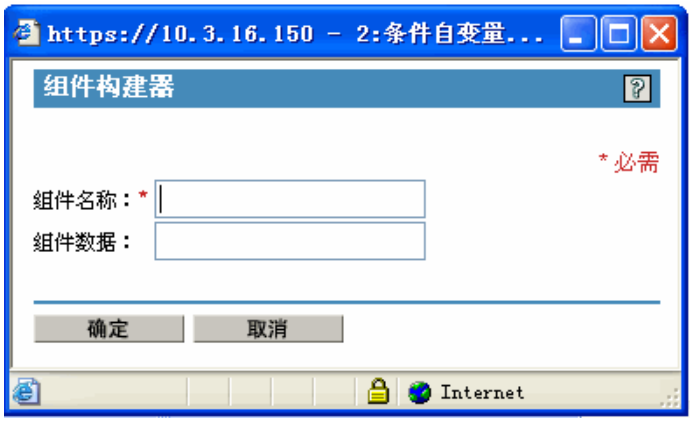


图 *3-21* 条件自变量组件构建器



3.2.5 修改策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 单击表示要修改的策略的图标。
3 选择要修改的策略，然后单击 " 编辑 "。

3.2.6 去除策略

从所选的策略集中去除该策略，但不要将其删除。

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 单击表示要去除的策略的图标。

若要查看与策略集不关联的策略，请执行以下操作：

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。

2 单击 " 查看所有策略 " 图标 。

若要将去除的策略重新添加到策略集，请执行以下操作：

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 单击要添加策略的策略集。
3 单击 " 插入 "。
4 选择 " 使用现有策略 "，然后单击 " 浏览 " 按钮。
5 浏览至要添加的策略。

提示 **:** 请确保您在正确的树枝中查看该策略。

6 单击 " 确定 "。
7 单击 " 关闭 "。

3.2.7 重命名策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 单击表示要重命名的策略的图标。
3 单击 " 重命名 "，然后重命名该策略。
4 单击 " 确定 "。
5 单击 " 关闭 "。

3.2.8 删除策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 单击表示要删除的策略的图标。
3 选择要删除的策略，然后单击 " 删除 "。

3.2.9 从 XML 文件导入策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 单击表示要导入的策略的图标。
3 选择该策略，然后单击 " 编辑 "。
4 单击 " 插入 " 按钮，然后选择 *Import an XML file containing DirXML® Script* （导入包含 DirXML® 底稿的 XML 文件）。
5 浏览至要导入的策略文件并选择该文件，然后单击 " 确定 "。

3.2.10 将策略导出至 XML 文件

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 单击要导出策略的相应图标。
3 选择该策略，然后单击 " 编辑 "。
4 单击 " 另存为 " 按钮，然后选择保存 DirXML 底稿 XML 文件的位置。

5 单击 " 保存 "。

3.2.11 创建策略参考

策略参考允许您创建单个策略，并可以在多个位置参考它。如果您具有一个用于多个驱动程序或策略的策略，则创建参考可以简化对该策略的管理。

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 单击要作为参照添加的策略的相应图标。
3 选择该策略，然后单击 " 编辑 "。
4 单击 " 插入 " 按钮，然后选择 " 向包含 DirXML 底稿的策略追加参照 "。
5 浏览至要参照的策略对象并选择该对象，然后单击 " 确定 "。

3.2.12 使用预定义规则

iManager 包含二十个预定义规则。可以导入并使用这些规则，也可以创建自己的规则。这些规则包含管理员使用的常见任务。需要提供特定于个人环境的信息才可以自定义这些规则。

- “命令转换 - 创建部门树枝 - 第 1 部分和第 2 部分”在第 208 页
- “命令转换 - 发布者删除 - 禁用”在第 209 页
- “创建 - 必需特性”在第 210 页
- “创建 - 发布者 - 使用模板”在第 211 页
- “创建 - 设置默认特性值”在第 212 页
- “创建 - 设置默认口令”在第 213 页
- “事件转换 - 范围过滤 - 包括子树”在第 214 页
- “事件转换 - 范围过滤 - 排除子树”在第 215 页
- “输入或输出转换 - 将电话号码格式重新从 (nnn) nnn-nnnn 设置为 nnn-nnn-nnnn”在第 215 页
- “输入或输出转换 - 将电话号码格式重新从 nnn-nnn-nnnn 设置为 (nnn) nnn-nnnn”在第 216 页
- “匹配 - 已镜像的发布者”在第 217 页
- “匹配 - 已镜像的订购者 - LDAP 格式”在第 218 页
- “匹配 - 按特性值”在第 219 页
- “布局 - 已镜像的发布者”在第 220 页
- “布局 - 已镜像的订购者 - LDAP 格式”在第 221 页
- “布局 - 发布者平面文件”在第 222 页
- “布局 - 订购者平面文件 - LDAP 格式”在第 223 页
- “布局 - 部门发布者”在第 224 页
- “布局 - 部门订购者 - LDAP 格式”在第 225 页

要访问这些预定义规则，请执行以下操作：

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。

2 单击表示要在其中添加预定义规则的策略的图标。

3 选择一个策略，然后单击 " 编辑 "。

4 单击 " 插入 "，然后选择要使用的预定义规则。

命令转换 - 创建部门树枝 - 第 1 部分和第 2 部分

如果目标数据存储区中没有部门树枝，请进行创建。对驱动程序中的订购者命令转换策略或发布者命令转换策略实施此规则。

使用预定义规则包含两个步骤：在命令转换策略集中创建策略，然后导入此预定义规则。如果要为已有的命令转换策略添加此规则，请转到 “导入预定义规则”在第 208 页 。

创建策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。

2 单击发布者通道或订购者通道中的 " 命令转换策略 " 对象。

3 单击 " 插入 "。

4 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1 在规则构建器中，单击 " 插入 "。

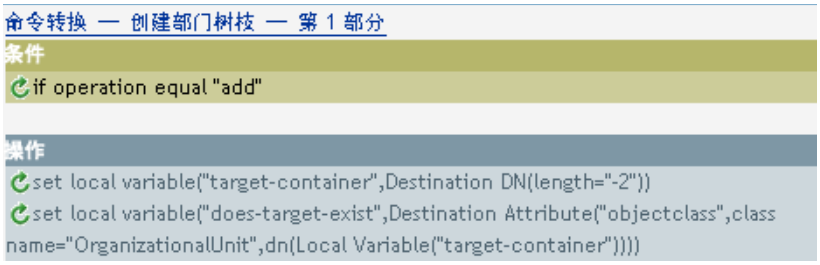
2 选择 " 命令转换 - 创建部门树枝 - 第 1 部分 "。

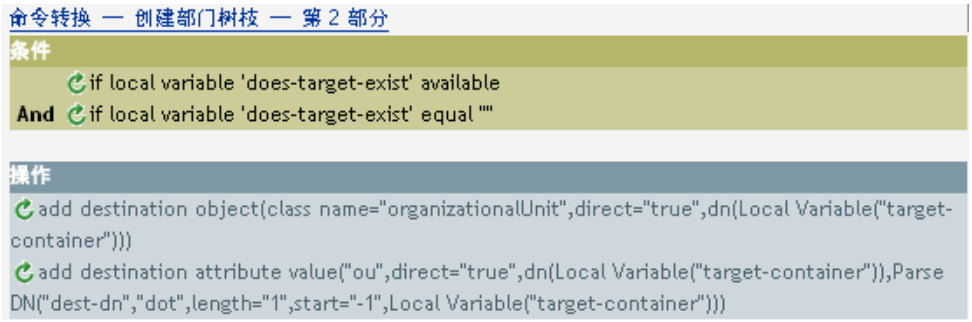
3 单击 " 插入 "。

4 选择 " 命令转换 - 创建部门树枝 - 第 2 部分 "。

5 单击 " 确定 "。

在特定于环境的规则中没有要更改的信息。





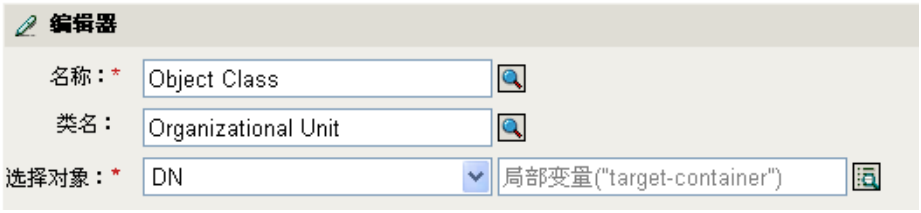
重要：确保按顺序列出规则。必须先执行第 1 部分，然后再执行第 2 部分。

该规则的运行逻辑

如果对象的目标位置不存在，则使用此规则。此规则将创建树枝并将对象放置在树枝中，这样就不会因为无法放置对象而被禁止。

第 1 部分查找 " 添加 " 操作。查找到 " 添加 " 操作时，将设置两个局部变量。第一个局部变量名为 target-container。将 target-container 的值设为目标 DN。第二个局部变量名为 does-target-exist。将 does-target-exist 的值设为 objectclass 的目标特性值。将类设置为 OrganizationalUnit。而 OrganizationalUnit 的 DN 设置为局部变量 target-container。

图 ***3-22*** 创建树枝



第 2 部分检查局部变量 does-target-exist 是否可用。同时还检查局部变量 does-target-exist 的值是否设为空值。如果该值为空，则创建一个组织单元对象。将组织单元的 DN 设为局部变量 target-container 的值。同时还将添加 OU 特性的值。将 OU 特性的值设为新组织单元名称，该名称通过分析局部变量 target-container 的值获得。

命令转换 - 发布者删除 - 禁用

将用户对象的 " 删除 " 操作转换为 " 修改 " 操作，将禁用 eDirectory™ 中的目标用户对象。对驱动程序中的发布者命令转换策略实施此规则。

使用预定义规则包含两个步骤：在命令转换策略集中创建策略，然后导入此预定义规则。如果要为已有的命令转换策略添加此规则，请转到导入预定义规则 (在第 210 页)。

创建策略

1. 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2. 在发布者通道中单击 " 命令转换策略 " 对象。
3. 单击 " 插入 "。
4. 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1. 在规则构建器中，单击 " 插入 "。
2. 选择 " 命令转换 - 发布者删除 - 禁用 "。
3. 单击 " 确定 "。

在特定于个人环境的规则中没有要更改的信息。

该规则的运行逻辑

通常为响应发生在已连接系统中的删除事件，需要将 " 删除 " 命令发送到 Identity Vault 时，使用此规则。在 Identity Vault 中禁用该用户对象，而不是将其删除。处理用户对象的 " 删除 " 命令时， Login Disabled （禁止登录）的目标特性值被设为 "True"，并去除与用户对象的关联，同时将禁止 " 删除 " 命令。该用户对象不能再登录到 Novell eDirectory 树，但并未被删除。

创建 - 必需特性

除非填充必需特性，否则将无法创建用户对象。对驱动程序中的订购者创建策略或发布者创建策略实施此规则。

使用预定义规则包含两个步骤：在创建策略集中创建策略，然后导入预定义规则。如果要为已有的创建策略添加此规则，请转到 “导入预定义规则”在第 211 页 。

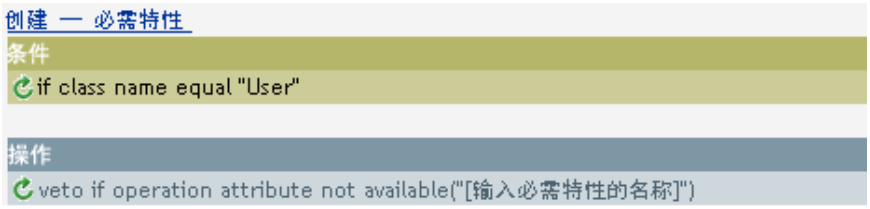
创建策略

1. 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2. 在发布者通道或订购者通道中单击创建策略对象。
3. 单击 " 插入 "。
4. 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1 在规则构建器中，单击 " 插入 "。
2 选择 " 创建 - 必需特性 "。
3 在规则构建器中，单击 " 创建 - 必需特性 "，编辑此规则。
4 从 " 输入名称 " 字段中删除 [输入所需特性的名称]。
5 单击 " 浏览 " 图标，然后浏览至创建用户对象所需的特性并选择该特性。
6 （可选）如果需要多个必需特性，则单击加号图标 添加新操作。
7 选择 " 操作特性不可用时禁止 "，然后再浏览至其它必需特性。
8 单击 " 确定 "。



该规则的运行逻辑

如果业务流程需要在创建目标用户对象前，用户在源用户对象中填充了特定特性，则使用此规则。如果在源数据存储区中创建了一个用户对象，除非在创建用户对象时提供了必需特性，否则该规则将禁止在目标数据存储区中创建该对象。可以有一个或多个必需特性。

创建 - 发布者 - 使用模板

在创建用户对象时，允许使用 Novell eDirectory 模板对象。对驱动程序中的发布者创建策略实施该规则。

使用预定义规则包含两个步骤：在创建策略集中创建策略，然后导入预定义规则。如果要为已有的创建策略添加此规则，请转到 “导入预定义规则”在第 211 页 。

创建策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 在发布者通道或订购者通道中单击创建策略对象。
3 单击 " 插入 "。
4 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

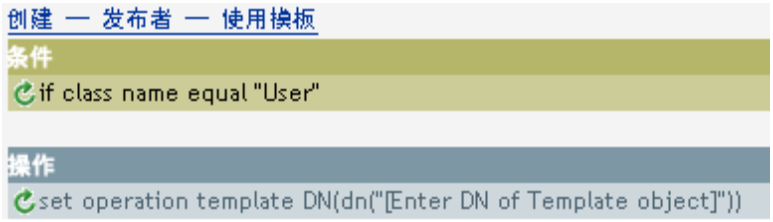
1 在规则构建器中，单击 " 插入 "。
2 选择 " 创建 - 发布者 - 使用模板 "。
3 在规则构建器中，单击 " 创建 - 发布者 - 使用模板 "，编辑此规则。
4 从 " 输入 DN" 字段删除 [输入模板对象的 DN]。

5 单击 " 编辑自变量 " 图标，起动自变量构建器。

6 在 " 名词 " 列表中选择 " 文本 "，然后单击 " 添加 "。

7 在编辑器中单击 " 浏览 " 图标，浏览至模板对象并选择此对象，然后单击 " 确定 "。

8 单击 " 确定 "。



该规则的运行逻辑

如果要在 Identity Vault 中基于模板对象创建用户，则使用此规则。如果具有与用户必需特性相同的特性，则使用模板可节省时间。只需填写模板对象中的信息，然后在创建用户对象时， Identity Manager 即可使用模板中的特性值创建该用户对象。

在创建用户对象期间，该规则执行 " 设置操作模板 DN" 操作，该模板指示 Identity Manager 在创建对象时要使用参考模板。

创建 - 设置默认特性值

您可以为创建用户对象时所指派的特性设置默认值。对驱动程序中的订购者创建策略或发布者创建策略实施此规则。

使用预定义规则包含两个步骤：在创建策略集中创建策略，然后导入预定义规则。如果要为已有的创建策略添加此规则，请转到 “导入预定义规则”在第 212 页 。

创建策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。

2 在发布者通道或订购者通道中单击创建策略对象。

3 单击 " 插入 "。

4 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1 在规则构建器中，单击 " 插入 "。

2 选择 " 创建 - 设置默认特性值 "。

3 在规则构建器中，单击 " 设置默认特性值 "，编辑此规则。

4 从 " 输入特性名称 " 字段中删除 [输入特性名]。

5 单击 " 浏览 " 图标，然后浏览至要创建的特性并选择此特性。

6 从 " 输入自变量值 " 字段中删除 [输入默认特性值]。

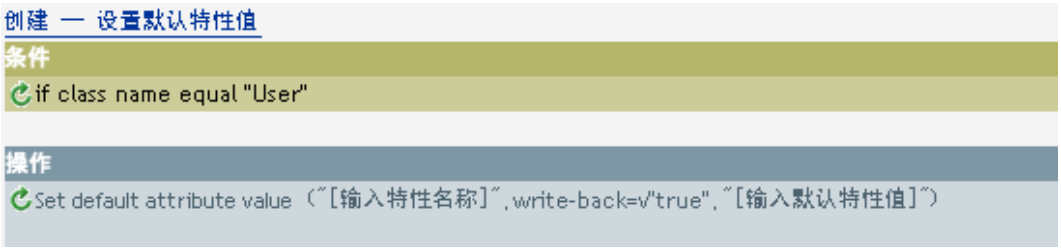
7 单击 " 编辑自变量 " 图标以起动自变量值列表构建器。

8 选择值的数据类型。

9 单击 " 编辑自变量 " 图标，起动自变量构建器。

10 通过自变量构建器创建特性值，然后单击 " 确定 "。

11 单击 " 确定 "。



该规则的运行逻辑

如果要在创建用户对象时填充默认特性值，则使用此规则。仅在创建用户对象时，源对象未提供该特性值时，该规则添加指定特性值。

如果要定义多个特性值，则右键单击该操作，然后单击 " 新建 ">" 操作 "。选择此操作，设置默认特性值，然后执行上述步骤，将值指派给该特性。

创建 - 设置默认口令

创建用户对象时，该规则为用户对象设置了一个默认口令。对驱动程序中的订购者创建策略或发布者创建策略实施此规则。

使用预定义规则包含两个步骤：在创建策略集中创建策略，然后导入预定义规则。如果要为已有的创建策略添加此规则，请转到 “导入预定义规则”在第 213 页 。

创建策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。

2 在发布者通道或订购者通道中单击创建策略对象。

3 单击 " 插入 "。

4 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

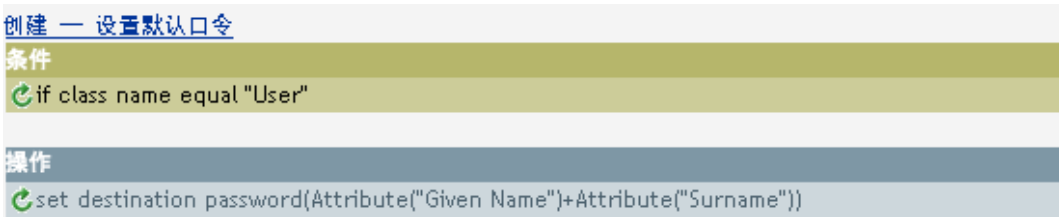
导入预定义规则

1 在规则构建器中，单击 " 插入 "。

2 选择 " 创建 - 设置默认口令 "。

3 单击 " 确定 "。

在特定于个人环境的规则中没有要更改的信息。



该规则的运行逻辑

如果要使创建的用户对象具有默认口令，则使用此规则。创建用户对象时，为用户对象设置的口令是用户对象的 Given Name （名）特性加上 Surname （姓氏）特性。

可以通过编辑自变量更改默认口令的值。使用自变量构建器可以将口令设置为所需的任意其它值。

事件转换 - 范围过滤 - 包括子树

排除发生在特定子树外的所有事件。对驱动程序中的订购者事件转换策略或发布者事件转换策略实施此规则。

使用预定义规则包含两个步骤：在事件转换策略集中创建策略，然后导入此预定义规则。如果要为已有的事件转换策略添加此规则，请转到 “导入预定义规则”在第 214 页 。

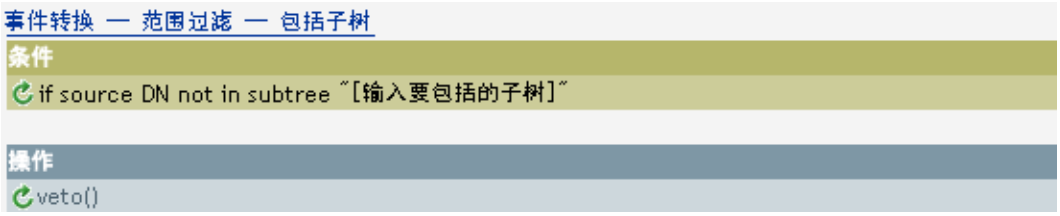
创建策略

1. 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2. 在发布者通道或订购者通道中单击 " 事件转换策略 " 对象。
3. 单击 " 插入 "。
4. 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1. 在规则构建器中，单击 " 插入 "。
2. 选择 " 事件转换 - 范围过滤 - 包括子树 "。
3. 在规则构建器中，单击 " 事件转换 - 范围过滤 - 包括子树 "，编辑此规则。
4. 删除 " 值 " 字段中的 [输入要包括的子树]。
5. 单击 " 浏览 " 按钮，在 Identity Vault 中浏览至树中要同步事件的部分，然后单击 " 确定 "。
6. 单击 " 确定 "。



该规则的运行逻辑

如果仅想在 Identity vault 和已连接系统之间同步特定子树，则使用此规则。如果事件发生在 Identity Vault 特定部分之外的其它地方，则禁用此规则。可以通过复制和粘贴 “If Source DN”在第 242 页 条件，添加要同步的附加子树。

事件转换 - 范围过滤 - 排除子树

排除发生在特定子树中的所有事件。对驱动程序中的订购者事件转换策略或发布者事件转换策略实施此规则。

使用预定义规则包含两个步骤：在事件转换策略集中创建策略，然后导入此预定义规则。如果要为已有的事件转换策略添加此规则，请转到 “导入预定义规则”在第 215 页 。

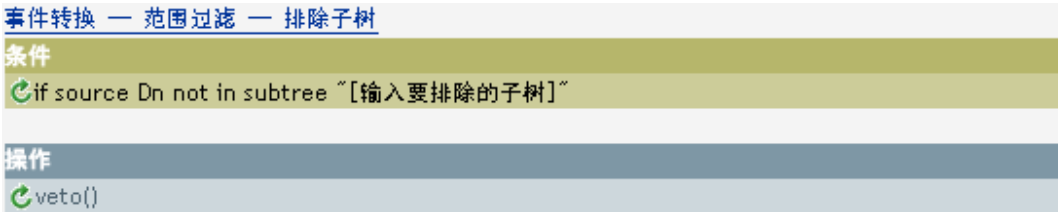
创建策略

1. 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2. 在发布者通道或订购者通道中单击 " 事件转换策略 " 对象。
3. 单击 " 插入 "。
4. 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1. 在规则构建器中，单击 " 插入 "。
2. 选择 " 事件转换 - 范围过滤 - 排除子树 "。
3. 在规则构建器中，单击 " 事件转换 - 范围过滤 - 排除子树 "，编辑此规则。
4. 删除 " 值 " 字段中的 [输入要排除的子树]。
5. 单击 " 浏览 " 按钮，在 Identity Vault 中浏览至不希望同步事件的树部分，然后单击 " 确定 "。
6. 单击 " 确定 "。



该规则的运行逻辑

如果不想对 Identity Vault 或已连接系统中的部分进行同步，则使用此规则。如果事件发生在 Identity Vault 的特定部分，则禁用此规则。可以通过复制和粘贴 If Source DN 条件添加要排除的附加子树。

输入或输出转换 - 将电话号码格式重新从 **(nnn) nnn-nnnn** 设置为 **nnn-nnn-nnnn**

转换电话号码的格式。对驱动程序中的输入或输出转换策略实施该规则。通常，如果将此规则用于输入转换，则需要随后使用此规则在输出转换中将电话号码的格式由 nnn-nnn-nnnn 重设为 (nnn) nnn-nnnn，反之亦然，可以在两种格式间自由转换。

使用预定义规则包含两个步骤：在输入或输出转换策略集中创建策略，然后导入预定义规则。如果要为已有的输入或输出转换策略添加此规则，请转到 “导入预定义规则”在第 216 页 。

创建策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 在发布者或订购者通道中单击 " 输入或输出转换策略 " 对象。
3 单击 " 插入 "。
4 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1 在规则构建器中，单击 " 插入 "。
2 选择 " 输入或输出转换 - 将电话号码格式重新从 (nnn) nnn-nnnn 设置为 nnn-nnn-nnnn"。
3 在规则构建器中单击 " 输入或输出转换 - 将电话号码格式重新从 (nnn) nnn-nnnn 设置为 nnn-nnn-nnnn"，编辑此规则。
4 定义重设电话号码格式时希望出现的条件。
5 单击 " 确定 "。

该规则的运行逻辑

希望重设电话号码的格式时将使用该规则。该规则将在当前操作中查找所有与 (nnn) nnn-nnnn 模式相匹配的 phone （电话）特性的值，并用 nnn-nnn-nnnn 逐一进行替换。

输入或输出转换 **-** 将电话号码格式重新从 **nnn-nnn-nnnn** 设置为 **(nnn) nnn-nnnn**

转换电话号码的格式。对输入或输出转换策略实施该规则。通常，如果对输出转换使用此规则，则将对输入转换使用 " 将电话号码的格式从 (nnn) nnn-nnnn 重设为 nnn-nnn-nnnn" 规则；反之亦然，可以在两种格式间自由转换。

使用预定义规则包含以下两个步骤：在输入或输出转换策略集中创建一个策略，然后导入预定义规则。如果要为已有的输入或输出转换策略添加此规则，请转到 “导入预定义规则” 在第 217 页 。

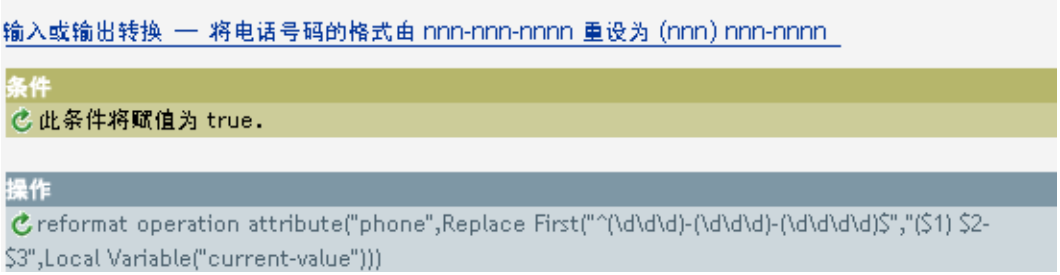
创建策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 在发布者或订购者通道中单击 " 输入或输出转换策略 " 对象。
3 单击 " 插入 "。
4 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1 在规则构建器中，单击 " 插入 "。
2 选择 " 输入或输出转换 - 将电话号码格式重新从 nnn-nnn-nnnn 设置为 (nnn) nnn-nnnn"。
3 在规则构建器中，单击 " 输入或输出转换 - 将电话号码格式重新从 nnn-nnn-nnnn 设置为 (nnn) nnn-nnnn"，以编辑该规则。
4 定义重设电话号码格式时希望出现的条件。
5 单击 " 确定 "。



该规则的运行逻辑

希望重设电话号码的格式时将使用该规则。该规则将在当前操作中查找所有与 (nnn) nnn-nnnn 模式相匹配的 phone （电话）特性的值，并用 nnn-nnn-nnnn 逐一进行替换。

匹配 - 已镜像的发布者

根据已连接系统中的对象的名称和位置，在 Identity Vault 中查找这些对象的匹配项。对驱动程序中的发布者匹配策略实施该规则。

使用预定义规则包含两个步骤：在匹配策略集中创建一个策略，然后导入预定义规则。如果要为已有的匹配策略添加此规则，请转到 “导入预定义规则”在第 217 页 。

创建策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 在发布者通道上，单击 " 匹配策略 " 对象。
3 单击 " 插入 "。
4 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

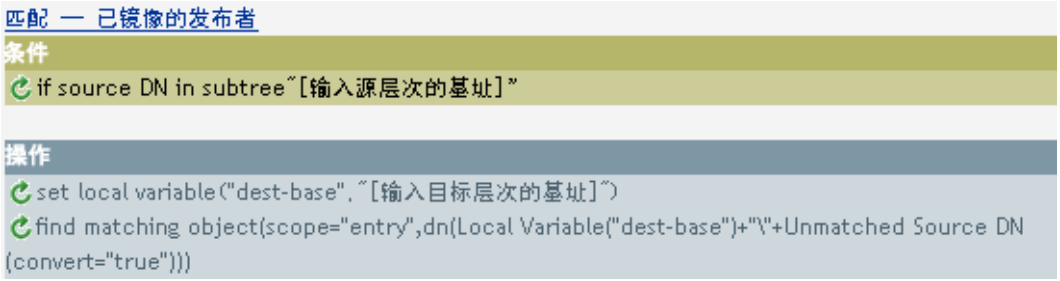
1 在规则构建器中，单击 " 插入 "。
2 选择 " 匹配 - 已镜像的发布者 "。
3 在规则构建器中，单击 " 匹配 - 已镜像的发布者 "，以编辑该规则。
4 从 " 值 " 字段删除 [输入源层次的基础]。
5 浏览至源层次中希望开始匹配的树枝，然后单击 " 确定 "。
6 单击 " 确定 "。
7 从 " 输入字符串 " 字段删除 [输入目标层次的基础]。

8 单击 " 编辑自变量 " 图标，起动自变量构建器。

9 在 " 名词 " 列表中选择 " 文本 "，然后单击 " 添加 "。

10 在编辑器中单击 " 浏览 " 图标，浏览至目标层次中希望在其中匹配源结构的树枝，并选择该树枝，然后单击 " 确定 "。

11 单击 " 确定 "。



该规则的运行逻辑

如果已连接系统中位于指定的源子树中的对象发生 Add 事件，该规则将构造一个 DN，表示 Identity Vault 中与指定的目标子树相关的相同对象名称和位置。如果目标对象存在且属于所需的对象类，则认为该对象是匹配项。必须提供源 （已连接系统）子树和目标 (Identity Vault) 子树的 DN。

匹配 **-** 已镜像的订购者 **- LDAP** 格式

根据 Identity Vault 中的对象的名称和位置，在已连接系统中查找这些对象的匹配项，该匹配项使用 LDAP 格式的 DN。对驱动程序中的订购者匹配策略实施该规则。

使用预定义规则包含两个步骤：在匹配策略集中创建一个策略，然后导入预定义规则。如果要为已有的匹配策略添加此规则，请转到 “导入预定义规则”在第 218 页 。

创建策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。

2 在订购者通道上，单击 " 匹配策略 " 对象。

3 单击 " 插入 "。

4 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1 在规则构建器中，单击 " 插入 "。

2 选择 " 匹配 - 已镜像的订购者 - LDAP 格式 "。

3 在规则构建器中，单击 " 匹配 - 已镜像的订购者 - LDAP 格式 "，以编辑该规则。

4 从 " 值 " 字段删除 [输入源层次的基础]。

5 浏览至源层次中希望开始匹配的树枝，然后单击 " 确定 "。

6 单击 " 确定 "。

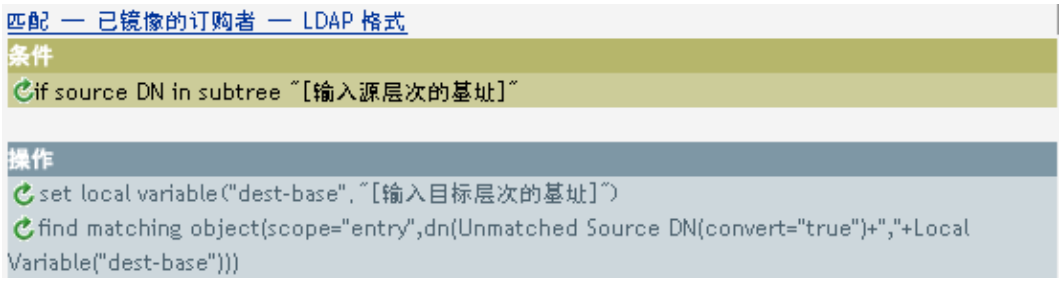
7 从 " 输入字符串 " 字段删除 [输入目标层次的基础]。

8 单击 " 编辑自变量 " 图标，起动自变量构建器。

9 在 " 名词 " 列表中选择 " 文本 "，然后单击 " 添加 "。

10 在编辑器中单击 " 浏览 " 图标，浏览至目标层次中希望在其中匹配源结构的树枝，并选择该树枝，然后单击 " 确定 "。

11 单击 " 确定 "。



该规则的运行逻辑

在 Identity Vault 中如果位于指定的源子树中的对象发生添加事件，则该规则将构造 DN，表示与已连接系统中指定的目标子树相关的相同对象的名称和位置。如果目标对象存在且属于所需的对象类，则认为该对象是匹配项。必须提供源 (Identity Vault) 子树和目标 （已连接系统）子树的 DN。已连接系统必须使用 LDAP 格式的 DN。

匹配 - 按特性值

按指定的特性值查找对象的匹配项。对驱动程序中的订购者匹配策略或发布者匹配策略实施该规则。

使用预定义规则包含以下两个步骤：在匹配策略集中创建策略，然后导入预定义规则。如果要为已有的匹配策略添加此规则，请转到 “导入预定义规则”在第 219 页 。

创建策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。

2 在发布者通道上，单击 " 匹配策略 " 对象。

3 单击 " 插入 "。

4 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1 在规则构建器中，单击 " 插入 "。

2 选择 " 匹配 - 按特性值 "。

3 在规则构建器中，单击 " 匹配 - 按特性值 "，以编辑该规则。

4 从 " 输入 DN" 字段中删除 [输入要开始搜索的基本 DN]。

5 单击 " 编辑自变量 " 图标，起动自变量构建器。

6 在 " 名词 " 列表中选择 " 文本 "，然后单击 " 添加 "。

7 在编辑器中单击 " 浏览 " 图标，浏览至要在其中开始搜索的树枝，并选择该树枝，然后单击 " 确定 "。

8 从 " 输入匹配特性 " 字段删除 [输入特性名称以进行匹配]。

9 单击 " 编辑自变量 " 图标以起动匹配特性构建器。

10 单击 " 浏览 " 图标并选择希望匹配的特性。可以选择一个或多个要匹配的特性，然后单击 " 确定 "。

11 单击 " 确定 "。

该规则的运行逻辑

源数据存储区中的对象上发生添加事件时，该规则将在目标数据存储区中搜索具有与指定特性相同的值的对象。必须提供在已连接系统中进行搜索的子树基址的 DN 和要匹配的特性名称。

布局 - 已镜像的发布者

根据已连接系统中对象的名称和位置，在 Identity Vault 中放置对象。对驱动程序中的发布者布局策略实施该规则。

使用预定义规则包含两个步骤：在布局策略集中创建策略，然后导入预定义规则。如果您希望在现有的布局策略中添加此规则，请跳至 “导入预定义规则”在第 220 页 。

创建策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。

2 在发布者通道上，单击 " 布局策略 " 对象。

3 单击 " 插入 "。

4 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1 在规则构建器中，单击 " 插入 "。

2 选择 " 布局 - 已镜像的发布者 "。

3 在规则构建器中，单击 " 布局 - 已镜像的发布者 " 以编辑该规则。

4 从 " 值 " 字段删除 [输入源层次的基础]。

5 浏览至源层次中希望处理对象的树枝，并选择该树枝，然后单击 " 确定 "。

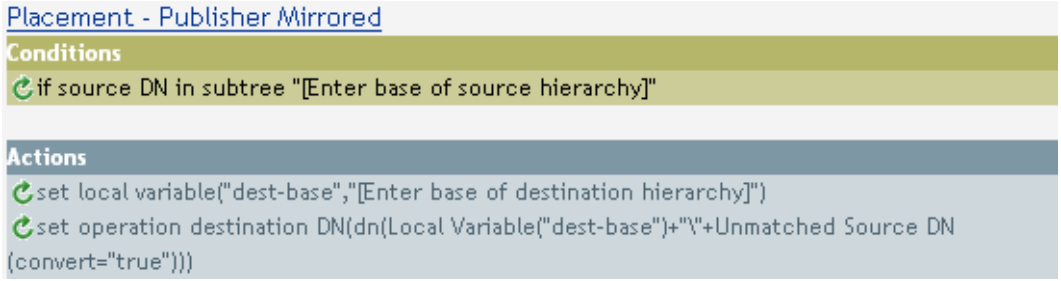
6 从 " 输入字符串 " 字段删除 [输入目标层次的基础]。

7 单击 " 编辑自变量 " 图标，起动自变量构建器。

8 在 " 名词 " 列表中选择 " 文本 "，然后单击 " 添加 "。

9 在编辑器中单击 " 浏览 " 图标，浏览至目标层次中希望在其中放置该对象的树枝，并选择该树枝，然后单击 " 确定 "。

10 单击 " 确定 "。



该规则的运行逻辑

如果用户对象位于已连接系统中指定的源子树中，则以相同的相对名称和位置将对象放置在 Identity Vault 中。必须提供源 （已连接系统）子树和目标 (Identity Vault) 子树的 DN。

布局 - 已镜像的订购者 - **LDAP** 格式

使用 Identity Vault 中特定点处的已镜像结构将对象放置在数据存储区中。对驱动程序中的布局策略实施该规则。只能在订购者通道上实施该规则。

使用预定义规则包含两个步骤：在布局策略集中创建策略，然后导入预定义规则。如果要为已有的布局策略添加此规则，请转到 “导入预定义规则”在第 221 页 。

创建策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。

2 在订购者通道上，单击 " 布局策略 " 对象。

3 单击 " 插入 "。

4 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1 在规则构建器中，单击 " 插入 "。

2 选择 " 布局 - 已镜像的订购者 - LDAP 格式 "。

3 在规则构建器中单击 " 布局 - 已镜像的订购者 - LDAP 格式 " 以编辑该规则。

4 从 " 值 " 字段删除 [输入源层次的基础]。

5 浏览至源层次中希望处理对象的树枝，并选择该树枝，然后单击 " 确定 "。

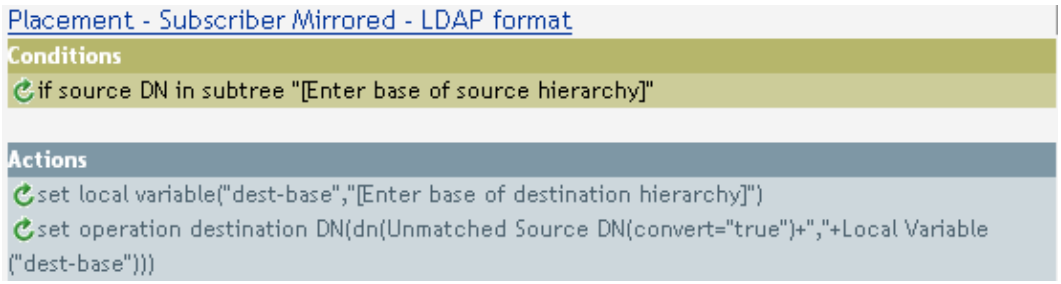
6 从 " 输入字符串 " 字段删除 [输入目标层次的基础]。

7 单击 " 编辑自变量 " 图标，起动自变量构建器。

8 在 " 名词 " 列表中选择 " 文本 "，然后单击 " 添加 "。

9 在编辑器中单击 " 浏览 " 图标，浏览至目标层次中希望在其中放置该对象的树枝，并选择该树枝，然后单击 " 确定 "。

10 单击 " 确定 "。



该规则的运行逻辑

如果用户对象位于指定的源子树中，则将对象以相同的相对名称和位置放置在 Identity Vault 中。必须提供源 (Identity Vault) 子树和目标 （已连接系统）子树的 DN。已连接系统必须使用 LDAP 格式的 DN。

布局 - 发布者平面文件

将对象从数据存储区放置到 Identity Vault 中的某个树枝中。对驱动程序中的发布者布局策略实施该规则。

使用预定义规则包含两个步骤：在布局策略集中创建策略，然后导入预定义规则。如果要为已有的布局策略添加此规则，请转到 “导入预定义规则”在第 222 页 。

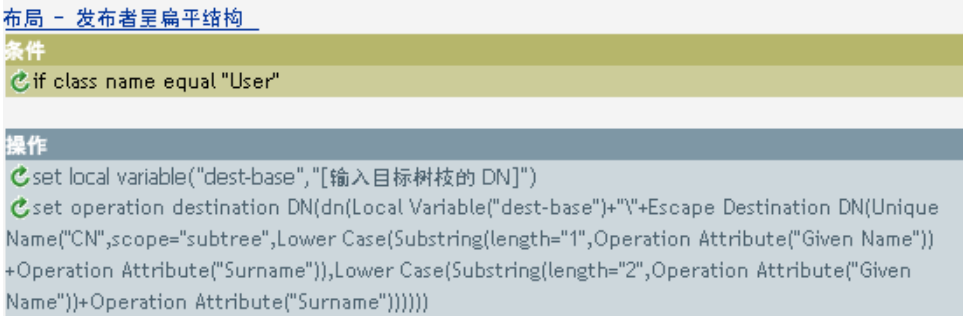
创建策略

1. 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2. 在发布者通道上，单击 " 布局策略 " 对象。
3. 单击 " 插入 "。
4. 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1. 在规则构建器中，单击 " 插入 "。
2. 选择 " 布局 - 发布者平面文件 "。
3. 在规则构建器中，单击 " 布局 - 发布者平面文件 " 以编辑该规则。
4. 从 " 输入字符串 " 字段删除 [输入目标树枝的 DN]。
5. 单击 " 编辑自变量 " 图标，起动自变量构建器。
6. 在 " 名词 " 列表中选择 " 文本 "，然后单击 " 添加 "。
7. 在编辑器中单击 " 浏览 " 图标，浏览至希望在其中放置所有用户对象的目标树枝，并选择该树枝，然后单击 " 确定 "。
8. 单击 " 确定 "。



该规则的运行逻辑

该规则将所有用户对象放置在目标 DN 中。该规则将目标树枝的 DN 设置为局部变量 dest-base。然后该规则将目标 DN 设置为 dest-base\CN 特性。用户对象的 CN 特性是 Given Name（名）特性的前两个字母加上 Surname （姓氏）特性，均为小写。该规则使用斜线格式。

布局 **-** 订购者平面文件 **- LDAP** 格式

将 Identity Vault 中的对象放置到数据存储区的某个树枝中。对驱动程序中的订购者布局策略实施该规则。

使用预定义规则包含两个步骤：在布局策略集中创建策略，然后导入预定义规则。如果要为已有的布局策略添加此规则，请转到 “导入预定义规则”在第 223 页 。

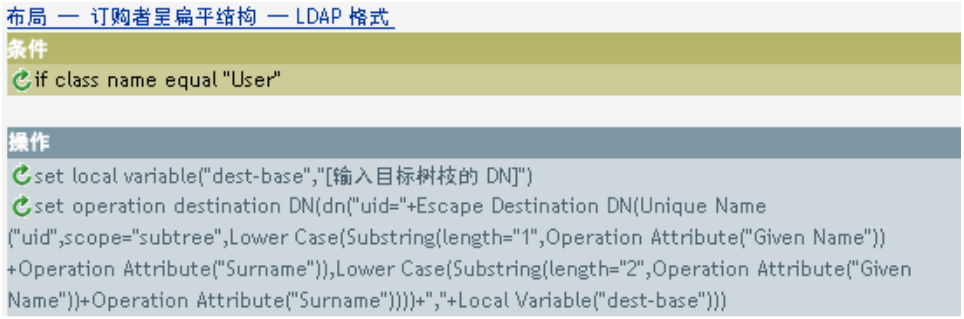
创建策略

1. 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2. 在订购者通道上，单击 " 布局策略 " 对象。
3. 单击 " 插入 "。
4. 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1. 在规则构建器中，单击 " 插入 "。
2. 选择 " 布局 - 订购者平面文件 - LDAP 格式 "。
3. 在规则构建器中，单击 " 布局 - 订购者平面文件 - LDAP 格式 " 以编辑该规则。
4. 从 " 输入字符串 " 字段删除 [输入目标树枝的 DN]。
5. 单击 " 编辑自变量 " 图标，起动自变量构建器。
6. 在 " 名词 " 列表中选择 " 文本 "，然后单击 " 添加 "。
7. 在编辑器中，添加要在其中放置所有用户对象的目标树枝。确保将该树枝指定为 LDAP 格式，然后单击 " 确定 "。
8. 单击 " 确定 "。



该规则的运行逻辑

该规则将所有用户对象放置在目标 DN 中。该规则将目标树枝的 DN 设置为局部变量 dest-base。然后将目标 DN 设置为 uid=unique name (dest-base)。用户对象的 uid 特性是 Given Name （名）特性的前两个字母加上 Surname （姓氏）特性，均为小写。该规则使用 LDAP 格式。

布局 - 部门发布者

根据 OU 特性的值，将对象从数据存储区中一个树枝放入 Identity Vault 中的多个树枝。对驱动程序中的发布者布局策略实施该规则。

使用预定义规则包含两个步骤：在布局策略集中创建策略，然后导入预定义规则。如果要为已有的布局策略添加此规则，请转到 “导入预定义规则”在第 224 页 。

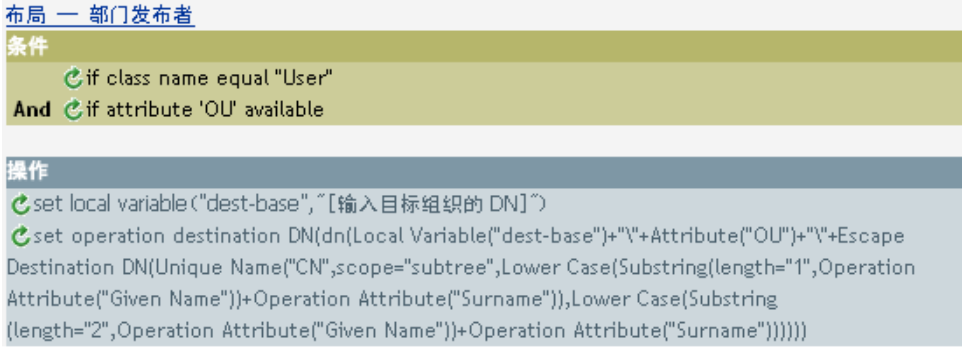
创建策略

1. 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2. 在发布者通道上，单击 " 布局策略 " 对象。
3. 单击 " 插入 "。
4. 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

将起动规则构建器。

导入预定义规则

1. 在规则构建器中，单击 " 插入 "。
2. 选择 " 布局 - 部门发布者 "。
3. 单击 " 布局 - 部门发布者 " 以编辑该规则。
4. 从 " 输入字符串 " 字段删除 [输入目标组织的 DN]。
5. 单击 " 编辑自变量 " 图标，起动自变量构建器。
6. 在 " 名词 " 列表中选择 " 文本 "，然后单击 " 添加 "。
7. 在编辑器中单击 " 浏览 " 图标，浏览至 Identity Vault 中的父树枝，并选择该树枝。确保所有部门树枝都是该 DN 的子树枝，然后单击 " 确定 "。
8. 单击 " 确定 "。



该规则的运行逻辑

该规则将用户对象放置在适当的部门树枝中，具体取决于储存在 OU 特性中的值。如果需要放置用户对象，该对象也具有可用的 OU 特性，则将用户对象放置在 dest-base\OU 特性的值 \CN 特性中。

dest-base 是一个局部变量。该 DN 必须是部门树枝的相对根路径。它可以是组织或组织单元。储存在 OU 特性中的值必须是局部变量 dest-base 的子树枝名称。

OU 特性的值必须是子树枝的名称。如果该 OU 特性不存在，则不执行该规则。

用户对象的 CN 特性是 Given Name （名）特性的前两个字母加上 Surname （姓氏）特性，均为小写。该规则使用斜线格式。

布局 **-** 部门订购者 **- LDAP** 格式

根据 OU 特性的值，将对象从 Identity Vault 中一个树枝放置到数据存储区中的多个树枝。对驱动程序中的布局策略实施该规则。只能在订购者通道上实施该规则。

使用预定义规则包含两个步骤：在布局策略集中创建策略，然后导入预定义规则。如果要为已有的布局策略添加此规则，请转到 “导入预定义规则”在第 225 页 。

创建策略

1. 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2. 在订购者通道上，单击 " 布局策略 " 对象。
3. 单击 " 插入 "。
4. 命名此策略，并确保使用策略构建器实施此策略，然后单击 " 确定 "。

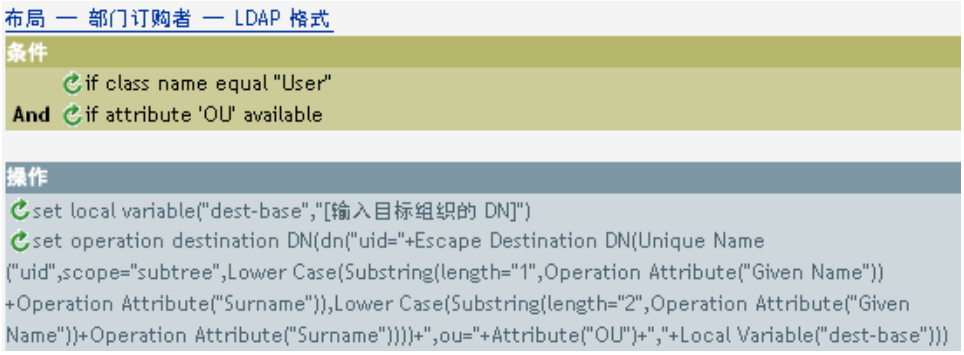
将起动规则构建器。

导入预定义规则

1. 在规则构建器中，单击 " 插入 "。
2. 选择 " 布局 - 部门订购者 - LDAP 格式 "。
3. 在规则构建器中，单击 " 布局 - 部门订购者 - LDAP 格式 " 以编辑该规则。
4. 从 " 输入字符串 " 字段删除 [输入目标组织的 DN]。
5. 单击 " 编辑自变量 " 图标，起动自变量构建器。
6. 在 " 名词 " 列表中选择 " 文本 "，然后单击 " 添加 "。

7 在编辑器中，将父树枝添加到数据存储区中。必须将该父树枝指定为 LDAP 格式。确保所有部门树枝都是该 DN 的子树枝，然后单击 " 确定 "。

8 单击 " 确定 "。



该规则的运行逻辑

该规则将用户对象放置在适当的部门树枝中，具体取决于储存在 OU 特性中的值。如果需要放置某用户对象并且其 OU 特性可用，则将该用户对象放置到 dest-base (uid=unique name,ou=value of OU attribute)。

dest-base 是一个局部变量。该 DN 必须是部门树枝的相对根路径。它可以是组织或组织单元。储存在 OU 特性中的值必须是局部变量 dest-base 的子树枝名称。

OU 特性的值必须是子树枝的名称。如果该 OU 特性不存在，将不执行该规则。

用户对象的 uid 特性是 Given Name （名）特性的前两个字母加上 Surname （姓氏）特性，均为小写。该规则使用 LDAP 格式。

3.3 正则表达式

正则表达式是匹配遵循某种模式的文本字符串的公式。正则表达式由普通字符和元字符组成。普通字符包括大写字母、小写字母和数字。元字符则具有特殊的含义。下表包含一些最常用的元字符及其含义。

表 ***3-1*** 常用正则表达式

元字符	说明
.	与任意单个字符相匹配。
$	匹配行尾。
^	匹配行首。
*	匹配之前出现了零次或多次的某一字符。
\	文字转义符。允许搜索任意元字符。例如，\$ 会查找 $1000，而不是查找与行的结尾相匹配的字符。
[ ]	与方括号中的任一字符相匹配。
[0-9]	匹配用连字符连接的一组字符。此示例将匹配所有数字。

元字符	说明
[A-Za-z]	同时匹配多个范围。此示例将匹配所有大写和小写字母。

自变量构建器旨在使用 Java* 中定义的正则表达式。有关详细信息，请访问 Java 万维网站点 (http://java.sun.com/j2se/1.4.2/docs/api/java/util/regex/Pattern.html)。

3.4 XPath 1.0 表达式

某些条件、操作和标记的自变量使用 XPath 1.0 表达式。 XPath 是一种语言，可以为 XSLT 和 XPointer 之间共享的功能提供通用的语法和语义。它主要用于 XML 文档的寻址部分，但也提供了处理字符串、数字和布尔值的基本工具。

XPath 规范要求嵌入的应用程序提供的环境应包括由应用程序定义的若干信息。在 DirXML 底稿 （请参见 “DirXML 底稿”在第 11 页 ）中， XPath 在以下环境中求值：

- 环境节点为当前操作。
- 环境位置和大小均为 1。
- 可用变量
  - 以下变量可用作 Identity Manager 中样式表的参数 （当前为 fromNDS、srcQueryProcessor、 destQueryProcessor、 srcCommandProcessor、destCommandProcessor 和 dnConverter）。
  - 全局配置变量。
  - 局部策略变量。
  - 如果不同的变量源之间存在名称冲突，则优先顺序为局部变量、样式表参数、全局变量。
- 在策略要素中声明的名称空间。
- 可用函数
  - 所有内置 XPath 1.0 函数
  - NXSL 提供的 Java 扩展函数
    - 必须在策略要素上声明将 Java 类与前缀相关联的名称空间声明。

有关详细信息，请访问 W3 万维网站点 (http://www.w3.org/TR/1999/REC-xpath-19991116)。

3.5 条件

本节包含有关使用 " 策略构建器 " 界面时的所有可用条件的详细参考信息。

- “If Local Variable”在第 235 页
- “If Named Password”在第 237 页
- “If Operation”在第 237 页
- “If Operation Attribute”在第 238 页
- “If Operation Property”在第 240 页
- “If Password”在第 240 页
- “If Source Attribute”在第 241 页
- “If Source DN”在第 242 页
- “If XPath Expression”在第 244 页

3.5.1 If Association

对当前操作或当前对象的关联值进行测试。

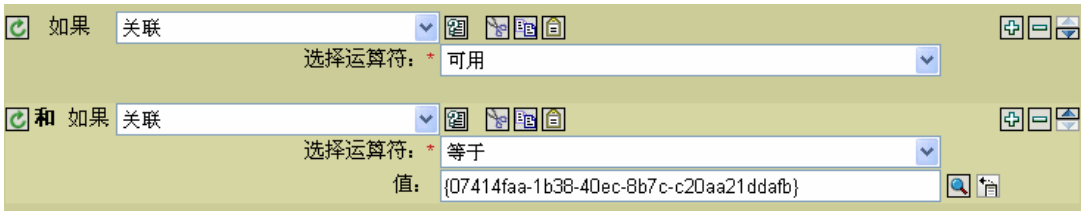
字段

操作符与满足条件的情形

操作符	满足条件的情形
associated	当前对象已建立一个关联。
available	存在由当前操作指定的非空关联值。
equal	当前操作指定的关联值与 If Association 的内容完全相同。
not-associated	当前对象未建立关联。
not available	与当前对象间的关联不可用。
not-equal	当前操作指定的关联值不等于 If Association 的内容。

示例

本示例测试关联是否可用。如果满足条件，将执行所定义的操作。



3.5.2 If Attribute

对当前操作或源数据存储区中的当前对象的特性值进行测试。此条件从逻辑上可以视为 If Operation Attribute 或 If Source Attribute，因为只有在源数据存储区或操作中满足条件此测试才能成功。

字段

名称

指定要测试的特性的名称。

操作符

选择此条件的测试类型。

比较方式

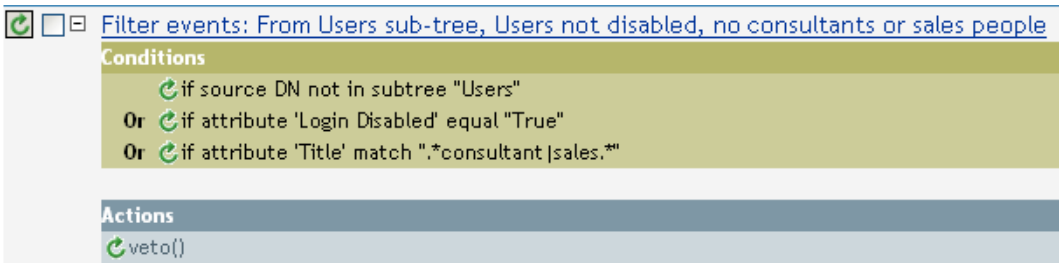
选择比较方式。请参见 “比较方式”在第 303 页 。

操作符与满足条件的情形

操作符	满足条件的情形
available	在当前操作或源数据存储区中存在一个指定特性的可用值。
equal	在当前操作或源数据存储区中，存在指定特性的可用值，它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

示例

下面的示例在过滤禁用的或具有特定职务的用户对象时使用 If Attribute 条件。此策略是 Policy to Filter Events （过滤事件策略），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。



此条件查找 Title （职务）特性的值为 consultant 或 sales 的所有用户对象。

3.5.3 If Class Name

对当前操作中的对象类名称进行测试。

字段

操作符

选择此条件的测试类型。

比较方式

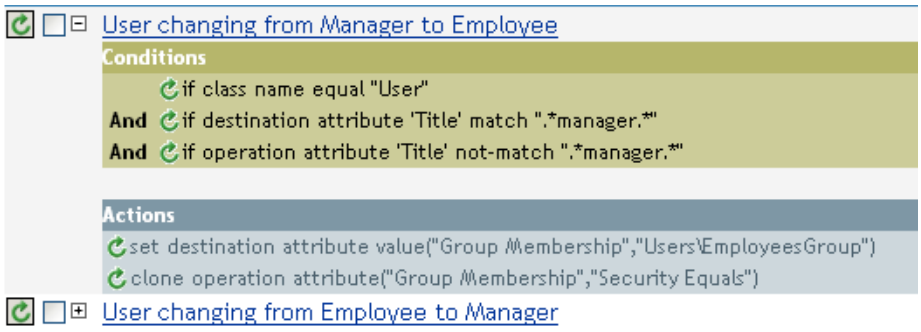
选择比较方式。请参见 “比较方式”在第 303 页 。

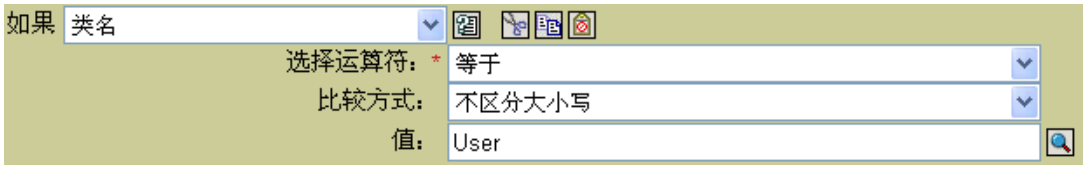
操作符与满足条件的情形

操作符	满足条件的情形
available	当前操作中有一个可用的对象类名称。
equal	当前操作中有一个可用的对象类名称，它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

示例

本示例将根据用户对象的职务使用 If Class Name 条件管理他们的组成员资格。此策略是 Govern Groups for User Based on Title Attribute （根据职务特性管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





检查当前对象的类名称是否为 User。

3.5.4 If Destination Attribute

对目标数据存储区中的当前对象的特性值进行测试。

字段

名称

指定要测试的特性的名称。

操作符

选择此条件的测试类型。

比较方式

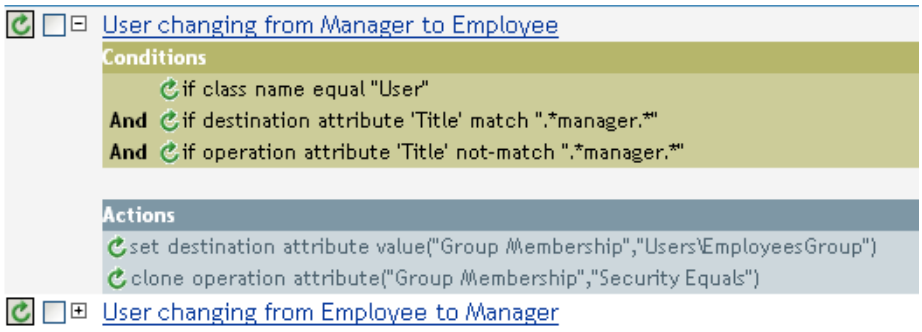
选择比较方式。请参见 “比较方式”在第 303 页。

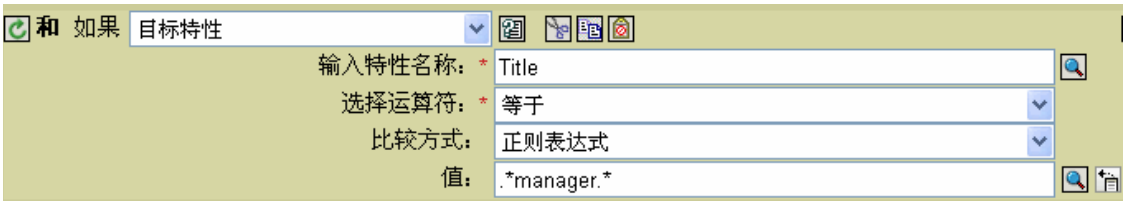
操作符与满足条件的情形

操作符	满足条件的情形
available	目标数据存储区中存在一个指定特性的可用值。
equal	在目标数据存储区中存在一个指定特性的可用值，它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

示例

本示例将根据用户对象的职务使用 If Attribute 条件管理他们的组成员资格。此策略是 Govern Groups for User Based on Title Attribute （根据职务特性管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页。





此策略检查 Title 特性的值是否包含 manager。

3.5.5 If Destination DN

对当前操作中的目标 DN 进行测试。执行的测试具体取决于指定的操作符。

字段

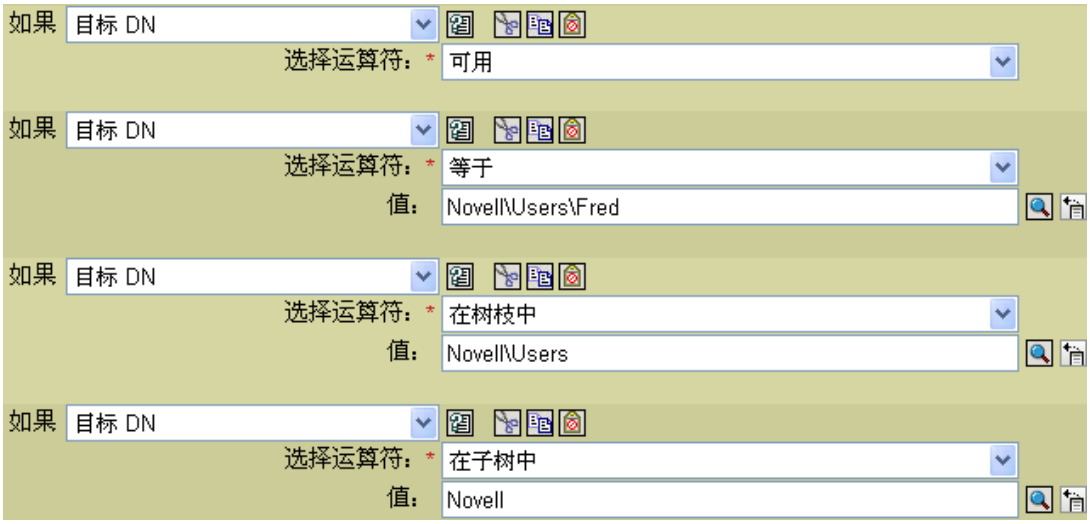
操作符

选择此条件的测试类型。

操作符与满足条件的情形

操作符	满足条件的情形
available	存在可用的目标 DN。
equal	存在可用的目标 DN，它等于使用适用于目标数据存储区的 DN 格式的语义进行比较时得到的值。
in-container	存在可用的目标 DN，当使用适用于目标数据存储区的 DN 格式的语义与该目标 DN 进行比较时，它表示由值指定的树枝中的某个对象。
in-subtree	存在可用的目标 DN，当使用适用于目标数据存储区的 DN 格式的语义与该目标 DN 进行比较时，它表示由值指定的子树中的某个对象。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。
not-in-container	若为 In-container，则将返回 False。
not-in-subtree	若为 In-subtree，则返回 False。

示例



3.5.6 If Entitlement

对当前操作或 Identity Vault 中当前对象的权利进行测试。

字段

名称

指定所选条件下要测试的权利的名称。

操作符

选择此条件的测试类型。

比较方式

选择比较方式。请参见 “比较方式”在第 303 页 。

操作符与满足条件的情形

操作符	满足条件的情形
available	命名权利在当前操作或 Identity Vault 中均可用。
changing	当前操作包含对命名权利的更改 （修改特性或添加特性）。
changing-from	当前操作包含的更改将去除命名权利的某个值 （去除值），该权利的一个值等于使用指定的比较方式进行比较时得到的值。
changing-to	当前操作包含向命名权利添加值 （添加值或添加特性）的更改。它的一个值等于使用指定的比较方式进行比较时得到的值。
equal	在目标数据存储区中存在一个指定特性的可用值，它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-changing	若为 Changing，则返回 False。
not-changing-from	若为 Changing-from，则返回 False。
not-changing-to	若为 Changing-to，则返回 False。
not-equal	如果等于，则返回 False。

示例

3.5.7 If Global Configuration Value

对全局配置变量进行测试。

字段

名称

指定所选条件下要测试的全局变量的名称。

操作符

选择此条件的测试类型。

比较方式

选择比较方式。请参见 “比较方式”在第 303 页 。

操作符与满足条件的情形

操作符	满足条件的情形
available	存在具有指定名称的全局配置变量。

操作符	满足条件的情形
equal	存在具有指定名称的全局配置变量，其值等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

示例

3.5.8 If Local Variable

对局部变量进行测试。

字段

名称

指定所选条件下要测试的局部变量的名称。

操作符

选择此条件的测试类型。

比较方式

选择比较方式。请参见 “比较方式”在第 303 页 。

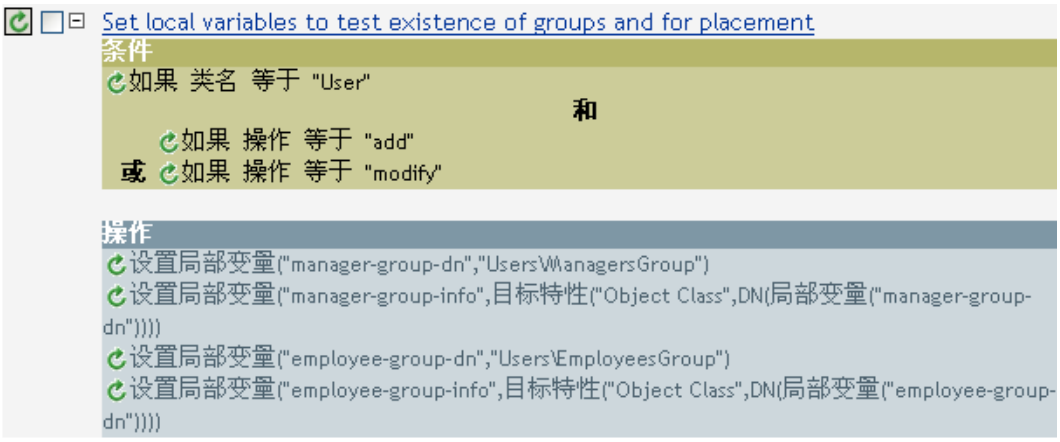
操作符与满足条件的情形

操作符	满足条件的情形
available	存在具有指定名称的局部变量，它由策略中早期规则的操作定义。
equal	存在具有指定名称的局部变量，其值等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

示例

下面的示例根据职务，向相应的 Employee 或 Manager 组中添加用户对象。在需要时，它也能创建组并设置与该组相对应的安全性。此策略是 Govern Groups for User Based on Title Attribute （根据职务特性管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。

此策略包含互相依存的五条规则。



为了使局部变量条件生效，第一条规则将设置四个不同的局部变量，用于测试组及组的位置。

该规则寻找的条件将会查看局部变量 manager-group-info 是否可用，以及 manager-group-info 是否与组不等效。如果这些条件都满足，将添加组的目标对象。

3.5.9 If Named Password

对当前操作中具有指定名称的口令进行测试。

字段

名称

指定所选条件下要测试的命名口令的名称。

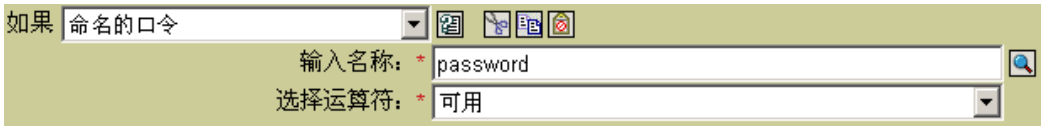
操作符

选择此条件的测试类型。

操作符与满足条件的情形

操作符	满足条件的情形
available	存在具有指定名称的口令。
not available	如果可用，则返回 False。

示例



3.5.10 If Operation

对当前操作的名称进行测试。

字段

操作符

选择此条件的测试类型。

操作符与满足条件的情形

操作符	满足条件的情形
equal	当前操作的名称与 If Operation 的内容完全相等。
not-equal	如果等于，则返回 False。

值

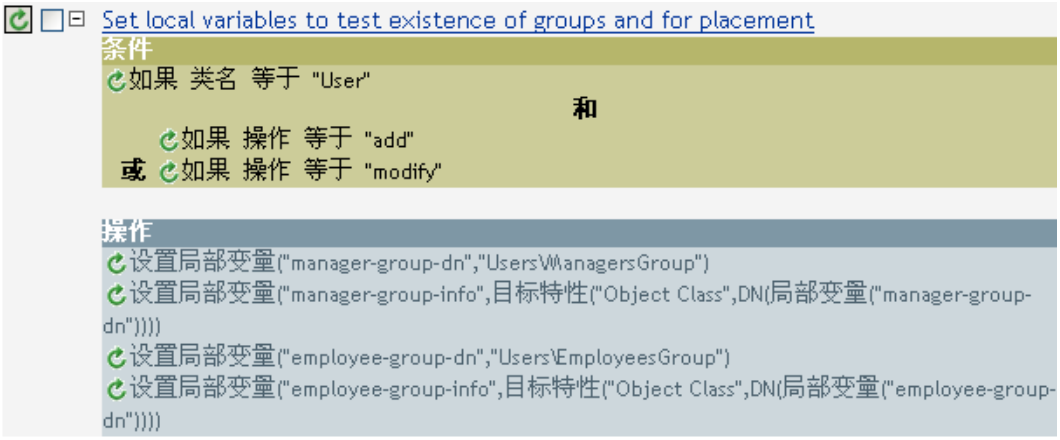
下列值是在此条件下 Metadirectory 引擎所搜索的操作：

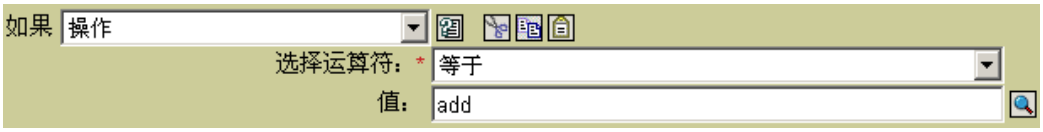
- add
- add-association
- check-object-password
- delete

- get-named-password
- modify
- modify-association
- modify-password
- move
- init-params
- instance

示例

下面的示例根据职务，向相应的 Employee 或 Manager 组中添加用户对象。在需要时，它也能创建组并设置与该组相对应的安全性。该策略名称为 Govern Groups for User Based on Title Attribute （根据职务特性管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





此条件检查是否已进行添加或修改操作。如果发生了这些操作之一，则此条件将设置局部变量。

3.5.11 If Operation Attribute

对当前操作中的特性值进行测试。执行的测试具体取决于指定的操作符。

字段

名称

指定要测试的特性的名称。

操作符

选择此条件的测试类型。

比较方式

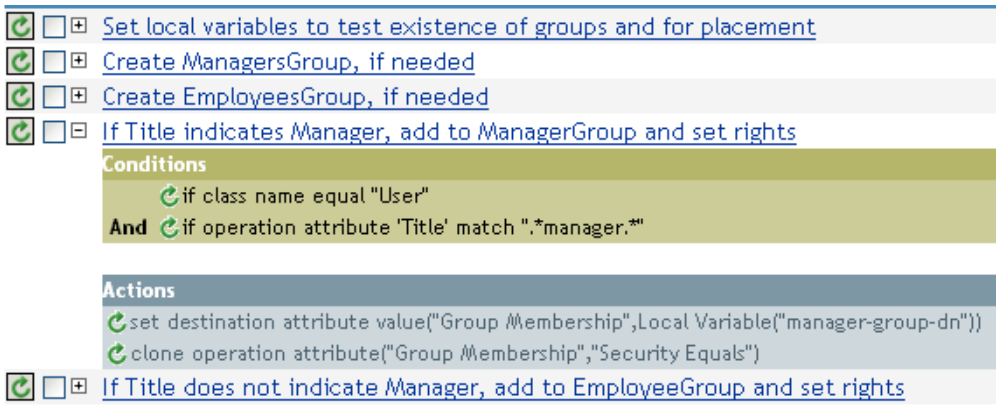
选择比较方式。请参见 “比较方式”在第 303 页 。

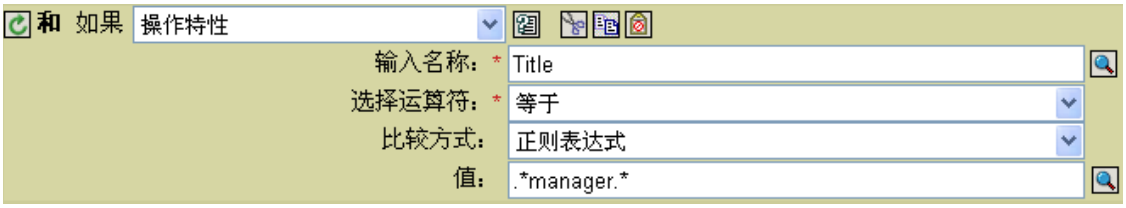
操作符与满足条件的情形

操作符	满足条件的情形
available	当前操作 （添加特性，添加值、特性）中存在可用于指定特性的值。
changing	当前操作包含对指定特性的更改 （修改特性或添加特性）。
changing-from	当前操作包含去除指定特性的某个值 （去除值）的更改。它等于使用指定的比较方式进行比较时得到的值。
changing-to	当前操作包含的更改将向指定的特性添加一个值 （添加值或添加特性）。它等于使用指定的比较方式进行比较时得到的值。
equal	当前操作中存在可用于指定特性的值 （非去除值）。它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-changing	若为 Changing，则返回 False。
not-changing-from	若为 Changing-from，则返回 False。
not-changing-to	若为 Changing-to，则返回 False。
not-equal	如果等于，则返回 False。

示例

下面的示例根据职务，向相应的 Employee 或 Manager 组中添加用户对象。在需要时，它也能创建组并设置与该组相对应的安全性。该策略名称为 Govern Groups for User Based on Title Attribute （根据职务特性管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





此条件检查 Title （职务）特性是否等于正则表达式 .*manager*。这意味着它将寻找以下条件的职务：即 manager 的前面有零个或多个字符，后面有一个字符。如果用户对象的职务为 sales managers，则将找到一个匹配项。

3.5.12 If Operation Property

对当前操作的一个操作属性进行测试。

字段

名称

指定所选条件下要测试的操作属性的名称。

操作符

选择此条件的测试类型。

比较方式

选择比较方式。请参见 “比较方式”在第 303 页 。

操作符与满足条件的情形

操作符	满足条件的情形
available	当前操作中有一个具有指定名称的操作属性。
equal	当前操作中存在具有指定名称的操作属性，其值等于使用指定的比较方式进行比较时提供的内容。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

示例

3.5.13 If Password

对当前操作中的口令进行测试。

字段

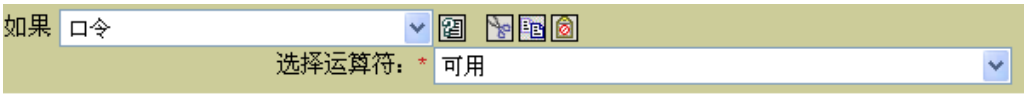
操作符

选择此条件的测试类型。

操作符与满足条件的情形

操作符	满足条件的情形
available	当前操作中存在一个可用口令。
not available	如果可用，则返回 False。

示例



3.5.14 If Source Attribute

对源数据存储区中当前对象的特性值进行测试。

字段

名称

指定所选条件下要测试的源特性的名称。

操作符

选择此条件的测试类型。

比较方式

选择比较方式。请参见 “比较方式”在第 303 页 。

操作符与满足条件的情形

操作符	满足条件的情形
available	源数据存储区中存在一个指定特性的可用值。
equal	源数据存储区中存在一个指定特性的可用值。它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

字段

名称

指定所选条件下要测试的源特性的名称。

操作符

选择此条件的测试类型。

比较方式

选择比较方式。请参见 “比较方式”在第 303 页 。

操作符与满足条件的情形

操作符	满足条件的情形
available	源数据存储区中存在一个指定特性的可用值。
equal	源数据存储区中存在一个指定特性的可用值。它等于使用指定的比较方式进行比较时得到的值。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。

示例

3.5.15 If Source DN

对当前操作中的源 DN 进行测试。

字段

操作符

选择此条件的测试类型。

操作符与满足条件的情形

操作符	满足条件的情形
available	存在可用的源 DN。
equal	存在可用的源 DN，与指定值 in-container 的内容相等。存在可用的源 DN，表示由指定值标识的树枝中的对象。
in-subtree	存在可用的源 DN，表示由指定值标识的子树中的对象。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。
not-in-container	若为 In-container，则将返回 False。
not-in-subtree	若为 In-subtree，则返回 False。

字段

操作符

选择此条件的测试类型。

操作符与满足条件的情形

操作符	满足条件的情形
available	存在可用的源 DN。
equal	存在可用的源 DN，与指定值 in-container 的内容相等。存在可用的源 DN，表示由指定值标识的树枝中的对象。
in-subtree	存在可用的源 DN，表示由指定值标识的子树中的对象。
not available	如果可用，则返回 False。
not-equal	如果等于，则返回 False。
not-in-container	若为 In-container，则将返回 False。
not-in-subtree	若为 In-subtree，则返回 False。

示例

此示例使用 If Source DN 条件检查用户对象是否在源 DN 中。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见“事件转换 - 范围过滤 - 排除子树”在第 215 页。

此条件检查源 DN 是否位于用户树枝中。如果此对象属于该树枝，则将禁止此对象。

3.5.16 If XPath Expression

对 XPath 1.0 表达式的求值结果进行测试。

字段

操作符

选择此条件的测试类型。

操作符与满足条件的情形

操作符	满足条件的情形
true	XPath 表达式的求值结果为 True。
false	如果为 True，则返回 False。

示例

3.6 操作

本节包含可通过策略构建器界面执行的所有操作的详细参考信息。

- “添加关联”在第 246 页
- “添加目标特性值”在第 246 页
- “添加目标对象”在第 247 页
- “添加源特性值”在第 248 页
- “添加源对象”在第 249 页
- “追加 XML 要素”在第 250 页
- “追加 XML 文本”在第 251 页
- “中断”在第 252 页
- “清除目标特性值”在第 252 页
- “清除操作属性”在第 253 页
- “清除 SSO 身份凭证”在第 253 页
- “清除源特性值”在第 254 页

- “跟踪讯息”在第 281 页
- “禁止”在第 282 页
- “操作特性不可用时禁止”在第 282 页

3.6.1 添加关联

向具有指定关联的 Identity Vault 发送添加关联命令。

字段

方式

选择应在当前操作添加此操作，还是直接将其写入 Identity Vault。

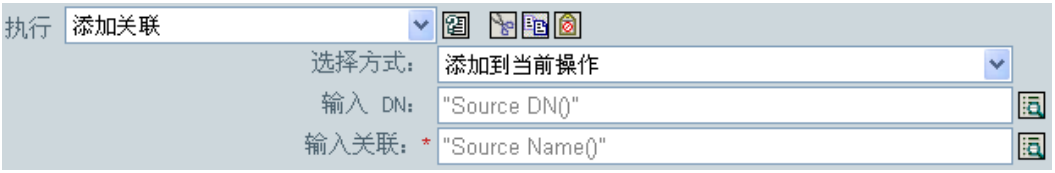
DN

指定目标对象的 DN，若保留空白则使用当前对象的 DN。

关联

指定要添加的关联值。

示例



3.6.2 添加目标特性值

向目标数据存储区中的某对象的特性添加一个值。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

值类型

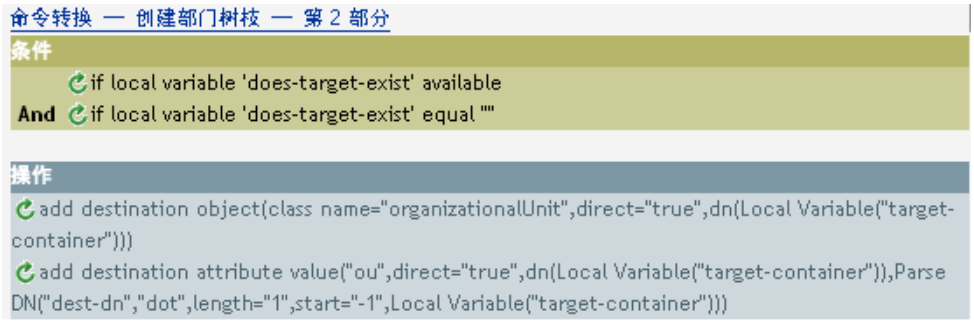
选择要添加的特性值的语法。

值

指定要添加的特性值。

示例

本示例在 OU 特性中添加目标特性值，该值由所创建的局部变量创建。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “命令转换 - 创建部门树枝 - 第 1 部分和第 2 部分”在第 208 页 。



3.6.3 添加目标对象

在目标数据存储区中创建指定类型的新对象。

字段

类名称

指定要创建对象的类名称。

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

DN

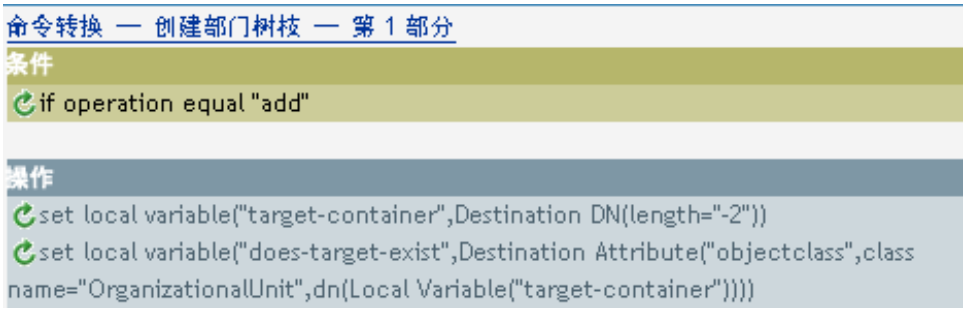
指定要创建的对象的 DN。

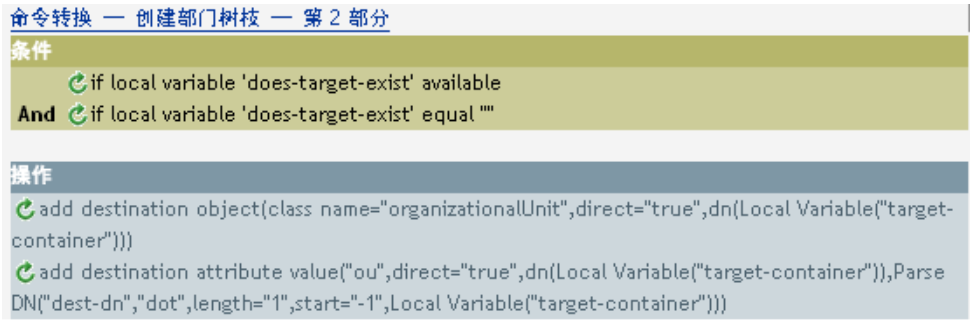
注释

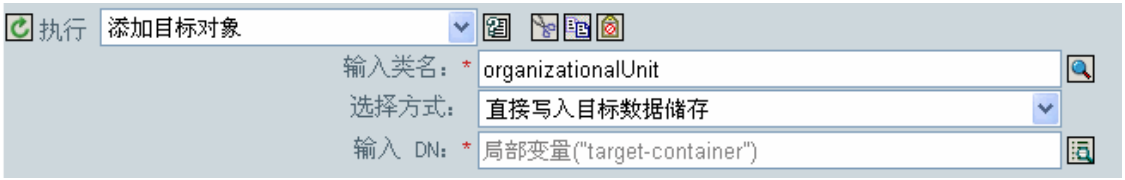
在创建对象的过程中添加的特性值，必须使用同一 DN 在后续操作 （“添加目标特性值”在第 246 页）中执行。

示例

本示例创建所需的部门树枝。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见预定义规则中的 “命令转换 - 创建部门树枝 - 第 1 部分和第 2 部分”在第 208 页。







OU 对象已创建。在此操作之后，通过目标特性值操作创建此 OU 特性的值。

3.6.4 添加源特性值

在源数据存储区中，向对象的指定特性添加指定值。目标对象为当前对象、 DN 或关联。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

值类型

选择要添加的特性值的语法。

值

指定要添加的特性值。

示例

3.6.5 添加源对象

在源数据存储区中创建指定类型的对象。在创建对象的过程中添加的特性值，必须使用同一 DN 在后续操作 （添加源特性值 (在第 248 页)）中执行。

字段

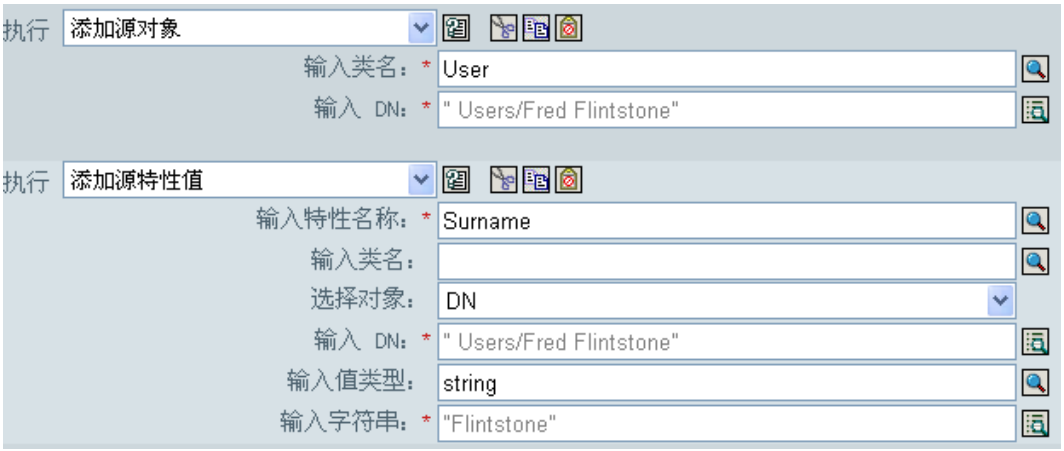
类名称

指定要添加对象的类名称。

DN

指定要添加对象的 DN。

示例



字段

类名称

指定要添加到源数据存储区的对象的类名称。

DN

指定要添加到源数据存储区的新对象的 DN。

3.6.6 追加 XML 要素

将一个要素追加到由 XPath 表达式选择的一组要素中。

字段

名称

指定 XML 要素的标签名。如果先前已在此策略中定义了名称空间前缀，则特性名称中可以包含该前缀。

XPATH 表达式

指定一个 XPath 1.0 表达式，该表达式会返回一个节点集，其中包含要追加新要素的要素。

示例

3.6.7 追加 XML 文本

将文本追加到由 XPath 表达式选择的一组要素中。

字段

XPATH 表达式

返回节点集的 XPath 1.0 表达式，该节点集包含应追加新要素的要素。

字符串

指定要追加的文本。

示例

3.6.8 中断

使用当前策略结束对当前操作的处理。

示例

3.6.9 清除目标特性值

从目标数据存储区的对象中去除已命名特性的所有值。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

示例

3.6.10 清除操作属性

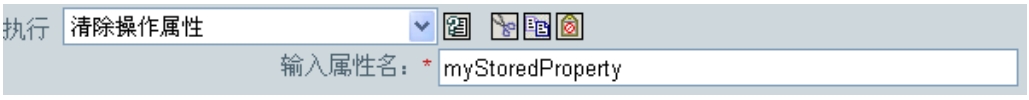
清除当前操作的所有操作属性。

字段

属性名

指定要清除的操作属性的名称。

示例



3.6.11 清除 SSO 身份凭证

清除一次签到身份凭证，以便取消对象供应。此操作是身份凭证供应策略的一部分。有关详细信息，请参见第 4 章 “Novell 身份凭证供应策略”在第 305 页 。

字段

身份凭证储存对象 **DN**

指定储存库对象的 DN。

目标用户 **DN**

指定目标用户的 DN。

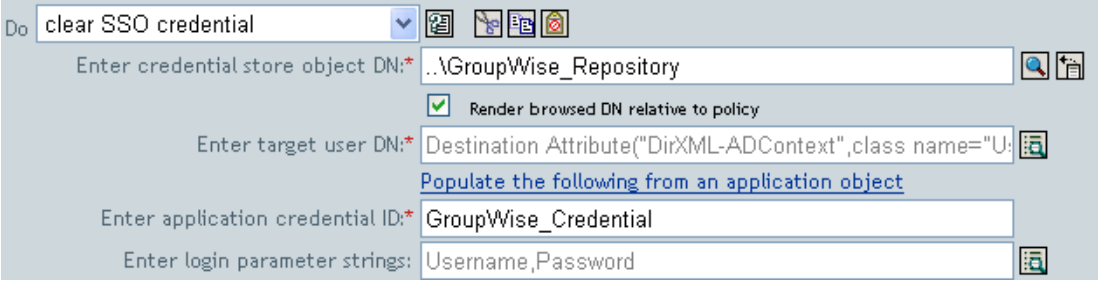
应用程序身份凭证 **ID**

指定储存在应用程序对象中的应用程序身份凭证。

登录参数字符串

指定应用程序的所有登录参数。登录参数是储存在应用程序对象中的鉴定密钥。

示例



3.6.12 清除源特性值

在源数据存储区中，将某对象一个特性的所有值去除。

字段

特性名称

指定该特性的名称。

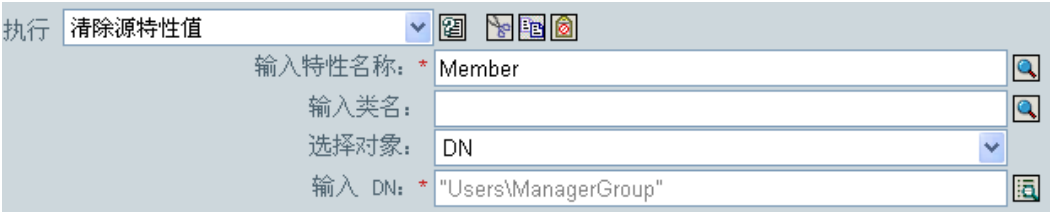
类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

示例



3.6.13 通过 XPath 表达式克隆

将一个 XPath 表达式选定的 XML 节点集的深层拷贝追加到由另一个 XPath 表达式选定的一组要素中。

字段

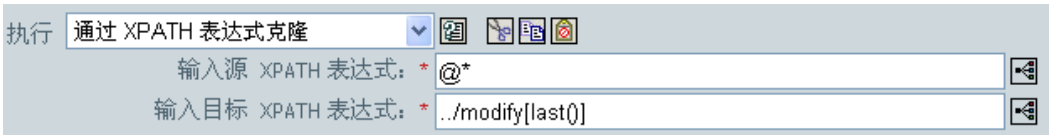
源 **XPATH** 表达式

指定返回节点集的 XPath 1.0 表达式，该节点集包含要复制的节点。

目标 **XPATH** 表达式

指定 XPath 1.0 表达式，该表达式返回的节点集中包含追加已复制节点的要素。

示例



3.6.14 克隆操作特性

将当前操作中所有具体出现的某一特性复制到当前操作中的另一特性中。

字段

源名称

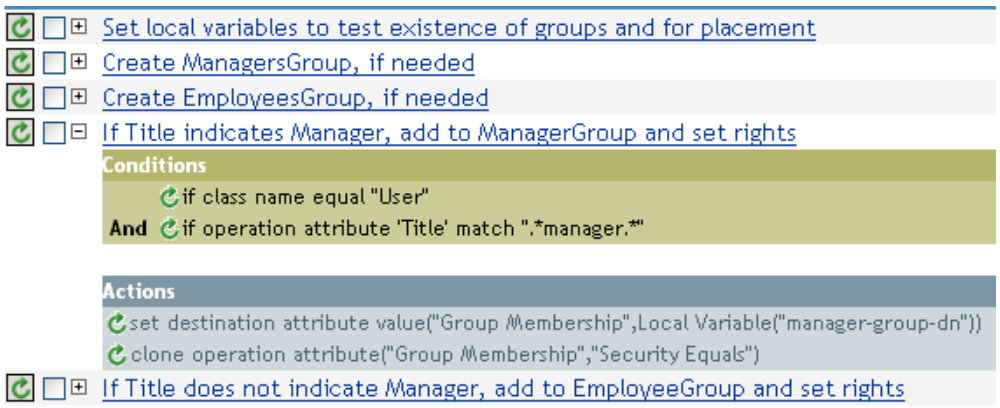
指定被复制的特性的名称。

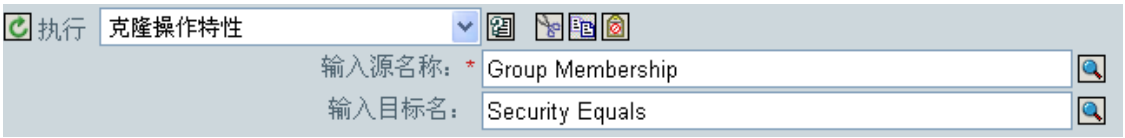
目标名称

指定要复制到的特性的名称。

示例

下面的示例根据职务，向相应的 Employee 或 Manager 组中添加用户对象。必要时还可以创建组并设置等效于该组的安全性。所用的策略是 Govern Groups for User Based on Title Attribute （根据职务特性管理用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





克隆操作特性从 Group Membership （组成员资格）特性获取信息，并将该信息添加到 Security Equals （安全性等效）特性，使两特性的值相同。

3.6.15 删除目标对象

删除目标数据存储区中的一个对象。

字段

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

选择要在目标数据存储区中删除的目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

示例

3.6.16 删除源对象

在源数据存储区中删除对象。

字段

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

选择要从源数据存储区中删除的目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

示例

3.6.17 查找匹配对象

在目标数据存储区中查找当前对象的匹配项。

字段

范围

选择搜索范围。范围可能是 entry （项）、 subordinates （从属项）或 subtree （子树）。

DN

指定作为搜索基础的 DN。

匹配特性

指定要搜索的特性值。

注释

只有当前操是添加操作时， " 查找匹配对象 " 才有效。

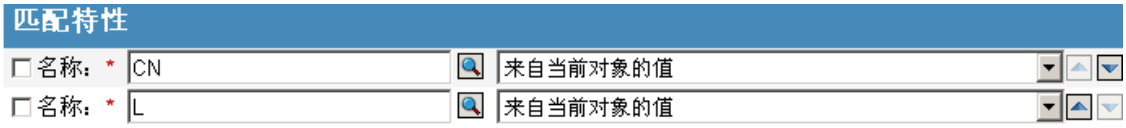
范围为 "entry" （项）时， DN 自变量是必需的，在其它情况下该自变量是可选的。范围为 "subtree" （子树）或 "subordinates" （从属项）时，至少需要一个匹配特性。如果范围为 entry 且存在多个指定的匹配特性，则结果为未定义。如果目标数据存储区为已连接的应用程序，会在返回的每个成功匹配的当前操作中添加关联。如果当前操作已存在非空关联，则不执行查询，因此可以根据同一规则将多个 " 查找匹配对象 " 操作连接在一起。

如果目标数据存储区是 Identity Vault，则应设置当前操作的目标 DN 特性。如果当前操作已存在非空目标 DN 特性，则不执行查询，因此可以根据同一规则将多个 " 查找匹配对象 " 操作连接在一起。如果只返回一个结果且该结果尚未关联，则将当前操作的目标 DN 设置为匹配对象的源 DN。如果只返回一个结果且该结果已关联，则将当前操作的目标 DN 设置为单一字符 ￼。如果返回多个结果，则将当前操作的目标 DN 设置为单一字符 &#xFFFD;。

示例

本示例使用 CN 和 L 特性匹配用户对象。此规则搜索的开始位置是 Users 树枝，并将储存在 OU 特性中的信息添加至 DN。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “匹配 - 按特性值”在第 86 页 。

单击 " 自变量构建器 " 图标时，将出现 " 匹配特性构建器 "。在此构建器中，指定您想要匹配的特性。本示例使用了 CN 和 L 特性。



3.6.18 对于每个

对节点集中的每个节点重复一组操作。

字段

节点集

指定节点集。

操作

指定要在节点集的各个节点上执行的操作。

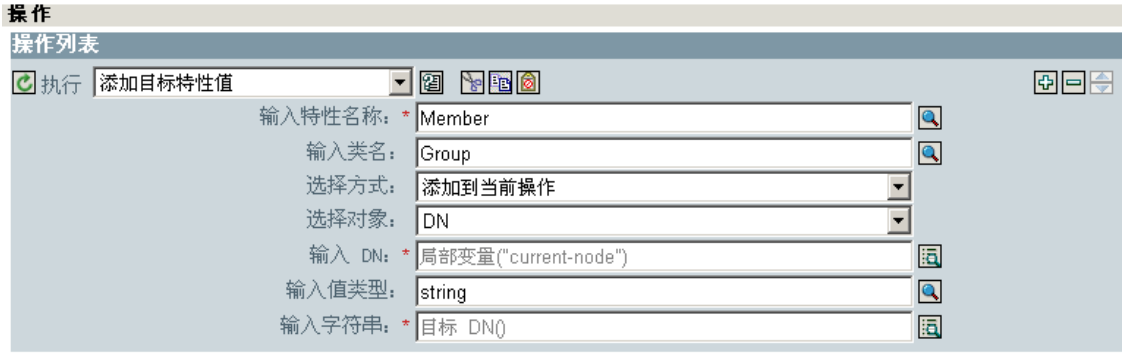
注释

如果使用局部变量，则当前节点对于操作的每个迭代分别为不同的值。

如果节点集中的某个节点是权利，则对其隐式执行 “实施权利”在第 260 页 操作。

示例

在以下示例中，自变量操作构建器用于提供操作自变量：



3.6.19 生成事件

将用户定义的事件发送到 Novell Audit。

字段

ID

事件的 ID。当使用 java.lang.Integer 的 parseInt 方法分析提供的值的语法时，该值必须生成介于 1000-1999 之间的整数。

级别

事件的级别。

级别	说明
记录紧急事件	引起 Metadirectory 引擎或驱动程序关闭的事件。

级别	说明
记录警报	需要立即注意的事件。
记录关键	引起 Metadirectory 引擎或驱动程序部分出现故障的事件。
记录错误	说明可由 Metadirectory 引擎或驱动程序处理的错误的事件。
记录警告	否定事件，并不表示出现问题。
记录通知	可供管理员了解或改进使用和操作的肯定或否定事件。
记录信息	任何重要的肯定事件。
记录调试	用于技术支持或工程师调试 Metadirectory 引擎或驱动程序时使用的相关事件。

字符串

指定事件中包含的由用户定义的字符串、整数和二进制值。这些值由命名字符串构建器提供。

标记	说明
target	要处理的对象。
target-type	指定目标预定义格式的整数。当前 target-type 的预定义值为： ◆ 0 = 无 ◆ 1 = 斜线表示法 ◆ 2 = 点表示法 ◆ 3 = LDAP 表示法
subTarget	要处理的目标的子组件。
text1	此处输入的文本储存在 text1 事件字段中。
text2	此处输入的文本储存在 text2 事件字段中。
text3	此处输入的文本储存在 text3 事件字段中。
value	此处输入的任何数字都将储存在 value 事件字段中。
value3	此处输入的任何数字都将储存在 value3 事件字段中。
data	此处输入的数据将储存在 BLOB 事件字段中。

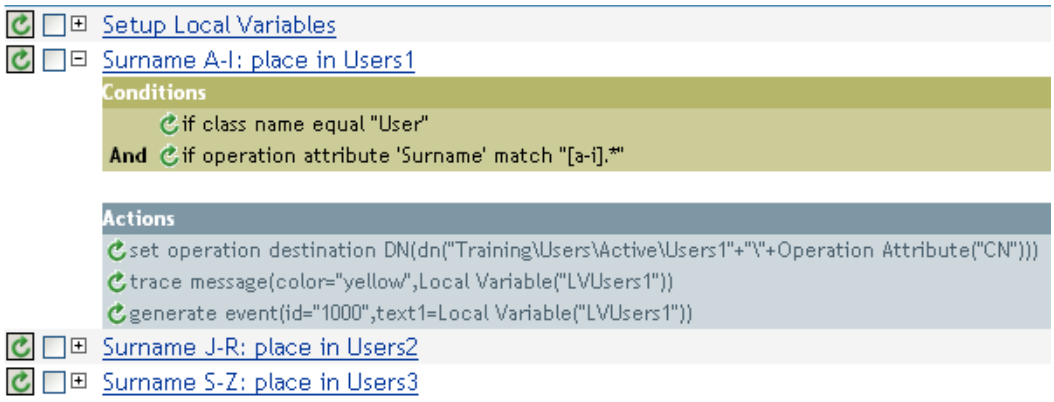
注释

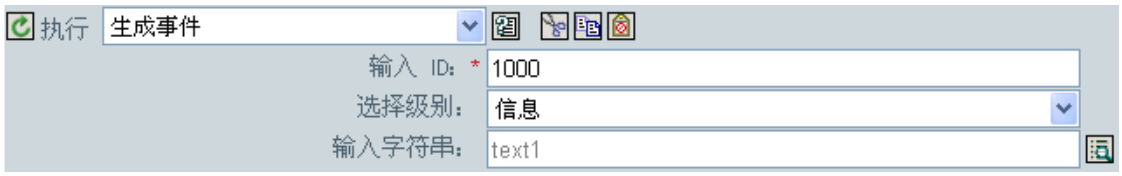
Novell Audit 事件结构包含一个 target、一个 subTarget、三个字符串 （text1、 text2、 text3）、两个整数 （value、 value3）以及一个通用字段 (data)。如果您的环境不支持更大的数据字段，则文本字段最多可以输入 256 个字节，数据字段最多可以包含 3 KB 的信息。

示例

本示例包含四条规则，它们根据 Surname （姓氏）特性的首字符实施对用户对象的布局策略，并生成一条跟踪讯息和一个自定义 Novell Audit 事件。 " 生成事件 " 操作用于向 Novell

Audit 发送事件。策略名称为 Policy to Place by Surname （按姓氏布局策略），可从 Novell 支持万维网站点下载。详情请见 “可下载的 Identity Manager 策略”在第 31 页 。





在以下示例中，命名字符串构建器用于提供字符串自变量。

" 生成事件 " 正在创建一个 ID 为 1000 的事件并显示由局部变量 LVUser1 生成的文本。局部变量 LVUser1 就是字符串 User:Operation Attribute"cn"+"added to the"+"Training\Users\Active\Users1"+"container"。此事件将读取 User:jsmith added to the Trainging\Users\Active\Users1 container。

3.6.20 实施权利

指定实施权利的操作，以便向授予或撤销权利的代理报告这些权利的状态。

字段

节点集

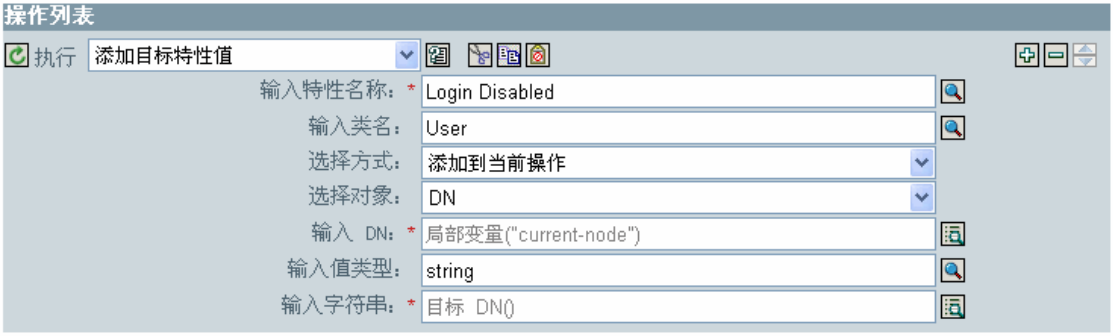
包含由指定操作实施的权利的节点集。

操作

实施指定权利的操作。

示例

在以下示例中，自变量操作构建器用于提供操作自变量：



3.6.21 移动目标对象

移动目标数据存储区中的对象。

字段

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

类名称

（可选）指定要移动的对象的类名称。若保留空白则使用当前对象的类名称。

要移动的对象

选择要移动的对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

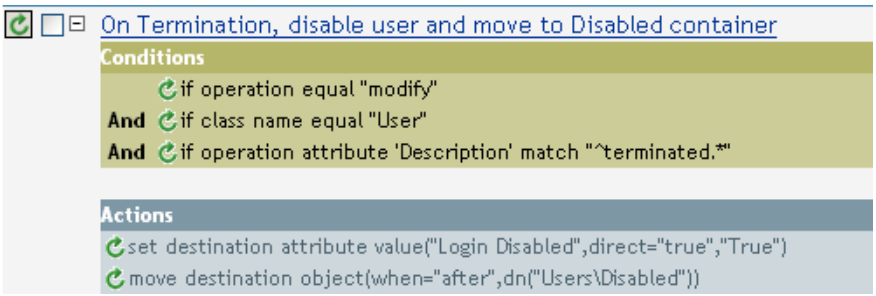
树枝

选择接收对象的树枝。此树枝由 DN 或关联指定。

示例

本示例包含一条规则，该规则在 Description （说明）特性指示用户的帐户已终止时将禁用这些帐户，同时将它们移动至禁用的树枝中。此策略名为 Disable User Account and Move

When Terminated （终止时禁用用户帐户并移动），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。



此策略将检查该事件是否为用户对象上的修改事件，以及 Description （说明）特性中是否包含值 terminated。如果是这种情况，它会将 Login Disabled （禁止登录）特性设置为 true，并将此对象移入 User\Disabled （用户 \ 已禁用）树枝中。

3.6.22 移动源对象

移动源数据存储区中的对象。

字段

要移动的对象

选择要移动的对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

选择树枝

选择接收对象的树枝。此树枝由 DN 或关联指定。

示例

3.6.23 重新设置操作特性的格式

在当前操作中，使用某种模式重设特性的所有值的格式。

字段

名称

指定该特性的名称。

值类型

指定新特性值的语法。

值

指定一个值，作为特性值新格式的模式。如果需要使用原始值构造新值，则必须通过引用局部变量 current-value 获取该原始值。

示例

本示例重设了电话号码的格式，由 (nnn)-nnn-nnnn 更改为 nnn-nnn-nnnn。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “输入或输出转换 - 将电话号码格式重新从 (nnn) nnn-nnnn 设置为 nnn-nnn-nnnn”在第 215 页 。

" 重新设置操作特性的格式 " 操作更改了电话号码的格式。此规则使用自变量构建器和正则表达式更改信息的显示方式。

3.6.24 去除关联

将去除关联命令发送至 Identity Vault。

字段

方式

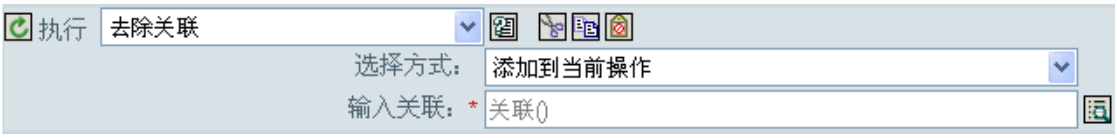
选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

关联

指定要去除的关联值。

示例

本示例执行了删除操作并禁用用户对象。然后转换事件。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “命令转换 - 发布者删除 - 禁用”在第 209 页 。



对用户对象执行删除操作时， Login Disabled （禁止登录）特性值将被设置为 true，并将关联从该对象中去除。由于已连接的应用程序中的关联对象已不存在，因此将去除该关联。

3.6.25 去除目标特性值

去除目标数据存储区中对象的特性值。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

选择对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

值类型

指定新特性值的语法。

值

指定新特性的值。

示例

3.6.26 去除源特性值

在源数据存储区中，从对象的命名特性中去除指定的值。

字段

特性名称
: 指定该特性的名称。

类名称
: （可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

对象
: 选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

值类型
: 指定要去除的特性值的语法。

值
: 指定要去除的特性值。

示例

3.6.27 重命名目标对象

在目标数据存储区中重命名对象。

字段

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

字符串

指定对象的新名称。

示例

3.6.28 重命名操作特性

重命名当前操作中所有具体出现的某一特性。

字段

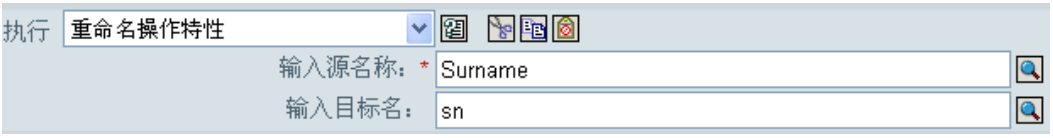
源名称

指定原始特性名称。

目标名称

指定新的特性名称。

示例



3.6.29 重命名源对象

重命名源数据存储区中的某一对象。

字段

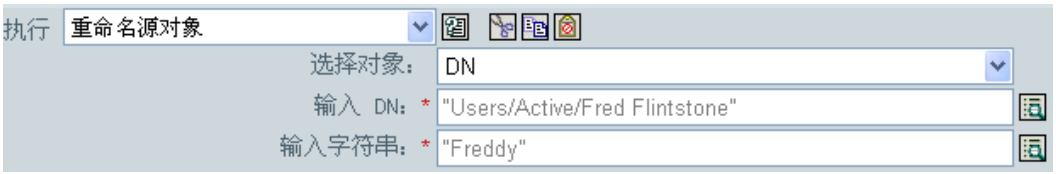
选择对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

字符串

指定对象的新名称。

示例



3.6.30 发送电子邮件

发送电子邮件通知。

字段

ID

（可选）在发送讯息的 SMTP 系统中指定用户 ID。

服务器

指定 SMTP 服务器的名称。

口令

（可选）指定 SMTP 服务器帐户口令。

重要：口令特性的值以明文形式储存。

类型

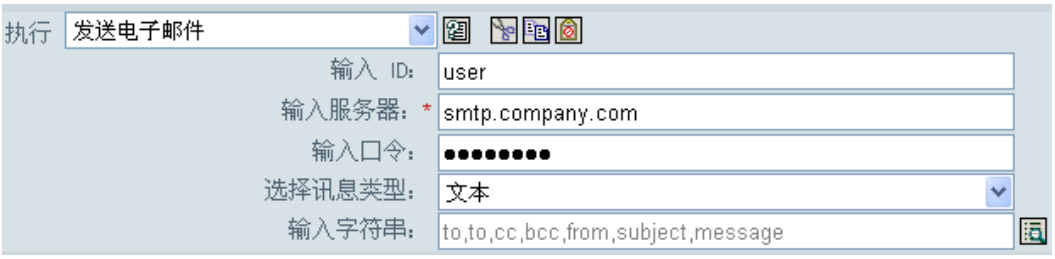
选择电子邮件讯息的类型。

字符串

指定包含各种电子邮件地址、主题和内容的值。下表列出了有效的命名字符串自变量：

字符串名称	说明
to	将地址添加到电子邮件收件人列表中；允许添加多个地址。
cc	将地址添加到 " 抄送 " 电子邮件收件人列表中；允许添加多个地址。
bcc	将地址添加到 " 暗送 " 电子邮件收件人列表中；允许添加多个地址。
from	指定电子邮件的发送地址。
reply-to	指定电子邮件讯息的回复地址。
subject	指定电子邮件主题。
message	指定电子邮件讯息的内容。
encoding	指定电子邮件讯息使用的字符编码。

示例



在以下示例中，命名字符串构建器用于提供字符串自变量：

3.6.31 通过模板发送电子邮件

使用模板生成电子邮件通知。

字段

通知 **DN**

指定 SMTP 通知配置对象的斜线格式 DN。

模板 **DN**

指定电子邮件模板对象的斜线格式 DN。

口令

（可选）指定 SMTP 服务器帐户口令。

重要：口令特性的值以明文形式储存。

字符串

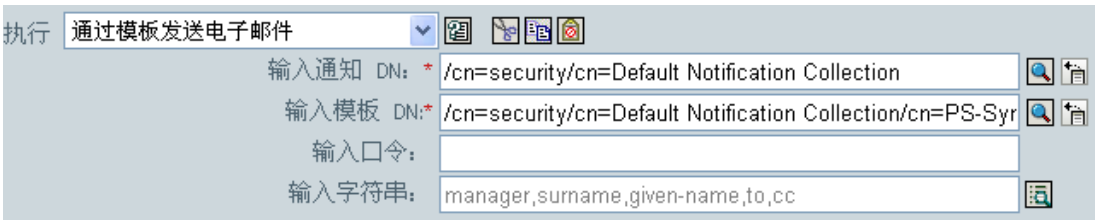
指定电子邮件讯息的附加字段。下表包含了保留的字段名称，用于指定各种电子邮件地址：

字符串名称	说明
to	将地址添加到电子邮件收件人列表中；允许添加多个地址。
cc	将地址添加到 " 抄送 " 电子邮件收件人列表中；允许添加多个地址。
bcc	将地址添加到 " 暗送 " 电子邮件收件人列表中；允许添加多个地址。

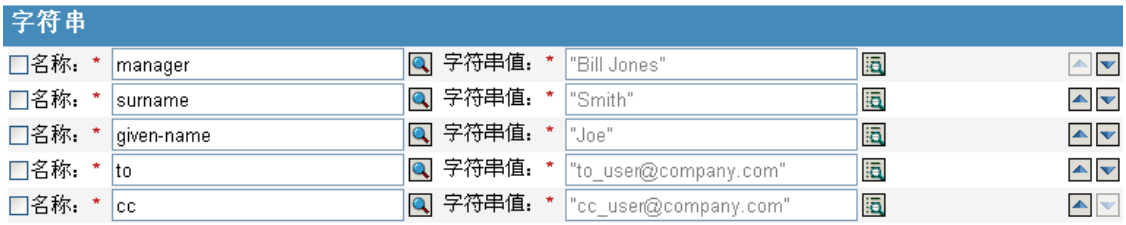
字符串名称	说明
reply-to	指定电子邮件讯息的回复地址。
encoding	指定电子邮件讯息使用的字符编码。

每个模板可能还定义了其它字段，这些字段可在电子邮件讯息的主题和正文中进行替换。

示例



在以下示例中，命名字符串构建器用于提供字符串自变量：



3.6.32 设置默认特性值

如果某特性无值，则在当前操作中为该特性添加默认值 （也可以将默认值添加至源数据存储区中的当前对象）。当前操作是 " 添加 " 时它才有效。

字段

特性名称

指定默认特性的名称。

写回

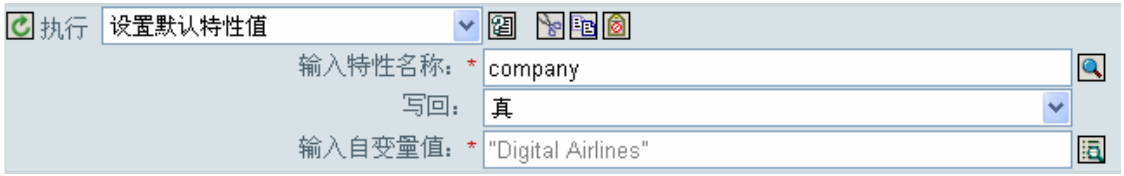
选择是否还要将默认值写回源数据存储区中。

值

指定特性的默认值。

示例

本示例设置 company （公司）特性的默认值。也可以设置任意特性的值。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “创建 - 设置默认特性值”在第 212 页 。



若要构建该值，请起动自变量值列表构建器。有关此构建器的更多信息，请参见 “自变量值列表构建器”在第 203 页 。可以根据需要设置此值。在这种情况下，我们使用自变量构建器并将文本设置为公司的名称。

3.6.33 设置目标特性值

向目标数据存储区中的某对象的特性添加一个值，并去除该特性的所有其它值。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定目标数据存储区中目标对象的类名称。若保留空白则使用当前对象的类名称。

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

值类型

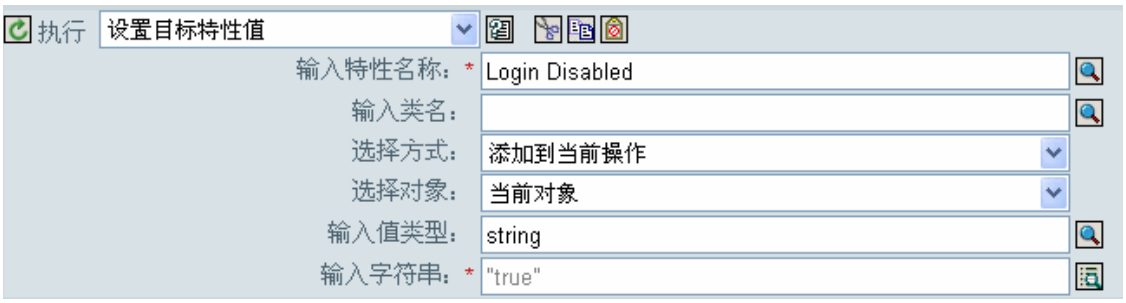
选择要设置的特性值的语法。

值

指定要设置的特性值。

示例

本示例执行了删除操作并禁用用户对象。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “命令转换 - 发布者删除 - 禁用”在第 209 页 。



此规则将 Login Disabled （禁止登录）特性的值设置为 true。它使用自变量构建器为此特性的值添加文本 true。有关此构建器的更多信息，请参见 “自变量构建器”在第 201 页 。

3.6.34 设置目标口令

设置目标数据存储区中当前对象的口令。

字段

方式

选择要将此操作添加到当前操作中，还是当前操作之前或之后，或是直接写入目标数据存储区。

对象

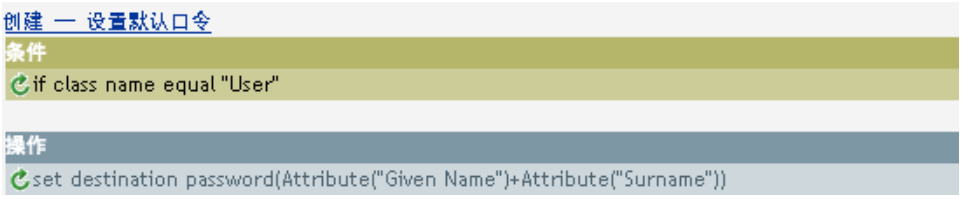
选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

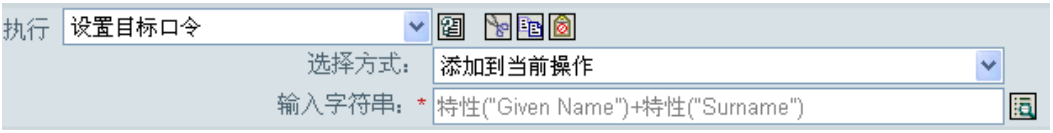
字符串

指定要设置的口令。

示例

本示例为已创建的用户对象设置默认口令。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “创建 - 设置默认口令”在第 213 页 。





创建用户对象时，此口令设置为 Given Name （名）特性加 Surname （姓氏）特性。

3.6.35 设置局部变量

设置局部变量。

字段

变量名

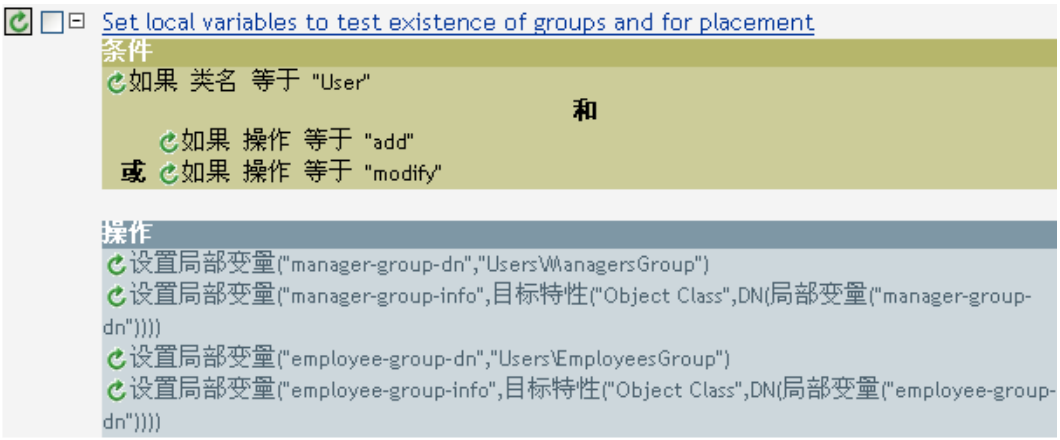
指定新局部变量的名称。

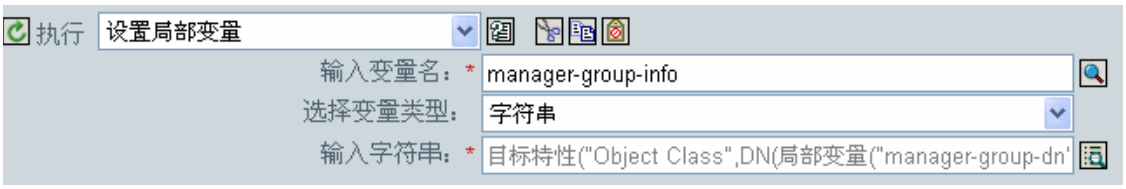
变量类型

选择局部变量的类型。可以为字符串、 XPath 1.0 节点集或 Java 对象。

示例

下面的示例根据职务，向相应的 Employee 或 Manager 组中添加用户对象。必要时还可以创建组并设置等效于该组的安全性。该策略名称为 Govern Groups for User Based on Title （根据 Title （职务）控制用户组），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





将局部变量设置为用户对象的目标特性 Object Class 中的值加上局部变量 manager-group-info。自变量构建器用于构造局部变量。有关详细信息，请参见“自变量构建器”在第 201 页。

3.6.36 设置操作关联

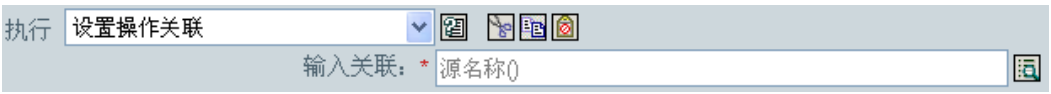
设置当前操作的关联值。

字段

关联

提供新的关联值。

示例



3.6.37 设置操作类名

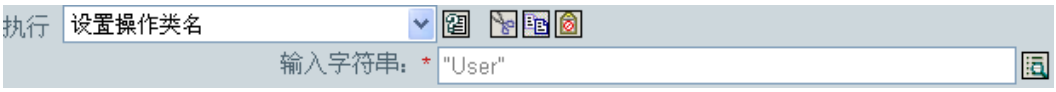
设置当前操作的对象类名称。

字段

字符串

指定新的类名称。

示例



3.6.38 设置操作目标 DN

设置当前操作的目标 DN。

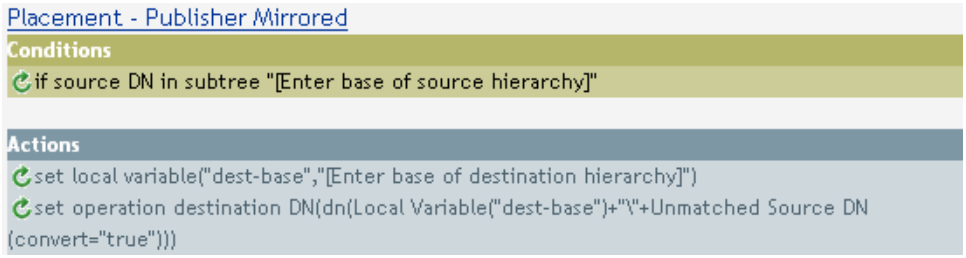
字段

DN

指定新的目标 DN。

示例

本示例使用从已连接系统镜像得到的结构将对象放置到 Identity Vault 中。您需要在源数据存储区和目标数据存储区中定义镜像的起始点。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “创建 - 设置默认特性值”在第 75 页 。



该规则将操作目标 DN 设置为目标基位置加上源 DN 组成的局部变量。

3.6.39 设置操作属性

设置操作属性。操作属性是储存在操作中的命名值。它通常用于提供其它环境，处理操作结果的策略可能需要这些环境。

字段

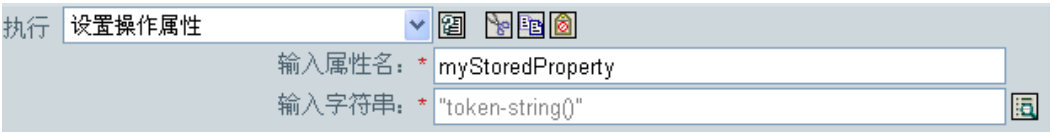
属性名

指定操作属性的名称。

字符串

指定操作属性的名称。

示例



3.6.40 设置操作源 DN

设置当前操作的源 DN。

字段

DN

指定新的源 DN。

示例

3.6.41 设置操作模板 DN

将当前操作的模板 DN 设置为指定值。当前操作是 " 添加 " 时此操作才有效。

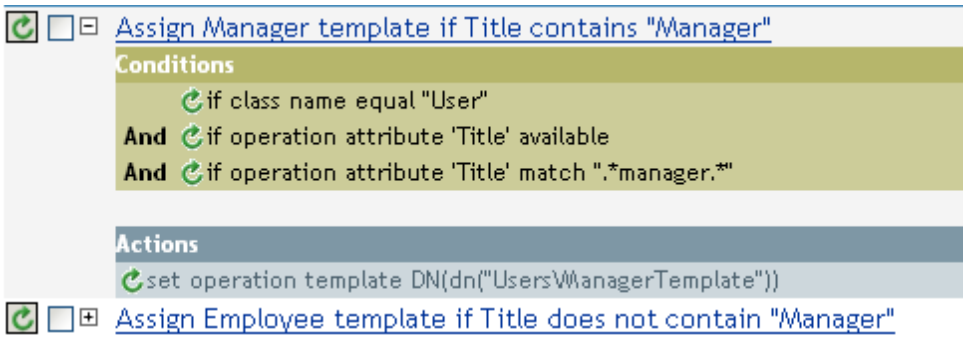
字段

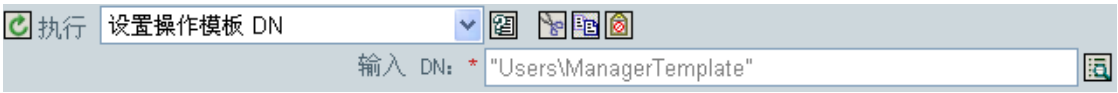
DN

指定模板 DN。

示例

如果 Title （职务）特性中包含文字 Manager，本示例将应用 Manager （经理）模板。策略名称为 Policy:Assign Template to User Based on Tile （根据职务为用户指派模板），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





Manager （经理）模板适用于具有可用 Title （职务）特性且在职务中某处包含文字 "manager" 的所有用户对象。此策略使用正则表达式查找所有可能的匹配项。

3.6.42 设置源特性值

向源数据存储区中的某对象的特性添加一个值，并去除该特性的所有其它值。

字段

特性名称

指定该特性的名称。

类名称

（可选）指定源数据存储区中的目标对象的类名称。若保留空白则使用当前对象的类名称。

对象

选择目标对象。此对象可以是当前对象，也可以是通过 DN 或关联指定的对象。

值类型

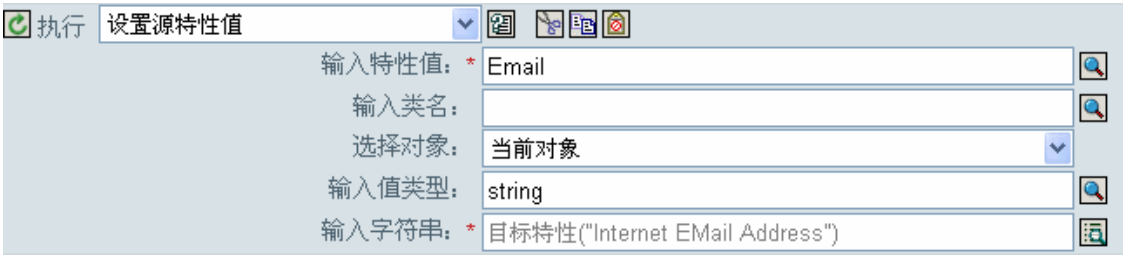
选择特性值的语法。

值

指定要设置的特性值。

示例

本示例在检测到电子邮件地址发生更改时，将其设置回原来的地址。策略名称为 Policy:Reset Value of the E-mail Attribute （重设置电子邮件特性的值），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。



此操作采用目标特性因特网电子邮件地址的值，并将电子邮件的源特性也设置为此值。

3.6.43 设置源口令

设置源数据存储区中的当前对象的口令。

字段

字符串

指定要设置的口令。

示例

3.6.44 设置 SSO 身份凭证

创建用户对象或修改口令后，设置 SSO 身份凭证。此操作是身份凭证供应策略的一部分。有关详细信息，请参见第 4 章 “Novell 身份凭证供应策略”在第 305 页。

字段

身份凭证储存对象 **DN**

指定储存库对象的 DN。

目标用户 **DN**

指定目标用户的 DN。

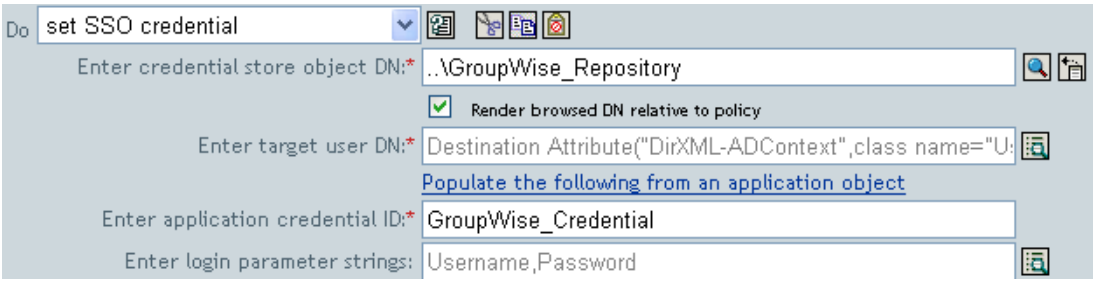
应用程序身份凭证 **ID**

指定储存在应用程序对象中的应用程序身份凭证。

登录参数字符串

指定应用程序的所有登录参数。登录参数是储存在应用程序对象中的鉴定密钥。

示例



3.6.45 设置 SSO 通行口令

设置供应用户对象时的 Novell SecureLogin® 通行口令和答案。此操作是身份凭证供应策略的一部分。有关详细信息，请参见第 4 章 “Novell 身份凭证供应策略”在第 305 页。

字段

身份凭证储存对象 **DN**

指定储存库对象的 DN。

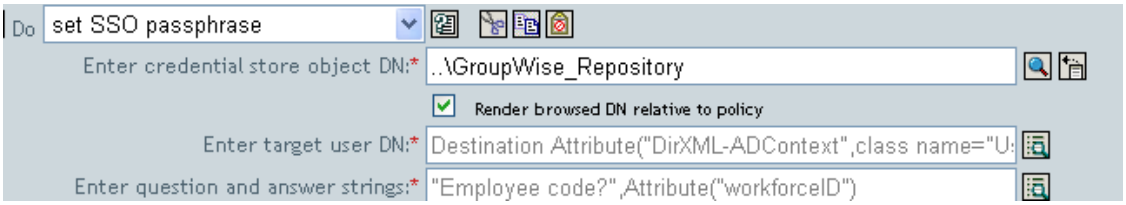
目标用户 **DN**

指定目标用户的 DN。

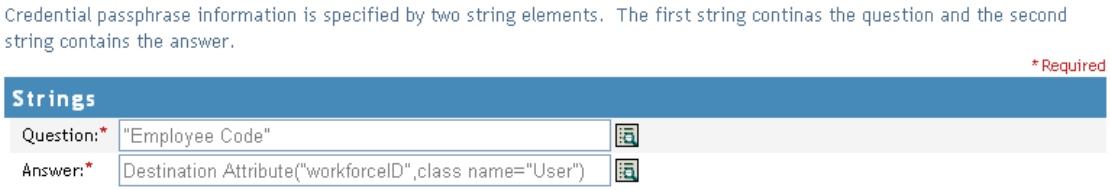
问题和答案字符串

指定 SecureLogin 通行口令的问题和答案。

示例



SecureLogin 通行口令的问题和答案以字符串的形式储存在策略中。单击 *Edit these strings*（编辑这些字符串）图标，起动字符串构建器。指定通行口令的问题和答案。



3.6.46 设置 XML 特性

设置由 XPath 表达式选择的一组要素上的 XML。

字段

名称

指定 XML 特性的名称。如果先前已在此策略中定义了名称空间前缀，则特性名称中可以包含该前缀。

XPATH 表达式

XPath 1.0 表达式，该表达式将返回一个节点集，其中包含设置此 XML 特性的要素。

字符串

指定 XML 特性的值。

示例

3.6.47 设置 SSO 身份凭证

创建用户对象或修改口令后，设置 SSO 身份凭证。此操作是身份凭证供应策略的一部分。有关详细信息，请参见第 4 章 “Novell 身份凭证供应策略”在第 305 页 。

字段

身份凭证储存对象 **DN**

指定储存库对象的 DN。

目标用户 **DN**

指定目标用户的 DN。

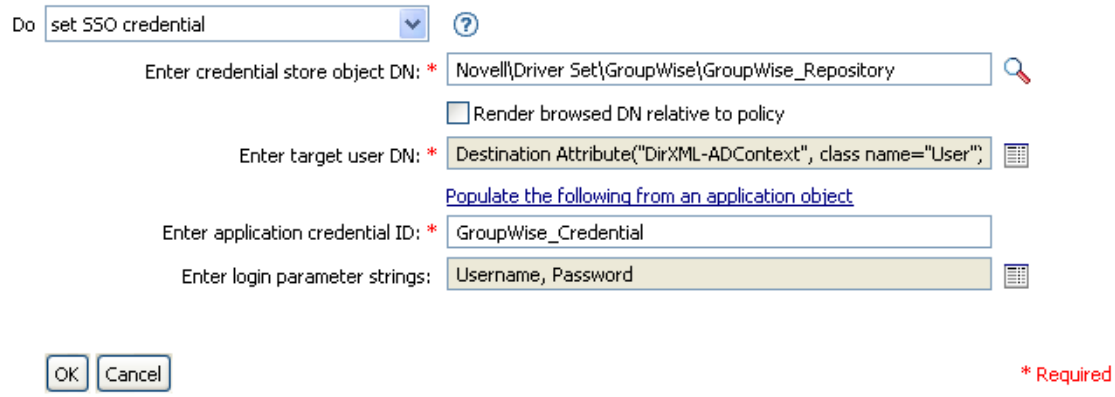
应用程序身份凭证 **ID**

指定储存在应用程序对象中的应用程序身份凭证。

登录参数字符串

指定应用程序的所有登录参数。登录参数是储存在应用程序对象中的鉴定密钥。

示例



3.6.48 状态

生成状态通知。

字段

级别

指定通知的状态级别。

讯息

使用自变量构建器提供状态讯息。

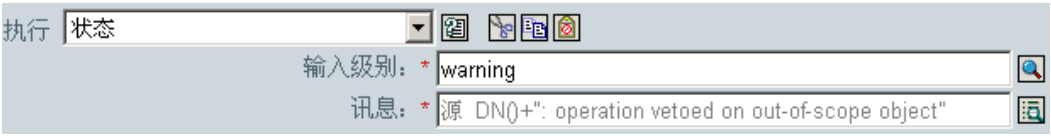
注释

如果是重试级别，则该策略将立即暂停对输入文档的处理，并安排重试当前正在处理的事件。

如果是致命级别，则该策略将立即暂停对输入文档的处理，并开始关闭驱动程序。

如果当前操作具有事件 ID，则状态通知将使用该事件 ID，否则将不会报告事件 ID。

示例



3.6.49 去除操作特性

去除当前操作中所有具体出现的某一特性。

字段

名称

指定要去除的特性的名称。

示例

本示例在检测到电子邮件地址发生更改时，将其设置回原来的地址。策略名称为 Policy:Reset Value of the E-mail Attribute （重设置电子邮件的特性值），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。

此操作将去除 Email （电子邮件）特性。所保留的值是目标 Email （电子邮件）特性中的原有值。

3.6.50 去除 XPath

去除由 XPath 1.0 表达式选择的节点。

字段

XPATH 表达式

指定返回包含要去除节点的节点集的 XPath 1.0 表达式。

示例

3.6.51 跟踪讯息

将讯息发送至 DSTRACE。

字段

级别

指定讯息的跟踪级别。默认级别为 0。指定的跟踪级别低于或等于在驱动程序中配置的跟踪级别时，此讯息才出现。

有关如何在驱动程序中设置跟踪级别的信息，请参见 《*Novell Identity Manager 3.0* 管理指南》中的 “Viewing Identity Manager Processes （查看 Identity Manager 进程）”。

颜色

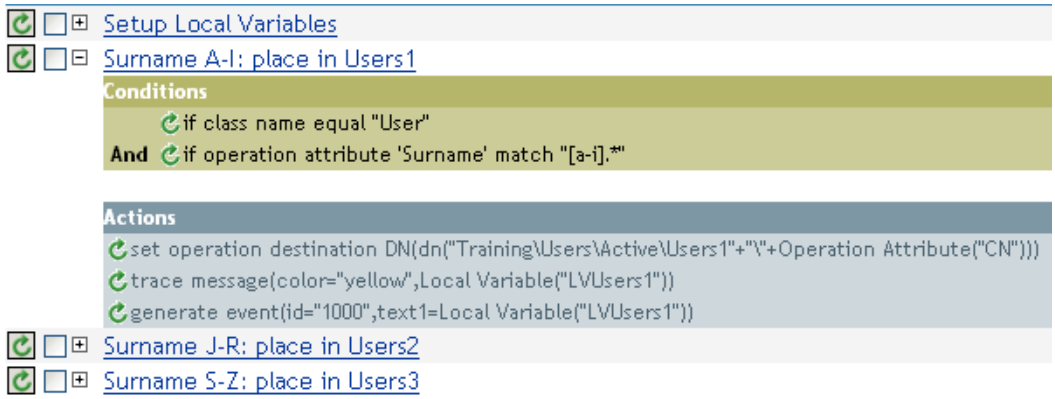
选择跟踪讯息的颜色。

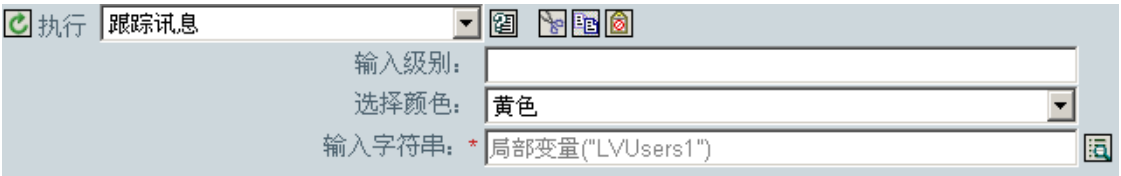
字符串

指定跟踪讯息的值。

示例

本示例包含四条规则，它们根据 Surname （姓氏）特性的首字符实施对用户对象的布局策略。并生成一条跟踪讯息和一个自定义 Novell Audit 事件。 " 跟踪讯息 " 操作用于将跟踪讯息发送至 DSTRACE。策略名称为 Policy to Place by Surname （按姓氏布局策略），可从 Novell 支持万维网站点下载。详情请见 “可下载的 Identity Manager 策略”在第 31 页。





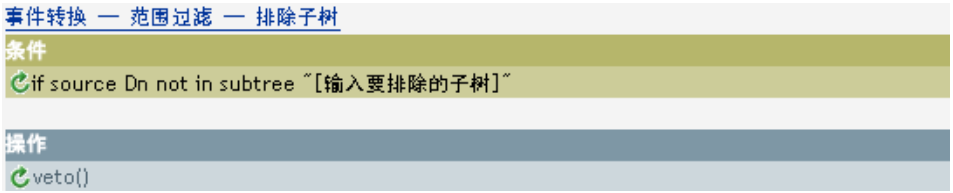
此操作将跟踪讯息发送至 DSTRACE。局部变量的内容为 LVUsers1，在 DSTRACE 中显示为黄色。

3.6.52 禁止

禁止当前操作。

示例

本示例将排除所有来自指定子树的事件。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见预定义规则中的 “事件转换 - 范围过滤 - 排除子树”在第 215 页 。



此操作将禁止所有来自指定子树的事件。

3.6.53 操作特性不可用时禁止

根据当前操作中特性的可用性，有条件地取消当前操作并结束对当前策略的处理。

字段

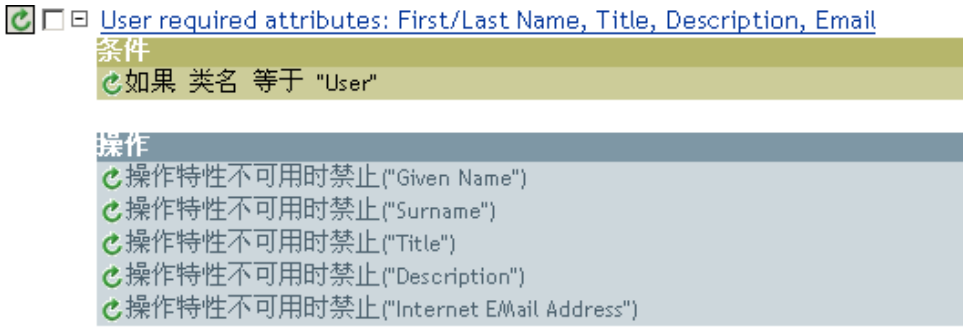
名称

指定该特性的名称。

示例

在 Given Name （名）、 Surname （姓）、 Title （职务）、 Description （说明）和 Internet EMail Address （因特网电子邮件地址）这些特性都可用时，本示例才创建所有的用户对象。策略名称为 Policy to Enforce the Presences of Attributes （强制验证特性存在策略），可

从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页。



如果以下特性：Given Name （名）、 Surname （姓氏）、 Title （职务）、 Description （说明）和 Internet EMail Address （因特网电子邮件地址）不可用，此操作将禁止创建对象。

3.7 名词标记

本节包含从策略构建器界面可以访问的所有名词标记的详细参照。

- “源名称”在第 292 页
- “文本”在第 292 页
- “唯一名称”在第 293 页
- “源 DN 不匹配”在第 295 页
- “XPath”在第 295 页

3.7.1 已添加的权利

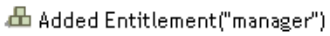
扩展到当前操作中授予的权利的值。

字段

名称

权利的名称。

示例



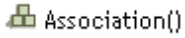
3.7.2 关联

从当前操作扩展到关联值。

示例

本示例运用了 Identity Manager 3.0 附带的预定义规则。有关预定义规则的更多信息，请参见“命令转换 - 发布者删除 - 禁用”在第 209 页 。

" 去除关联 " 操作使用 " 关联 " 标记检索当前操作的值。此规则将去除用户对象的关联，因此所有遇到的新事件都不会影响用户对象。



3.7.3 特性

扩展到当前操作和源数据存储区中的当前对象的某一特性的值。逻辑上，可以将它看作 " 操作特性 " 标记和 " 源特性 " 标记的组合。但不包括修改操作去除的值。

字段

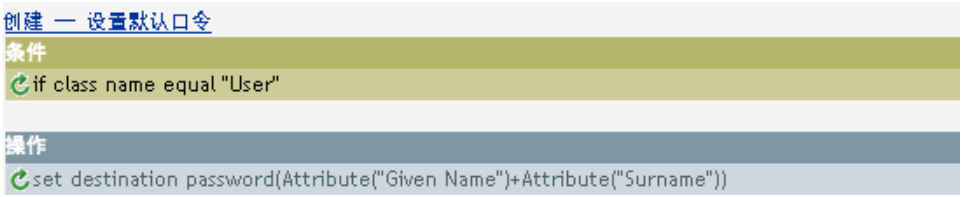
名称

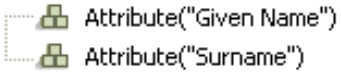
指定该特性的名称。

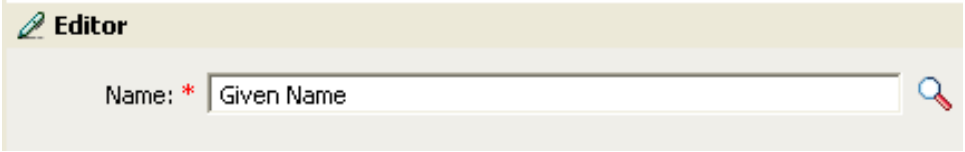
示例

本示例运用了 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “创建 - 设置默认口令”在第 213 页 。

" 设置目标口令 " 操作使用 " 特性 " 标记创建口令。该口令由 Given Name （名）特性和 Surname （姓）特性构成。在 " 自变量构建器编辑器 " 中可以浏览并选择想要使用的特性。



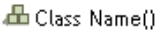




3.7.4 类名称

扩展到当前操作的对象类名称。

示例



3.7.5 目标特性

扩展到目标数据存储区中的当前对象、 DN 或关联的指定特性值。

字段

类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

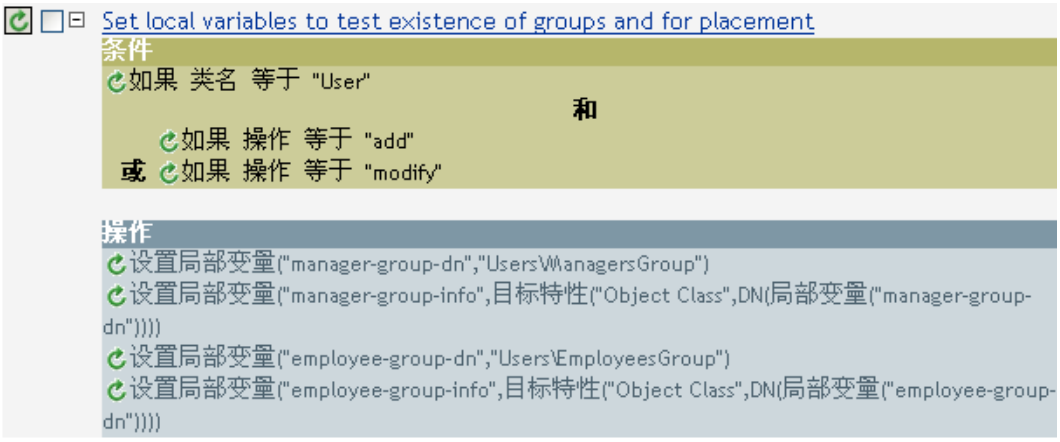
名称

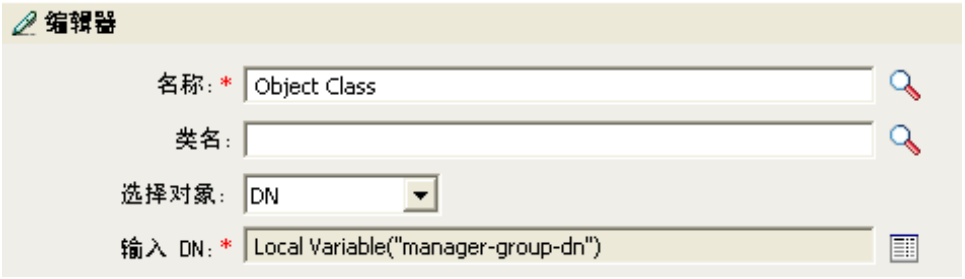
特性的名称。

示例

本示例属于 Govern Groups for User Based on Title （根据职务管理用户组）策略，可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页。

此策略使用自变量构建器创建 Destination Attribute （目标特性）。 " 设置局部变量 " 操作包含 " 目标特性 " 标记。





通过编辑器构建 " 目标特性 "。在本示例中，将特性设置为 Object Class。使用 DN 选择对象。 DN 的值是局部变量 manager-group-dn。

3.7.6 目标 DN

扩展到在当前操作中指定的目标 DN。

字段

转换

选择是否将 DN 转换为源数据存储区使用的格式。

开始

指定起始 RDN 索引：

- 索引 0 是最靠近根的 RDN

- 正索引是从最靠近根的 RDN 计算的偏移量
- 索引 -1 是最靠近叶的段
- 负索引是从最靠近叶的 RDN 到最靠近根的 RDN 的偏移量

长度

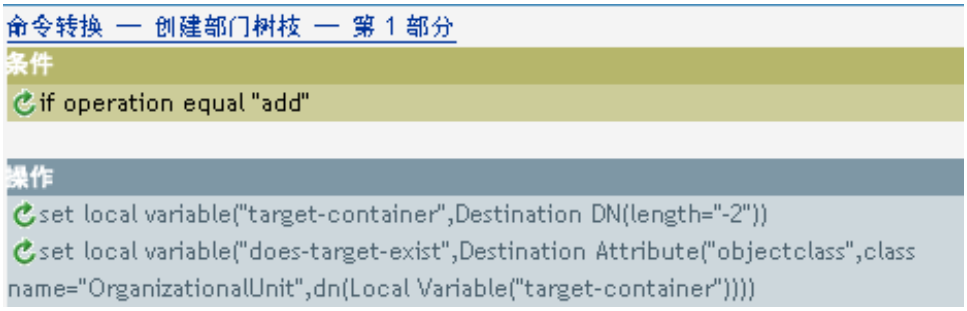
指定要包括的 RDN 数。负数可解释为 （段的总数 + 长度） + 1。例如，对于具有 5 个段的 DN，其长度 -1 = (5 + (-1)) + 1 = 5、 -2 = (5 + (-2)) + 1 = 4，依此类推。

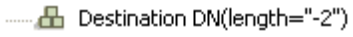
注释

如果起始索引和长度设置为默认值 {0,-1}，则使用整个 DN ；否则仅使用由起始索引和长度指定的 DN 部分。

示例

本示例使用 " 目标 DN" 标记设置局部变量 target-container 的值。如果部门树枝不存在，则该策略将为用户对象创建该树枝。此策略运用了 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “命令转换 - 创建部门树枝 - 第 1 部分和第 2 部分”在第 208 页 。

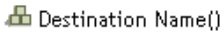




3.7.7 目标名称

扩展到目标 DN （在当前操作中指定）的无资格的相对判别名 (RDN)。

示例



3.7.8 权利

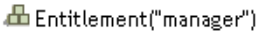
扩展到当前对象中授予的权利的值。

字段

名称

权利的名称。

示例



3.7.9 全局配置值

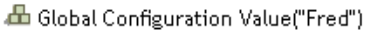
扩展到全局配置变量的值。

字段

名称

全局配置值的名称。

示例



3.7.10 局部变量

扩展到局部变量的值。

字段

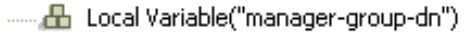
名称

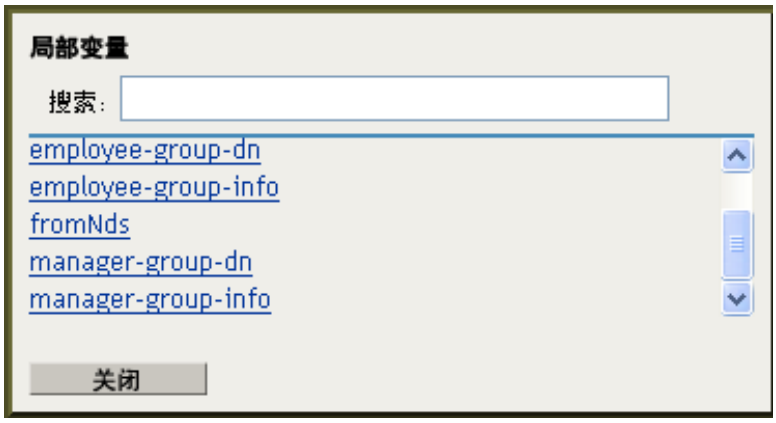
指定该局部变量的名称。

示例

本示例运用了 Govern Groups for User Based on Title （根据职务管理用户组）策略，可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。

" 添加目标对象 " 操作使用 " 局部变量 " 标记。





只有先前已在策略中使用过 " 设置局部变量 " 操作时，才可以使用局部变量。它将设置储存在局部变量中的值。在编辑器中，单击 " 浏览 " 图标，即可列出已定义的所有局部变量。选择正确的局部变量。

局部变量的值是 group-manager-dn。如果规则在它之前，则 " 设置局部变量 " 操作会将 group-manager-dn 定义为经理组 Users\ManagersGroup 的 DN。

3.7.11 命名口令

扩展到驱动程序的命名口令。

字段

名称

口令的名称。

示例

Named Password("password")

3.7.12 操作

扩展到当前操作的名称。

示例

Operation()

3.7.13 操作特性

扩展到当前操作的特性的值。但不包括修改操作去除的值。

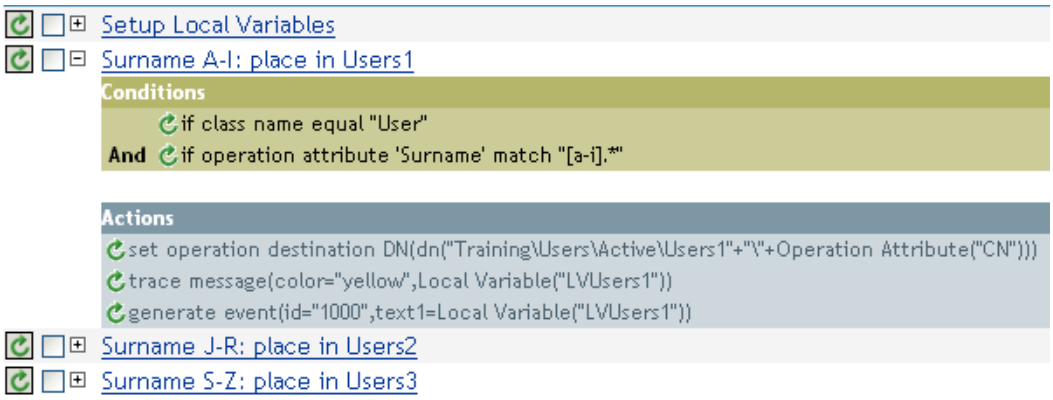
字段

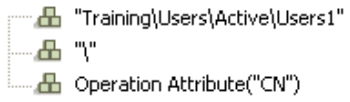
名称

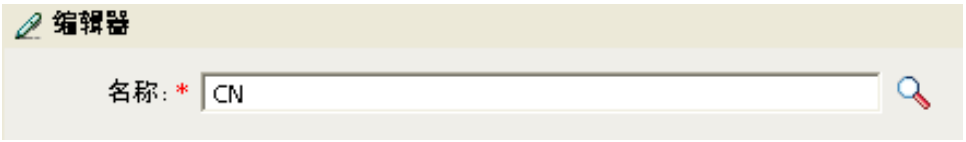
指定该特性的名称。

示例

本示例包含四条规则，它们根据 Surname （姓氏）特性的首字符实施对用户对象的布局策略。并生成一条跟踪讯息和一个自定义 Novell Audit 事件。策略名称为 Policy to Place by Surname （按姓氏布局策略），可从 Novell 支持万维网站点下载。详情请见 “可下载的 Identity Manager 策略”在第 31 页 。







" 设置操作目标 DN" 操作包含 " 操作特性 " 标记。 " 操作特性 " 标记将目标 DN 设置为 CN 特性。该规则采用 Training\Users\Active\Users 的环境，并添加一个 \ 以及 CN 特性的值。

3.7.14 操作属性

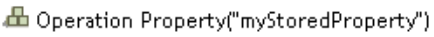
扩展到当前操作中指定的操作属性值。

字段

名称

指定操作属性的名称。

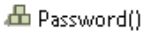
示例



3.7.15 口令

扩展到当前操作中指定的口令。

示例



3.7.16 已去除的特性

扩展到当前操作中正在去除的指定的特性值。它仅适用于修改操作。

字段

名称

指定该特性的名称。

示例

3.7.17 已去除的权利

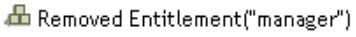
扩展到当前操作中已撤消的权利的值。

字段

名称

指定权利的名称。

示例



3.7.18 源特性

扩展到源数据存储区中的对象的某一特性的值。

字段

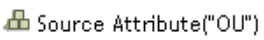
类名称

（可选）指定目标对象的类名称。若保留空白则使用当前对象的类名称。

名称

特性的名称。

示例



3.7.19 源 DN

扩展到当前操作的源 DN。

字段

转换

选择是否将 DN 转换为目标数据存储区使用的格式。

开始

指定起始 RDN 索引：

- 索引 0 是最靠近根的 RDN
- 正索引是从最靠近根的 RDN 计算的偏移量
- 索引 -1 是最靠近叶的段
- 负索引是从最靠近叶的 RDN 到最靠近根的 RDN 的偏移量

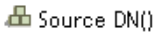
长度

指定要包括的 RDN 段数。负数可解释为 （段的总数 + 长度） + 1。例如，对于具有 5 个段的 DN，其长度 -1 = (5 + (-1)) + 1 = 5、 -2 = (5 + (-2)) + 1 = 4，依此类推。

注释

如果起始索引和长度设置为默认值 {0,-1}，则使用整个 DN ；否则仅使用由起始索引和长度指定的 DN 部分。

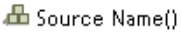
示例



3.7.20 源名称

扩展到目标 DN （在当前操作中指定）的无资格的相对判别名 (RDN)。

示例



3.7.21 文本

扩展到文本。

字段

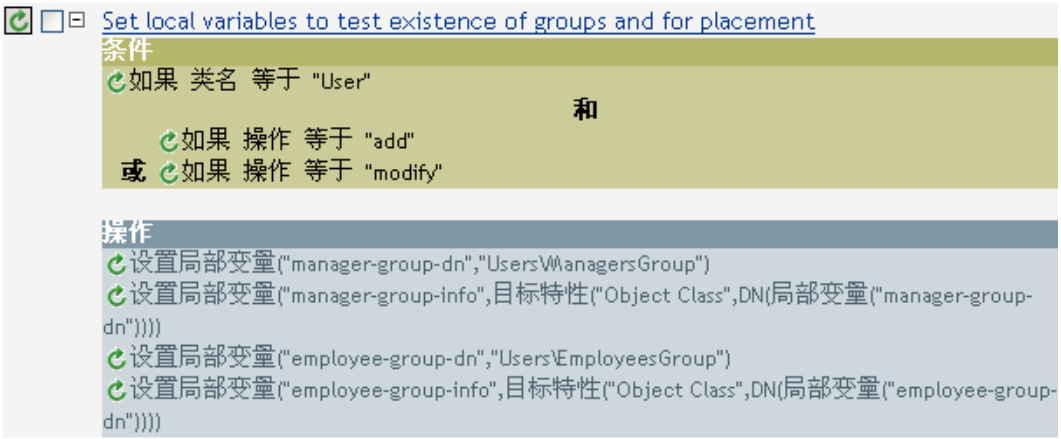
文本

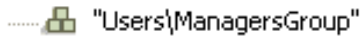
指定文本。

示例

本示例运用了 Govern Groups for User Based on Title （根据职务管理用户组）策略，可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。

" 设置局部变量 " 操作使用 " 文本 " 标记定义经理组的 DN。 " 文本 " 标记可以包含对象或纯文本。





" 文本 " 标记包含经理组的 DN。您可以浏览想要使用的对象，或将信息键入编辑器。

3.7.22 唯一名称

根据指定的准则扩展到基于模式的名称，该名称在目标数据存储区中是唯一的。

字段

名称

指定要检查唯一性的特性名称。

范围

指定检查唯一性的范围。

开始搜索

选择搜索的起始点。该起点可以为数据存储区的根，可以由 DN 指定，也可以为关联。

模式

指定使用自变量构建器生成唯一值时所用的模式。

计数器起点

指定查找唯一名称时需要使用的计数器的起始数字。

位数

指定计数器的宽度 （以位数为单位），默认为 1。选中 " 平板计数器 （初始为 0） " 复选框会在前面追加 0，以与位数长度相匹配。例如，如果位数宽度为 3，则将初始唯一值追加为 001，然后为 002，依次类推。

注释

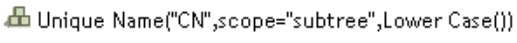
对提供的每一个模式，都会使用提供的特性名称、范围和搜索起点在目标数据存储区中执行一次查询。还将按顺序尝试每个指定模式，直到找到不会返回任何已找到对象的值为止。

如果已用尽所有指定模式，将在最终模式后追加一个计数器，然后重复尝试该模式 （每尝试一次，计数器都加一），直到此查询不返回任何实例。

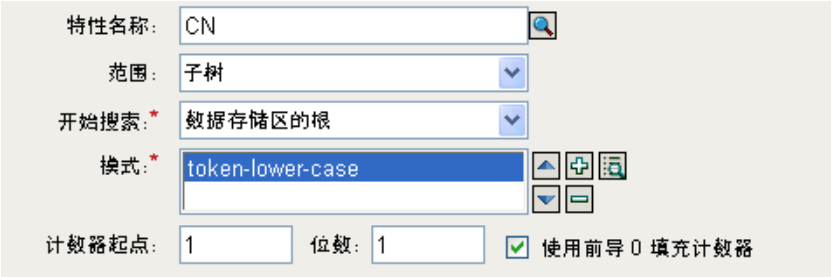
使用 " 计数器起点 " 字段可以为计数器设置不同的起点数字。计数器使用由 " 位数 " 字段指定的位数。如果该位数少于指定的位数，则使用 0 填充该计数器。如果该位数超过指定的位数，则无法生成唯一名称，且附加规则将返回一个错误状态。

如果目标数据存储区为 Identity Vault 且名称字段保留为空，则根据伪特性 "[Entry].rdn" 执行搜索，该特性表示对象的 RDN 而不考虑可能的命名特性。如果目标数据存储区是已连接的应用程序，则需要填写名称字段。

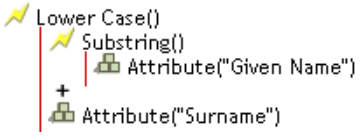
示例



在以下示例中， " 编辑器 " 窗格用于构造唯一名称自变量：



已构造了下面的模式来提供唯一名称：



如果此模式未生成唯一名称，将会追加一位，以增加到指定的位数。本示例中，通过追加位数会生成九个附加的唯一名称 (pattern1 - pattern9)，之后会发生错误。

3.7.23 源 DN 不匹配

扩展到当前操作中的源 DN 部分，它对应于与 If Source DN 条件的最近匹配项不相匹配的 DN 部分。

字段

转换

选择是否转换目标数据存储区使用的 DN 格式。

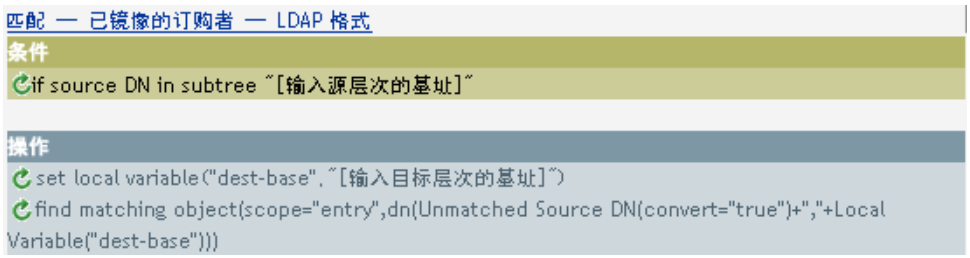
注释

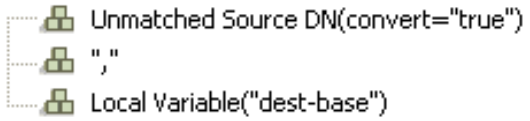
如果没有匹配项，则使用整个 DN。

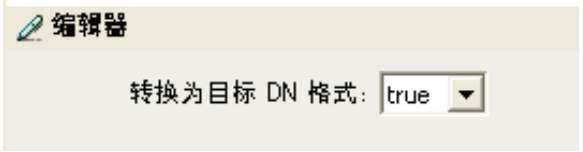
示例

本示例运用了 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “匹配 - 已镜像的订购者 - LDAP 格式”在第 218 页 。

" 查找匹配对象 " 操作使用 " 源 DN 不匹配 " 标记以 LDAP 格式构建匹配信息。它对源 DN 的不匹配部分进行匹配。







3.7.24 XPath

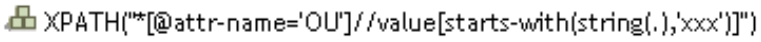
扩展到 XPath 1.0 表达式的求值结果。

字段

表达式

要计算的 XPath 1.0 表达式。

示例



3.8 动词标记

本部分包含使用策略构建器界面可用的所有动词的详细参照。

- “转义目标 DN”在第 296 页
- “转义源 DN”在第 297 页
- “小写”在第 297 页
- “语法分析 DN”在第 297 页
- “替换全部”在第 299 页
- “替换第一个”在第 300 页
- “子字符串”在第 301 页
- “大写字母”在第 302 页

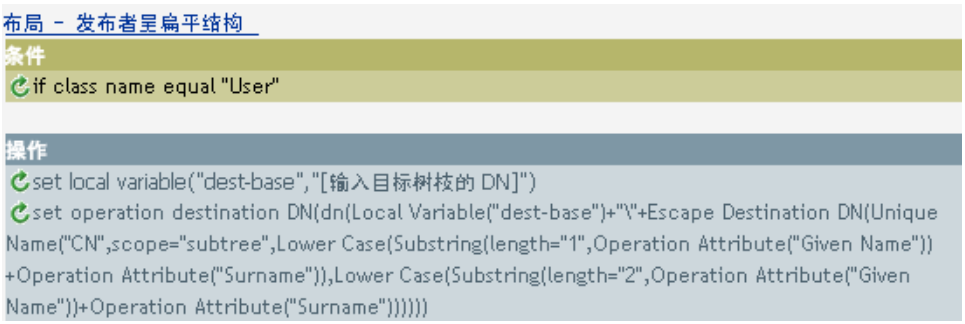
3.8.1 转义目标 DN

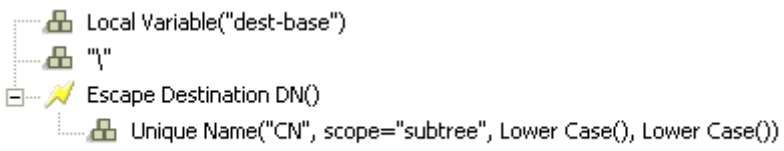
根据目标数据存储区的 DN 格式规则转义字符串。

示例

本示例运用了 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见“布局 - 发布者平面文件”在第 91 页。

" 设置操作目标 DN" 操作使用 " 转义目标 DN" 标记构建用户对象的目标 DN。



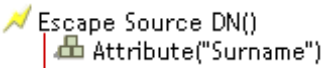


" 转义目标 DN" 标记采用 " 唯一名称 " 中的值，并将其设置为目标 DN 的格式。

3.8.2 转义源 DN

根据源数据存储区的 DN 格式规则转义字符串。

示例

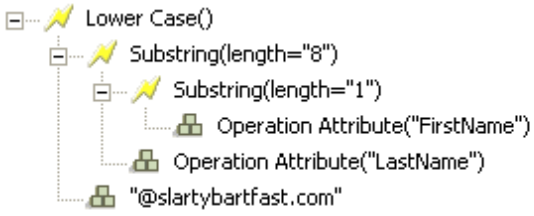


3.8.3 小写

将字符串中的字符转换为小写字母。

示例

此示例将电子邮件地址设置为 name@slartybartfast.com，其中的 name 等于 Given Name （名）的首字母加 Surname （姓氏）。策略名称为 Policy:Create E-mail from Given Name and Surname （使用名和姓氏创建电子邮件），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。



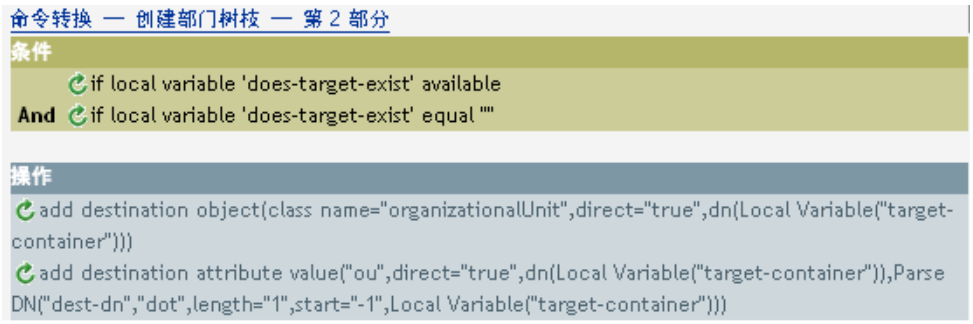
" 小写 " 标记将 " 设置目标特性值 " 操作中的所有信息设置为小写字母。

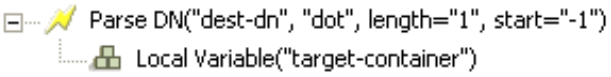
3.8.4 语法分析 DN

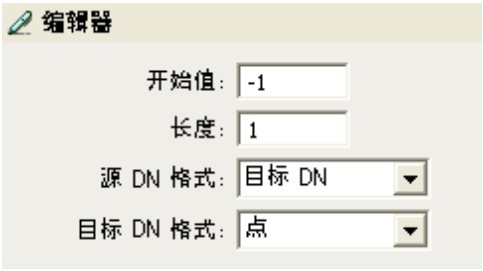
将 DN 转换为替换格式。

示例

在本示例中，使用 " 语法分析 DN" 标记构建 " 添加目标特性值 " 操作中的值。本示例运用了 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “命令转换 - 创建部门树枝 - 第 1 部分和第 2 部分”在第 208 页 。







" 语法分析 DN" 标记从源 DN 获取信息，然后将信息转换为点表示法。语法分析 DN 的信息储存在 OU 特性值中。

字段

开始

指定起始 RDN 索引：

- 索引 0 是最靠近根的 RDN
- 正索引是从最靠近根的 RDN 计算的偏移量
- 索引 -1 是最靠近叶的段
- 负索引是从最靠近叶的 RDN 到最靠近根的 RDN 的偏移量

长度

要包括的 RDN 的数量。负数可解释为 （段的总数 + 长度） + 1。例如，对于具有 5 个段的 DN，其长度 -1 = (5 + (-1)) + 1 = 5、 -2 = (5 + (-2)) + 1 = 4，依此类推。

源 **DN** 格式

指定对源 DN 进行语法分析所使用的格式。

目标 **DN** 格式

指定输出已经过语法分析的 DN 所使用的格式。

源 **DN** 分界符

如果源 DN 格式设置为自定义，则指定自定义源 DN 分界符集。

目标 **DN** 分界符

如果目标 DN 格式设置为自定义，则指定自定义目标 DN 分界符集。

注释

如果起始索引和长度设置为默认值 {0,-1}，则使用整个 DN ；否则仅使用由起始索引和长度指定的 DN 部分。

指定自定义 DN 格式时，构成分界符集的八个字符定义如下：

1. 已定义类型名称的布尔标志：0 表示未定义类型的名称，而 1 表示已定义类型的名称

2. Unicode 无映射字符布尔标志：0 表示不输出无法映射的 Unicode 字符，也不会将其解释为转义的十六进制字符串，如 \FEFF。 eDirectory 不接受以下 Unicode 字符：0xfeff、 0xfffe、0xfffd 和 0xffff。

3. 相对 RDN 分界符

4. RDN 分界符

5. 名称分隔符

6. 名称值分界符

7. 通配符字符

8. 转义字符

如果 RDN 分界符和相对 RDN 分界符是同一个字符，则名称方向为根右侧，否则其方向为根左侧。

如果分界符集超过八个字符，则将多余字符视为需要转义的字符，但这些字符没有其它特殊的含义。

3.8.5 替换全部

替换字符串中所有具体出现的某一正则表达式。

字段

正则表达式

指定与要替换的子字符串匹配的正则表达式。

替换为

指定替换字符串。

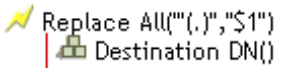
注释

有关创建正则表达式的详细信息，请参见：

- Sun 的 Java 万维网站点 (http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Pattern.html)
- Sun 的 Java 万维网站点 (http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Matcher.html#replaceAll (java.lang.String))

此处使用的模式选项为 CASE_INSENSITIVE、 DOTALL 和 UNICODE_CASE，但是可以通过使用适当的嵌入式转义符进行反转。

示例



3.8.6 替换第一个

替换字符串中第一次出现的某一正则表达式。

字段

正则表达式

指定与要替换的子字符串匹配的正则表达式。

替换为

指定替换字符串。

注释

将匹配实例替换为由 " 替换为 " 字段中指定的值指定的字符串。

有关创建正则表达式的详细信息，请参见：

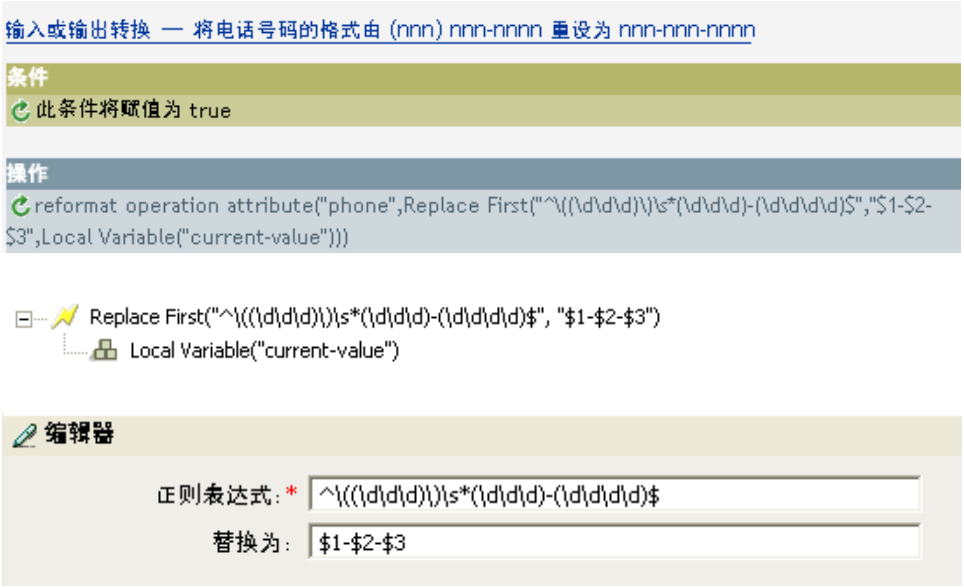
- Sun 万维网站点 (http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Pattern.html)
- Sun 万维网站点 (java.lang.String) (http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Matcher.html#replaceAll (java.lang.String))

此处使用的模式选项为 CASE_INSENSITIVE、 DOTALL 和 UNICODE_CASE，但是可以通过使用适当的嵌入式转义符进行反转。

示例

该示例将电话号码的格式由 (nnn)-nnn-nnnn 重设为 nnn-nnn-nnnn。所用的规则为 Identity Manager 3.0 附带的预定义规则。有关详细信息，请参见 “输入或输出转换 - 将电话号码格式重新从 (nnn) nnn-nnnn 设置为 nnn-nnn-nnnn”在第 215 页 。

" 重新设置操作特性的格式 " 操作将使用 " 替换第一个 " 标记。



正则表达式 ^\((\d\d\d)\)\s*(\d\d\d)-(\d\d\d\d)$ 表示 (nnn) nnn-nnnn，而正则表达式 $1-$2-$3 表示 nnn。此规则将电话号码的格式从 (nnn) nnn-nnnn 转换为 nnn-nnn-nnnn。

3.8.7 子字符串

抽取字符串的一部分。

字段

开始

指定起始字符的索引：

- 索引 0 表示第一个字符。
- 正索引表示距离字符串起始位置的偏移量。
- 索引 -1 表示最后一个字符。
- 负索引表示从字符串中最后一个字符开始的向前偏移量。

例如，如果起始索引指定为 -2，则从结尾第一个字符开始读取。如果起始索引指定为 -3，则从结尾第 2 个字符开始读取。

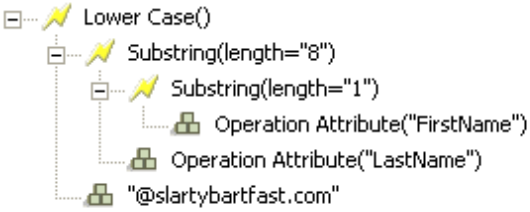
长度

子字符串中从开始位置起包含的字符数。负数可解释为 （字符总数 + 长度） + 1。例如， -1 表示全长或原始字符串长度。如果指定为 -2，则长度为全长 -1。对于具有 5 个字符的字符串，长度 -1 = (5 + (-1)) + 1 = 5 ； 长度 -2 = (5 + (-2)) + 1 = 4，依此类推。

示例

此示例将电子邮件地址设置为 name@slartybartfast.com，其中的 name 等于 Given Name （名）的首字母加 Surname （姓氏）。策略名称为 Policy:Create E-mail from Given Name and

Surname （使用名和姓氏创建电子邮件），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。



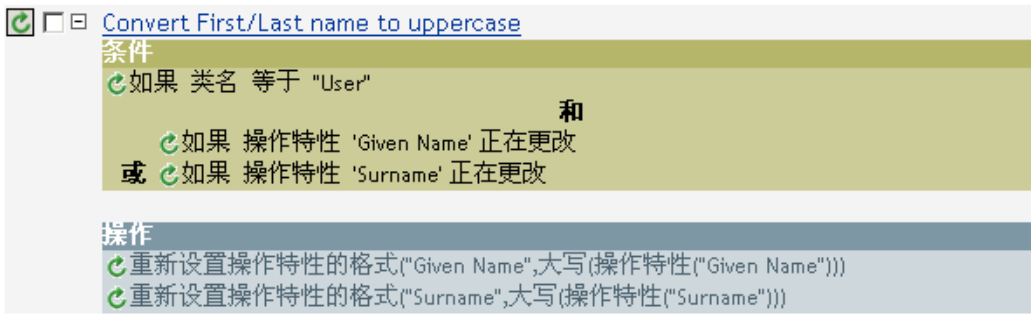
在 " 设置目标特性值 " 操作中，两次使用了 " 子字符串 " 标记。该标记使用 First Name （名）特性的第一个字符加上 Last Name （姓）特性的八个字符来构成一个子字符串。

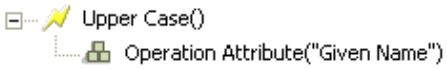
3.8.8 大写字母

将字符串中的字符转换为大写字母。

示例

在本示例中，将用户对象的名和姓特性转换成大写字母。策略名称为 Policy:Convert First/Last Name to Upper Case （将名 / 姓转换为大写字母），可从 Novell 支持万维网站点下载。有关详细信息，请参见 “可下载的 Identity Manager 策略”在第 31 页 。





3.9 值

本节包含常用策略构建器值的列表。

3.9.1 比较方式

方式	说明
case	逐个字符进行区分大小写比较。
nocase	逐个字符进行不区分大小写比较。
regex	整个字符串的正则表达式匹配项。默认情况下，不区分大小写，但可通过表达式中的转义进行更改。 请参见 Sun 的 Java 万维网站点 (http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Pattern.html) 和 Sun 的 Java 万维网站点 (http://java.sun.com/j2se/1.4/docs/api/java/util/regex/Matcher.html#matches())。 请注意，使用 CASE_INSENSITIVE、 DOTALL 和 UNICODE_CASE 模式选项，但可通过使用相应的嵌入转义反转这些模式选项。
src-dn	使用适用于源数据存储区的 DN 格式的语义进行比较。
dest-dn	使用适用于目标数据存储区的 DN 格式的语义进行比较。
numeric	数值比较。
octet	比较八位组 （Base64 编码）值。
structured	根据结构特性语法的比较规则，比较结构特性。

Novell 身份凭证供应策略

4

Identity Manager 3 的 Novell® 身份凭证供应策略能够同时将应用程序身份凭证供应给 Novell SecretStore® 和 Novell SecureLogin 身份凭证储存库，从而增强所有 Identity Manager 驱动程序的用户供应功能。此外，该产品还可以在需要无否决功能的环境中供应 SecureLogin 通行口令的问题和答案。

这些产品功能可以增强用户一次签到的体验，同时通过消除 SecureLogin 帐户信息的初始设置、向应用程序身份凭证提供附加的安全性，以及减少重复工作 （通常与供应用户的单一式签到身份凭证储存有关），来增加单一式签到技术的投资回报。此外，身份凭证供应策略可以使用 Identity Manager 策略自动取消供应应用程序身份凭证，以防止访问应用程序数据。

- “采用 Novell SecureLogin 的身份凭证供应策略”在第 305 页
- “实现采用 SecureLogin 的身份凭证供应策略”在第 307 页
- “Novell SecretStore 支持的身份凭证供应策略”在第 325 页
- “实施 SecretStore 支持的身份凭证供应策略”在第 327 页

4.1 采用 Novell SecureLogin 的身份凭证供应策略

可以使用身份凭证供应策略自动供应 SecureLogin 支持的应用程序身份凭证。此主题记录了在 Identity Manager 中配置对象和策略所需的步骤。它不包含任何 SecureLogin 组件的部署和配置信息。有关 SecureLogin 文档的信息，请参见 Novell SecureLogin 6.0 文档 (http://www.novell.com/documentation/securelogin60/index.html)。

要实现采用 SecureLogin 的身份凭证供应需要一个储存库对象、一个应用程序对象和多个策略。储存库和应用程序对象储存 SecureLogin 信息以供 Identity Manager 使用。如果使用这些策略，驱动程序就能使用身份凭证供应。有关详细信息，请参见 “实现采用 SecureLogin 的身份凭证供应策略”在第 307 页 。

也可配置下列选项：

- 身份凭证供应可由发布者通道、订购者通道提供，或由这两个通道同时提供。
- SecureLogin 同步可作为应用程序口令同步的一部分发生或由一些其它事件触发。
- 在不供应应用程序帐户的情况下，可以供应万维网服务身份凭证。
- 可以供应 SecureLogin 初始通行口令的问题和答案。

图 4-1 在第 306 页显示了一个典型而简洁的方案，其中包含向财务部门中的 SAP* 财务应用程序的新用户供应 SecureLogin 身份凭证。由于大多数应用程序只需要一个常用用户名和口令，而 SAP 用户供应需要更多的登录参数，因此使用此应用程序。

该部门通过 SAP HR 系统和 Identity Manager 向 Identity Vault 供应新用户。根据组织信息，将用户对象供应给在 Active Directory 上实现的部门鉴定树。在该树中，新用户鉴定到网络，因此 SecureLogin 身份凭证储存库也在该树中。因为 Identity Manager 随后将向用户供应各种财务应用程序，所以这些系统的用户身份凭证与 Active Directory 中的 SecureLogin 储存同步。

图 4-1 显示了供应的用户 Glen 的鉴定身份凭证。当 Glen 鉴定到他所在部门的 Active Directory 鉴定域并起动 SecureLogin 客户机时，他不需要输入或甚至不需要知道该系统上的口令就可以一次签到 SAP 财务帐户。

图 ***4-1*** 采用 *SecureLogin* 的身份凭证供应

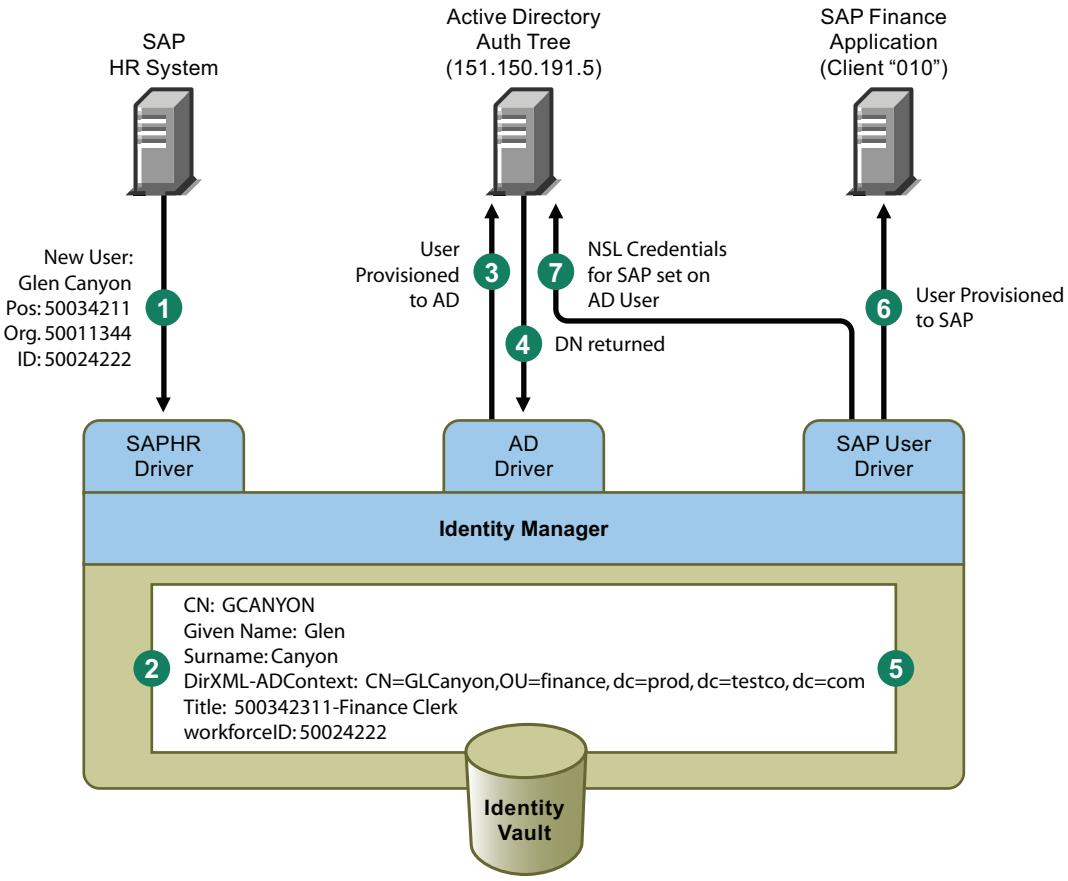


图 4-1 说明了下列步骤：

1. SAP HR 系统发布了有关新雇用员工 Glen Canyon 的数据。 Identity Manager SAP HR 驱动程序将处理此数据。
2. 在 Identity Vault 中创建新用户对象，其 CN 值为 GCANYON， workforceID 值为 50024222。由于该用户被指派给他所在公司的财务组织，因此他需要鉴定到 finance.prod.testco.com 域中的财务部门 Active Directory 服务器。同步该域的 Identity Manager Active Directory 驱动程序现在使用 Identity Vault 信息。
3. 将 Glen 供应给财务部门 Active Directory 服务器。
4. 配置该驱动程序以获得 Glen 完整的 LDAP 判别名：CN=GLCanyon,OU=finace,dc=prod,dc=testco,dc=com.
5. 该驱动程序将名称放入 Identity Vault 中的 GCANYON 用户的 DirXML-ADContext 特性中。

   现在所需的特性在 Identity Vault 中可用， SAP 用户管理驱动程序处理 GCANYON 对象的特性。
6. 因为 Glen 在财务组织，所以该驱动程序供应一个 SAP 财务服务器上的 SAP 用户帐户 GCANYON。

7. 成功创建该帐户后， SAP 用户管理驱动程序策略向 Glen 的 AD 用户帐户供应其 SAP 鉴定身份凭证。因为此命令为一个 " 添加 " 操作，所以这些策略也供应其 SecureLogin 通行口令问题和答案。

4.2 实现采用 SecureLogin 的身份凭证供应策略

可以任意自定义采用 SecureLogin 的身份凭证策略的实现。实现该策略的步骤有所不同，取决于安装 SecureLogin 的平台、供应的应用程序和所涉及的 Identity Manager 驱动程序。

若要实现采用 SecureLogin 的身份凭证供应策略，请参见下列主题：

- “满足采用 Novell SecureLogin 的身份凭证供应策略的要求”在第 307 页
- “扩展 Novell SecureLogin 的 LDAP 纲要”在第 307 页
- “确定 Novell SecureLogin 的部署配置参数”在第 308 页
- “创建 Novell SecureLogin 的储存库对象”在第 310 页
- “为 Novell SecureLogin 创建应用程序对象”在第 316 页
- “为 Novell SecureLogin 配置身份凭证供应策略”在第 322 页

4.2.1 满足采用 Novell SecureLogin 的身份凭证供应策略的要求

若要使用采用 SecureLogin 的身份凭证供应策略，则必须满足下列条件：

- 带有 Support Pack 1 的 Identity Manager 3.0
  - 必须安装在 eDirectory™ 8.7x 上；系统不支持 eDirectory 8.8。
  - 请校验 jsso.jar、 idmcp.jar 和 jnet.jar 是否位于 Identity Manager Java 库的标准位置。
- Novell SecureLogin 6.0 或更高版本

校验环境满足要求后，请转至 “扩展 Novell SecureLogin 的 LDAP 纲要”在第 307 页 。

4.2.2 扩展 Novell SecureLogin 的 LDAP 纲要

在 eDirectory 服务器上部署 SecureLogin 时，使用称为 ndsschema.exe 的工具扩展具有 SecureLogin 特性集的 eDirectory 纲要，这些特性用于储存用户和树枝对象的加密身份凭证、策略等。这些特性是：

- Prot:SSO Auth
- Prot:SSO Entry
- Prot:SSO Entry Checksum
- Prot:SSO Profile
- Prot:SSO Security Prefs
- Prot:SSO Security Prefs Checksum

这些特性特定于 eDirectory，需要这些特性以运行 SecureLogin 产品。 Identity Manager 3.0 Support Pack 1 中提供的供应 API 使用 LDAP 名称空间执行其功能，以便可以使用任意 SecureLogin 身份凭证储存。如果要为以上列出的特性提供 LDAP 映射，则必须使用与 SecureLogin 产品一起提供的第二种工具。该工具的名称是 ldapschema.exe，用于在 eDirectory 环境中向 eDirectory 特性提供 LDAP 名称空间映射。

运行 ldapschema.exe 之后，通过检查 iManager 中的 LDAP 组特性映射表校验映射。

1. 在 iManager 中，单击 "LDAP">"LDAP 选项 "。
2. 选择与承载 SecureLogin 的 eDirectory 服务器相关联的 LDAP 组。
3. 在 "LDAP 组 " 属性页中，选中 " 特性映射表 " 选项并校验已将上述特性映射到下列 *Primary LDAP Attributes*（LDAP 主属性）：
   - protocom-SSO-Auth-Data
   - protocom-SSO-Entries
   - protocom-SSO-Entries-Checksum
   - protocom-SSO-Profile
   - protocom-SSO-Security-Prefs
   - protocom-SSO-Security-Prefs-Checksum

扩展纲要后，请转至 “确定 Novell SecureLogin 的部署配置参数”在第 308 页。

4.2.3 确定 Novell SecureLogin 的部署配置参数

若要提供图 4-1 中阐述的部署方案中描述的同步功能，则首先要收集所有与 Identity Manager 和 SecureLogin 环境相关的业务处理信息。您可以打印表 4-1 “SecureLogin 的身份凭证供应策略工作表”在第 308 页并将其用作记录信息的工作表。

表 ***4-1*** *SecureLogin* 的身份凭证供应策略工作表

所需的配置信息	信息
1) 将为 SecureLogin 一次签到供应配置哪些应用程序？	
2) 请校验在鉴定服务器上预配置的 SecureLogin 应用程序定义以及这些定义是否可由供应给这些系统的新用户继承。	
3) SecureLogin 储存库服务器的 DNS 名称或 IP 地址。	
4) SecureLogin 储存库服务器的 SSL LDAP 端口。	
5) SecureLogin 储存库服务器管理员的完全限定的 LDAP 判别名。	
6) SecureLogin 储存库服务器的管理员口令。	
7) 从 SecureLogin 服务器导出的 SSL 证书的完整路径和名称。对于 Identity Manager 服务器而言，该证书必须是本地的。	
8) 确定是多个驱动程序使用一个 SecureLogin 储存库，还是每个驱动程序使用单独的储存库。	
9) 每个 SecureLogin 应用程序的应用程序 ID。	
10) 查找每个应用程序所需的所有鉴定密钥。例如，用户名、口令、客户机和语言。每个应用程序的鉴定密钥可能会有所不同。	

所需的配置信息	信息
11) 确定是否可以使用静态值设置所有的鉴定密钥值。	
12) 对于每个用户不同 （或可能不同）的非静态值，请记录非静态信息 （事件信息或 Identity Vault 特性值）的源。	
13) 如果实施 SecureLogin 供应的驱动程序也将口令同步到目标应用程序，请确定 SecureLogin 供应发生的时间是在目标应用程序服务器中设置口令之前还是之后。	
14) 储存库和应用程序对象所储存的驱动程序对象的名称。（可以为不同的驱动程序。）	
15) 确定目标应用程序的用户对象 DN。	
16) 如果您正在实现 SecureLogin 通行口令，请确定通行口令的问题和答案。	问题：答案：

供应配置数据的示例

下面的示例数据使用供应方案向 Finance Active Directory 鉴定树中的用户供应 SAP 财务服务器的用户 SecureLogin 身份凭证：

表 ***4-2*** *SecureLogin* 身份凭证供应策略工作表的示例

所需的配置信息	信息
1) 将为 SecureLogin 一次签到供应配置哪些应用程序？	SAP 财务应用程序
2) 请校验在鉴定服务器上预配置的 SecureLogin 应用程序定义以及这些定义是否可由供应给这些系统的新用户继承。	已校验
3) SecureLogin 储存库服务器的 DNS 名称或 IP 地址。	151.150.191.5
4) SecureLogin 储存库服务器的 SSL LDAP 端口。	636
5) SecureLogin 储存库服务器管理员的完全限定的 LDAP 判别名。	cn=admin,ou=prod,dc=testco,dc=.com
6) SecureLogin 储存库服务器的管理员口令。	dixml
7) 从 SecureLogin 服务器导出的 SSL 证书的完整路径和名称。对于 Identity Manager 服务器而言，该证书必须是本地的。	c:\novell\nds\FinanceAD.cer
8) 确定是多个驱动程序使用一个 SecureLogin 储存库，还是每个驱动程序使用单独的储存库。	在本示例中，只有一个储存库。
9) 每个 SecureLogin 应用程序的应用程序 ID。	SAP - 151.150.191.27

所需的配置信息	信息
10) 查找每个应用程序所需的所有鉴定密钥。例如，用户名、口令、客户机和语言。每个应用程序的鉴定密钥可能会有所不同。	SAP 客户机 010 登录参数客户机 SAP 客户机 010 登录参数语言 SAP 客户机 010 登录参数用户名 SAP 客户机 010 登录参数口令
11) 确定是否可以使用静态值设置所有的鉴定密钥值。	SAP 客户机 010 登录参数客户机："010" SAP 客户机 010 登录参数语言："EN"
12) 对于每个用户不同 （或可能不同）的非静态值，请记录非静态信息 （事件信息或 Identity Vault 特性值）的源。	SAP 客户机 010 登录参数用户名：Identity Vault 特性 "sapUsername"SAP 客户机 010 登录参数口令：事件 <password>
13) 如果实施 SecureLogin 供应的驱动程序也将口令同步到目标应用程序，请确定 SecureLogin 供应发生的时间是在目标应用程序服务器中设置口令之前还是之后。	之后
14) 储存库和应用程序对象所储存的驱动程序对象的名称。（可以为不同的驱动程序。）	SAP 驱动程序
15) 确定目标应用程序的用户对象 DN。	Identity Vault 特性 "DirXML-ADContext"
16) 如果您将供应 SecureLogin 的通行口令，请确定通行口令的问题和答案。	问题：" 员工代码是什么？ " 答案：Identity Vault 特性 "workforceID"

其它环境信息：

- 财务部门 AD 树将成为所有财务应用程序的 SecureLogin 储存库。
- 所有财务部门供应的驱动程序都位于名为 Finance Drivers 的驱动程序集中。
- 如果 Identity Vault 的 employeeStatus 特性值设置为 "I"，则必须删除 SAP 用户帐户，该帐户的 SecureLogin 身份凭证也要从 Active Directory 用户中去除。

确定所有配置数据后，请转至 “创建 Novell SecureLogin 的储存库对象”在第 310 页 。

4.2.4 创建 Novell SecureLogin 的储存库对象

储存库对象储存 SecureLogin 的静态配置信息。储存库信息独立于使用应用程序身份凭证的应用程序。无论已连接的系统 （例如：SAP、 PeopleSoft*、 Notes* 等）是什么，此信息都适用于所有的供应事件。储存库对象可以在 Designer 或 iManager 中创建。

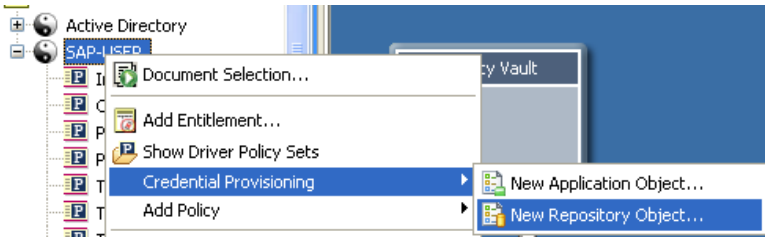
- “在 Designer 中，创建 Novell SecureLogin 的储存库对象”在第 310 页
- “在 iManager 中，创建 Novell SecureLogin 的储存库对象”在第 313 页

在 Designer 中，创建 Novell SecureLogin 的储存库对象

在 Designer 中创建储存库对象的方法有很多种，下面的方法是其中之一：

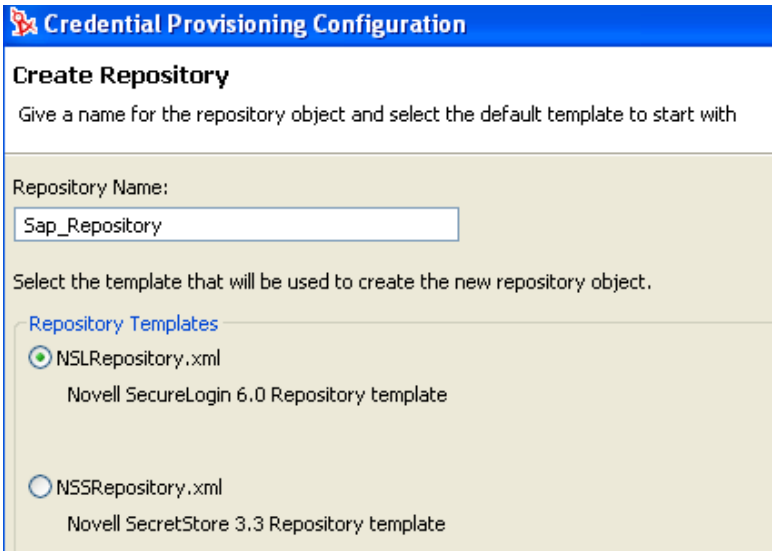
1 在大纲视图中，右击您希望在其中储存储存库对象的驱动程序对象。

2 单击 *Credential Provisioning* > *New Repository Object*（身份凭证供应 > 新建储存库对象）。



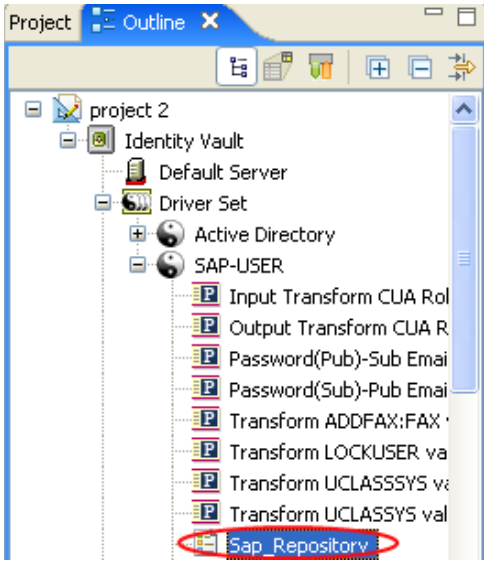
3 指定储存库对象的名称。

4 选择 *NSLRepository.xml*，以使用 SecureLogin 模板。

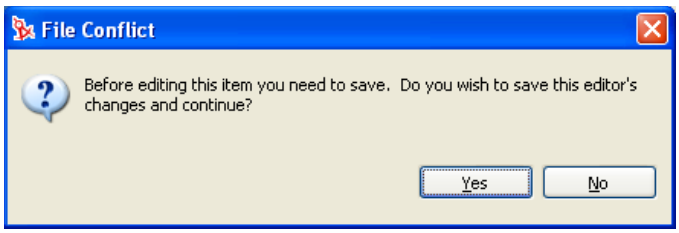


5 单击 " 确定 "。

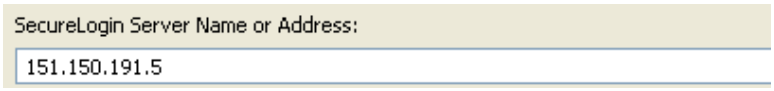
6 在 " 大纲 " 视图中，双击储存库对象以添加配置信息。



7 单击 " 是 " 保存新的储存库对象。

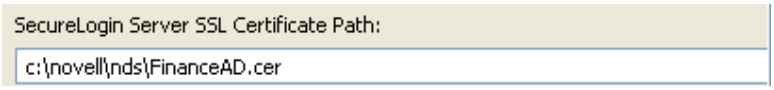


8 指定 SecureLogin 服务器的 DNS 名称或 IP 地址。请参见工作表项 3)。



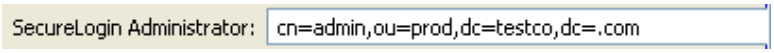
9 指定 SecureLogin 服务器的 SSL 端口。请参见工作表项 4)。

10 指定从 SecureLogin 服务器导出的 SSL 证书的完整路径。此路径必须包括证书名称，而且对于 Identity Manager 服务器而言必须是本地的。请参见工作表项 7)。



SecureLogin 服务器可以在多种平台上运行。有关如何导出 SSL 证书的信息，请参考特定于平台的文档。

11 指定 SecureLogin 管理员的完全限定的 LDAP 判别名。请参见工作表项 5)。

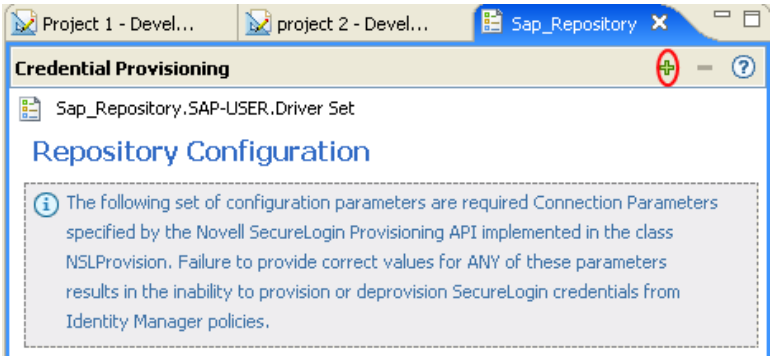


12 单击 " 设置口令 "。

13 指定并输入 SecureLogin 管理员口令两次，然后单击 " 确定 "。请参见工作表项 6)。

14 查看信息，然后单击 " 保存 " 图标 保存信息。

15 （可选）如果您希望为储存库对象创建其它的配置参数，请单击 *Add new item* （添加新项） 图标。

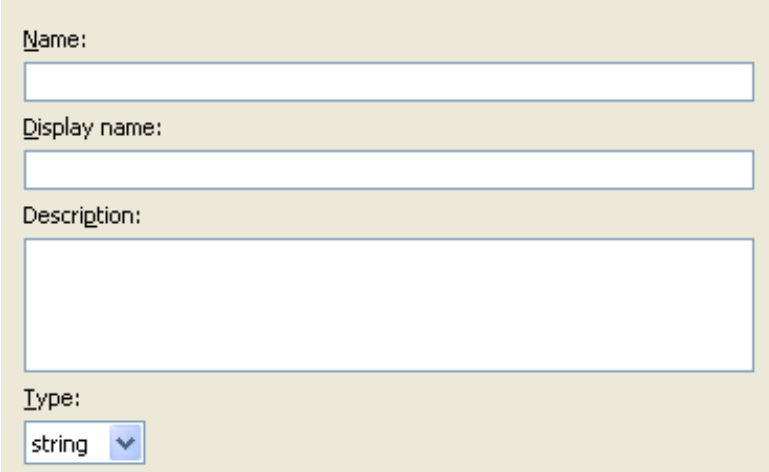


15a 指定参数的名称。

15b 指定参数的显示名称。

15c 指定参数说明以供参考。

参数以字符串的形式储存。



15d 单击 " 确定 "。

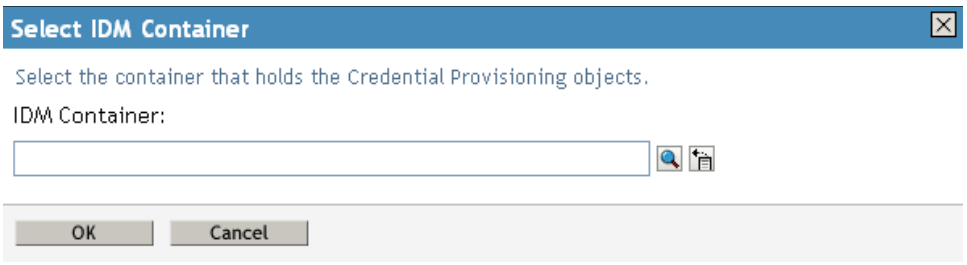
15e 单击 " 保存 " 图标 保存储存库对象。

创建储存库对象后，请转至 “为 Novell SecureLogin 创建应用程序对象”在第 316 页 。

在 **iManager** 中，创建 **Novell SecureLogin** 的储存库对象

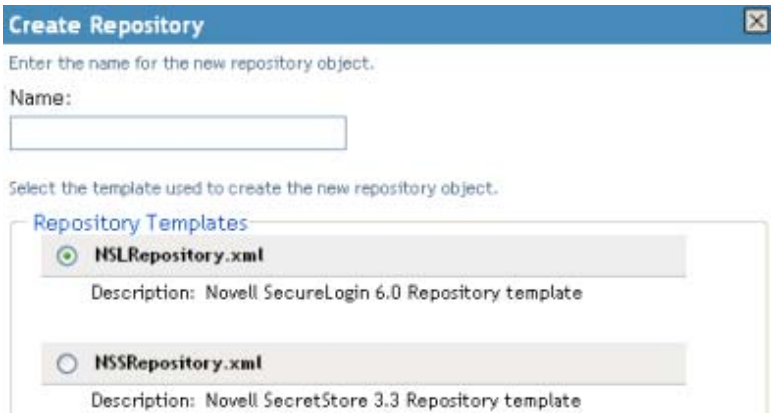
1 在 iManager 中，选择 " 身份凭证供应 ">" 配置 "。

2 浏览至将在其中储存储存库对象的驱动程序对象，并选择该对象，然后单击 " 确定 "。



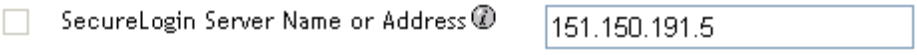
3 单击 " 新建 " 创建储存库。

4 指定储存库对象的名称，然后选择 *NSLRepository.xml* 以使用 SecureLogin 模板创建储存库。

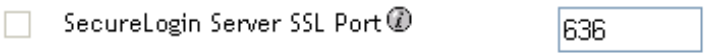


5 单击 " 确定 "。

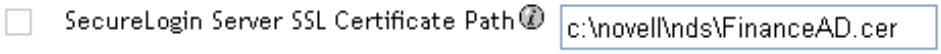
6 指定 SecureLogin 服务器的 DNS 名称或 IP 地址。请参见工作表项 3)。



7 指定 SecureLogin 服务器的 SSL 端口。请参见工作表项 4)。



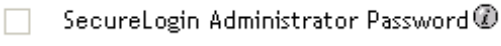
8 指定从 SecureLogin 服务器导出的 SSL 证书的完整路径。此路径必须包括证书名称，而且对于 Identity Manager 服务器而言必须是本地的。请参见工作表项 7)。



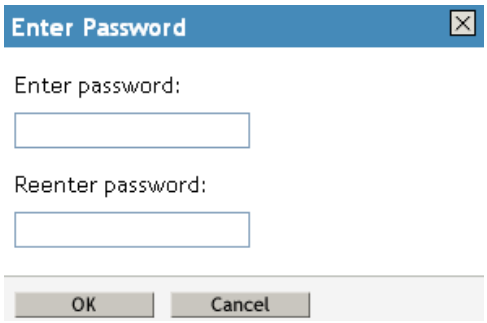
SecureLogin 服务器可以在多种平台上运行。有关导出 SSL 证书的步骤，请参考特定于平台的文档。

9 指定 SecureLogin 管理员的完全限定的 LDAP 判别名。请参见工作表项 5)。

10 单击 " 设置口令 "。

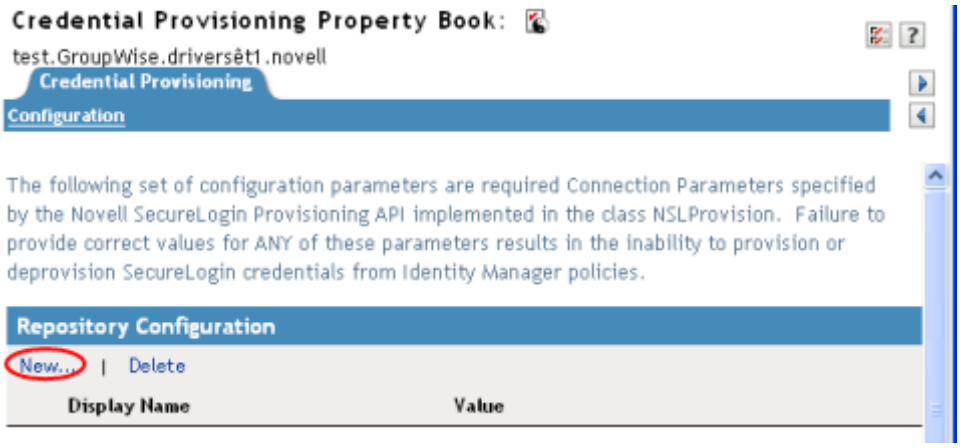


11 指定并输入 SecureLogin 管理员口令两次，然后单击 " 确定 "。请参见工作表项 6)。



12 查看指定的值，然后单击 " 确定 "。

13 （可选）如果您需要为储存库创建其它的配置参数，请单击 " 新建 "。

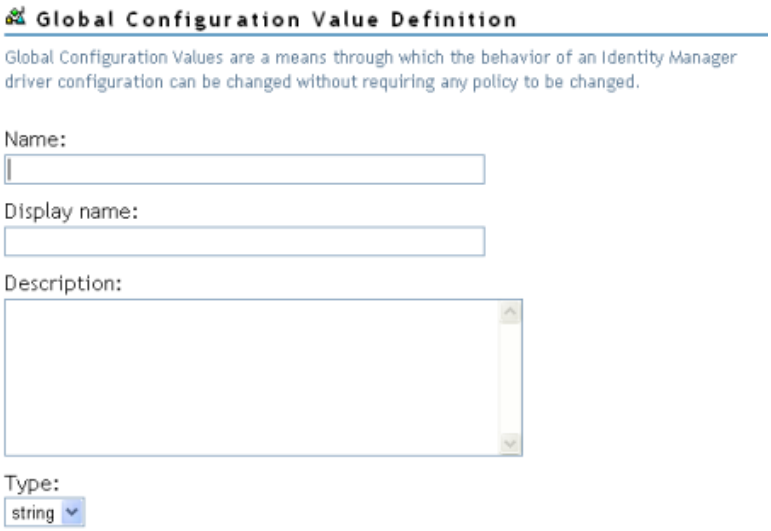


13a 指定参数的名称。

13b 指定参数的显示名称。

13c 指定用作参照的参数说明。

参数以字符串的形式储存。



13d 单击 " 确定 "。

创建储存库对象后，请转至 “在 iManager 中，创建 Novell SecureLogin 的应用程序对象”在第 319 页 。

4.2.5 为 Novell SecureLogin 创建应用程序对象

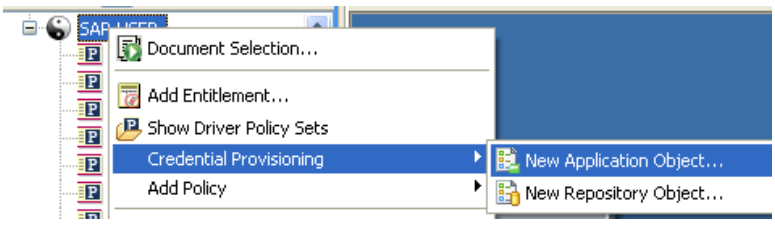
应用程序对象储存 SecureLogin 的应用程序鉴定参数值。应用程序信息特定于使用应用程序身份凭证的应用程序（例如：GroupWise® 客户机信息或 SAP 数据库客户机信息）。可以在 Designer 或 iManager 中创建应用程序对象。

- “在 Designer 中，创建 Novell SecureLogin 的应用程序对象”在第 316 页
- “在 iManager 中，创建 Novell SecureLogin 的应用程序对象”在第 319 页

在 Designer 中，创建 Novell SecureLogin 的应用程序对象

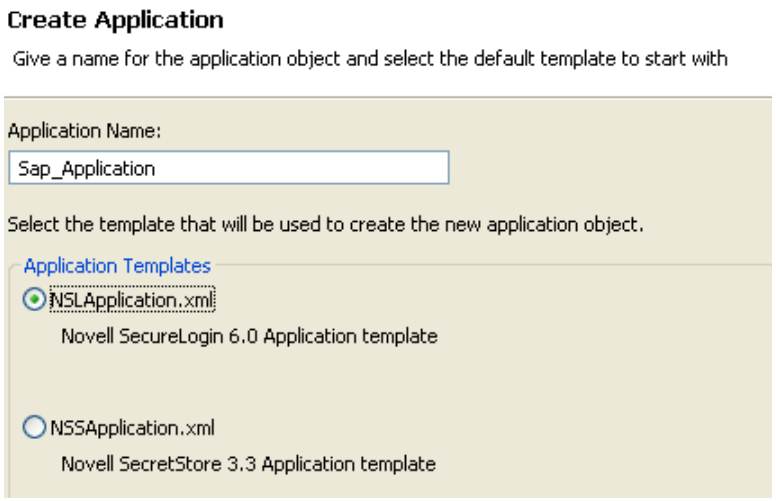
在 Designer 中，用于创建应用程序对象的方法有很多种，下面的方法是其中之一：

1. 在 " 大纲 " 视图中，右击要储存应用程序对象的驱动程序对象。
2. 单击 " 身份凭证供应 ">" 新建应用程序对象 "。



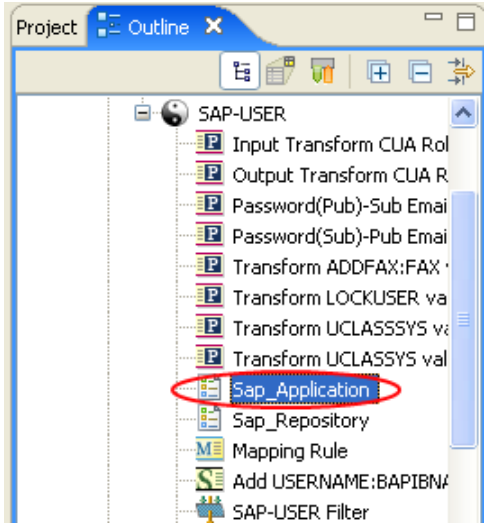
3. 指定应用程序对象的名称。

4 选择 *NSLApplication.xml*，以使用 SecureLogin 模板。

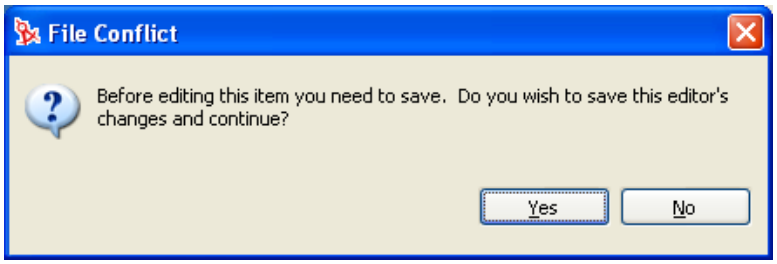


5 单击 " 确定 "。

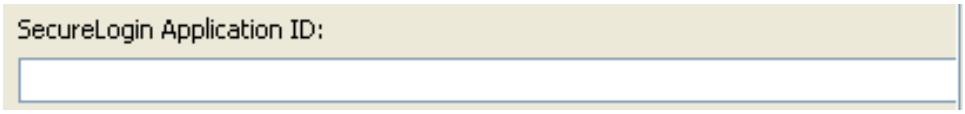
6 在 " 大纲 " 视图中，双击应用程序对象以添加配置信息。



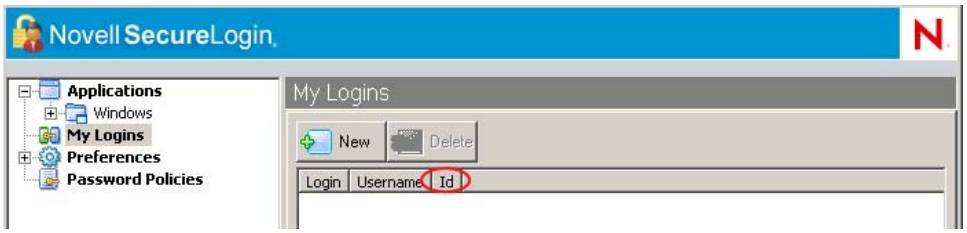
7 单击 " 是 " 保存新的应用程序对象。



8 指定 SecureLogin 应用程序 ID。请参见工作表项 9)。

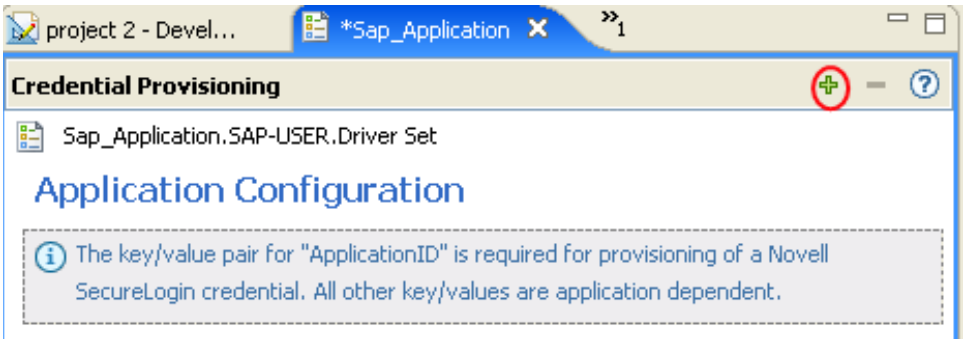


若要在 SecureLogin 中查找应用程序 ID，请单击 *My Logins*（我的登录）。应用程序 ID 储存在 *Id* 字段中。



9 单击 " 保存 " 图标 保存应用程序。

10 单击 " 添加新项 " 图标 ，添加应用程序所需的鉴定密钥。

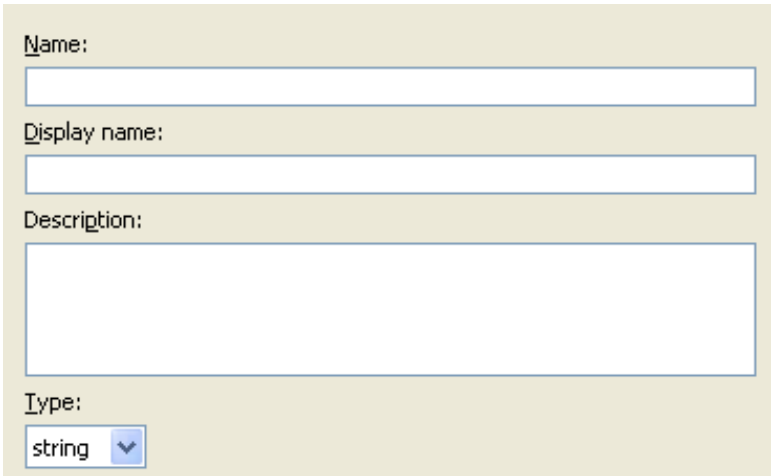


10a 指定鉴定密钥的名称。

10b 指定鉴定密钥的显示名称。

10c 指定鉴定密钥的说明以供参考。

鉴定密钥以字符串的形式储存。



10d 单击 " 确定 "。

10e 对每个需要输入的新鉴定密钥重复步骤 10。

若要查找应用程序的鉴定密钥，请在应用程序中手动创建一个用户的 SecureLogin 身份凭证，然后以此用户身份登录。用户登录后，鉴定密钥的信息将显示在 SecureLogin 管理窗口的 " 我的登录 " 下。

11 如果鉴定密钥值是所有用户身份凭证共享的静态值，请指定该值。

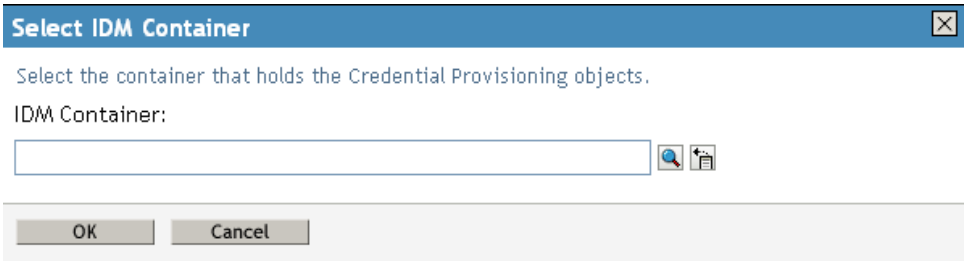
12 单击 " 保存 " 图标 保存应用程序。

创建应用程序对象后，请转至 “为 Novell SecureLogin 配置身份凭证供应策略”在第 322 页 。

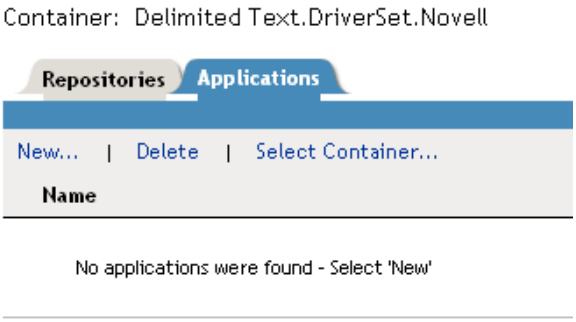
在 **iManager** 中，创建 **Novell SecureLogin** 的应用程序对象

1 在 iManager 中，选择 " 身份凭证供应 ">" 配置 "。

2 浏览至要储存应用程序对象的驱动程序对象并选择该对象，然后单击 " 确定 "。

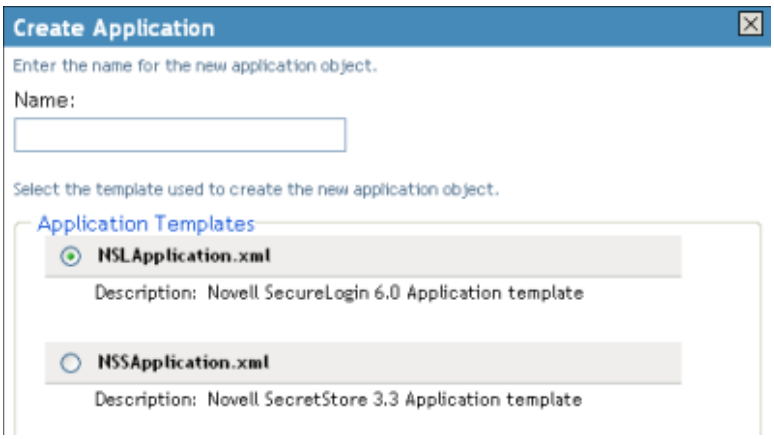


3 选择 " 应用程序 " 选项卡，然后单击 " 新建 "。



4 指定应用程序对象的名称。

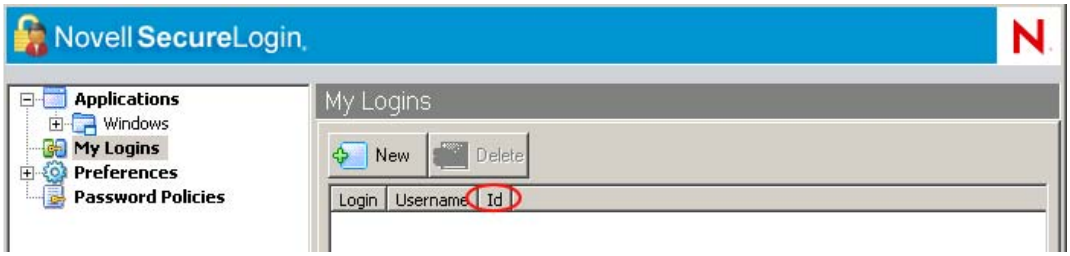
5 选择 *NSLApplication.xml* 以使用 SecureLogin 模板创建应用程序。



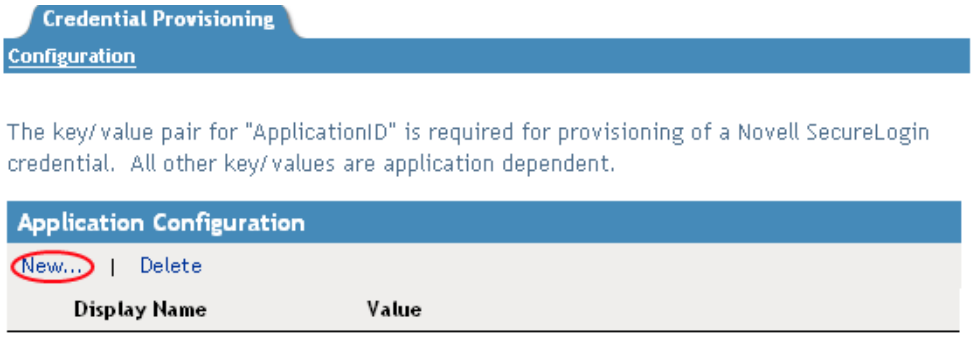
6 单击 " 确定 "。

7 指定 *SecureLogin Application ID* （SecureLogin 应用程序 ID）。请参见工作表项 9)。

若要在 SecureLogin 中查找应用程序 ID，请单击 " 我的登录 "。应用程序 ID 储存在 *Id* 字段中。



8 单击 " 新建 " 创建鉴定密钥参数。请参见工作表项 10)。

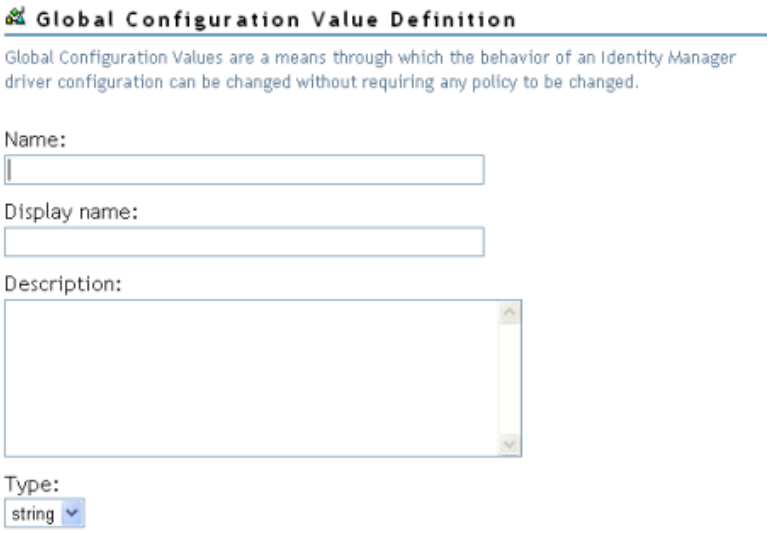


8a 指定鉴定密钥的名称。

8b 指定鉴定密钥的显示名称。

8c 指定鉴定密钥的说明以供参考。

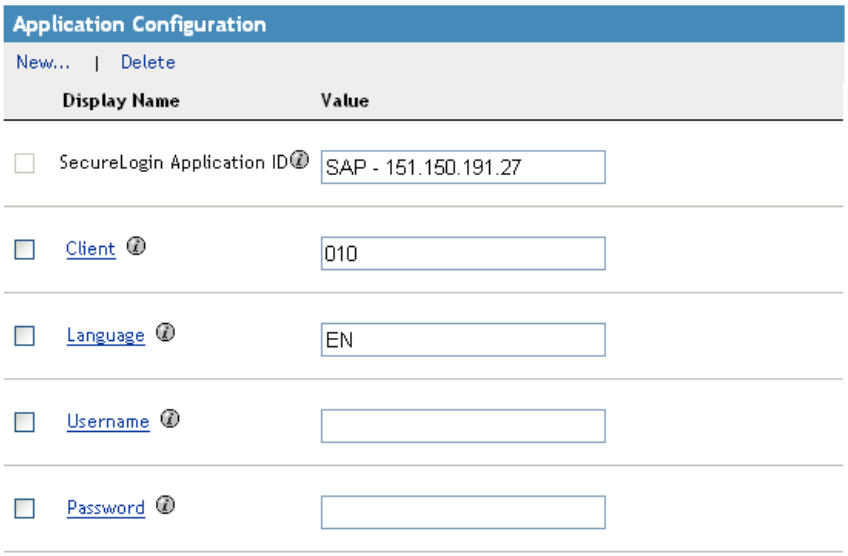
鉴定密钥以字符串的形式储存。



若要查找应用程序的鉴定密钥，请在应用程序中手动创建一个用户的 SecureLogin 身份凭证，然后以此用户身份登录。用户登录后，鉴定密钥的信息将显示在 SecureLogin 管理窗口的 " 我的登录 " 下。

8d 单击 " 确定 "。

8e 如果鉴定密钥值为静态，则指定该值，然后单击 " 确定 "。



创建应用程序对象后，请转至 “为 Novell SecureLogin 配置身份凭证供应策略”在第 322 页 。

4.2.6 为 Novell SecureLogin 配置身份凭证供应策略

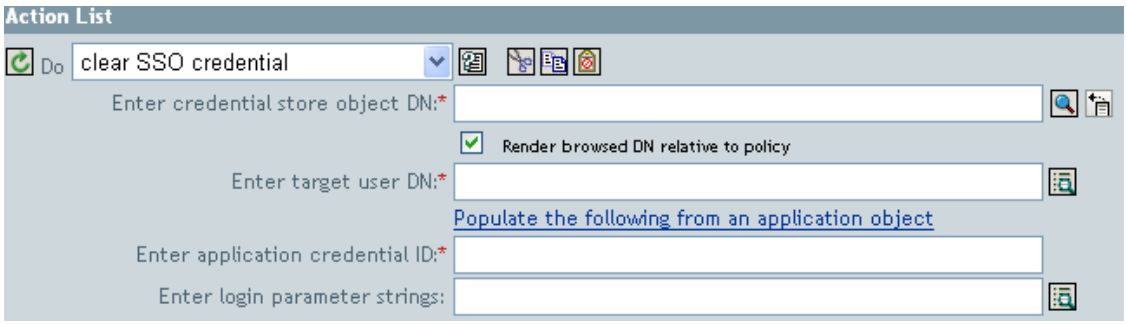
创建储存库和应用程序对象后，需要创建一些用于供应 SecureLogin 信息的策略。这些策略使用储存在储存库和应用程序对象中的信息。在策略构建器中，有三种操作可用于供应 SecureLogin 身份凭证：

- “清除 SSO 身份凭证”在第 322 页
- “设置 SSO 身份凭证”在第 322 页
- “设置 SSO 通行口令”在第 323 页

清除 SSO 身份凭证

" 清除 SSO 身份凭证 " 操作允许清除 SSO 身份凭证，因此可以取消对对象的供应。

图 *4-2* 清除 *SSO* 身份凭证

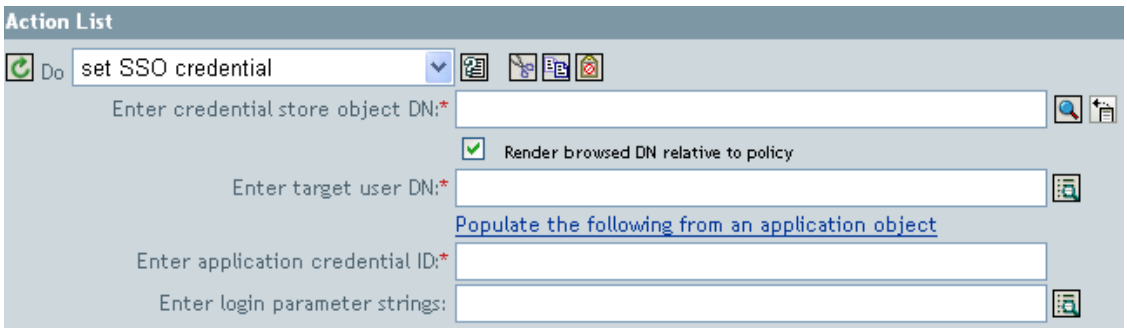


- 输入身份凭证储存对象 **DN**：浏览至储存库对象并选择该对象。
- 输入目标用户 **DN**：使用自变量构建器创建目标用户的 DN。请参见工作表项 15)。
- 输入应用程序身份凭证 **ID**：指定应用程序 ID。请参见工作表项 9)。
- 输入登录参数字符串：起动字符串构建器，然后输入应用程序的每个鉴定密钥。请参见工作表项 10)。

设置 SSO 身份凭证

创建用户对象或修改口令后， " 设置 SSO 身份凭证 " 操作允许设置 SSO 身份凭证。

图 *4-3* 设置 *SSO* 身份凭证



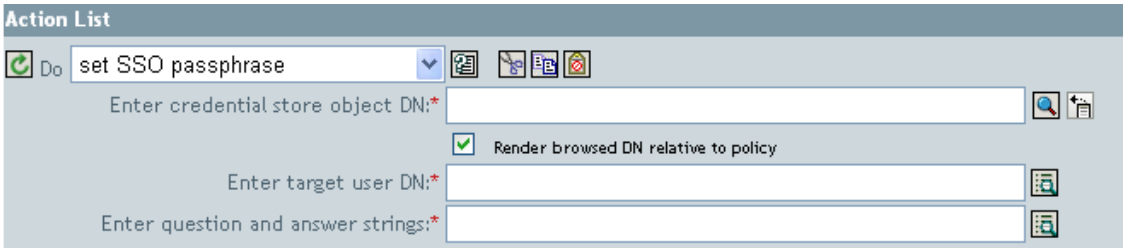
- 输入身份凭证储存对象 **DN**：浏览至储存库对象并选择该对象。

- 输入目标用户 **DN**：使用自变量构建器创建目标用户的 DN。请参见工作表项 15)。
- 输入应用程序身份凭证 **ID**：指定应用程序 ID。请参见工作表项 9)。
- 输入登录参数字符串：起动字符串构建器，然后输入应用程序的鉴定密钥每个。请参见工作表项 10)。

设置 **SSO** 通行口令

" 设置 SSO 通行口令 " 操作允许创建 SecureLogin 通行口令，并在供应用户对象时对其负责。

图 ***4-4*** 设置 *SSO* 通行口令



- 输入身份凭证储存对象 **DN**：浏览至储存库对象并选择该对象。
- 输入目标用户 **DN**：使用自变量构建器创建目标用户的 DN。请参见工作表项 15)。
- 输入问题和答案的字符串：起动字符串构建器，然后输入通行口令的问题和答案。请参见工作表项 16)。

身份凭证供应策略的示例

可以实施和自定义这些供应策略以满足环境的需要。下面的示例说明如何对图 4-1 在第 306 页演示的方案实施这些策略。

在此财务方案中，在 SAP 中成功设置口令后会出现 SecureLogin 供应。大多数的必需参数都是静态配置的，可用于储存库对象和应用程序对象中的所有策略。但是，也存在一些非静态数据参数 （sapUsername、口令、 DirXML-ADContext 和 workforceID），它们仅在完成 SAP 用户管理驱动程序的 <add> 或 <modify-password> 命令并将 <output> 状态文档从 SAP 用户管理驱动程序 Shim 返回之后才可以使用。<ouput> 文档不再包含任何订购者通道操作特性，并丢失了命令的用户环境，因此阻止查询对象。因此，必须执行以下操作：

- 请确保由 SAP 用户驱动程序的订购者创建策略实施所出现的非静态数据参数。
- 在发出 SAP 用户驱动程序 Shim 的订购者命令之前，超速缓存供应运算所需的非静态参数。
- 在成功完成命令后，检索 SecureLogin 供应要使用的超速缓存的数据。

注释：策略样本在 Identity Manager 3.0 Support Pack 1 媒体中为 XML 格式。其文件名分别为 SampleInputTransform.xml、 SampleSubCommandTransform.xml 和 SampleSubEventTransform.xml。根据所在平台，可以分别在以下目录中找到这些文件：

- linux\setup\utilities\cred_prov
- nt\dirxml\utilities\cred_prov
- nw\dirxml\utilities\cred_prov

如果在安装实用程序的过程中选择了身份凭证供应样本策略，则这些文件将安装在 Identity Manager 服务器上。根据所在平台的不同，这些样本策略将分别安装在以下位置：

- Windows：C:\Novell\NDS\DirXMLUtilities （默认平台；用户可在安装期间进行更改）
- NetWare® ： SYS:\System\DirXmlUtilities
- Linux (eDir 8.7)：/usr/lib/dirxml/rules/credprov

这些样本策略提供了开发适用于您的环境的策略的起始点。

操作数据超速缓存

进行所需的数据超速缓存运算时可用的机制是 <operation-data> 要素。您可能需要使用 <add> 或 <modify-password> 命令供应 SecureLogin 帐户，因此实施非静态数据超速缓存策略的逻辑位置位于订购者命令转换策略中。以下示例显示典型的 SecureLogin 供应 <operation-data> 要素：

<operation-data> <nsl-sync-data> <nsl-target-user-dn> cn=GLCANYON,ou=finance,dc=prod,dc=testco,dc=com </nsl-target-user-dn> <nsl-app-username>GCANYON</nsl-app-username> <password><!-- content suppressed --></password> <nsl-passphrase-answer>50024222</nsl-passphrase-answer> </nsl-sync-data> </operation-data>

在图 4-1 在第 306 页的财务部门方案样本中，需要以下值来填充操作数据有效负载：

- 使用 Identity Vault 中的 DirXML-ADContext 特性值填充 <nsl-target-user-dn> 要素，该特性值由 Active Directory 驱动程序设置。若要确保 AD 驱动程序在设置此值时通知 SAP 用户驱动程序，请确保将 DirXML-ADContext 作为通知特性添加至订购者过滤器。
- 使用 sapUsername 特性值填充 <nsl-app-username> 要素。使用 <add> 命令时，该特性由 SAP 用户驱动程序的创建策略生成，并可作为操作特性使用。在 SAP 用户驱动程序中， SAP 用户名值为关联值的一部分。这意味着对于口令修改事件而言，需从关联分析名称。
- 使用 <add> 或 <modify-password> 命令中的 <password> 要素值填充口令要素。
- 使用 Identity Vault 中的 workforceID 特性值填充 <nsl-passphrase-answer> 要素，该特性值由 SAP HR 驱动程序设置。虽然在向 Identity Vault 初始供应的阶段就应该设置该值，但将 workforceID 作为通知特性添加至订购者过滤器也仍然可行。

SecureLogin 供应

在本方案中，可以在其中进行操作数据检索，并将检索结果用于 SecureLogin 身份凭证供应的第一个可用位置位于驱动程序的输入转换策略中。本样本方案实施以下三种策略：

- 在成功同步口令后，实施设置 SecureLogin 身份凭证策略。
- 设置 SecureLogin 通行口令和答案
- 如果删除了应用程序用户 （未删除 Identity Vault 对象），则去除 SecureLogin 身份凭证

> 注释：在 SampleInputTransform.xml 文件中存在样本策略，该策略在成功同步口令后设置 SecureLogin 身份凭证。该文件位于 Identity Manager 3.0 Support Pack 1 媒体的身份凭证供应文件夹中。

设置 SecureLogin 身份凭证策略需要确保仅当返回的命令状态为成功，且以前设置的 <operation-data> 要素仍存在时，才发生供应。

取消 **SecureLogin** 供应

很多方案都可以采用以下策略：删除已连接应用程序的用户帐户，但保留 Identity Vault 帐户。在财务方案中，将用户的 Identity Vault employeeStatus 特性值设置为 "I" 时，需要删除 SAP 用户帐户并取消 SecureLogin 身份凭证供应。为了处理这种情况， SAP 用户驱动程序的订购者事件转换包含了将修改特性值转换为对象删除的策略。由于完成删除命令后仍然需要 Active Directory 帐户名称，因此需要在 <delete> 命令中设置 <operation-data> 事件，以使该事件可用于输入转换策略中的取消 SecureLogin 供应策略。

<operation-data> <nsl-sync-data> <nsl-target-user-dn> cn=GLCANYON,ou=finance,dc=prod,dc=testco,dc=com </nsl-targer-user-dn> </nsl-sync-data> </operation-data>

在文件 SampleSubEventTransform.xml 的身份凭证供应策略样本中，可以找到将 <modify> 事件转换为 <delete> 并创建该要素的策略。

4.3 Novell SecretStore 支持的身份凭证供应策略

通过身份凭证供应策略，您可以向 Novell SecretStore 储存库中的用户对象供应应用程序身份凭证。它将应用程序服务器和用户身份凭证供应作为 Identity Manager 标准供应方案的一部分，这种能力为用户提供了更安全和同步的万维网一次签到体验。

本文档包含在 Identity Manager 中配置对象和策略时所需的步骤。但不包含任何 SecretStore 组件的部署和配置信息。有关 SecretStore 文档，请参见 Novell SecretStore 3.3.3 文档 (http://www.novell.com/documentation/secretstore33/index.html)。

在实施 SecretStore 支持的身份凭证供应策略时，需要一个储存库对象、一个应用程序对象并需要另外创建一些策略。为方便 Identity Manager 使用，储存库对象和应用程序对象储存了 SecretStore 信息。为使所有驱动程序都可以使用身份凭证供应，还使用了另外创建的策略。也可以配置以下选项：

- 身份凭证供应可由发布者通道、订购者通道提供，或由这两个通道同时提供。
- SecretStore 同步可以在应用程序口令同步过程中发生，也可以由其它事件触发。
- 在不供应应用程序帐户的情况下，可以供应万维网服务身份凭证。

图 4-5 显示的方案典型而简单，它向 GroupWise® 的新用户供应了一次签到身份凭证。该部门通过 SAP HR 系统和 Identity Manager 向 Identity Vault 供应新用户。根据组织信息，这些用户随后将供应给 eDirectory 上实现的部门鉴定树。新用户通过该树鉴定到网络，同时该树也是 GroupWise 安全身份凭证的储存库， Novell iChain® 或 Access Manager® 使用该储存库

从公司防火墙外部提供安全的一次签到功能。由于随后 Identity Manager 会将用户供应给 GroupWise，因此这些系统的身份凭证将与鉴定树中这些身份凭证的 SecretStore 特性同步。

图 ***4-5*** *SecretStore* 支持的身份凭证供应

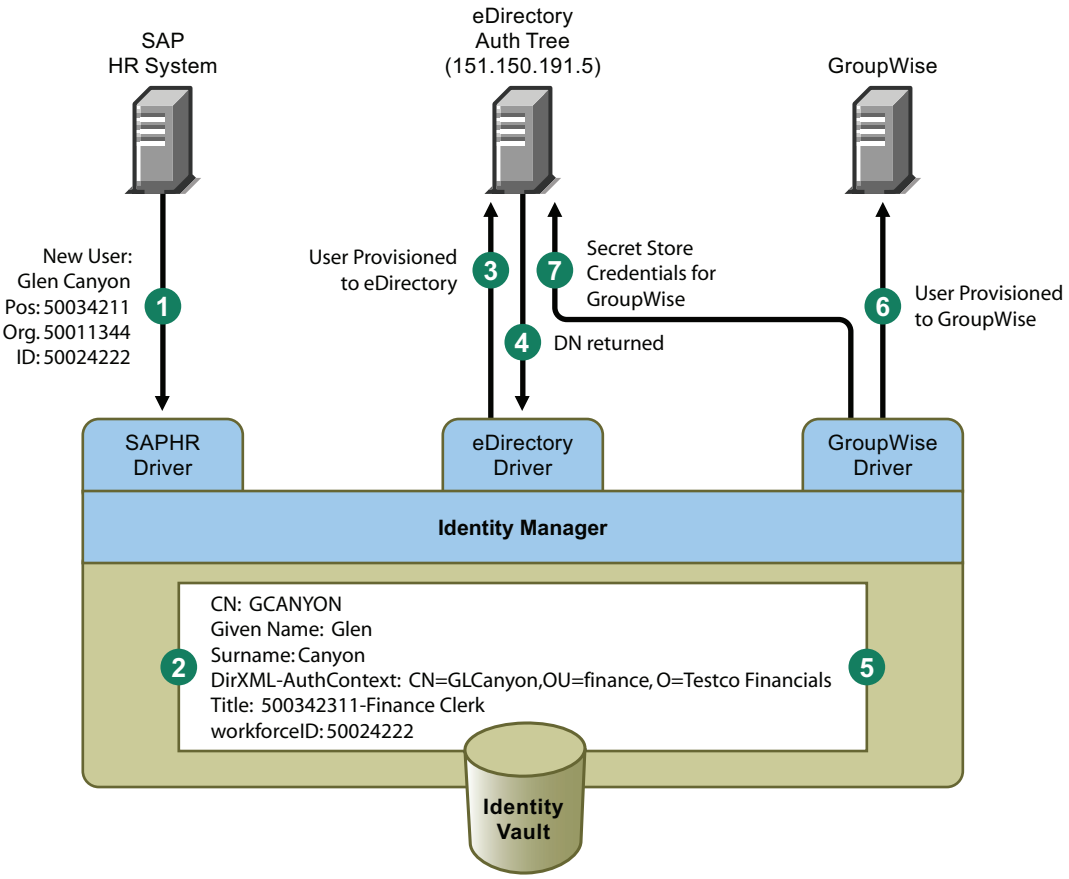


图 4-5 阐述了以下供应步骤：

1. SAP HR 系统发布了有关新雇用员工 Glen Canyon 的数据。 Identity Manager SAP HR 驱动程序将处理此数据。
2. 在 Identity Vault 中创建了一个新的用户对象，其 CN 值为 GCANYON， workforceID 值为 50024222。由于此用户被分配到其所在公司的财务组织，因此需要将其鉴定到财务部门的 eDirectory 服务器。现在，同步该域的 Identity Manager eDirectory 驱动程序将使用 Identity Vault 信息。
3. 将 Glen 供应给财务部门的 eDirectory 服务器。
4. 配置该驱动程序以获得 Glen 完整的 LDAP 判别名：CN=GLCanyon、 OU=finance、O=Testco Financials。
5. 在 Identity Vault 中，将 LDAP 名称置于 GCANYON 用户的 DirXML-AuthContext（用户对象的扩展名， DirXML-ADContext 的副本）特性中。

   由于所需特性现在在 Identity Vault 中可用，因此 GroupWise 驱动程序将开始处理 GCANYON 对象的特性。
6. 由于 Glen 隶属于财务组织，因此驱动程序在 GroupWise 域服务器的财务部门中为 GCANYON 供应了一个 GroupWise 帐户。

7. 成功创建帐户后， GroupWise 驱动程序策略为 Glen 的 eDirectory 用户帐户供应了他的 GroupWise 鉴定身份凭证。

Glen 从因特网鉴定到他所在公司的万维网站点后， iChain 服务器可以在不需要输入他的 GroupWise 身份凭证的前提下，使用 SecretStore 身份凭证将他的鉴定填写到他的安全 GroupWise 电子邮件帐户上，因而为公司的资源提供了额外的安全保障。

4.4 实施 SecretStore 支持的身份凭证供应策略

SecretStore 支持的身份凭证供应策略在实施时非常用户化。根据 SecretStore 安装的平台、供应的应用程序以及使用的 Identity Manager 驱动程序，实施该策略时的步骤将有所不同。

实施 SecretStore 支持的身份凭证供应策略：

- “满足 Novell SecretStore 支持的身份凭证供应策略的要求”在第 327 页
- “确定 Novell SecretStore 的部署配置参数”在第 327 页
- “创建 Novell SecretStore 的储存库对象”在第 330 页
- “创建 Novell SecretStore 的应用程序对象”在第 336 页
- “配置 Novell SecretStore 的身份凭证供应策略”在第 343 页

4.4.1 满足 Novell SecretStore 支持的身份凭证供应策略的要求

若要实施 SecretStore 支持的身份凭证供应策略，必须具备以下条件：

- 带有 Support Pack 1 的 Identity Manager 3.0
  - 必须安装在 eDirectory 8.7x 上，不支持 eDirectory 8.8
  - 要验证 jsso.jar、 idmcp.jar 和 jnet.jar 位于 Identity Manager Java 库的标准位置
- SecretStore 3.3 或更高版本

验证个人环境满足要求后，请按 “确定 Novell SecretStore 的部署配置参数”在第 327 页 继续。

4.4.2 确定 Novell SecretStore 的部署配置参数

若要提供图 4-5 中阐述的部署方案中描述的同步功能，首先需收集所有与 Identity Manager 和 SecretStore 环境相关的业务处理信息。可以打印表 4-3 “SecretStore 的身份凭证供应策略工作表”在第 327 页，并将其用作记录信息的工作表。

表 ***4-3*** *SecretStore* 的身份凭证供应策略工作表

所需的配置信息	信息
1) 将为万维网一次签到供应配置哪些应用程序？	
2) SecretStore 储存库服务器的 DNS 名称或 IP 地址。	
3) SecretStore 储存库服务器的 SSL LDAP 端口。	

所需的配置信息	信息
4) SecretStore 储存库服务器管理员的完全限定的 LDAP 判别名。	
5) SecretStore 储存库服务器的管理员口令。	
6) 从 SecretStore 服务器导出 SSL 证书的完整路径及该证书的名称。对于 Identity Manager 服务器而言，该证书必须是本地的。	
7) 确定多个驱动程序共同使用一个 SecretStore 储存库，还是由每个驱动程序单独使用一个储存库。	
8) 记录正在使用的 SecretStore 机密的类型。	有两种受支持的机密类型： ◆ A：应用程序机密 （SS_App：前缀） ◆ C：身份凭证集机密 （SS_CredSet：前缀）
9) 所供应的每个应用程序的应用程序 ID 或身份凭证集名称。	
10) 查找每个应用程序的所有所需鉴定密钥，例如用户名和口令等。每个应用程序的鉴定密钥可能会有所不同。	
11) 确定是否可以使用静态值设置所有的鉴定密钥值。	
12) 对于每个用户都不同或有可能不同的非静态值，请记录非静态信息 （事件信息或 Identity Vault 特性值）的源。	
13) 如果正在驱动程序上实施 SecretStore 供应，而该驱动程序的口令还与目标应用程序的口令同步，则请确定是在将口令设置到目标应用程序服务器之前还是之后供应 SecretStore。	
14) 储存库和应用程序对象所储存的驱动程序对象的名称。（可以为不同的驱动程序。）	
15) 确定目标应用程序的用户对象 DN。	

供应配置数据的示例

本供应方案将确定位于财务 eDirectory 鉴定树上的用户的以下示例数据，从而为财务部门的 GroupWise 域服务器供应用户的 SecretStore 身份凭证：

SecretStore 储存库信息

表 ***4-4*** *SecretStore* 的身份凭证供应策略工作表示例

所需的配置信息	信息
1) 将为万维网一次签到供应配置哪些应用程序？	GroupWise
2) SecretStore 储存库服务器的 DNS 名称或 IP 地址。	151.150.191.5

所需的配置信息	信息
3) SecretStore 储存库服务器的 SSL LDAP 端口。	636
4) SecretStore 储存库服务器管理员的完全限定的 LDAP 判别名。	cn=admin、 ou=finance、 o=Tesetco Financials
5) SecretStore 储存库服务器的管理员口令。	dixml
6) 从 SecretStore 服务器导出 SSL 证书的完整路径及该证书的名称。对于 Identity Manager 服务器而言，该证书必须是本地的。	c:\novell\nds\FinanceAD.cer
7) 确定多个驱动程序共同使用一个 SecretStore 储存库，还是由每个驱动程序单独使用一个储存库。	在本示例中，只有一个储存库。
8) 记录正在使用的 SecretStore 机密的类型。	有两种受支持的机密类型： ◆ A：应用程序机密 （SS_App：前缀） ◆ **C：身份凭证集机密 （SS_CredSet：前缀）**
9) 所供应的每个应用程序的应用程序 ID 或身份凭证集名称。	GroupWise_Credentials
10) 查找每个应用程序的所有所需鉴定密钥，例如用户名和口令等。每个应用程序的鉴定密钥可能会有所不同。	用户名口令
11) 确定是否可以使用静态值设置所有的鉴定密钥值。	本方案中没有静态信息。
12) 对于每个用户都不同或有可能不同的非静态值，请记录非静态信息 （事件信息或 Identity Vault 特性值）的源。	用户名：Identity Vault 特性 "CN" 口令：事件 <password>
13) 如果正在驱动程序上实施 SecretStore 供应，而该驱动程序的口令还与目标应用程序的口令同步，则请确定是在将口令设置到目标应用程序服务器之前还是之后供应 SecretStore。	之后
14) 储存库和应用程序对象所储存的驱动程序对象的名称。（可以为不同的驱动程序。）	GroupWise-Finance 驱动程序
15) 确定目标应用程序的用户对象 DN。	Identity Vault 特性 "DirXML-ADContext"

其它环境信息：

- 财务部门 eDirectory 树将作为所有财务应用程序的 SecretStore 储存库。
- 所有财务部门供应驱动程序都位于名为 " 财务驱动程序 " 的驱动程序集中。
- Identity Vault 特性的 employeeStatus 设置为 " 值 "I" 时，必须删除 GroupWise 帐户，GroupWise 用户帐户的 SecretStore 身份凭证也要从 eDirectory 用户中去除。

从所收集的数据中可以看出， SecretStore 储存库信息为供应财务部门应用程序的所有驱动程序提供全局信息。此外，可以静态配置所有供应信息，但 GroupWise 登录参数用户名、口令和目标用户 DN 除外。

确定所有参数后，请按 “创建 Novell SecretStore 的储存库对象”在第 330 页 继续。

4.4.3 创建 Novell SecretStore 的储存库对象

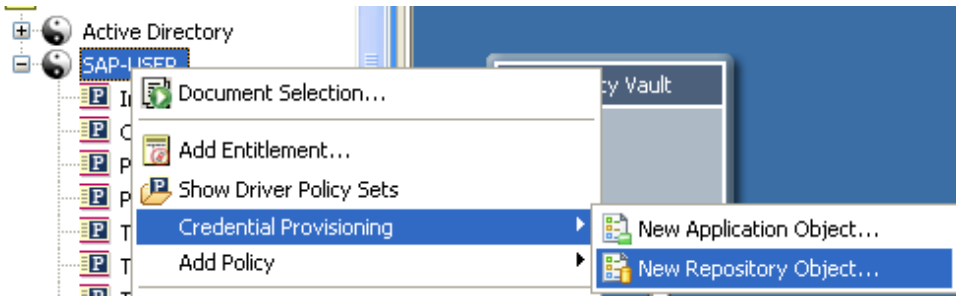
储存库对象储存 SecretStore 的静态配置信息。储存库信息独立于使用应用程序身份凭证的应用程序。无论连接什么系统（例如， SAP、 PeopleSoft、 Notes 等），所有供应事件都可应用此信息。储存库对象可以在 Designer 或 iManager 中创建。

- “在 Designer 中创建 Novell SecretStore 储存库对象”在第 330 页
- “在 iManager 中创建 Novell SecretStore 的储存库对象”在第 333 页

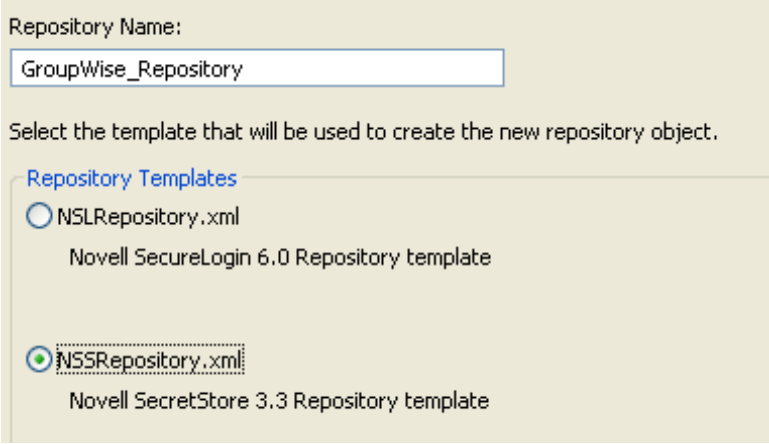
在 Designer 中创建 Novell SecretStore 储存库对象

在 Designer 中创建储存库对象的方法有很多种，下面的方法是其中之一：

1. 在 " 大纲 " 视图中，右击要储存储存库对象的驱动程序对象。
2. 单击 " 身份凭证供应 ">" 新建储存库对象 "。

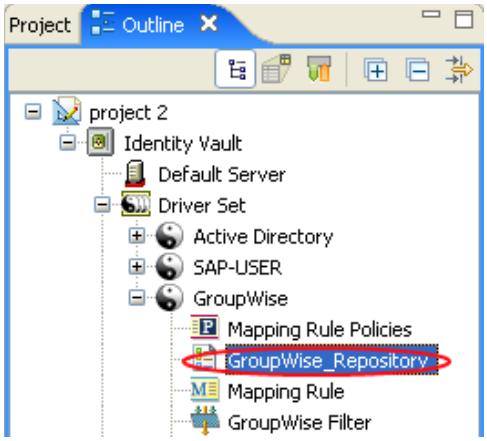


3. 指定储存库对象的名称。
4. 选择 *NSSRepository.xml* 以使用 SecretStore 模板。

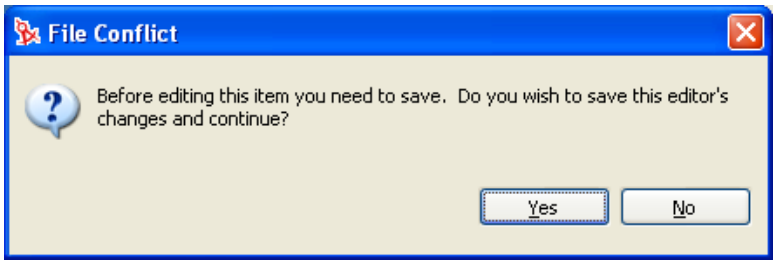


5. 单击 " 确定 "。

6 在 " 大纲 " 视图中，双击储存库对象以添加配置信息。



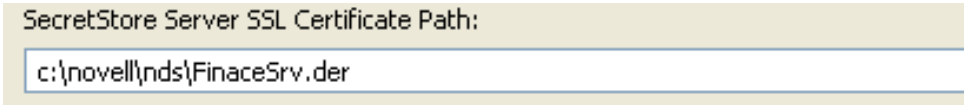
7 单击 " 是 " 保存新储存库对象。



8 指定 SecretStore 服务器的 DNS 名称或 IP 地址。请参见工作表项 2)。

9 指定 SecretStore 服务器的 SSL 端口。请参见工作表项 3)。

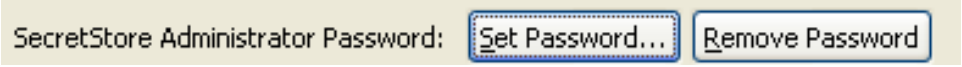
10 指定从 SecretStore 服务器导出 SSL 证书的完整路径。此路径必须包括证书名称，而且对于 Identity Manager 服务器而言必须是本地的。请参见工作表项 6)。



注释 : 有关如何导出 SSL 证书的信息，请参考 iManager 文档。

11 指定 SecretStore 管理员的完全限定的 LDAP 判别名。请参见工作表项 4)。

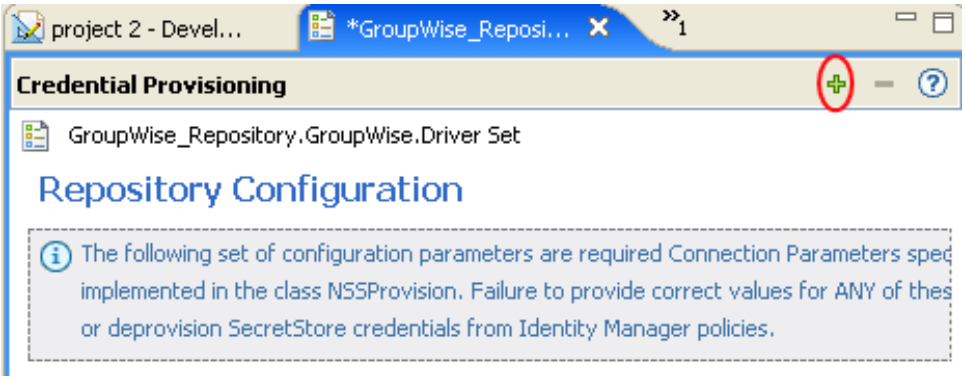
12 单击 " 设置口令 "。



13 指定并再次输入 SecretStore 管理员的口令，然后单击 " 确定 "。请参见工作表项 5)。

14 查看信息，然后单击 " 保存 " 图标 保存信息。

15 （可选）如果要创建储存库对象的其它配置参数，请单击 " 添加新项 " 图标 。

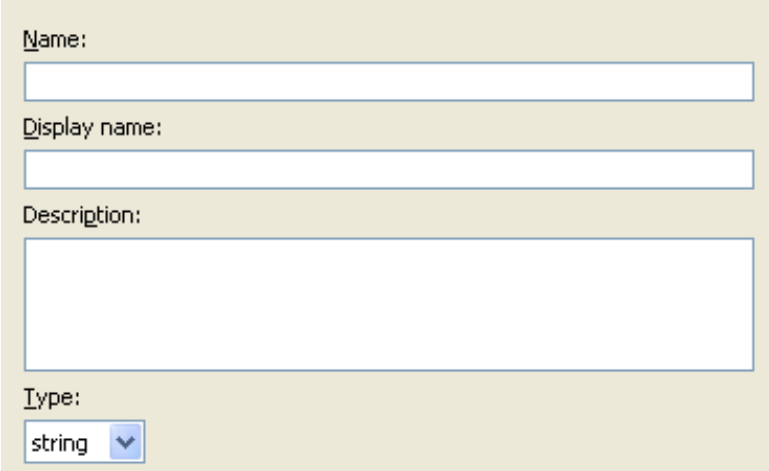


15a 指定参数的名称。

15b 指定参数的显示名称。

15c 指定参数说明以供参考。

参数以字符串的形式储存。



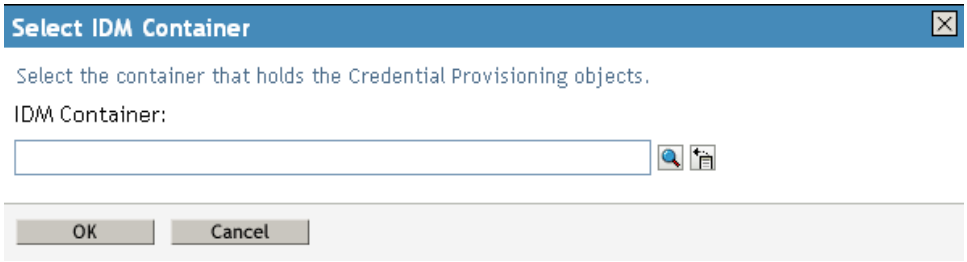
15d 单击 " 确定 "。

15e 单击 " 保存 " 图标 保存储存库对象。

创建储存库对象后，请转至 “在 Designer 中，创建 Novell SecureLogin 的应用程序对象”在第 316 页 。

在 **iManager** 中创建 **Novell SecretStore** 的储存库对象

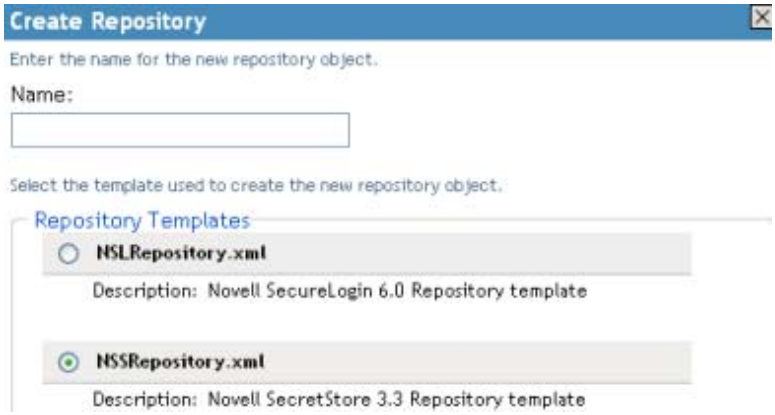
1. 在 iManager 中，选择 " 身份凭证供应 ">" 配置 "。
2. 浏览至将在其中储存储存库对象的驱动程序对象，并选择该对象，然后单击 " 确定 "。



3. 单击 " 新建 " 创建储存库。

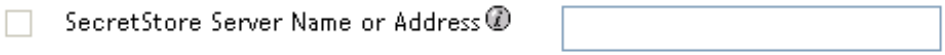
4. 指定储存库对象的名称。

5 选择 *NSSRepository.xml*，以使用 SecretStore 模板创建储存库。

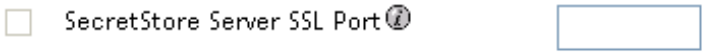


6 单击 " 确定 "。

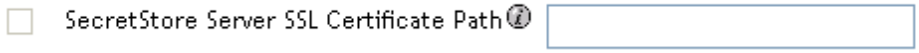
7 指定 SecretStore 服务器的 DNS 名称或 IP 地址。请参见工作表项 2)。



8 指定 SecretStore 服务器的 SSL 端口。请参见工作表项 3)。



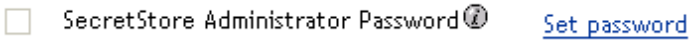
9 指定从 SecretStore 服务器导出 SSL 证书的完整路径。此路径必须包括证书名称，而且对于 Identity Manager 服务器而言必须是本地的。请参见工作表项 6)。



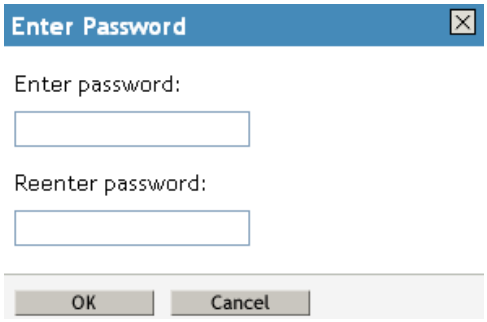
注释 : 有关如何导出 SSL 证书的信息，请参考 iManager 文档。

10 指定 SecretStore 管理员的完全限定的 LDAP 判别名。请参见工作表项 4)。

11 单击 " 设置口令 "。

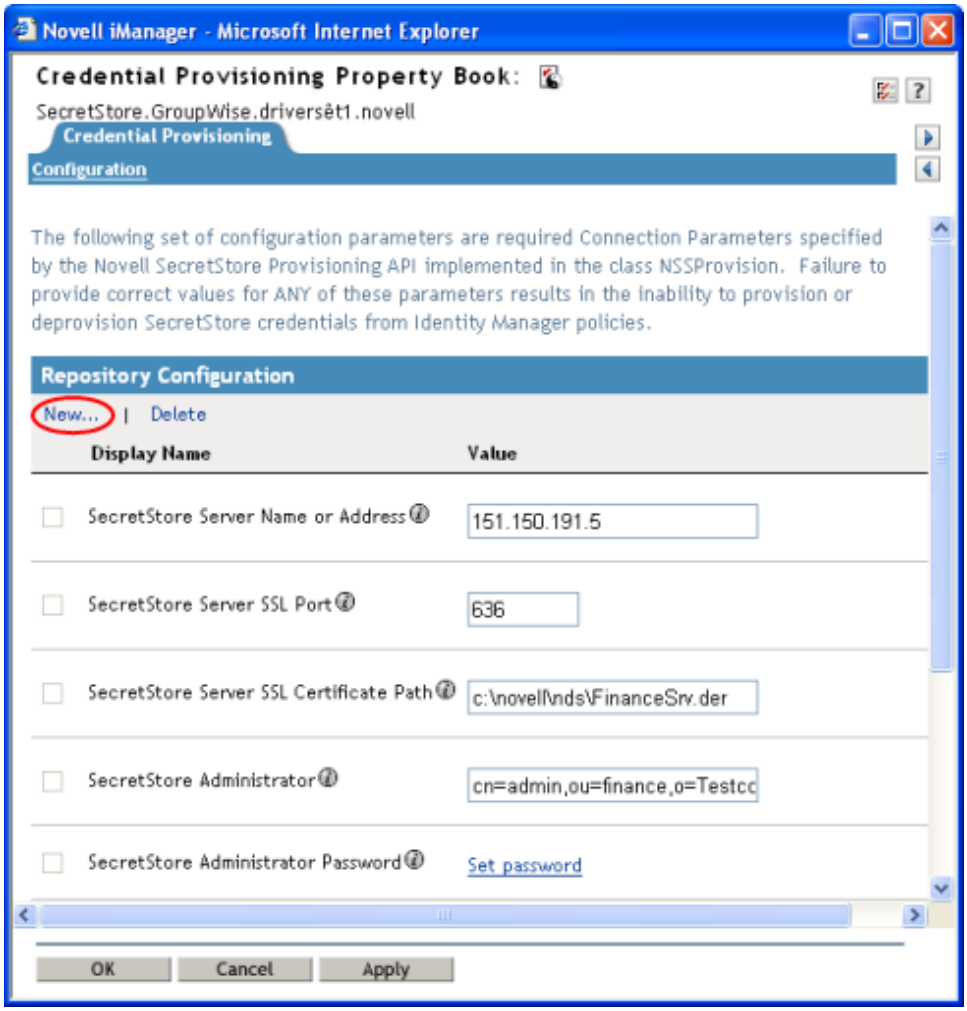


12 指定并再次输入 SecretStore 管理员的口令，然后单击 " 确定 "。请参见工作表项 5)。



13 查看指定的值，然后单击 " 确定 "。

14 （可选）如果要创建储存库对象的其它配置参数，请单击 " 新建 "。



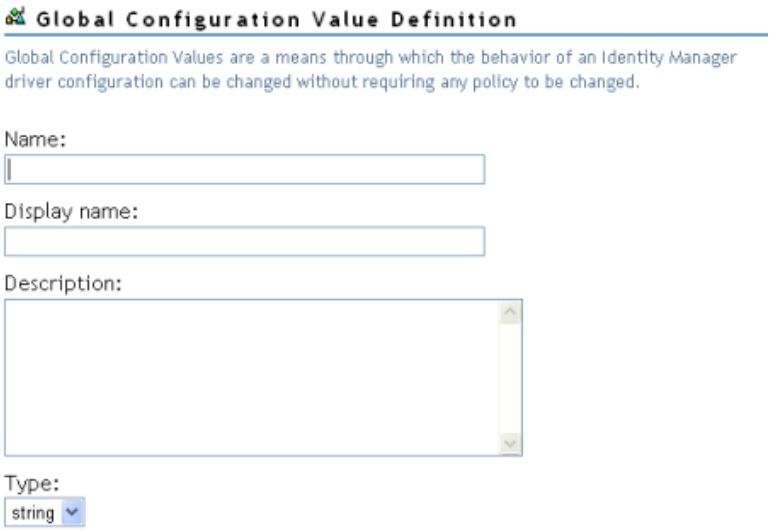
此示例信息来自图 4-1 在第 306 页中的方案。

14a 指定参数的名称。

14b 指定参数的显示名称。

14c 指定用作参照的参数说明。

参数以字符串的形式储存。



14d 单击 " 确定 "。

创建储存库对象后，请转至 “在 iManager 中，创建 Novell SecureLogin 的应用程序对象”在第 319 页 。

4.4.4 创建 Novell SecretStore 的应用程序对象

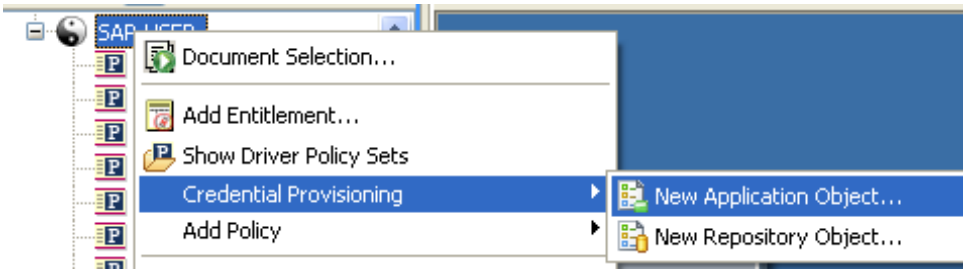
应用程序储存 SecretStore 的静态配置参数值。应用程序信息特定于使用应用程序身份凭证的应用程序 （例如， GroupWise 客户程序信息或 SAP 数据库客户程序信息）。可以在 Designer 或 iManager 中创建应用程序对象。

- “在 Designer 中创建 Novell SecretStore 的应用程序对象”在第 336 页
- “在 iManager 中创建 Novell SecretStore 的应用程序对象”在第 339 页

在 Designer 中创建 Novell SecretStore 的应用程序对象

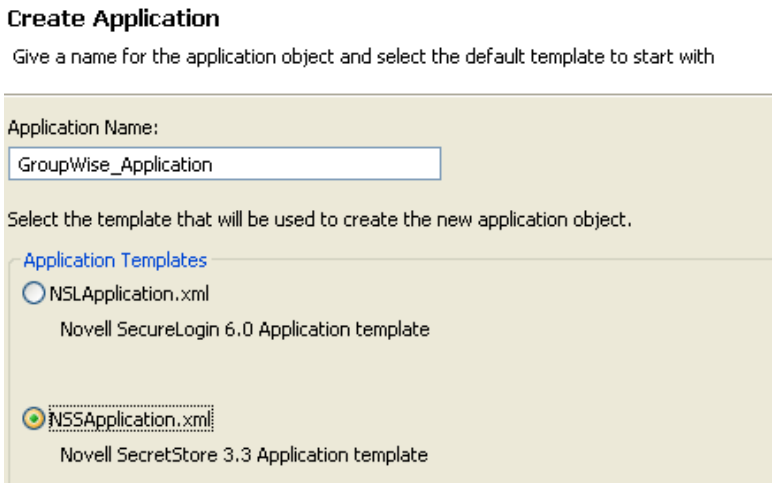
在 Designer 中创建应用程序对象的方法有很多种，下面的方法是其中之一：

1. 在 " 大纲 " 视图中，右击要储存应用程序对象的驱动程序对象。
2. 单击 " 身份凭证供应 ">" 新建应用程序对象 "。



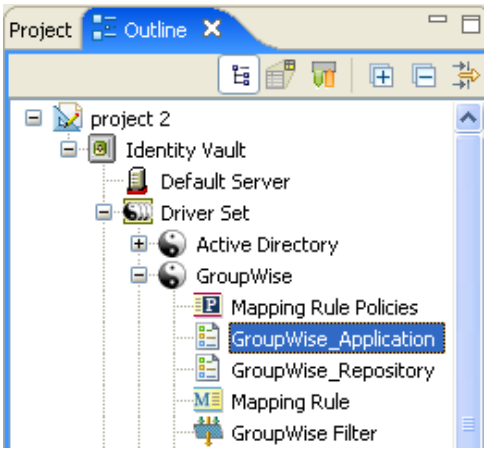
3 指定应用程序对象的名称。

4 选择 *NSSApplication.xml* 以使用 SecretStore 模板。

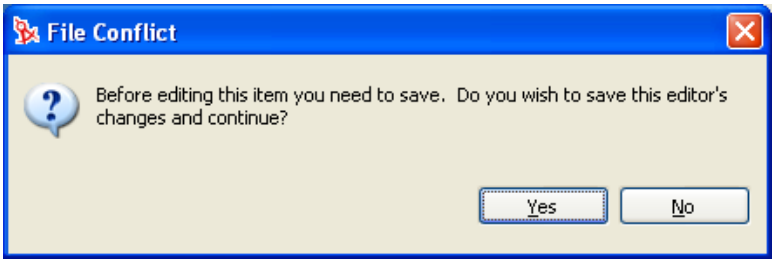


5 单击 " 确定 "。

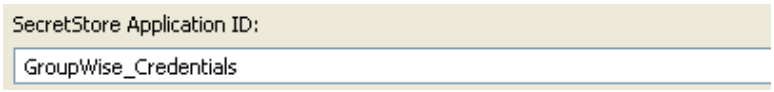
6 在 " 大纲 " 视图中，双击应用程序对象以添加配置信息。



7 单击 " 是 " 保存新的应用程序对象。

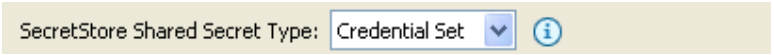


8 指定 SecretStore 应用程序 ID。请参见工作表项 9)。

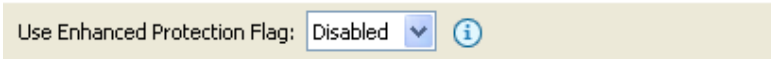


9 选择 "SecretStore 机密类型 "。请参见工作表项 8)。

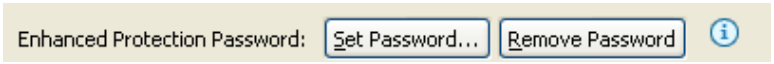
10 选择 "SecretStore 共享机密类型 "。请参见工作表项 8)。



11 选择 SecretStore" 使用增强型保护标志 " 为 " 禁用 " 还是 " 启用 "。



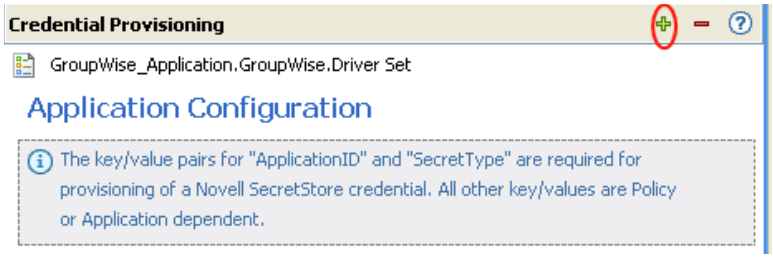
12 如果已启用 " 增强型保护口令 "，则单击 " 设置口令 " 以设置该口令。



13 指定并再次输入口令，然后单击 " 确定 "。

14 单击 " 保存 " 图标 保存应用程序。

15 单击 " 添加新项 " 图标 ，添加应用程序所需的鉴定密钥。

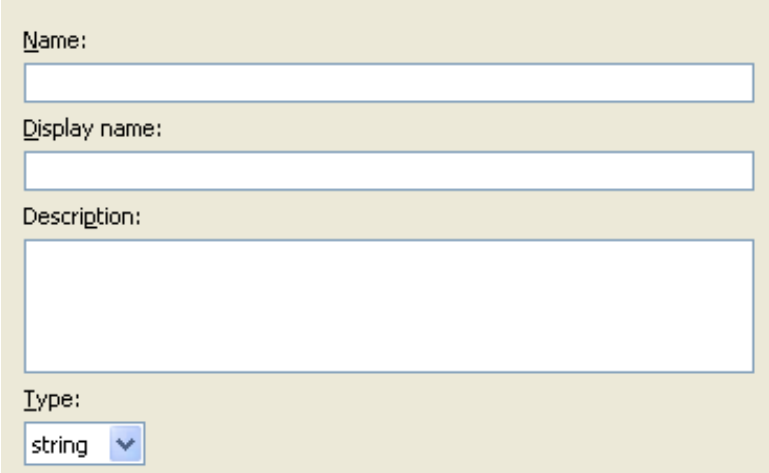


15a 指定鉴定密钥的名称。

15b 指定鉴定密钥的显示名称。

15c 指定鉴定密钥的说明以供参考。

鉴定密钥以字符串的形式储存。



15d 单击 " 确定 "。

15e 对每个需要输入的新鉴定密钥重复步骤 15。

16 如果鉴定密钥值是所有用户身份凭证共享的静态值，则请指定该值。

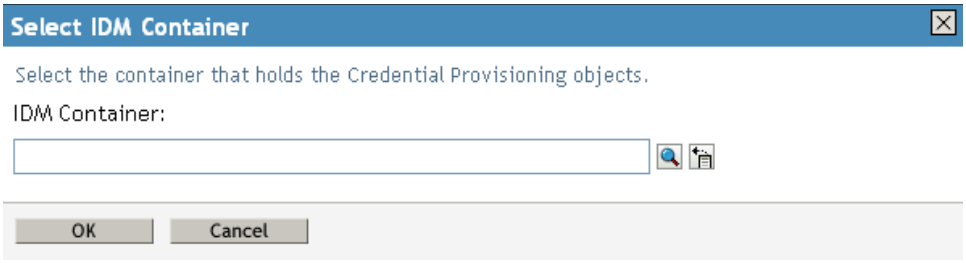
17 单击 " 保存 " 图标 保存应用程序。

创建应用程序对象后，请转至“配置 Novell SecretStore 的身份凭证供应策略”在第 343 页。

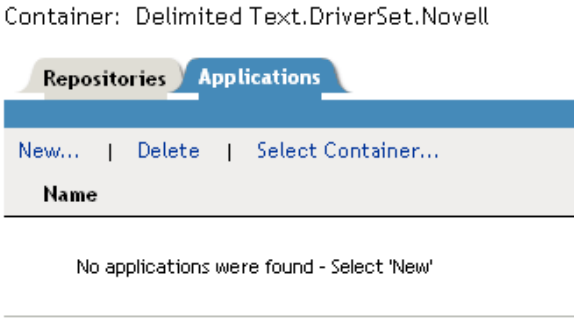
在 iManager 中创建 Novell SecretStore 的应用程序对象

1 在 iManager 中，选择 " 身份凭证供应 ">" 配置 "。

2 浏览至要储存应用程序对象的驱动程序对象并选择该对象，然后单击 " 确定 "。

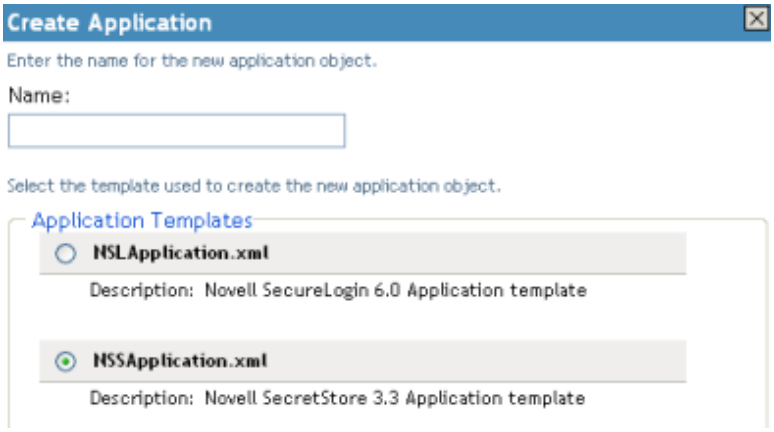


3 选择 " 应用程序 " 选项卡，然后单击 " 新建 "。



4 指定应用程序对象的名称

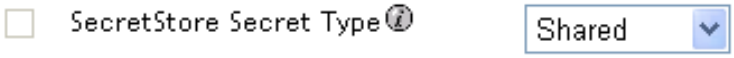
5 选择 *NSSApplication.xml*，以使用 SecretStore 模板创建应用程序。



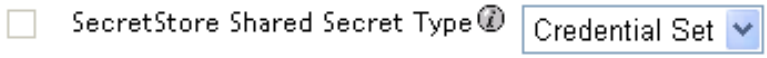
6 单击 " 确定 "。

7 指定 "SecretStore 应用程序 ID"。请参见工作表项 9)。

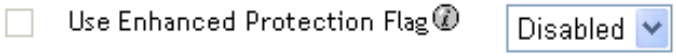
8 选择 "SecretStore 机密类型 "。请参见工作表项 7)。 SecretStore 类型为 " 共享 " 或 " 非共享 "。



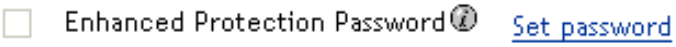
9 选择 "SecretStore 共享机密类型 "。请参见工作表项 8)。共享 SecretStore 类型为 " 身份凭证集 " 或 " 应用程序 "。



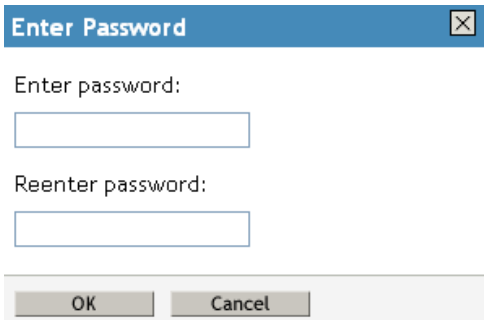
10 选择 SecretStore" 使用增强型保护标志 " 为 " 禁用 " 还是 " 启用 "。



11 如果已启用 " 增强型保护口令 "，则单击 " 设置口令 " 以设置该口令。



12 指定并再次输入口令，然后单击 " 确定 "。



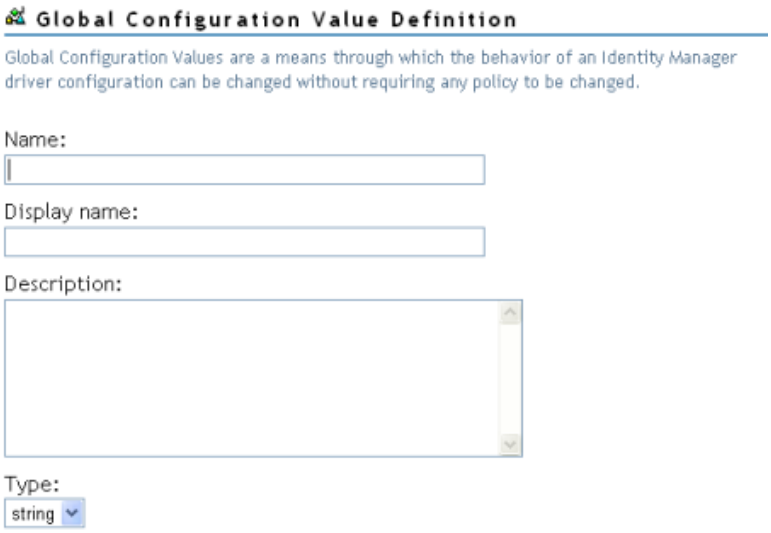
13 单击 " 新建 " 创建应用程序需要的鉴定密钥。请参见工作表项 10)。

13a 指定鉴定密钥的名称。

13b 指定鉴定密钥的显示名称。

13c 指定鉴定密钥的说明以供参考。

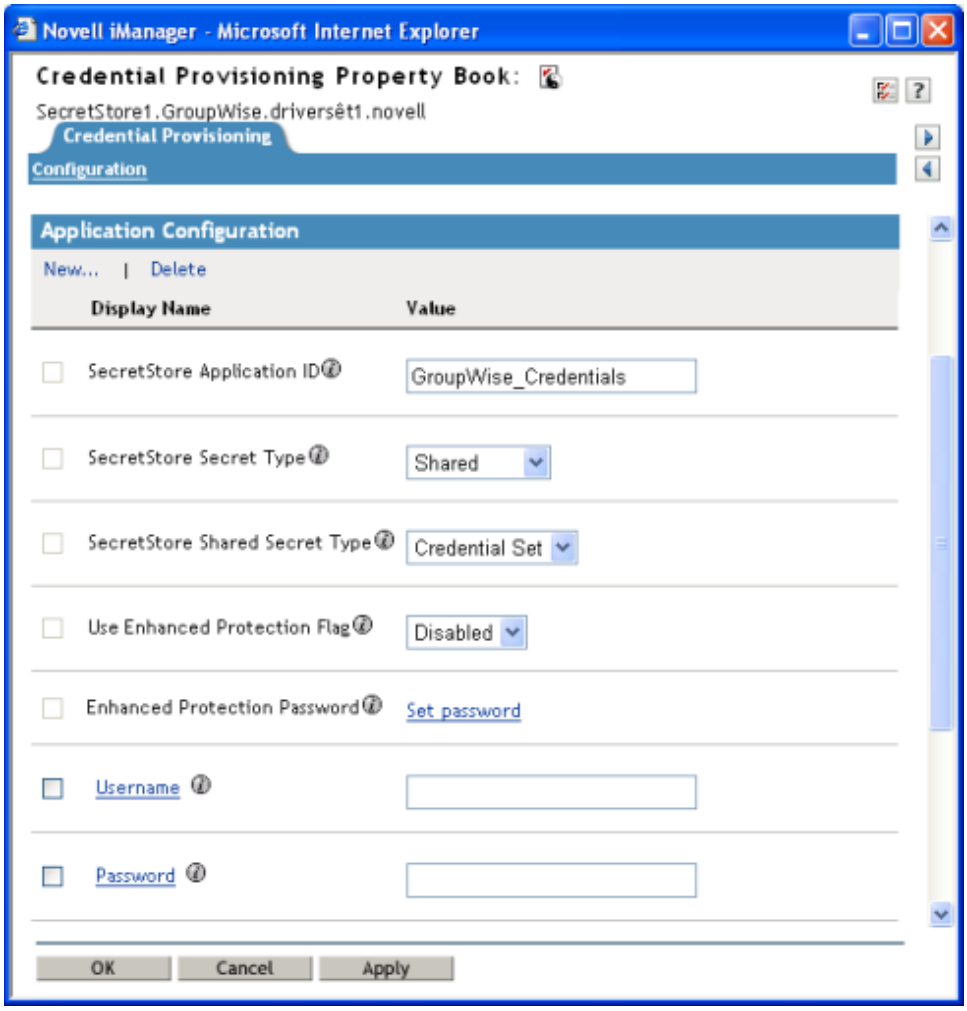
鉴定密钥以字符串的形式储存。



13d 单击 " 确定 "。

13e 对应用程序需要的每个鉴定密钥重复步骤 13。

14 如果鉴定密钥值为静态，则指定该值，然后单击 " 确定 "。



创建应用程序对象后，请转至“配置 Novell SecretStore 的身份凭证供应策略”在第 343 页。

4.4.5 配置 Novell SecretStore 的身份凭证供应策略

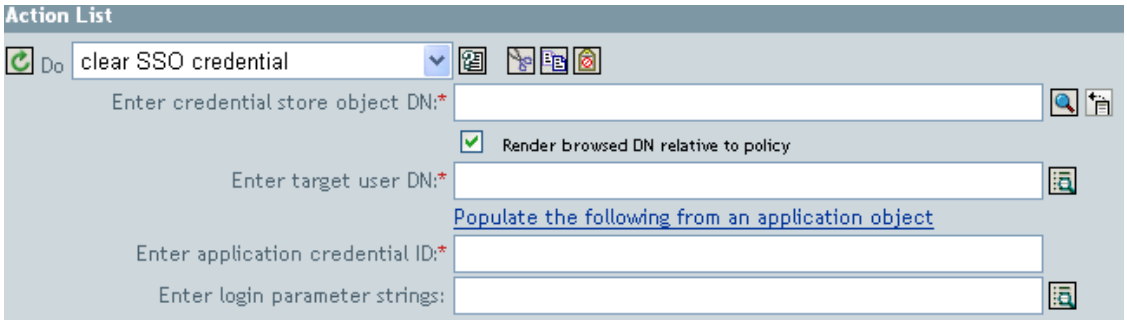
创建储存库和应用程序对象后，需要创建一些用于供应 SecretStore 信息的策略。这些策略使用储存在储存库和应用程序对象中的信息。在策略构建器中，有两种操作可用于供应 SecretStore 身份凭证：

- “清除 SSO 身份凭证”在第 322 页
- “设置 SSO 身份凭证”在第 322 页

清除 SSO 身份凭证

" 清除 SSO 身份凭证 " 操作允许清除 SSO 身份凭证，因此可以取消对对象的供应。

图 *4-6* 清除 *SSO* 身份凭证

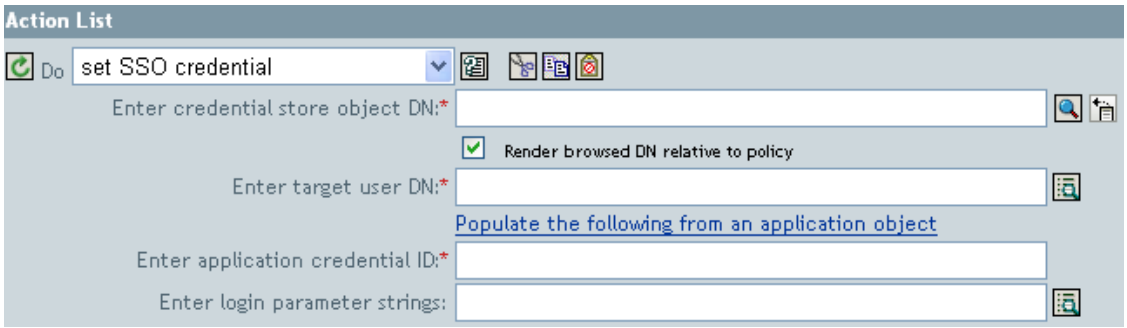


- 输入身份凭证储存对象 **DN**：浏览至储存库对象并选择该对象。
- 输入目标用户 **DN**：使用自变量构建器创建目标用户的 DN。请参见工作表项 15)。
- 输入应用程序身份凭证 **ID**：指定应用程序 ID。请参见工作表项 9)。
- 输入登录参数字符串：起动字符串构建器，然后输入应用程序的每个鉴定密钥。请参见工作表项 10)。

设置 SSO 身份凭证

创建用户对象或修改口令后， " 设置 SSO 身份凭证 " 操作允许设置 SSO 身份凭证。

图 *4-7* 设置 *SSO* 身份凭证



- 输入身份凭证储存对象 **DN**：浏览至储存库对象并选择该对象。
- 输入目标用户 **DN**：使用自变量构建器创建目标用户的 DN。请参见工作表项 15)。
- 输入应用程序身份凭证 **ID**：指定应用程序 ID。请参见工作表项 9)。
- 输入登录参数字符串：起动字符串构建器，然后输入应用程序的每个鉴定密钥。请参见工作表项 10)。

身份凭证供应策略的示例

可以根据满足环境的需要实施和自定义身份凭证供应策略。下面的示例说明如何对图 4-5 在第 326 页演示的方案实施这些策略。

在财务方案中，成功设置 GroupWise 中的口令之后将供应 SecretStore。大多数的必需参数都是静态配置的，可用于储存库对象和应用程序对象中的所有策略。但是也存在一些非静态数据参数 （CN、口令和 DirXML-ADContext），在完成 GroupWise 用户 <add> 或 <modify-password> 命令，并从 GroupWise 驱动程序 Shim 中返回 <output> 文档后，这些参数才可用。<output> 文档不再包含任何订购者操作特性，并丢失了命令的用户环境，从而阻止了对此对象的查询。因此，必须执行以下操作：

- 请确保 GroupWise 驱动程序的订购者创建策略可以强制非静态数据参数的存在。
- 在将订购者命令发送到 GroupWise 驱动程序 Shim 之前，超速缓存供应操作所需的非静态参数。
- 成功完成此命令后，请检索 SecretStore 供应要使用的已超速缓存数据。

注释：策略样本在 Identity Manager 3.0 Support Pack 1 媒体中为 XML 格式。其文件名分别为 SampleInputTransform.xml、 SampleSubCommandTransform.xml 和 SampleSubEventTransform.xml。这些文件位于以下目录中：

- linux\setup\utilities\cred_prov
- nt\dirxml\utilities\cred_prov
- nw\dirxml\utilities\cred_prov

如果在安装实用程序的过程中选择了身份凭证供应样本策略，则这些文件将安装在 Identity Manager 服务器上。根据所在平台的不同，这些样本策略将分别安装在以下位置：

- Windows：C:\Novell\NDS\DirXMLUtilities （默认平台；用户可在安装期间进行更改）
- NetWare：SYS:\System\DirXmlUtilities
- Linux (eDir 8.7)：/usr/lib/dirxml/rules/credprov

这些样本策略提供了开发适用于您的环境的策略的起始点。

操作数据超速缓存

进行所需的数据超速缓存运算时可用的机制是 <operation-data> 要素。您可能需要使用 <add> 或 <modify-password> 命令供应 SecretStore 帐户，因此实施非静态数据超速缓存策略的逻辑位置位于订购者命令转换策略中。以下示例显示典型的 SecretStore 供应要素：

<operation-data> <nss-sync-data> <nss-target-user-dn>cn=GLCANYON,ou=finance,o=Testco Financials </nss-target-user-dn> <nss-app-username>GCANYON</nsl-app-username> <password><!-- content suppressed --></password> <nss-passphrase-answer>50024222</nsl-passphrase-answer> </nss-sync-data> </operation-data>

在图 4-5 在第 326 页的财务部门方案样本中，需要以下值来填充操作数据有效负载：

- 使用 Identity Vault 中的 DirXML-ADContext 特性值填充 <nss-target-user-dn> 要素，该特性值由 eDirectory 驱动程序设置。要确保 eDirectory 驱动程序设置该值时通知 GroupWise 驱动程序，请确保将 DirXML-ADContext 作为通知特性添加到订购者过滤器中。
- 使用 Identity Vault 中的 CN 特性值填充 <nss-app-username> 要素。
- 使用 <add> 或 <modify-password> 命令中的 <password> 要素值填充口令要素。

SecretStore 供应

在方案样本中，可以检索操作数据且将其用作 SecretStore 身份凭证供应的第一个可用位置位于驱动程序输入转换策略中。在方案样本中，实施了两个策略：

- 在口令同步成功后设置 SecretStore 身份凭证
- 如果已删除应用程序用户 （未删除 Identity Vault 对象），则去除 SecretStore 身份凭证

注释 **:** SampleInputTransform.xml 文件中含有样本策略，此策略在口令同步成功后设置 SecretStore 身份凭证。该文件位于 Identity Manager 3.0 Support Pack 1 媒体的实用程序目录的 cred_prov 文件夹中。

设置 SecretStore 身份凭证策略需要确保仅当返回命令状态为成功，且存在先前设置的 <operation-data> 时才进行供应。

取消 **SecretStore** 供应

删除已连接应用程序的用户帐户，但保留 Identity Vault 帐户的策略可以用于多种方案。在财务方案中，将用户的 Identity Vault employeeStatus 特性值设置为 "I" 时，需要删除 GroupWise 帐户并取消 SecretStore 身份凭证供应。为了处理这种情况， GroupWise 驱动程序的订购者事件转换包含将修改特性值转换为对象删除的策略。因为删除命令完成后仍然需要 eDirectory 帐户名，所以需要在 <delete> 命令中设置 <operation-data> 事件，以便可将此事件用于输入转换策略中的取消 SecretStore 供应策略。

<operation-data> <nss-sync-data> <nss-target-user-dn>cn=GLCANYON,ou=finance,o=Testco Financials </nss-targer-user-dn> </nss-sync-data> </operation-data>

有关如何将 <modify> 事件转换为 <delete> 并创建此要素的策略以 XML 格式存在于名为 SampleSubEventTransform.xml 的文件中，此文件位于 Identity Manager 3.0 Support Pack 1 媒体的 utilities 目录的 cred_prov 文件夹中。

使用 XSLT 样式表定义策略

5

可以将策略作为 XSLT 样式表进行实施。 XSLT 是一种转换 XML 文档的标准语言。Metadirectory 引擎中的 XSLT 处理程序符合 1999 年 11 月 16 日提出的 W3C 推荐标准。有关相关说明，请参见以下内容：

- XSL 转换 (XSLT) （XSL Transformations (XSLT)）(http://www.w3.org/TR/1999/REC-xslt-19991116)
- XML 路径语言 (XPath) （XML Path Language (XPath)）(http://www.w3.org/TR/1999/REC-xpath-19991116)

以下各节介绍了在 Identity Manager 中使用样式表的具体实现情况。

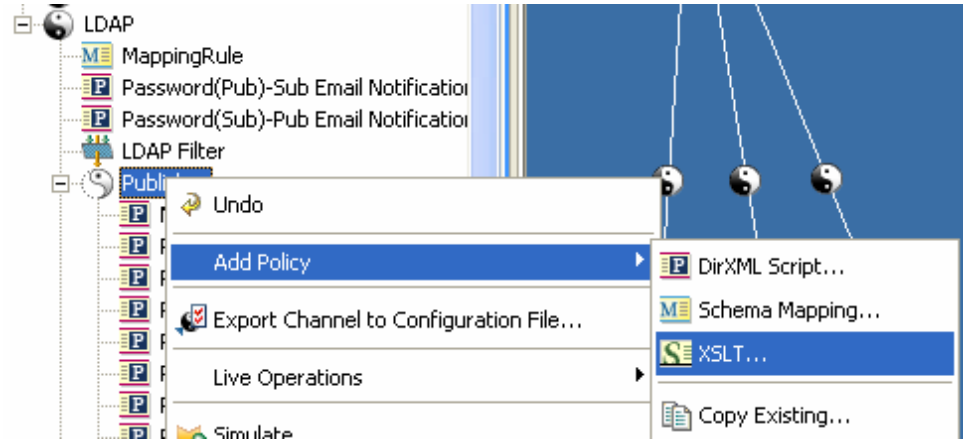
5.1 在 Designer 中管理 XSLT 样式表

使用 Designer 添加、修改并删除 XSLT 策略样式表。以下各节提供了有关在 Designer 中使用 XSLT 样式表的细节：

5.1.1 在 Designer 中添加 XSLT 策略

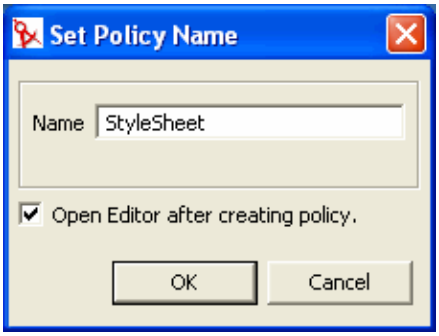
1. 在 Designer 中打开一个项目并选择 " 大纲 " 选项卡。
2. 选择样式表所使用的驱动程序和所在的位置。

3 右击并选择 " 添加策略 "> XSLT。

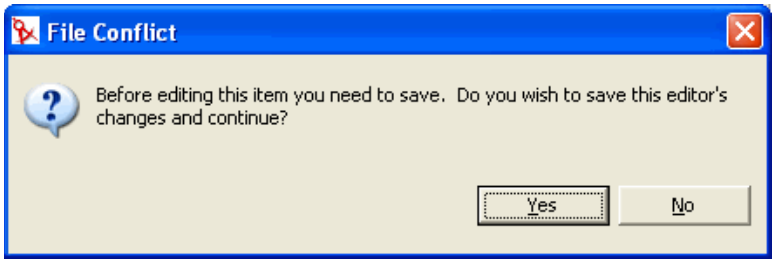


4 指定此样式表的名称。

5 选择 " 创建策略后打开编辑器 "，然后单击 " 确定 "。



6 选择 " 是 "，以在编辑新策略之前保存此项目。



7 将样式表信息添加到 add your custom templates here （在此处添加您的自定义模板）行的下面。

8 通过选择 " 文件 ">" 保存 " 保存此样式表。

5.2 在 iManager 中管理 XSLT 样式表

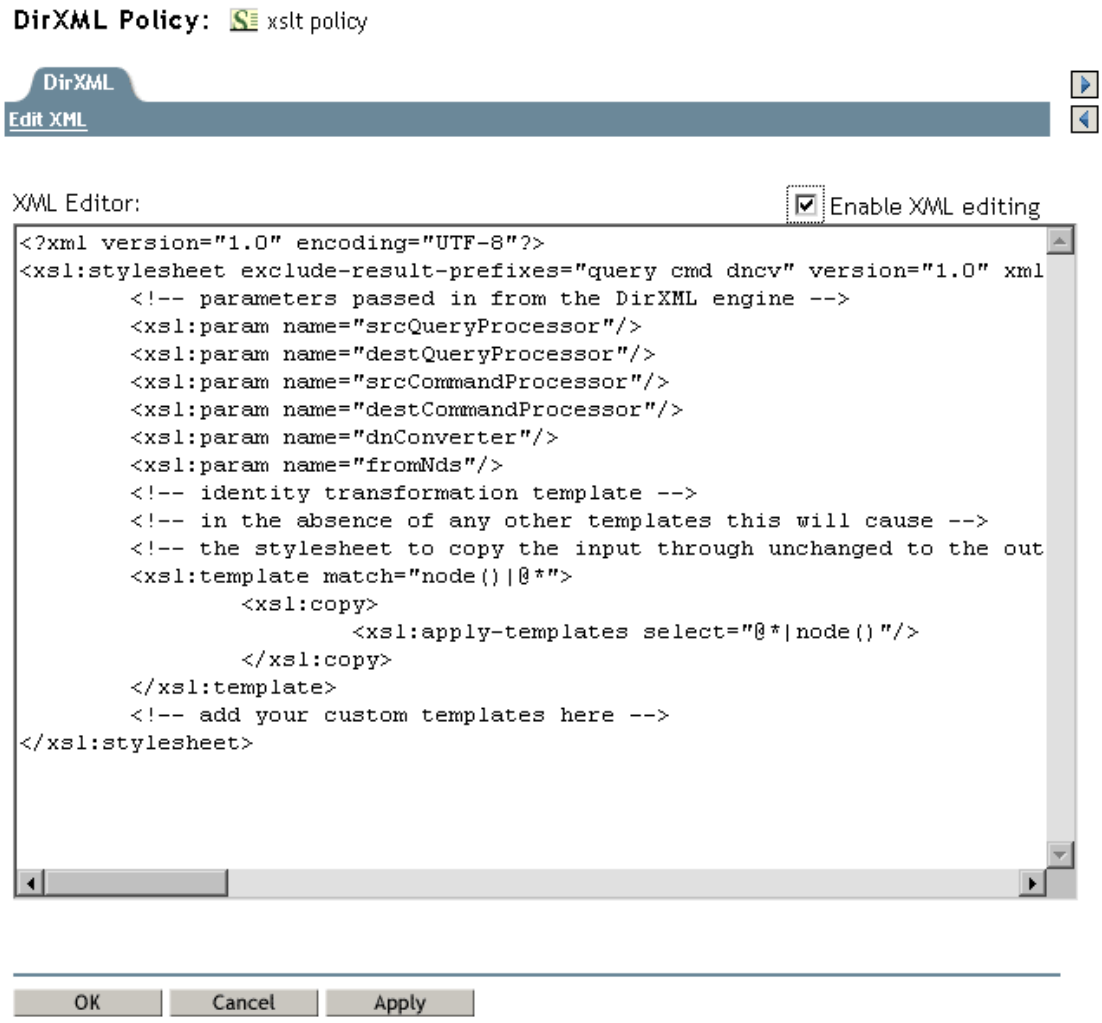
使用 iManager 添加、修改并删除 XSLT 策略样式表。以下各节提供了有关在 iManager 中使用 XSLT 样式表的细节。

- “在 iManager 中添加 XSLT 策略”在第 349 页

5.2.1 在 iManager 中添加 XSLT 策略

1 打开要管理的驱动程序的 "Identity Manager 驱动程序概述 "。
2 单击表示要定义的策略的图标。
3 单击 " 插入 "。
4 为新策略提供名称，选择 XSLT，然后单击 " 输入 "。

5 定义 XSLT 策略，然后单击 " 确定 "：



5.3 使用身份转换

在 iManager 或 Designer 中创建新样式表时，将使用实现身份转换的样式表来预先填充该样式表。如果没有附加模板，则身份转换将允许输入 XML 文档通过样式表，而不对该文档进行任何更改。通常通过添加附加模板来实现策略，以仅处理要更改的 XML。如果样式表正用于将文档转换为不同于 XDS 的 XML 词汇表，或者从此词汇表中转换出文档 （如 SOAP 和定界文本驱动程序的输入和输出转换），则您可能需要去除此身份模板。

5.4 使用 Identity Manager 传递的参数

使用 Metadirectory 引擎传递策略样式表时，这些样式表可以使用以下参数：

- srcQueryProcessor--- 实现 XdsQueryProcessor 界面的一个 Java 对象。此参数允许样式表查询源数据存储区以获取更多信息。
- destQueryProcessor--- 实现 XdsQueryProcessor 界面的一个 Java 对象。此参数允许样式表查询目标数据存储区以获取更多信息。

- srcCommandProcessor--- 实现 XdsCommandProcessor 界面的一个 Java 对象。此参数允许样式表将命令写回到事件源中。不可用于 DirXML 1.0。
- destCommandProcessor- 实现 XdsCommandProcessor 界面的一个 Java 对象。此参数允许样式表发出命令并直接将命令发送到目标数据存储区中。
- dnConverter--- 实现 XdsCommandProcessor 界面的一个 Java 对象。此参数允许样式表将 Identity Vault 对象的 DN 从一种格式转换成另一种格式。有关详细信息，请参见 Interface DNCoverter （界面 DNCoverter）(http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/api/com/novell/nds/dirxml/driver/DNConverter.html)。
- fromNds--- 一个布尔值，如果源数据存储区为 Identity Vault，则此值为 true ；如果源数据存储区为已连接应用程序，则此值为 false。

在 iManager 或 Designer 中创建新样式表时，将使用包含这些参数声明的样式表来预先填充该样式表。

在使用纲要映射策略、输入转换策略和输出转换策略时，如果使用查询和命令参数，将应用以下限制：

- 向应用程序 Shim 发出的查询必须符合应用程序 Shim 所需的形式。换言之，纲要名称必须位于应用程序名称空间内，且查询必须符合本地 Shim 使用的 XML 词汇表。未向查询中添加任何关联参照。
- 应用程序 Shim 的响应形式是由 Shim 未执行修改或纲要映射且未进行关联参照解析时返回的形式。
- 向 eDirectory™ 发出的查询必须为符合 eDirectory 的形式。换言之，纲要名称必须位于 eDirectory 名称空间内，且查询必须为 XDS。未解析关联参照。
- 应用程序 Shim 的响应形式是由 Shim 未执行修改或纲要映射时返回的形式。

查询处理程序

查询处理程序的使用取决于 Novell® XSLT 扩展功能的实现。若要进行查询，需要声明 XdsQueryProcessor 界面的名称空间。通过将以下内容添加到样式表的 <xsl:stylesheet> 或 <xsl:transform> 要素中来执行此操作。

```
xmlns:query="http://www.novell.com/nxsl/java/
com.novell.nds.dirxml.driver.XdsQueryProcessor"
```

在 iManager 或 Designer 中创建新样式表时，将使用名称空间声明来预先填充此样式表。有关查询处理程序的更多信息，请参见 Class XdsQueryProcessor （XdsQueryProcessor 类）(http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/api/com/novell/nds/dirxml/driver/XdsQueryProcessor.html)

以下示例使用一个查询处理程序 （特别长的行已换行且不以 < 开始）：

```
<!-- Query object name queries NDS for the passed object name -->

<xsl:template name="query-object-name">
   <xsl:param name="object-name"/>

<!-- build an xds query as a result tree fragment -->
   <xsl:variable name="query">
            <query>
```

```
                <search-class class-name="{ancestor-or-self:
                      :add/@class-name}"/>

<!-- NOTE: depends on CN being the naming attribute -->
                <search-attr attr-name="CN">
                    <value><xsl:value-of select="$object-name"/
                         ></value>
                </search-attr>
<!-- put an empty read attribute in so that we don't get -->
<!-- the whole object back                                -->
                <read-attr/>
            </query>
   </xsl:variable>

<!-- query NDS -->
<xsl:variable name="result" select="query:query($destQuery
     Processor,$query)"/>

<!-- return an empty or non-empty result tree fragment -->
<!-- depending on result of query                      -->
   <xsl:value-of select="$result//instance"/>
</xsl:template>
```

下面是另一个示例。

```
<?xml version="1.0"?>
<xsl:transform
            version="1.0"
       xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
       xmlns:cmd="http://www.novell.com/nxsl/java
       com.novell.nds.dirxml.driver.XdsCommandProcessor"
>
<xsl:param name="srcCommandProcessor"/>

<xsl:template match="node()|@*">
     <xsl:copy>
            <xsl:apply-templates select="@*|node()"/>
      </xsl:copy>
</xsl:template>

<xsl:template match="add">
      <xsl:copy>
            <xsl:apply-templates select="@*|node()"/>
      </xsl:copy>

      <!-- on a user add, add Engineering department to the source
object -->
      <xsl:variable name="dummy">
            <modify class-name="{@class-name} "dest-dn="{@src-dn}">
                  <xsl-copy-of select="association"/>
                  <modify-attr attr-name="OU">
                  <add-value>
                              <value type="string">Engineering</value>
```

```
                        </add-value>
                    </modify-attr>
                </modify>
            </xsl:variable>
            <xsl:variable name="dummy2"
               select="cmd:execute($srcCommandProcessor, $dummy)"/>
</xsl:template>

</xsl:transform>
```

5.5 使用扩展功能

XSLT 用于执行某些类型转换时是非常理想的工具，但用于其它类型转换 （例如，不常用字符串操作和迭代进程）时，却并不是非常理想。好在 Novell XSLT 处理程序配备了扩展功能，通过这些功能，样式表可以调用 Java 中实现的功能，还可以调用通过 JNI 访问的任何其它语言 （通过扩展）。

有关特定的示例，请参见使用查询处理程序的上述示例，以及使用 Java 说明字符串操作的以下示例 （特别长的行已换行且不以 < 开始）。

```
<!-- get-dn-prefix places the part of the passed dn that -->
<!-- precedes the last occurrence of '\' in the passed dn -->
<!-- in a result tree fragment meaning that it can be    -->
<!-- used to assign a variable value                     -->

<xsl:template name="get-dn-prefix" xmlns:jstring="http://
     www.novell.com/nxsl/java/java.lang.String">

   <xsl:param name="src-dn"/>

<!-- use java string stuff to make this much easier -->
   <xsl:variable name="dn" select="jstring:new($src-dn)"/>
   <xsl:variable name="index" select="jstring:lastIndexOf
       ($dn,'\')"/>
   <xsl:if test="$index != -1">
      <xsl:value-of select="jstring:substring($dn,0,$index)
          "/>
   </xsl:if>
</xsl:template>
```

5.6 创建口令示例：创建策略

下面的样式表可用于创建策略。此样式表创建用户，从用户的姓氏和 CN 特性中生成用户的口令并执行身份转换 （该转换可通过文档中的一切内容，但正尝试截获和转换的事件除外）。

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<!--   This stylesheet has an example of how to replace a create rule
with
```

```
        an XSLT stylesheet and supply an initial password for "User"
objects. -->

<xsl:transform xmlns:xsl="http://www.w3.org/1999/XSL/Transform
     "version="1.0">

<!-- ensure we have required NDS attributes -->
<xsl:template match="add">
   <xsl:if test="add-attr[@attr-name='Surname'] and
                 add-attr[@attr-name='CN']">
      <!-- copy the add through -->
      <xsl:copy>
         <xsl:apply-templates select="@*|node()"/>
         <!-- add a <password> element -->
         <xsl:call-template name="create-password"/>
      </xsl:copy>
   </xsl:if>

<!-- if the xsl:if fails, we don't have all the required attributes
      so we won't copy the add through, and the create rule will veto
the add -->

</xsl:template>

<xsl:template name="create-password">
   <password>
      <xsl:value-of select="concat(add-attr[@attr-name='Surname']/
value,
           '-',add-attr[@attr-name='CN']/value)"/>
   </password>
</xsl:template>

<!-- identity transform for everything we don't want to change -->

<xsl:template match="@*|node()">
    <xsl:copy>
        <xsl:apply-templates select="@*|node()"/>
    </xsl:copy>
</xsl:template>

</xsl:transform>
```

5.7 创建 eDirectory 用户示例：创建策略

此样式表可用于创建策略。它显示了如何从外部应用程序创建的项中创建 eDirectory 用户。在此示例中，首先在人力资源数据库中创建一个新雇佣的人员，随后再在网络中创建。它使用用户的名和姓在 eDirectory 树中生成唯一的 CN。尽管 eDirectory 仅要求 CN 在其特定的树枝中是唯一的，但此样式表可确保该 CN 在 eDirectory 树的所有树枝中都是唯一的。

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<!--  This stylesheet is an example of how to replace a create rule
```

```
with an
      XSLT stylesheet and that creates the User name from the Surname
and
      given Name attributes -->

<xsl:transform
   xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
   xmlns:query="http://www.novell.com/nxsl/java/
com.novell.nds.dirxml.driver.
         XdsQueryProcessor"
   >

<!-- This is for testing the stylesheet outside of Identity Manager so
things
     are pretty to look at -->
<xsl:strip-space elements="*"/>
<xsl:preserve-space elements="value,component"/>
<xsl:output method="xml" indent="yes"/>

<!-- Identity Manager always passes two stylesheet parameters to an
XSLT rule:
     an inbound and outbound query processor -->
<xsl:param name="srcQueryProcessor"/>
<xsl:param name="destQueryProcessor"/>

<!-- match <add> elements -->
<xsl:template match="add">

   <!-- ensure we have required NDS attributes we need for the name -->
   <xsl:if test="add-attr[@attr-name='Surname'] and
                 add-attr[@attr-name='Given Name']">

      <!-- copy the add through -->
      <xsl:copy>
         <!-- copy any attributes through except for the src-dn -->
         <!-- we'll construct the src-dn below so that the placement
rule will work -->
         <xsl:apply-templates select="@*[string(.) != 'src-dn']"/>

        <!-- call a template to construct the object name and place the
result in a variable -->
         <xsl:variable name="object-name">
            <xsl:call-template name="create-object-name"/>
         </xsl:variable>

         <!-- now create the src-dn attribute with the created name -->
         <xsl:attribute name="src-dn">
            <xsl:variable name="prefix">
               <xsl:call-template name="get-dn-prefix">
                  <xsl:with-param name="src-dn" select="string(@src-
dn)"/>
               </xsl:call-template>
            </xsl:variable>
            <xsl:value-of select="concat($prefix,'\',$object-name)"/>
```

```
         </xsl:attribute>

         <!-- if we have a "CN" attribute, set it to the constructed
name -->
         <xsl:if test="./add-attr[@attr-name='CN']">
            <add-attr attr-name="CN">
               <value type="string"><xsl:value-of select="$object-
name"/></value>
            </add-attr>
         </xsl:if>

         <!-- copy the rest of the stuff through, except for what we
have already copied -->
         <xsl:apply-templates select="*[name() != 'add-attr' or @attr-
name != 'CN'] |
                                      comment() |
                                       processing-instruction() |
                                        text()"/>

         <!-- add a <password> element -->
         <xsl:call-template name="create-password"/>

      </xsl:copy>
   </xsl:if>
   <!-- if the xsl:if fails, it means we don't have all the required
attributes
       so we won't copy the add through, and the create rule will veto
the add -->
</xsl:template>

<!-- get-dn-prefix places the part of the passed dn that precedes the
-->
<!-- last occurrance of '\' in the passed dn in a result tree fragment
-->
<!-- meaning that it can be used to assign a variable value
-->
<xsl:template name="get-dn-prefix" xmlns:jstring="http://
www.novell.com/nxsl/java/java.lang.String">
   <xsl:param name="src-dn"/>

   <!-- use java string stuff to make this much easier -->
   <xsl:variable name="dn" select="jstring:new($src-dn)"/>
   <xsl:variable name="index" select="jstring:lastIndexOf($dn,'\')"/>
   <xsl:if test="$index != -1">
      <xsl:value-of select="jstring:substring($dn,0,$index)"/>
   </xsl:if>
</xsl:template>

<!-- create-object-name creates a name for the user object and places
the   -->
<!-- result in a result tree fragment
-->
<xsl:template name="create-object-name">
```

```
   <!-- first try is first initial followed by surname -->
   <xsl:variable name="given-name" select="add-attr[@attr-name='Given
Name']/value"/>
   <xsl:variable name="surname" select="add-attr[@attr-
name='Surname']/value"/>
   <xsl:variable name="prefix" select="substring($given-name,1,1)"/>
   <xsl:variable name="object-name" select="concat($prefix,$surname)"/
>

   <!-- then see if name already exists in NDS -->
   <xsl:variable name="exists">
      <xsl:call-template name="query-object-name">
         <xsl:with-param name="object-name" select="$object-name"/>
      </xsl:call-template>
   </xsl:variable>

   <!-- if exists, then try 1st fallback, else return result -->
   <xsl:choose>
      <xsl:when test="$exists != ''">
         <xsl:call-template name="create-object-name-2"/>
      </xsl:when>
      <xsl:otherwise>
         <xsl:value-of select="$object-name"/>
      </xsl:otherwise>
   </xsl:choose>

</xsl:template>

<!-- create-object-name-2 is the first fallback if the name created by
-->
<!-- create-object-name already exists
-->
<xsl:template name="create-object-name-2">

   <!-- first try is first name followed by surname -->
   <xsl:variable name="given-name" select="add-attr[@attr-name='Given
Name']/value"/>
   <xsl:variable name="surname" select="add-attr[@attr-
name='Surname']/value"/>
   <xsl:variable name="object-name" select="concat($given-
name,$surname)"/>

   <!-- then see if name already exists in NDS -->
   <xsl:variable name="exists">
      <xsl:call-template name="query-object-name">
         <xsl:with-param name="object-name" select="$object-name"/>
      </xsl:call-template>
   </xsl:variable>

   <!-- if exists, then try last fallback, else return result -->
   <xsl:choose>
      <xsl:when test="$exists != ''">
         <xsl:call-template name="create-object-name-fallback"/>
      </xsl:when>
```

```
      <xsl:otherwise>
         <xsl:value-of select="$object-name"/>
      </xsl:otherwise>
   </xsl:choose>

</xsl:template>

<!-- create-object-name-fallback recursively tries a name created by
-->
<!-- concatenating the surname and a count until NDS doesn't find
-->
<!-- the name. There is a danger of infinite recursion, but only if
-->
<!-- there is a bug in NDS                                          -
->
<xsl:template name="create-object-name-fallback">
   <xsl:param name="count" select="1"/>

   <!-- construct the a name based on the surname and a count -->
   <xsl:variable name="surname" select="add-attr[@attr-
name='Surname']/value"/>
   <xsl:variable name="object-name" select="concat($surname,'-
',$count)"/>

   <!-- see if it exists in NDS -->
   <xsl:variable name="exists">
      <xsl:call-template name="query-object-name">
         <xsl:with-param name="object-name" select="$object-name"/>
      </xsl:call-template>
   </xsl:variable>

   <!-- if exists, then try again recursively, else return result -->
   <xsl:choose>
      <xsl:when test="$exists != ''">
         <xsl:call-template name="create-object-name-fallback">
            <xsl:with-param name="count" select="$count + 1"/>
         </xsl:call-template>
      </xsl:when>
      <xsl:otherwise>
         <xsl:value-of select="$object-name"/>
      </xsl:otherwise>
   </xsl:choose>

</xsl:template>

<!-- query object name queries NDS for the passed object-name. Ideally,
this would   -->
<!-- not depend on "CN": to do this, add another parameter that is the
name of the   -->
<!-- naming attribute.
-->
<xsl:template name="query-object-name">
   <xsl:param name="object-name"/>
```

```
   <!-- build an xds query as a result tree fragment -->
   <xsl:variable name="query">
      <nds ndsversion="8.5" dtdversion="1.0">
         <input>
            <query>
              <search-class class-name="{ancestor-or-self::add/@class-
name}"/>
               <!-- NOTE: depends on CN being the naming attribute -->
               <search-attr attr-name="CN">
                  <value><xsl:value-of select="$object-name"/></value>
               </search-attr>
              <!-- put an empty read attribute in so that we don't get
the whole object back -->
               <read-attr/>
            </query>
         </input>
      </nds>
   </xsl:variable>

   <!-- query NDS -->
   <xsl:variable name="result"
select="query:query($destQueryProcessor,$query)"/>

   <!-- return an empty or non-empty result tree fragment depending on
result of query -->
   <xsl:value-of select="$result//instance"/>
</xsl:template>

<!-- create an initial password -->
<xsl:template name="create-password">
   <password>
      <xsl:value-of select="concat(add-attr[@attr-name='Surname']/
value,'-',add-attr[@attr-name='CN']/value)"/>
   </password>
</xsl:template>

<!-- identity transform for everything we don't want to mess with -->
<xsl:template match="@*|node()">
    <xsl:copy>
        <xsl:apply-templates select="@*|node()"/>
    </xsl:copy>
</xsl:template>

</xsl:transform>
```

管理过滤器

6

过滤器编辑器允许您管理过滤器。在过滤器编辑器中，定义应如何由发布者通道和订购者通道处理每个类和特性。

本节包含以下与过滤器相关的主题：

6.1 Designer 中的过滤器任务

本节包含有关在 Designer 中执行与过滤器相关的常见任务的说明：

6.1.1 访问过滤器编辑器

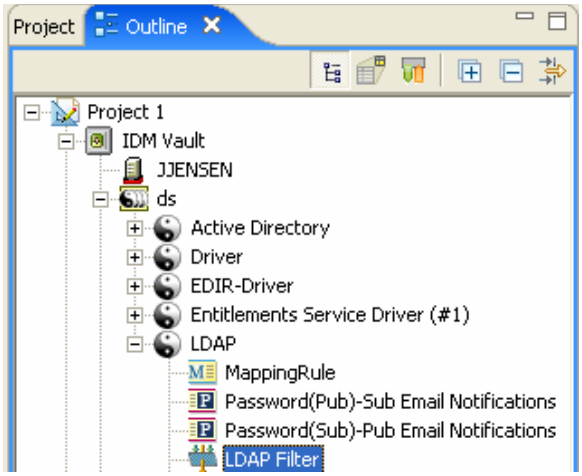
过滤器编辑器允许您编辑过滤器。可以通过三种方法访问过滤器编辑器：通过模型大纲、策略流和策略集视图。

模型大纲视图

1. 在打开的项目中，单击 " 大纲 " 选项卡。
2. 单击 " 显示模型大纲 " 图标。 
3. 选择要用于管理过滤器的驱动程序，然后单击右侧的加号。
4. 双击 " 过滤器 " 图标以起动过滤器编辑器。

   或者

右击并选择 " 编辑 "。

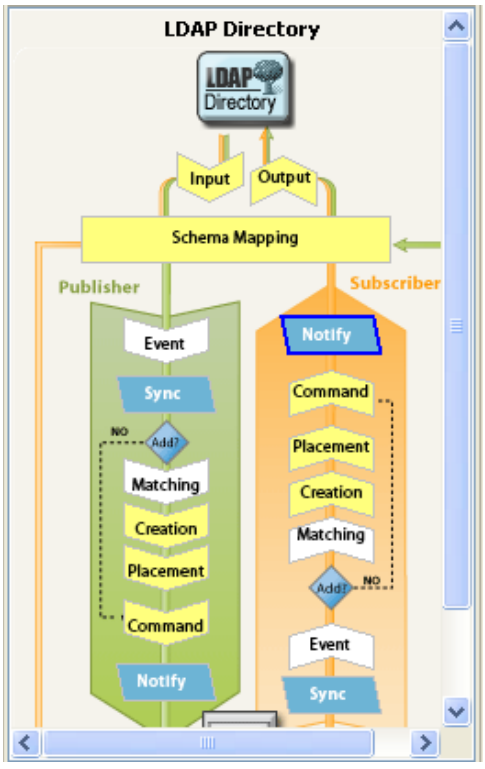


策略流程视图

1 在打开的项目中，单击 " 大纲 " 选项卡。

2 选择 " 显示策略流 " 图标。

3 双击 " 同步 " 图标或 " 通知 " 图标以起动过滤器编辑器。



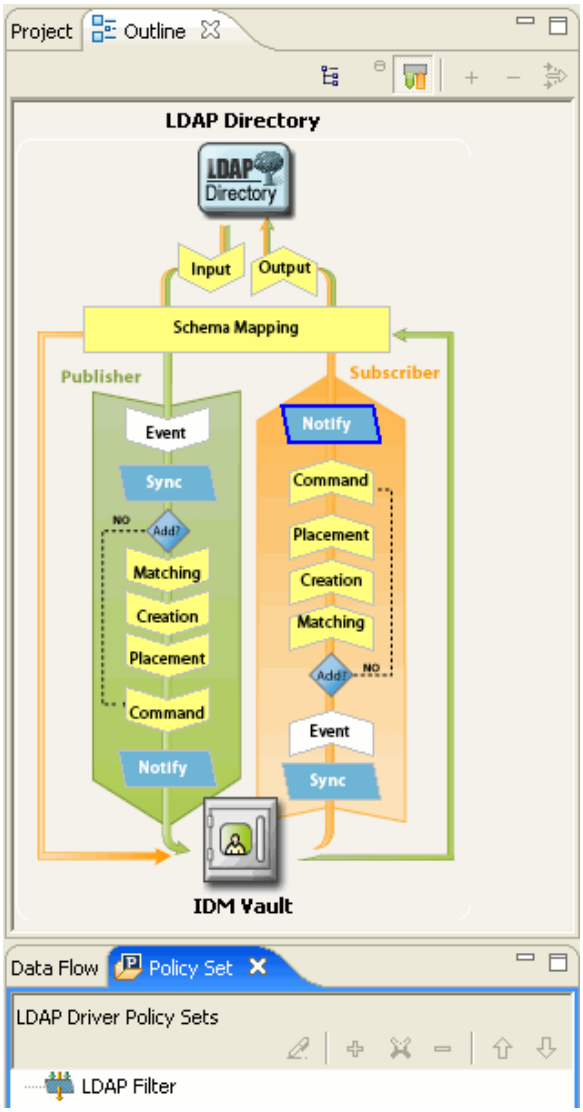
4 当过滤器出现在 " 策略流 " 下面的 " 策略集管理器 " 中时，双击该过滤器以起动过滤器编辑器。

或者

右击并选择 " 编辑策略 ">" 过滤器 "。

策略集视图

1 双击此过滤器策略。



键盘支持

表 *6-1* 过滤器编辑器键盘支持功能

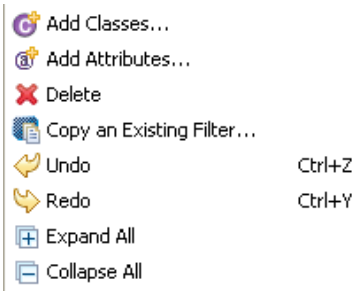
操作	说明
向上箭头	在过滤器编辑器中上移光标。
向下箭头	在过滤器编辑器中下移光标。
向左箭头	折叠显示的信息

操作	说明
向右箭头	展开显示的信息。
Insert	添加类。
Ctrl+Insert	添加特性。
Delete	删除选定的项目。
Enter	访问编辑方式。再次按 Enter 可提交更改。
Esc	退出编辑方式。

6.1.2 编辑过滤器

过滤器编辑器允许您创建并编辑过滤器。要显示上下文菜单，请右击某项目。

图 *6-1* 过滤器选项



- “去除或添加类和特性”在第 364 页
- “修改多个特性”在第 365 页
- “复制现有的过滤器”在第 365 页
- “设置特性的默认值”在第 365 页
- “更改过滤器设置”在第 366 页

去除或添加类和特性

通过去除或添加类和特性，确定在已连接数据存储区和 Identity Vault 之间同步的对象。

去除类或特性

如果不想同步类或特性，最好从过滤器中彻底地去除类或特性。可以使用两种方法从过滤器中添加或去除特性和类：

- 右击想要去除的类或特性，然后选择 " 删除 "。
- 选择想要去除的类或特性，然后单击右上角的 " 删除 " 图标 ✖。

添加类

1 右击过滤器编辑器，然后单击 " 添加类 "。

或者

单击右上角的 " 添加类 " 图标

2 浏览并选择想要添加的类，然后单击 " 确定 "。

3 更改选项以同步信息。

4 要保存这些更改，请单击 " 文件 ">" 保存 "。

添加特性

1 在过滤器编辑器中右击，然后单击 " 添加特性 "。

或者

单击右上角的 " 特性 " 图标 。

2 浏览并选择要添加的特性，然后单击 " 确定 "。

3 更改选项以同步信息。

4 要保存这些更改，请单击 " 文件 ">" 保存 "。

修改多个特性

过滤器编辑器允许您同时修改多个特性。按住 Ctrl 键并选择多个特性；更改选项时将更改所有选定特性的选项。

复制现有的过滤器

您可以从其它驱动程序中复制一个现有的过滤器，然后在当前使用的驱动程序中使用该过滤器。

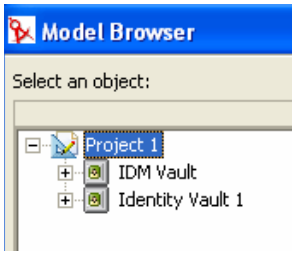
1 单击 " 复制现有过滤器 " 图标

或者

在过滤器编辑器中右击，然后单击 " 复制现有过滤器 "。

2 浏览至要复制的过滤器对象并选择该对象，然后单击 " 确定 "。

如果项目中有多个 Identity Vault，则可以从其它 Identity Vault 中复制过滤器。如果正在浏览至其它对象并选择了其它对象，则可以浏览至其它 Identity Vault 并使用其储存的过滤器。



设置特性的默认值

将新特性添加到过滤器时，可以定义新特性的默认值。

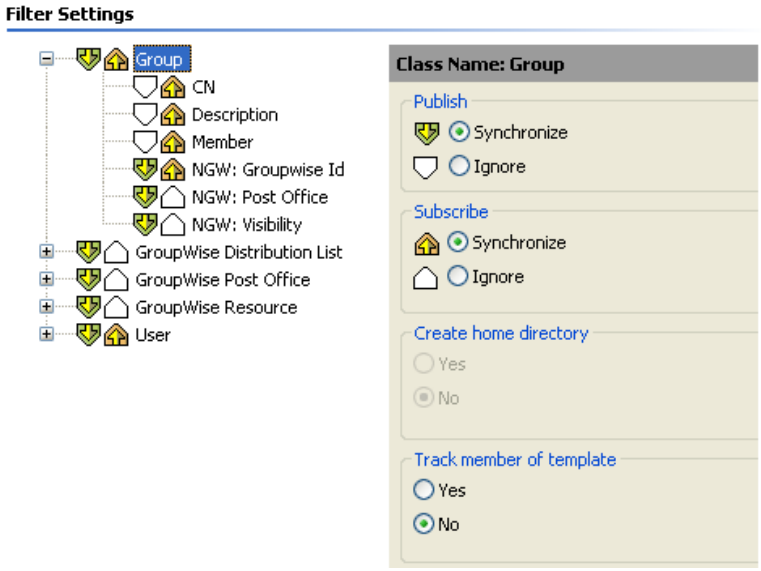
1 单击右上角的 " 设置新特性的默认值 " 图标 。

2 选择希望新特性具有的选项，然后单击 " 确定 "。

更改过滤器设置

过滤器编辑器提供了更改选项，以更改 Identity Vault 和已连接系统之间同步信息的方式。过滤器具有针对类和特性的不同设置。

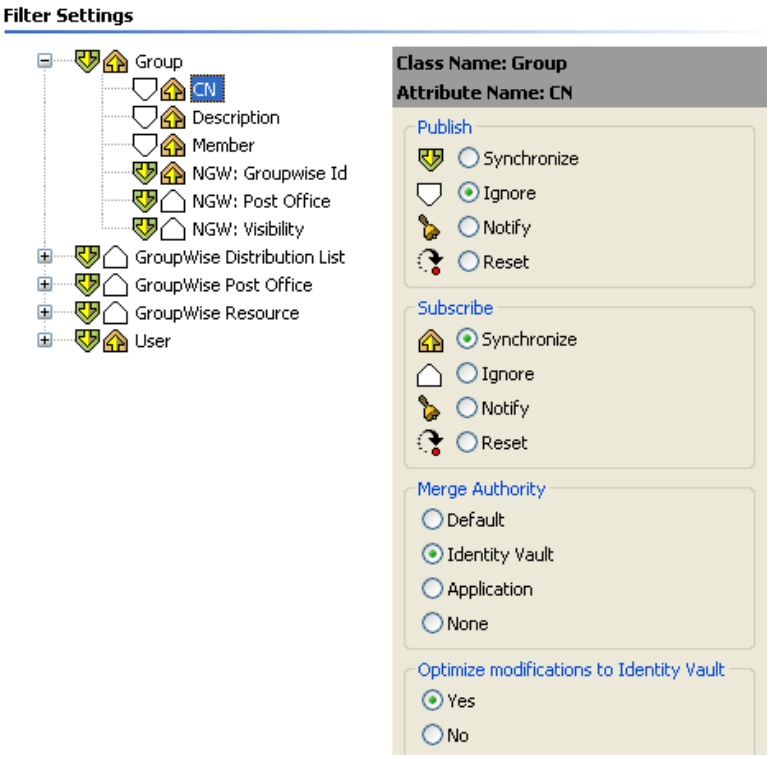
1 在过滤器编辑器中选择一个类。



2 更改所选类的过滤器设置。

选项	定义
发布者	• **同步：** 允许将该类从已连接系统同步到 Identity Vault。 • **忽略：** 不将该类从已连接系统同步到 Identity Vault。
订购者	• **同步：** 允许将该类从 Identity Vault 同步到已连接系统。 • **忽略：** 不将该类从 Identity Vault 同步到已连接系统。
创建用户主目录	• **是：** 自动创建用户主目录。 • **否：** 不创建用户主目录。
跟踪模板成员	• **是：** 确定发布者通道从模板创建对象时是否保持 Member of Template （模板成员）特性。 • **否：** 不跟踪 Member of Template （模板成员）特性。

3 选择特性。



4 更改选定特性的过滤器设置。

选项	定义
发布者	• **同步：**报告并自动同步对该对象的更改。 • **忽略：**既不报告也不自动同步对该对象的更改。 • **通知：**报告但不自动同步对该对象的更改。 • **重设置：**将对象值重设置为由相反通道指定的值。（可以在发布者通道或订购者通道设置该值，但不可以在两个通道均设置该值。）
订购者	• **同步：**报告并自动同步对该对象的更改。 • **忽略：**既不报告也不自动同步对该对象的更改。 • **通知：**报告但不自动同步对该对象的更改。 • **重设置：**将对象值重设置为由相反通道指定的值。（可以在发布者通道或订购者通道设置该值，但不可以在两个通道均设置该值。）

选项	定义
合并权限	◆ **默认行为：**如果未在任一通道中同步某特性，则不发生合并。 如果在一个通道 （而未在另一个通道）中同步某特性，则去除该通道目标的所有现有值，并由该通道源的值代替。如果源具有多个值而目标只能容纳一个值，则目标侧仅使用其中的一个值。 如果在两个通道中均同步某特性，且两侧都只能容纳一个值，则只要 Identity Vault 存在值，已连接应用程序便可获得 Identity Vault 值。在这种情况下， Identity Vault 从已连接应用程序获得值 （如果有）。 如果在两个通道中均同步某特性，且只有一侧能容纳多个值，则如果单值侧的值不在多值侧，此值将被添加到多值侧。如果单值侧没有值，则可以选择值添加到单值侧。 这始终是有效的行为。 ◆ **Identity Vault：**如果某特性在订购者通道中同步而未在发布者通道中同步，则 Identity Vault 的行为方式与默认行为相同。 在订购者通道中同步时，这是有效的行为。 ◆ **应用程序：**如果某特性在发布者通道中同步而未在订购者通道中同步，则应用程序的行为方式与默认行为相同。 在发布者通道中同步时，这是有效的行为。 ◆ *无*：无论同步与否，都不发生合并。
对 Identity Manager 的优化修改	◆ **是：**在发布者通道中检查对该特性的更改，以确定在 Identity Vault 中所做的最少更改。 ◆ **否：**不检查更改。

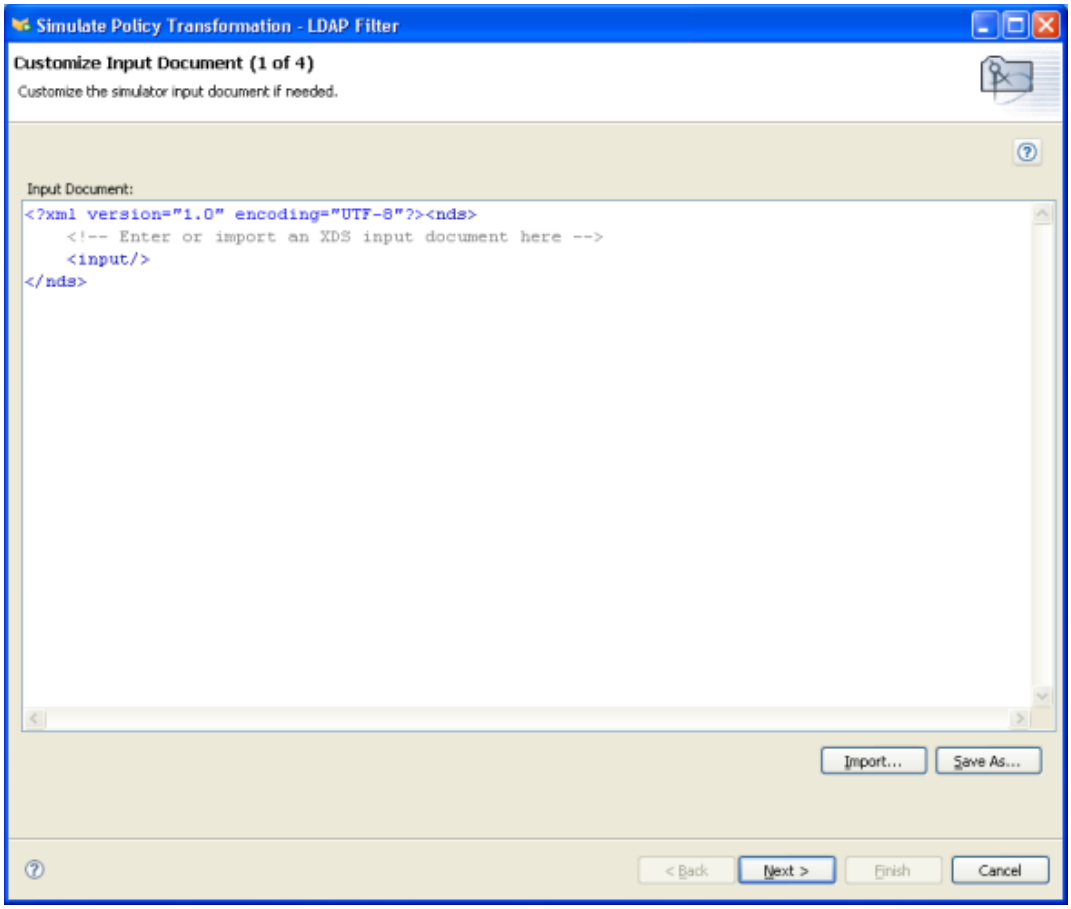
5 单击 " 保存 " 图标 以保存这些更改。

6.1.3 测试过滤器

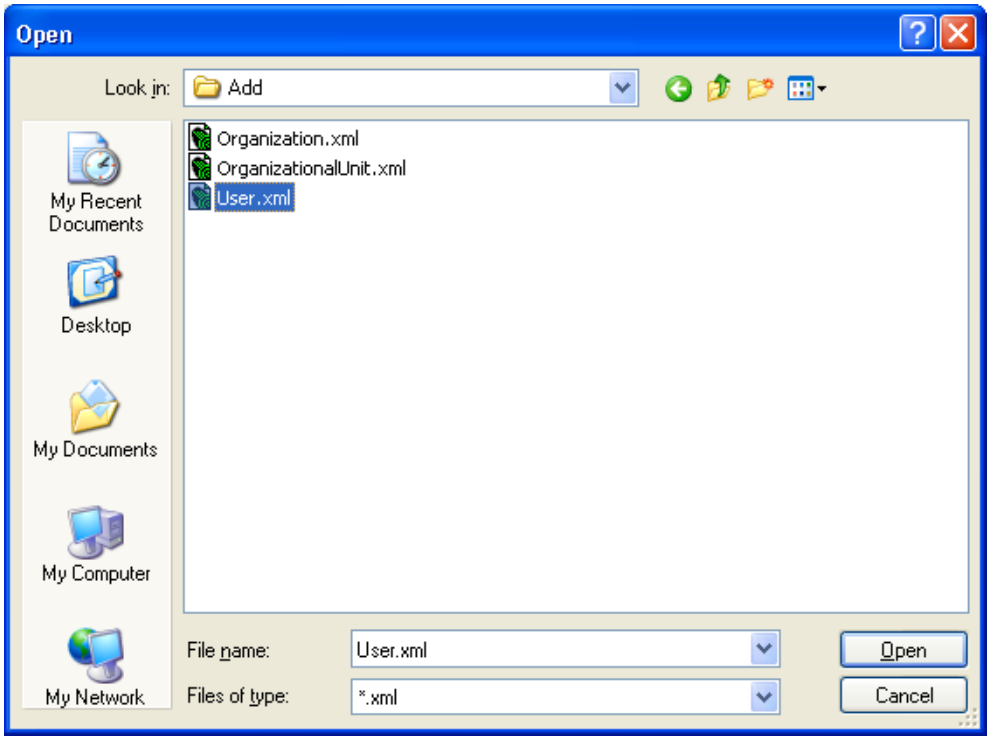
Designer 附带了一个名为策略模拟器的工具，利用该工具，无需在生产环境中实施策略便可对这些策略进行测试。修改策略后，您可以通过过滤器编辑器起动策略模拟器，以测试策略。

1 在工具栏中单击 " 起动策略模拟器 " 图标 。

2 选择 " 导入 " 以浏览至模拟事件的文件。

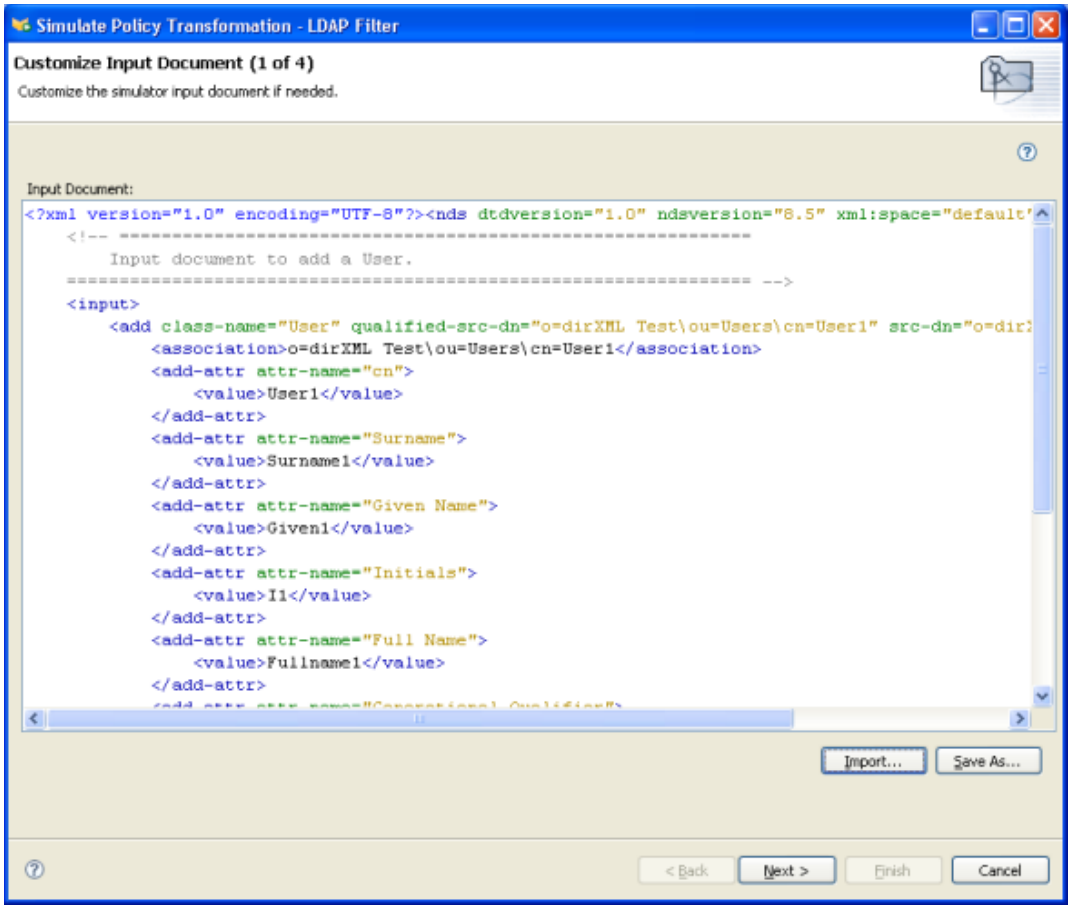


3 选择文件，然后单击 " 打开 "。本示例使用 com.novell.designer.idm.policy\simulation\add\User.xml 文件，该文件模拟用户对象的添加事件。



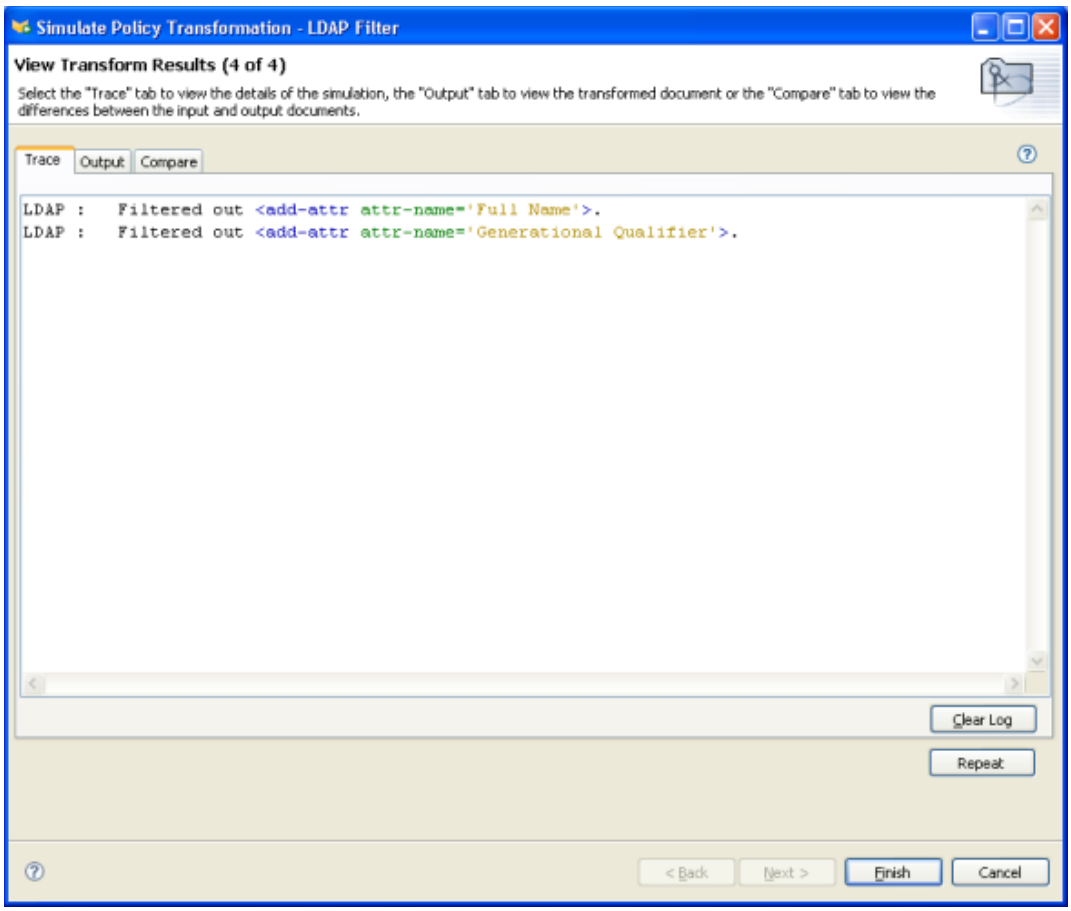
策略模拟器显示用户添加事件的输入文档。

4 单击 " 下一步 " 开始进行模拟。

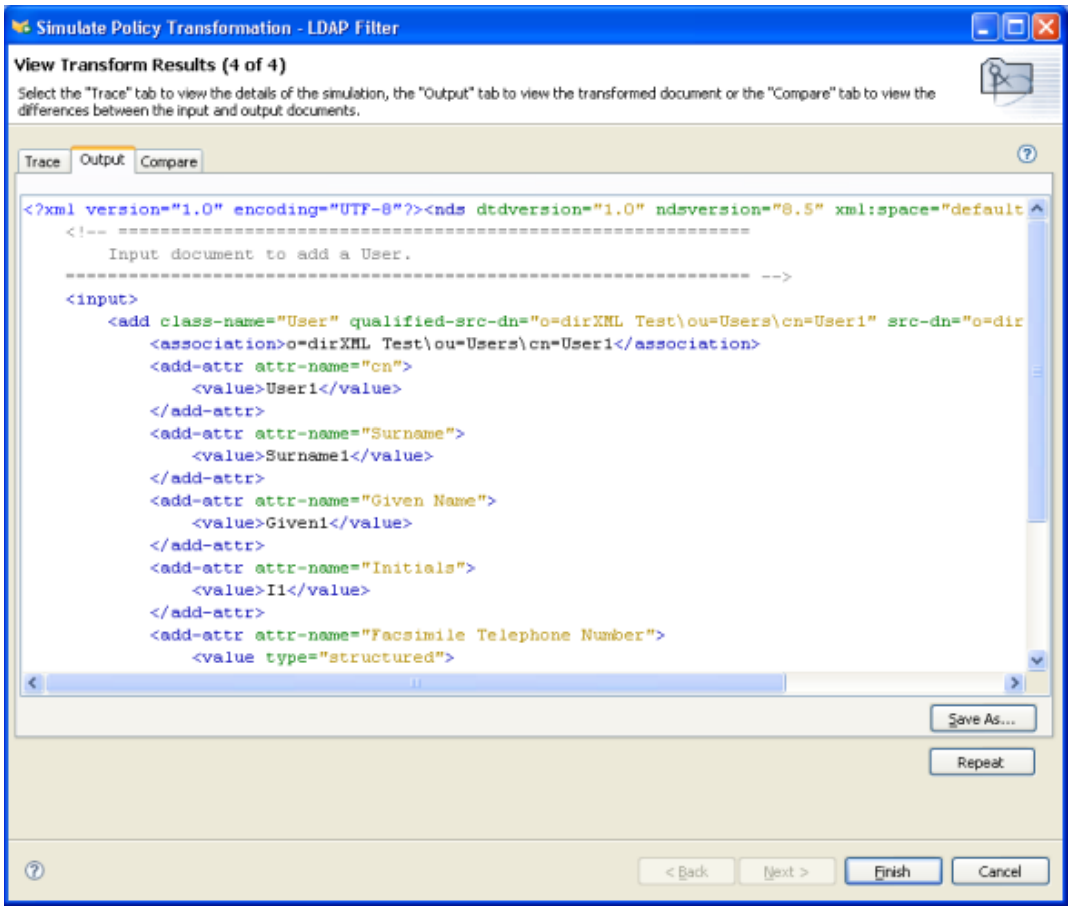


策略模拟器显示添加事件的日志、输出文档和输入文档与生成的输出文档的比较。

5 选择 " 跟踪 " 选项卡将显示添加事件的结果，如 DSTRACE 中所示。

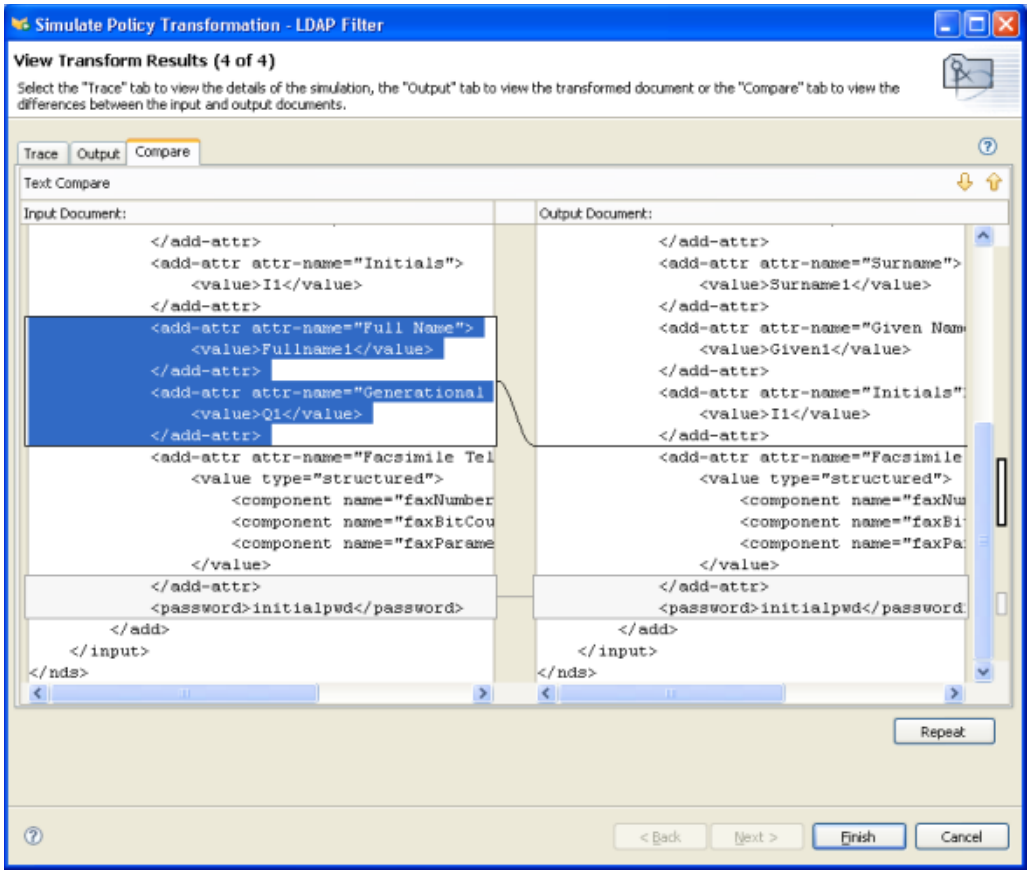


6 选择 " 输出 " 选项卡以查看对输入文档执行过滤时生成的输出文档。输入文档就是用户添加事件。



可以编辑输入文档和输出文档。如果要保留更改，请单击 " 另存为 "。

7 选择 " 比较 " 选项卡以对输入文档的文本与生成的输出文档进行比较。



8 单击 " 重复 " 以选择其它输入文档并查看该事件的结果。

9 完成测试过滤器时，请单击 " 完成 " 以关闭策略模拟器。

6.1.4 查看过滤器 XML 源

在 Designer 中，您可以使用 XML 编辑器或文本编辑器来查看、编辑和验证 XML。

- “查看 XML 源”在第 374 页
- “编辑 XML 源”在第 377 页
- “验证 XML 源”在第 379 页

查看 **XML** 源

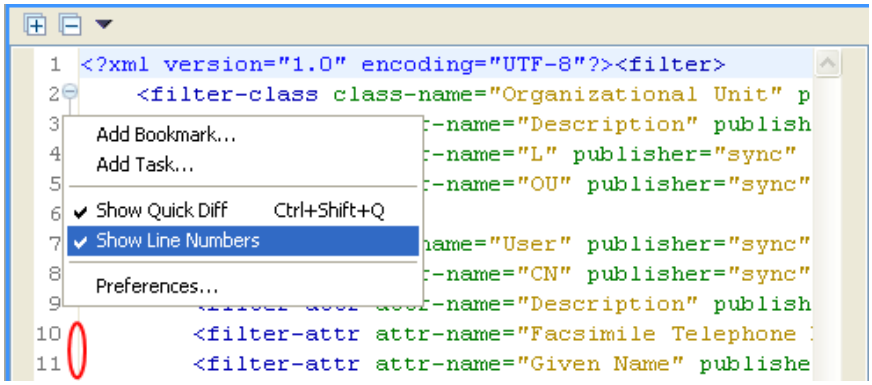
可以查看 XML 格式或 XML 树格式的 XML 源。

要打开 XML 源视图，请执行以下操作：

1 在过滤器编辑器工作空间的底部单击 "XML 源 "。

XML 编辑器可以显示行号。要查看行号，可以右击左边距，然后选择 " 显示行号 "。

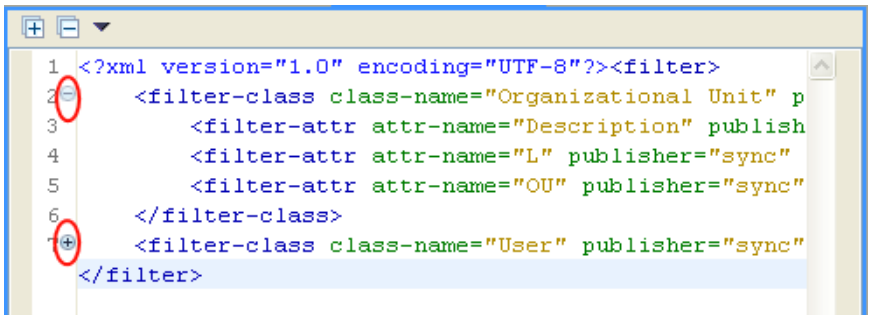
图 *6-2* 过滤器显示行号



XML 编辑器可以按函数展开或折叠 XML。如果函数中包含大量 XML，可以通过单击左上角的减号图标折叠 XML。要展开所有 XML 函数，请单击左上角的加号图标。

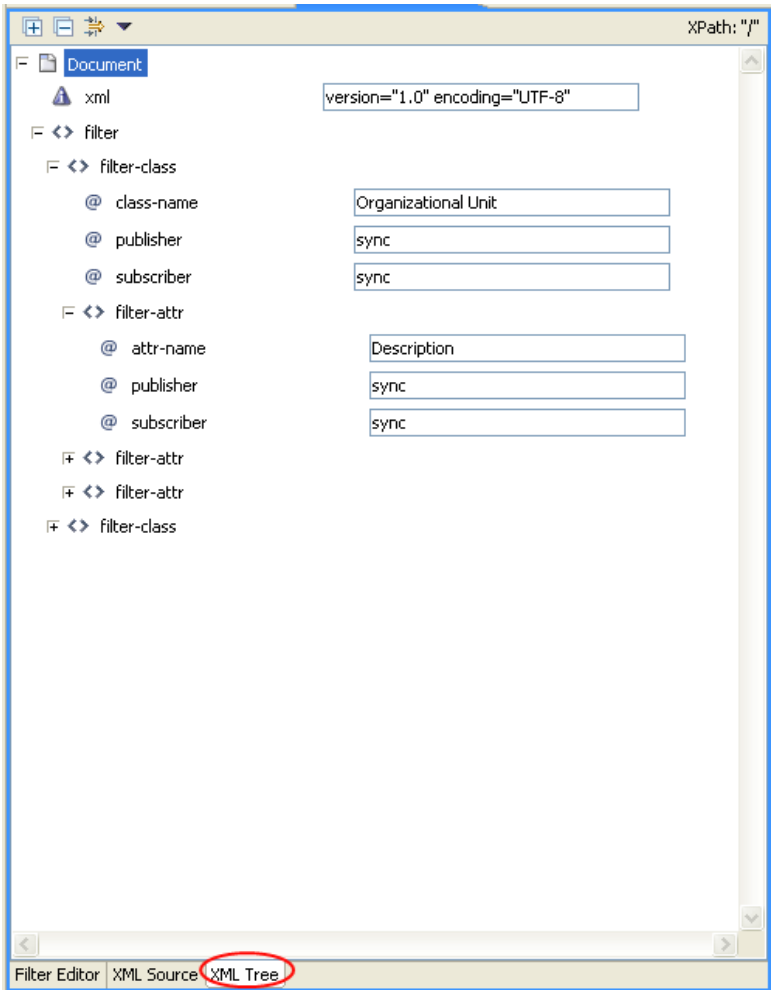
在左边距中，每个要素都有其各自的加号或减号图标。

图 ***6-3*** *过滤器 XML 加号或减号*



要以树格式查看 XML，请执行以下操作：

1 在过滤器编辑器工作空间的底部单击 "XML 树 "。

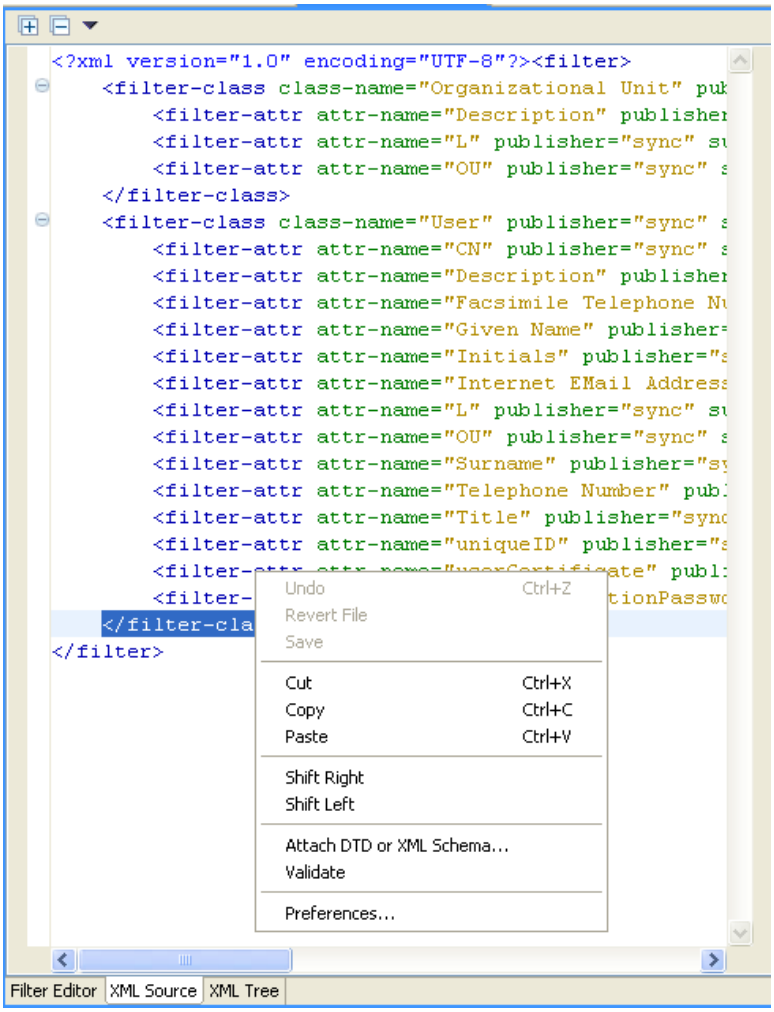


要查看整个树视图，请展开列出的每一项。

编辑 **XML** 源

可以通过 XML 编辑器来编辑 XML。可以在此处进行更改，也可以通过 GUI 界面来进行更改。

图 ***6-4*** 编辑过滤器的 *XML* 源



加载的默认编辑器与 .xml 文件类型相关联。如果找不到默认编辑器，则装载系统文本编辑器。 XML 源视图的功能取决于所加载的编辑器。

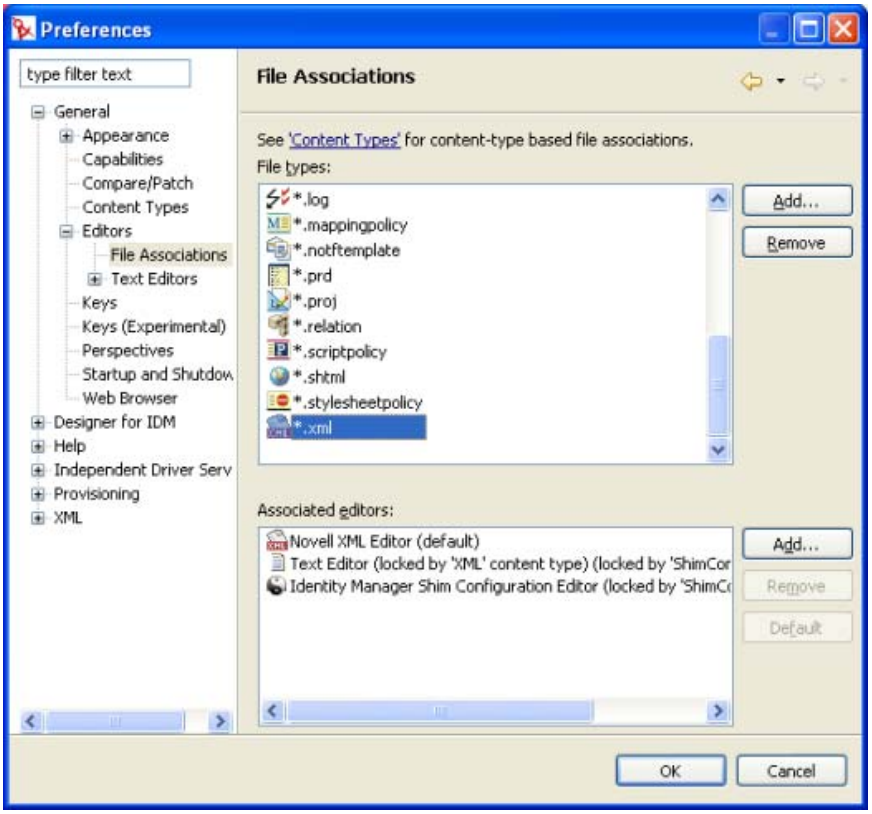
右击以显示 XML 编辑器所包含的功能列表。

- 复原：复原上一操作。
- 还原文件：将文件还原为所保存的上一版本。
- 保存：保存文件。
- 剪切：剪切所选信息。
- 复制：将选定的信息复制到剪贴板。
- 粘贴：将信息粘贴到文档中。
- 右移：将本行向右缩排。

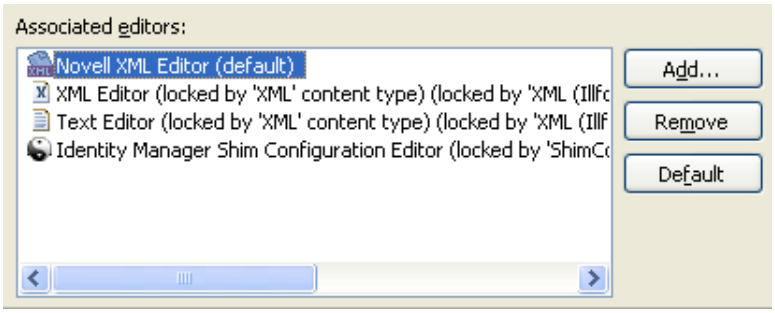
- 左移：将本行向左缩排。
- 挂接 **DTD** 或 **XML** 纲要：挂接 DTD 或 XML 纲要文件以验证策略。
- 验证：验证 XML 代码。
- 自选设置：设置 XML 编辑器的自选设置。

要为 XML 源视图选择不同的 XML 编辑器：

1. 单击主菜单中的 " 窗口 ">" 自选设置 "。
2. 单击 " 常规 ">" 编辑器 ">" 文件关联 "。
3. 从文件类型列表中选择 **xml*。



4. 从 " 关联的编辑器 " 中，选择希望使用的编辑器 （例如， Novell XML 编辑器）。（如果列表中没有您需要的编辑器，可以单击 " 添加 "，然后将其添加到列表中。）



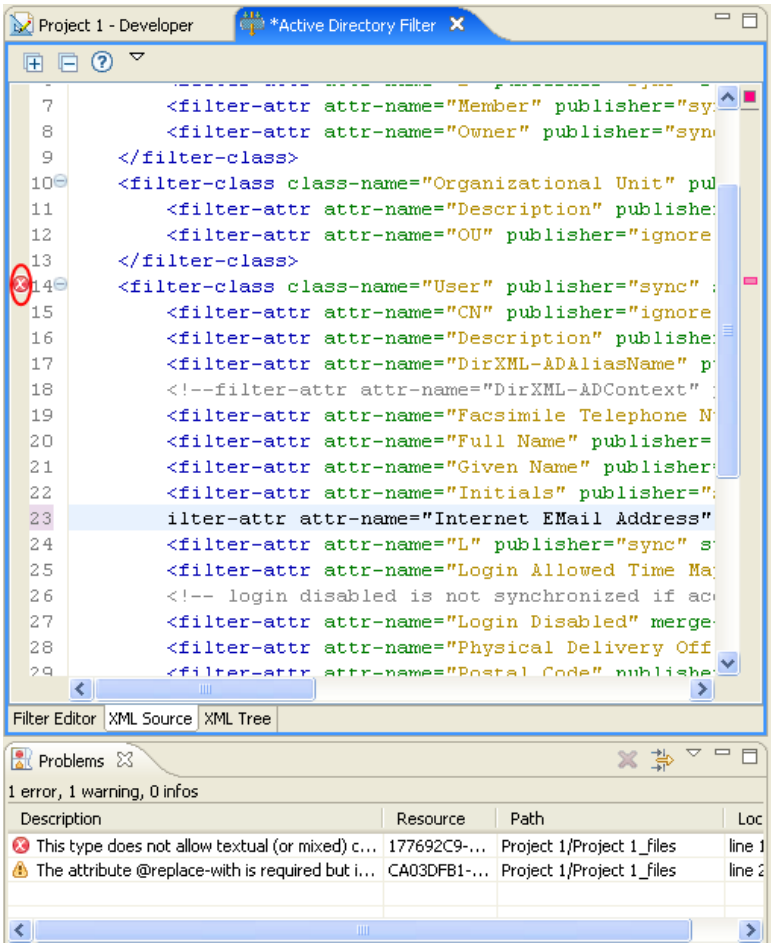
5. 单击 " 确定 "。

6 关闭并重新打开过滤器编辑器。将在 "XML 源 " 视图中加载默认编辑器。

验证 **XML** 源

XML 编辑器可以验证 XML 代码。右击并选择 " 验证 "。如果存在错误，则出现错误的行将显示一个红色的 x。窗口底部的解释将提供有关该问题的详细信息。

图 *6-5* 验证过滤器



本示例中，丢失了起始标签和 <filter-attr> 的首字母。

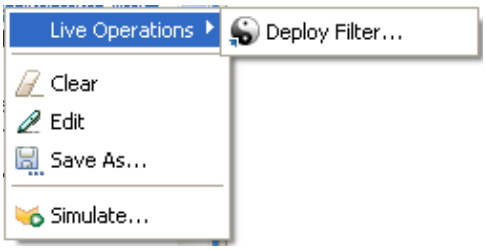
6.1.5 附加的过滤器选项

右击过滤器对象时，在 " 大纲 " 视图、 " 策略流程 " 视图和 " 策略集 " 视图中将出现多个选项。

- “" 大纲 " 视图的附加选项”在第 380 页
- “策略流程视图的附加选项”在第 380 页
- “" 策略集 " 视图的附加选项”在第 380 页

" 大纲 " 视图的附加选项

1 在 " 大纲 " 视图中，右击过滤器对象。



- **Live Operations** （实时操作） >" 部署过滤器 "：将过滤器部署到 Identity Vault 中。
- 清除：从过滤器策略中删除所有内容，但保留对象。
- 编辑：起动过滤器编辑器。有关详细信息，请参见 “编辑过滤器”在第 364 页 。
- 另存为：将过滤器另存为 .xml 文件。
- 模拟：起动策略模拟器。有关详细信息，请参见 “测试过滤器”在第 368 页 。

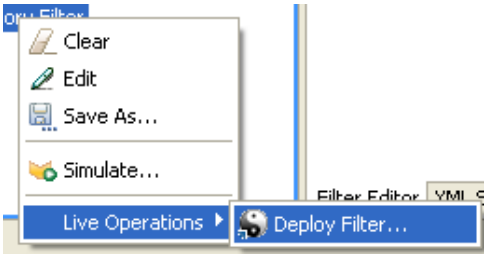
策略流程视图的附加选项

1 在 " 策略流程 " 视图中，右击过滤器对象。

- " 编辑策略 ">" 过滤器 "：起动过滤器编辑器。有关详细信息，请参见 “编辑过滤器”在第 364 页 。
- 模拟：起动策略模拟器。有关详细信息，请参见 “测试过滤器”在第 368 页 。

" 策略集 " 视图的附加选项

1 在 " 策略集 " 视图中，右击过滤器对象。



- 清除：从过滤器策略中删除所有内容，但保留对象。
- 编辑：起动过滤器编辑器。有关详细信息，请参见 “编辑过滤器”在第 381 页 。
- 保存：将过滤器另存为 Xml 文件。
- 模拟：起动策略模拟器。有关详细信息，请参见 “测试过滤器”在第 368 页 。

- " 实时操作 ">" 部署过滤器 "：允许您将过滤器部署到 Identity Vault 中。

6.2 iManager 中的过滤器任务

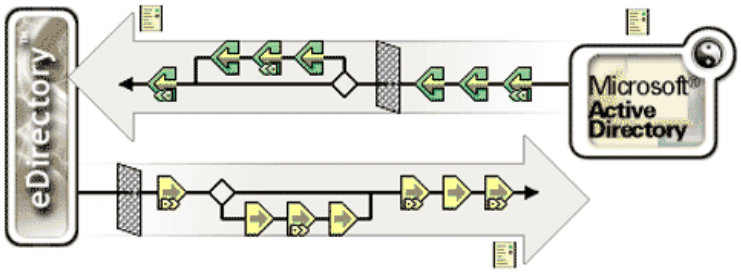
本节包含有关在 iManager 中执行与过滤器相关的常见任务的说明：

- “访问过滤器”在第 381 页
- “编辑过滤器”在第 381 页

6.2.1 访问过滤器

1. 在 iManager 中，展开 *Identity Manager* 职能，然后单击 "Identity Manager 概述 "。
2. 选择 " 搜索整个树 " 或 " 在树枝中搜索 "，然后单击 " 搜索 "。
3. 单击要访问的过滤器所属的驱动程序。将打开 "Identity Manager 驱动程序概述 "：

图 *6-6* 驱动程序概述



4. 单击发布者或订购者通道上的过滤器图标。它是同一个对象。

6.2.2 编辑过滤器

过滤器编辑器提供了编辑选项，以编辑在 Identity Vault 和已连接系统之间同步信息的方式。以下是编辑过滤器时最常见的任务列表：

- “从过滤器中去除类或特性”在第 381 页
- “添加类”在第 382 页
- “添加特性”在第 382 页
- “复制过滤器”在第 382 页
- “设置模板”在第 382 页
- “更改过滤器设置”在第 382 页

从过滤器中去除类或特性

1. 选择该类或特性，然后单击 " 删除 "。

添加类

1 单击 " 添加类 "。
2 更改选项以同步信息。
3 单击 " 应用 "。

添加特性

1 单击 " 添加特性 "。
2 更改选项以同步信息。
3 单击 " 应用 "。

复制过滤器

允许您将过滤器从现有的驱动程序复制到当前使用的驱动程序中。

1 单击 " 过滤器复制自 "。
2 浏览至要从中复制过滤器的驱动程序，然后单击 " 确定 "。

设置模板

允许您为添加到过滤器的特性设置默认值。

1 单击 " 设置模板 "。
2 选中希望新特性具有的选项，然后单击 " 确定 "。

创建特性后，可以更改它们的值。

更改过滤器设置

过滤器编辑器提供了更改选项，以更改 Identity Vault 和已连接系统之间同步信息的方式。过滤器具有针对类和特性的不同设置。

1 在过滤器编辑器中选择一个类。
2 更改所选类的过滤器设置。

选项	定义
发布者	◆ **同步：** 允许将该类从已连接系统同步到 Identity Vault。 ◆ **忽略：** 不将该类从已连接系统同步到 Identity Vault。
订购者	◆ **同步：** 允许将该类从 Identity Vault 同步到已连接系统。 ◆ **忽略：** 不将该类从 Identity Vault 同步到已连接系统。
创建用户主目录	◆ **是：** 自动创建用户主目录。 ◆ **否：** 不创建用户主目录。
跟踪模板成员	◆ **是：** 确定发布者通道从模板创建对象时是否保持 Member of Template （模板成员）特性。 ◆ **否：** 不跟踪 Member of Template （模板成员）特性。

3 选择特性。

4 更改选定特性的过滤器设置。

选项	定义
发布者	◆ **同步：**报告并自动同步对该对象的更改。 ◆ **忽略：**既不报告也不自动同步对该对象的更改。 ◆ **通知：**报告但不自动同步对该对象的更改。 ◆ **重设置：**将对象值重设置为由相反通道指定的值。（可以在发布者通道或订购者通道设置该值，但不可以在两个通道均设置该值。）
订购者	◆ **同步：**报告并自动同步对该对象的更改。 ◆ **忽略：**既不报告也不自动同步对该对象的更改。 ◆ **通知：**报告但不自动同步对该对象的更改。 ◆ **重设置：**将对象值重设置为由相反通道指定的值。（可以在发布者通道或订购者通道设置该值，但不可以在两个通道均设置该值。）
合并权限	◆ **默认行为：**如果未在任一通道中同步某特性，则不发生合并。 如果在一个通道 （而未在另一个通道）中同步某特性，则去除该通道目标的所有现有值，并由该通道源的值代替。如果源具有多个值而目标只能容纳一个值，则目标侧仅使用其中的一个值。 如果在两个通道中均同步某特性，且两侧都只能容纳一个值，则只要 Identity Vault 存在值，已连接应用程序便可获得 Identity Vault 值。在这种情况下， Identity Vault 从已连接应用程序获得值 （如果有）。 如果在两个通道中均同步某特性，且只有一侧能容纳多个值，则如果单值侧的值不在多值侧，此值将被添加到多值侧。如果单值侧没有值，则可以选择值添加到单值侧。 这始终是有效的行为。 ◆ **Identity Vault：**如果某特性在订购者通道中同步而未在发布者通道中同步，则 Identity Vault 的行为方式与默认行为相同。 在订购者通道中同步时，这是有效的行为。 ◆ **应用程序：**如果某特性在发布者通道中同步而未在订购者通道中同步，则应用程序的行为方式与默认行为相同。 在发布者通道中同步时，这是有效的行为。 ◆ *无*：无论同步与否，都不发生合并。
对 Identity Manager 的优化修改	◆ **是：**在发布者通道中检查对该特性的更改，以确定在 Identity Vault 中所做的最少更改。 ◆ **否：**不检查更改。

5 单击 " 确定 " 以保存更改。

管理纲要映射策略

7

纲要映射策略在 Identity Vault 名称空间和应用程序名称空间之间映射类名称和特性名称。双向应用相同的纲要映射策略。在 Metadirectory 引擎和应用程序 Shim 之间的任一通道中以任一方向传递的所有文档都通过纲要映射策略进行传递。

每个驱动程序都有一个纲要映射策略。

本节包含以下与过滤器相关的主题：

7.1 Designer 中的纲要映射策略任务

本节包含有关在 Designer 中执行与纲要映射策略相关的常见任务的指南：

7.1.1 访问纲要映射编辑器

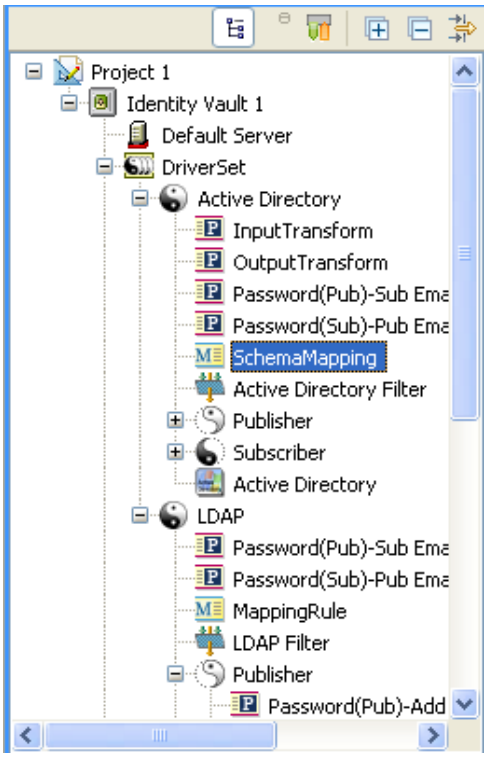
使用纲要映射编辑器可以编辑纲要映射策略。在 Designer 中，可以通过三种不同的方法访问纲要映射编辑器：通过 " 大纲 " 视图、通过 " 策略流程 " 视图或通过 " 策略集 " 视图。

" 大纲 " 视图

1. 在打开的项目中，单击 " 大纲 " 选项卡。
2. 单击 " 显示模型大纲 " 图标。 
3. 选择您希望在其上管理纲要映射策略的驱动程序，然后单击右侧的加号。
4. 双击 " 纲要映射 " 图标以起动纲要映射编辑器。

   或者

右击并选择 " 编辑 "。

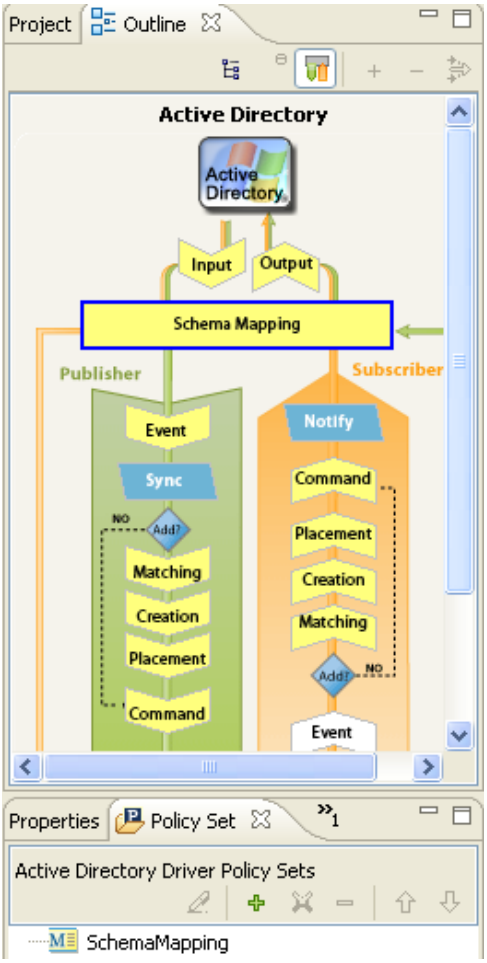


策略流程视图

1 在打开的项目中，单击 " 大纲 " 选项卡。
2 单击 *Show Policy Flow* （显示策略流程）图标。
3 双击 " 纲要映射 " 策略以起动纲要映射编辑器。

或者

右击并选择 " 编辑策略 " 以起动纲要映射编辑器。

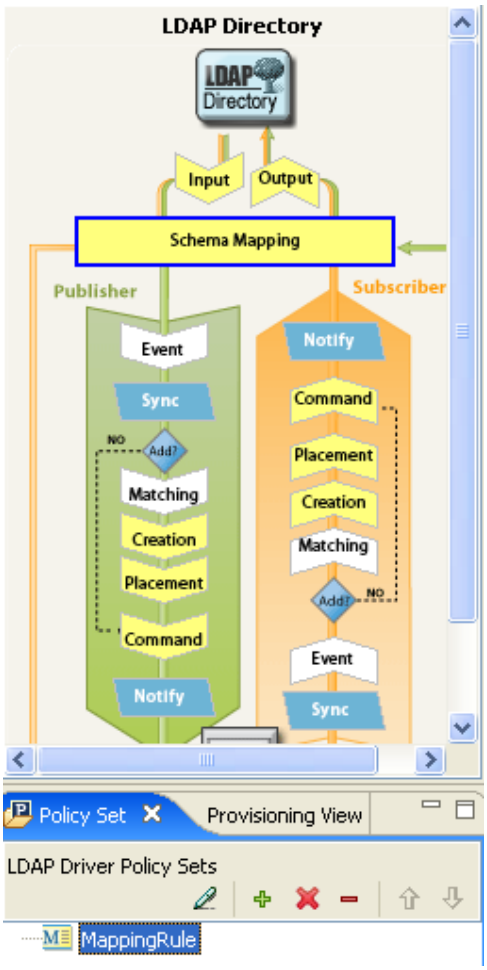


策略集视图

1 在 " 策略集 " 视图中，双击 " 纲要映射 " 策略。

或者

右击 " 纲要映射 " 策略并选择 " 编辑 "。



键盘支持

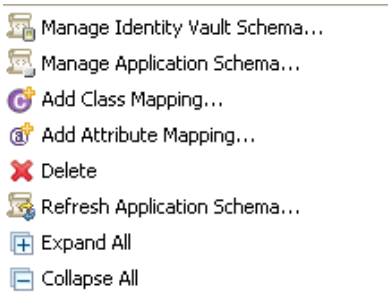
表 ***7-1*** 纲要映射编辑器的键盘支持

操作	说明
向上箭头	在纲要映射编辑器中，向上移动光标。
向下箭头	在纲要映射编辑器中，向下移动光标。
向左箭头	折叠显示的信息
向右箭头	展开显示的信息。
Insert	添加类。
Ctrl+Insert	添加特性。
Delete	删除选定的项目。
Enter	访问编辑方式。再次按 Enter 可提交更改。
Esc	退出编辑方式。

7.1.2 编辑纲要映射策略

使用纲要映射编辑器可以创建和编辑纲要映射策略。要显示上下文菜单，请右击某项目。

图 *7-1* 纲要映射编辑器的上下文菜单



- “去除或添加类和特性”在第 389 页
- “刷新应用程序纲要”在第 390 页
- “编辑项”在第 391 页
- “排序各项”在第 391 页
- “管理纲要”在第 391 页

去除或添加类和特性

- “去除类或特性”在第 389 页
- “添加类”在第 390 页
- “添加特性”在第 390 页

去除类或特性

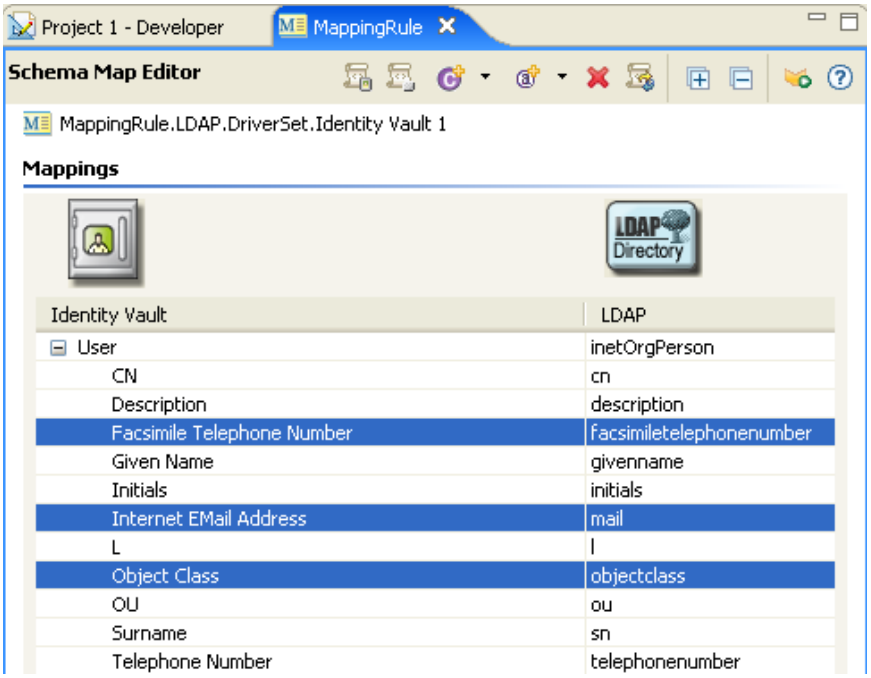
如果不希望某一类或特性映射到已连接系统中的类或特性，最好将该类或特性从纲要映射策略中彻底去除。在纲要映射策略中，可以通过三种不同的方法添加或去除特性和类：

- 选择希望去除的类或特性，然后右击并单击 " 删除 "。
- 选择想要去除的类或特性，然后单击右上角的 " 删除 " 图标 ✖。
- 选择希望去除的类或特性，然后按 Delete 键。

您可以同时选择多个要删除的类或特性。

1 按住 Ctrl 并使用鼠标选择每一项。

2 按 Delete 键删除这些项。



添加类

1 右击纲要映射编辑器，然后单击 *Add Class Mapping*（添加类映射）。

或者

选择右上角的 " 添加类映射 " 图标 。

2 从 Identity Vault 的下拉列表中，选择您希望添加的类。

3 从已连接系统的下拉列表中，选择您希望添加的类。

4 要保存更改，请单击 " 文件 ">" 保存 "。

添加特性

1 右击纲要映射编辑器，然后单击 *Add Attribute Mapping*（添加特性映射）。

或者

选择右上角的 " 添加特性映射 " 图标 。

2 从 Identity Vault 的下拉列表中，选择您希望添加的特性。

3 从已连接系统的下拉列表中，选择您希望添加的特性。

4 要保存更改，请单击 " 文件 ">" 保存 "。

刷新应用程序纲要

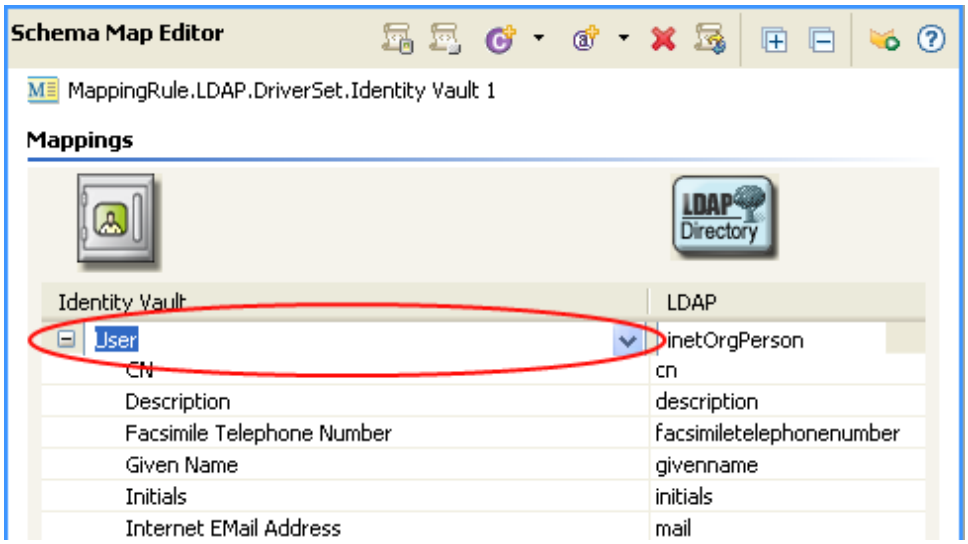
如果您修改了应用程序中的纲要，则需要将这些更改反映在纲要映射策略中。要使新的纲要可用，请单击工具栏中的 " 刷新应用程序纲要 " 图标 。

创建新的类或特性映射后，您可以在下拉列表中看到已连接应用程序的新纲要。

编辑项

要编辑映射，请双击选定的行。即出现一个现场编辑器，允许您编辑映射。

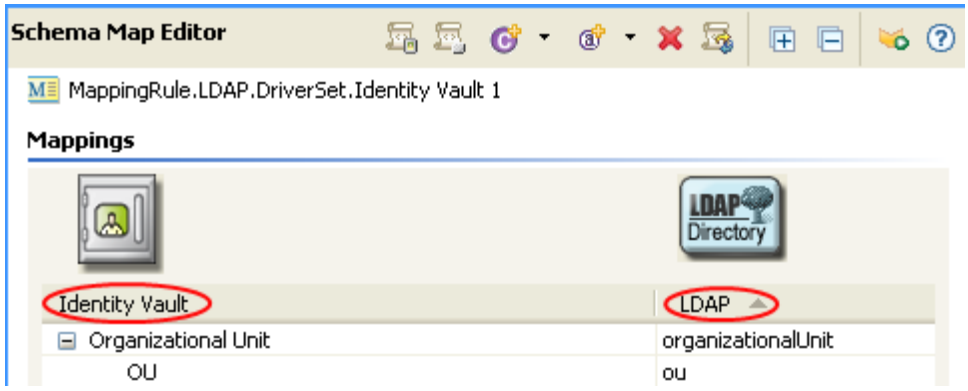
图 *7-2* 纲要映射编辑器



排序各项

使用纲要编辑器可以基于 Identity Manager 或已连接系统按升序排列各项。要排序，请单击任意一列的标题。

图 *7-3* 对纲要映射编辑器中的各项进行排序



管理纲要

使用 Designer 可以管理 Identity Vault 纲要和任意已连接系统的纲要。您可以导入、修改纲要，还可以将更改过的纲要部署回 Identity Vault 或已连接系统。要管理 Identity Vault 纲要，请右击 " 纲要映射编辑器 "，然后单击 *Manage Identity Vault Schema*（管理 Identity Vault 纲要）。要管理已连接系统纲要，请右击 " 纲要映射编辑器 "，然后单击 *Manage Application Schema*（管理应用程序纲要）。有关如何管理纲要的信息，请参见 《*Designer for Identity Manager 3*：管理帮助》中的 “Managing Schema（管理纲要）”。

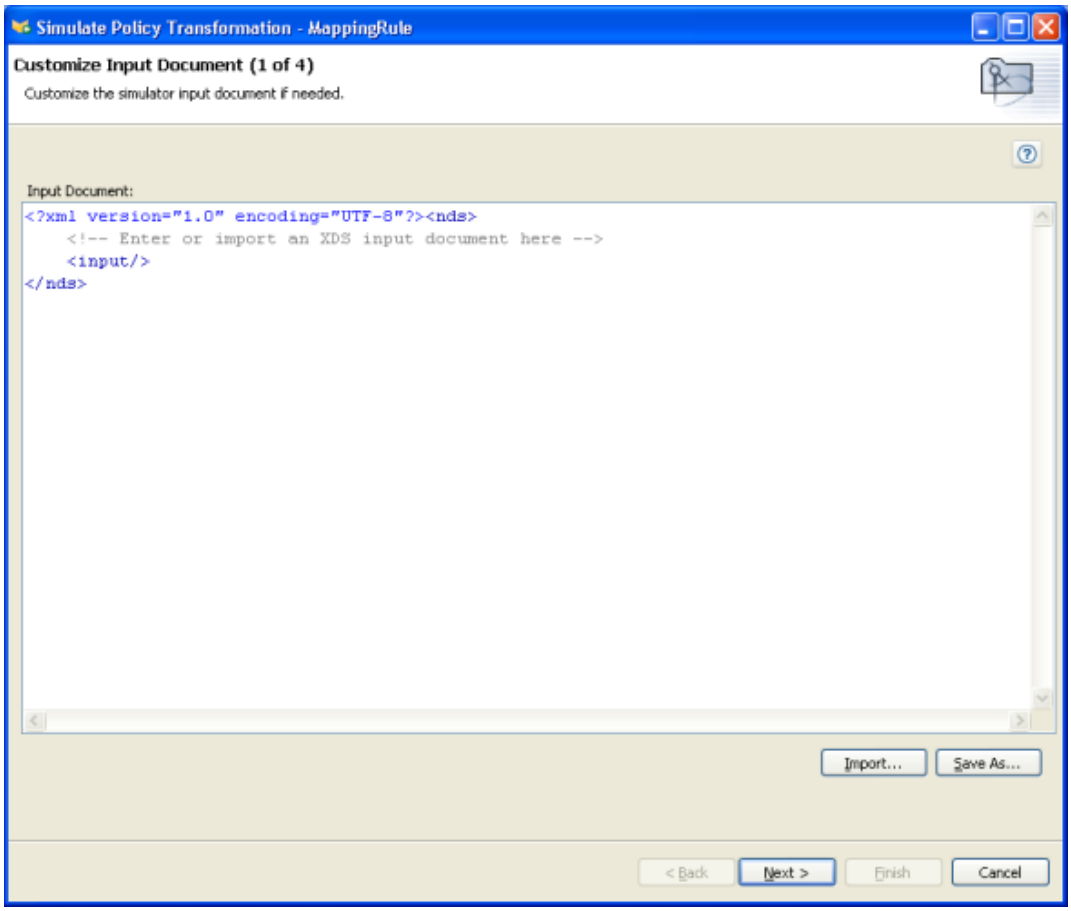
7.1.3 测试纲要映射策略

Designer 附带了一个称为 " 策略模拟器 " 的工具。它允许您在生产环境中不实施策略的情况下便可测试这些策略。修改完策略后，可以通过纲要映射编辑器起动策略模拟器以测试该策略。

要访问策略模拟器并测试纲要映射策略，请执行以下操作：

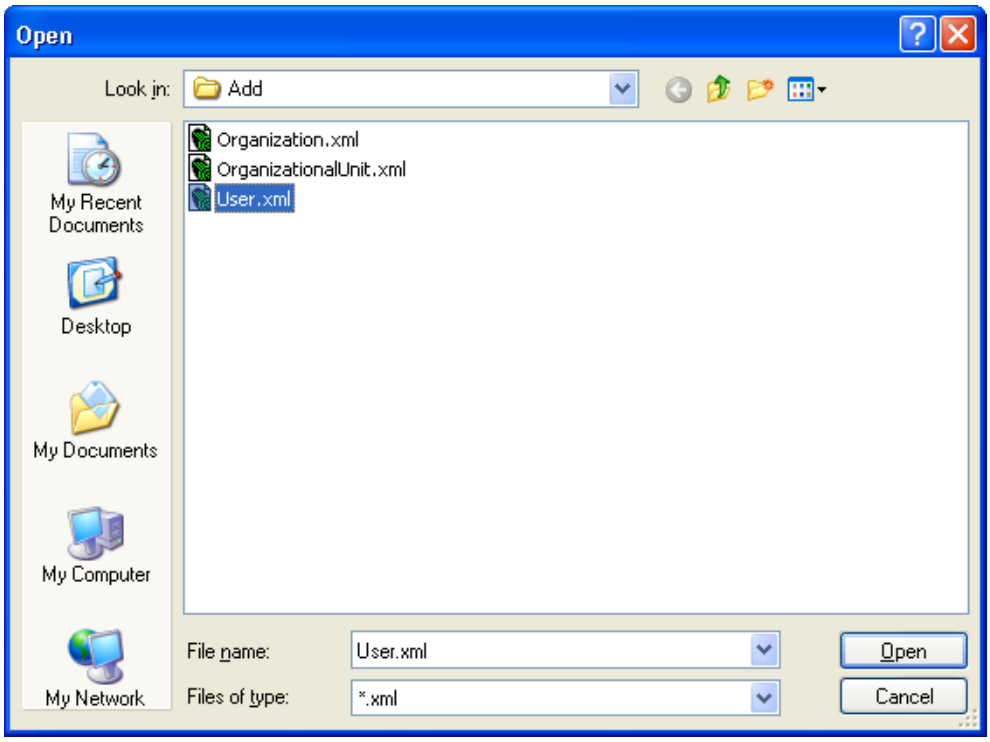
1 在工具栏中单击 " 起动策略模拟器 " 图标 。

2 选择 " 导入 " 以浏览至模拟事件的文件。



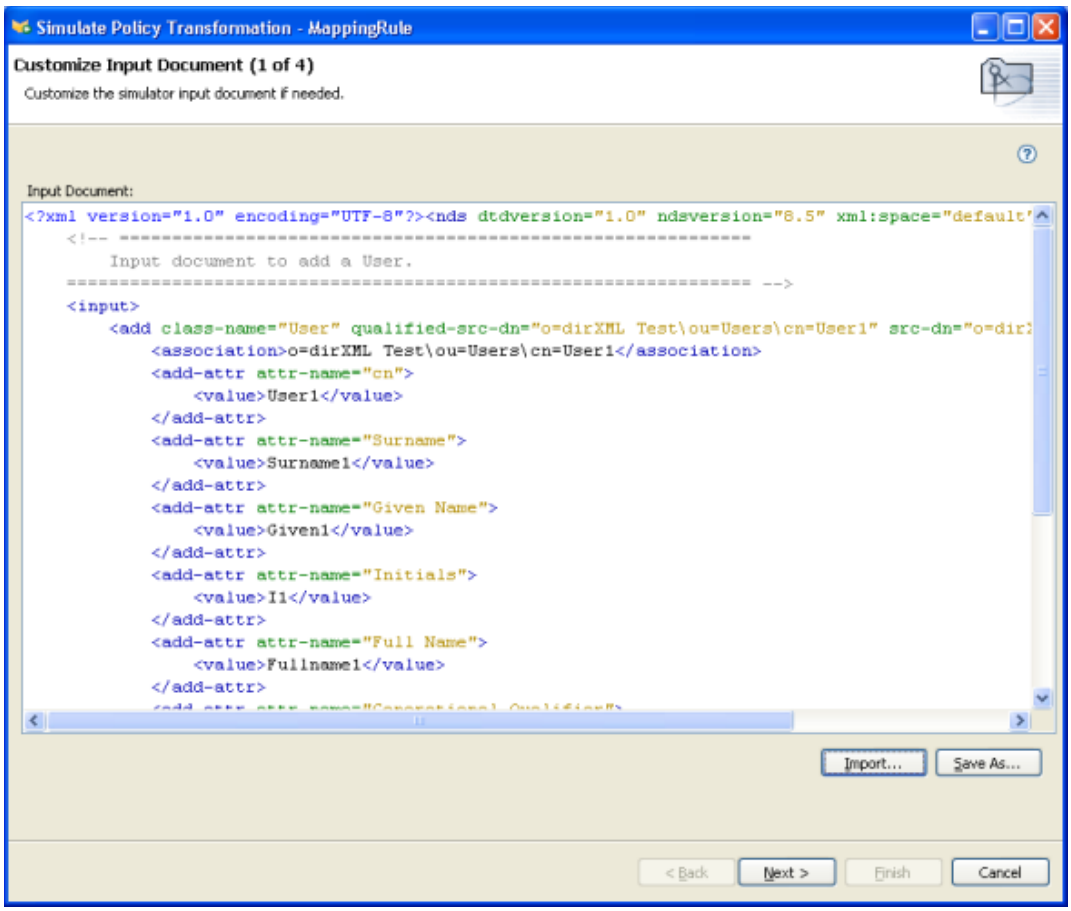
3 选择文件，然后单击 " 打开 "。

本示例使用的文件是 com.novell.designer.policy\simulation\add\user.xml，它模拟了用户对象的添加事件。

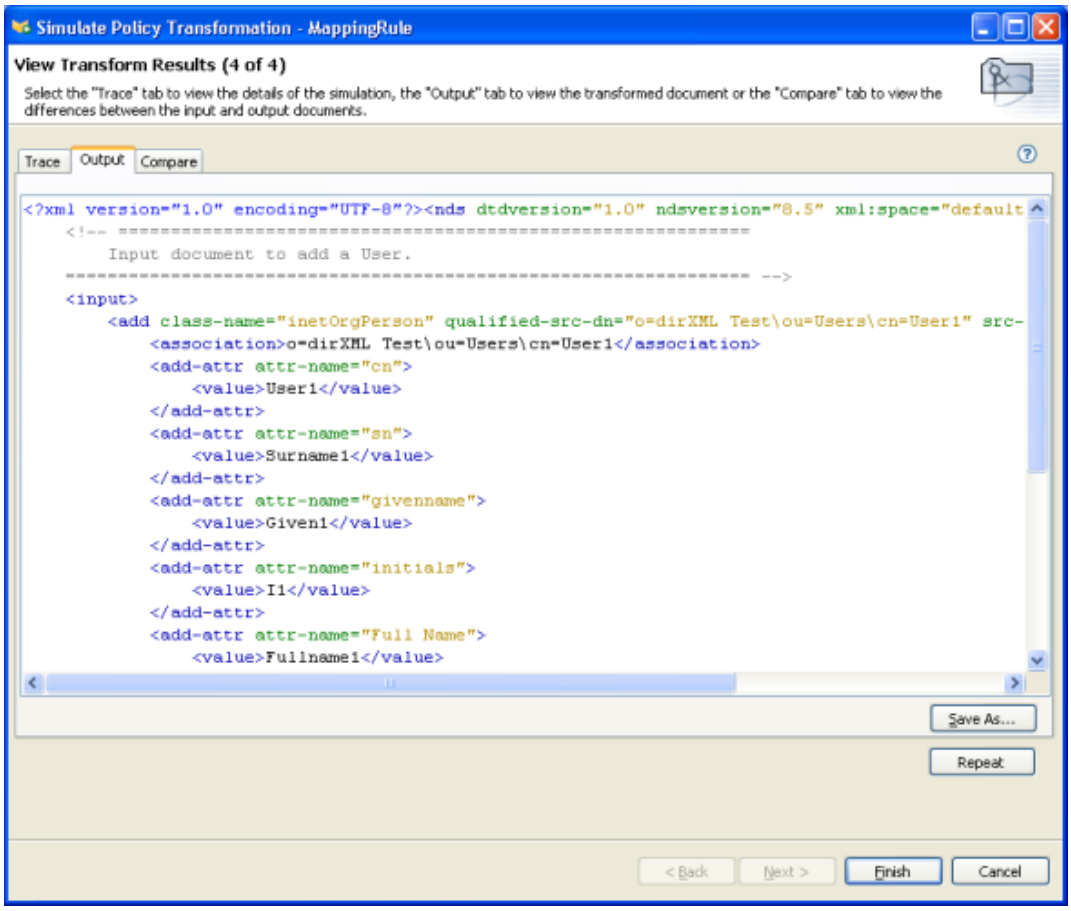


策略模拟器显示用户添加事件的输入文档。

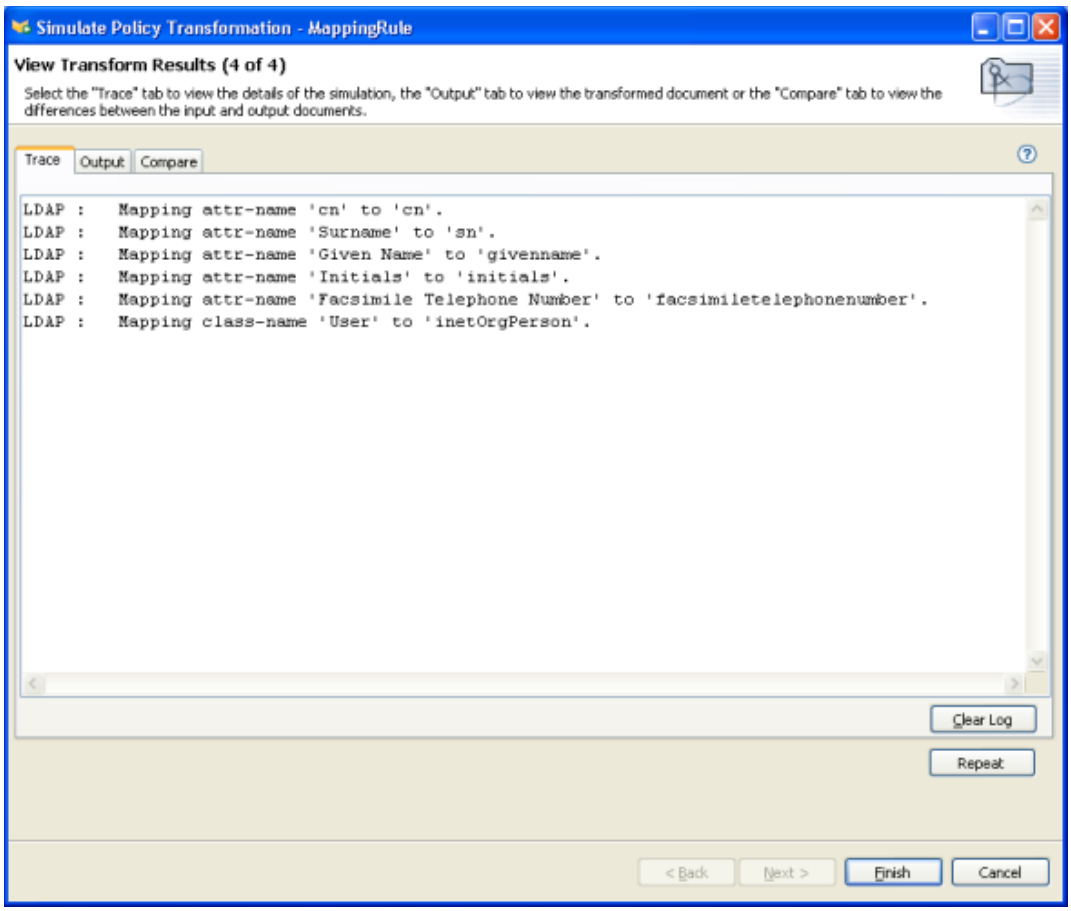
4 单击 " 下一步 " 开始进行模拟。



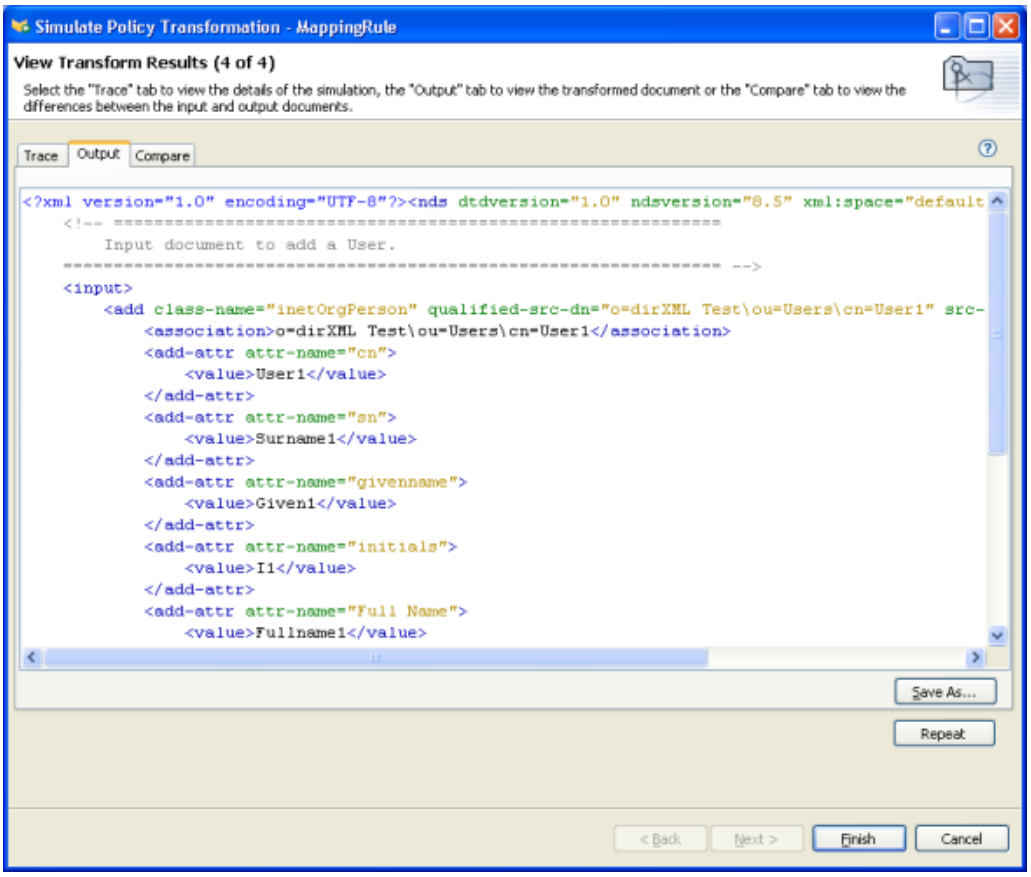
此策略模拟器显示添加事件的日志、输出文档，以及与生成的输出文档相比较的输入文档。



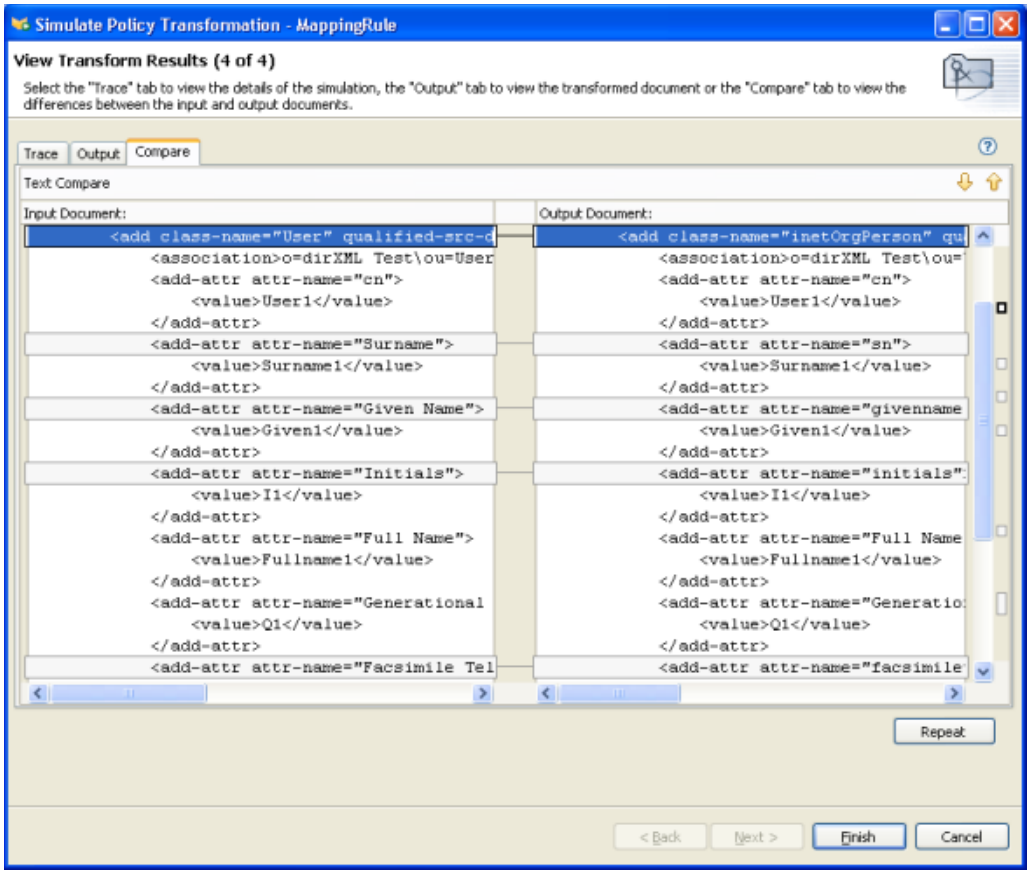
5 选择 " 跟踪 " 选项卡，通过 DSTRACE 查看添加事件的结果。



6 选择 " 输出 " 选项卡，查看对输入文档执行纲要映射策略所生成的输出文档。在此示例中，输出文档为用户添加事件。



7 选择 " 比较 " 选项卡，比较输入文档和生成的输出文档中的文本。



8 单击 " 重复 " 以选择其它输入文档并查看该事件的结果。

9 结束对纲要映射策略的测试后，请单击 " 完成 " 以关闭策略模拟器。

7.1.4 访问纲要映射策略 XML

在 Designer 中，您可以使用 XML 编辑器或文本编辑器来查看、编辑和验证 XML。

- “查看 XML 源”在第 398 页
- “编辑 XML 源”在第 402 页
- “验证 XML 源”在第 404 页

查看 **XML** 源

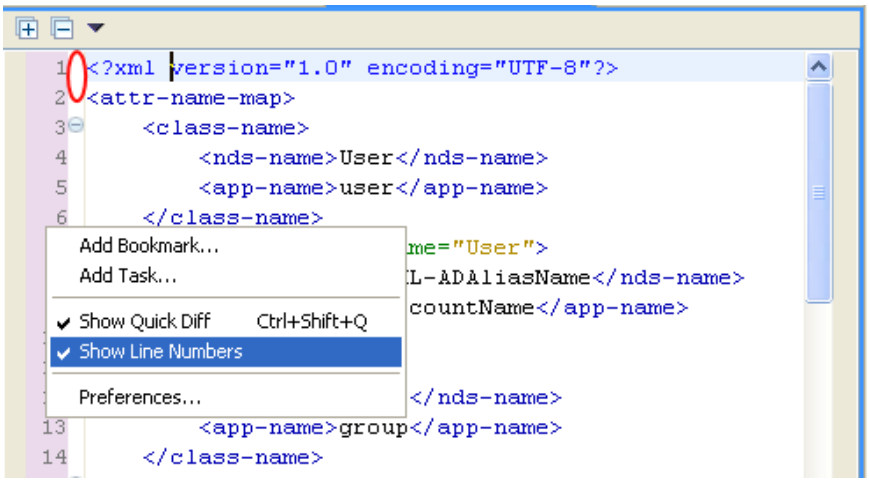
可以查看 XML 格式或 XML 树格式的 XML 源。

要打开 XML 源视图，请执行以下操作：

1 在纲要映射编辑器工作空间的底部，单击 "XML 源 " > 。

XML 编辑器可以显示行号。要查看行号，可以右击左边距，然后选择 " 显示行号 "。

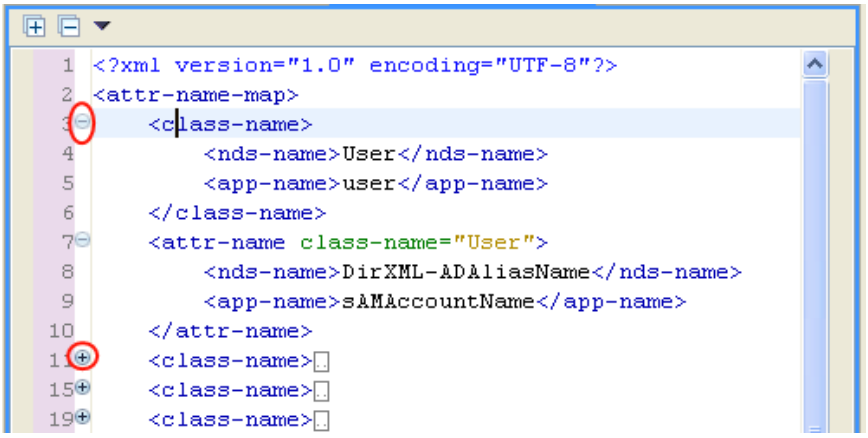
图 **7-4** 纲要映射策略的行号



XML 编辑器可以按函数展开或折叠 XML。如果函数中包含大量 XML，可以通过单击左上角的减号图标折叠 XML。要展开所有 XML 函数，请单击左上角的加号图标。

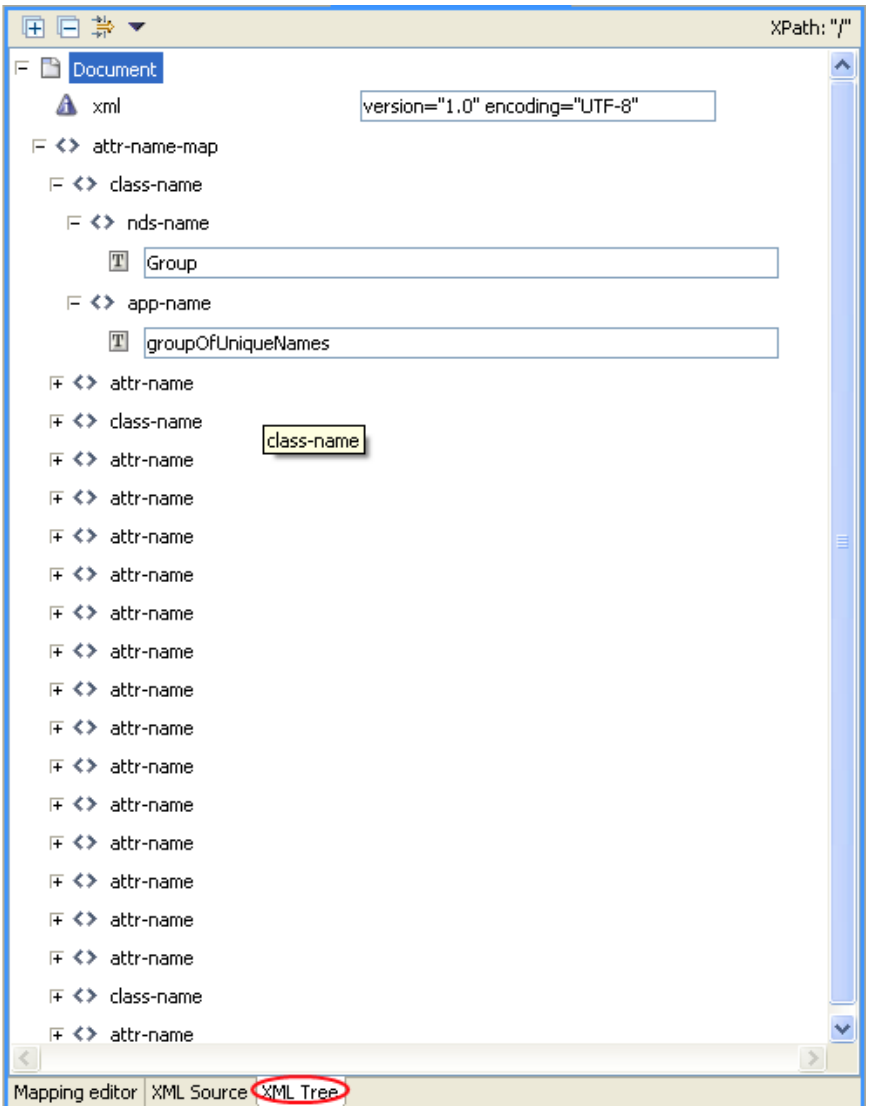
在左边距中，每个要素都有其各自的加号或减号图标。

图 **7-5** 纲要映射策略 *XML* 加号或减号



要以树格式查看 XML，请执行以下操作：

1 在纲要映射编辑器工作空间的底部，单击 "XML 树 "。

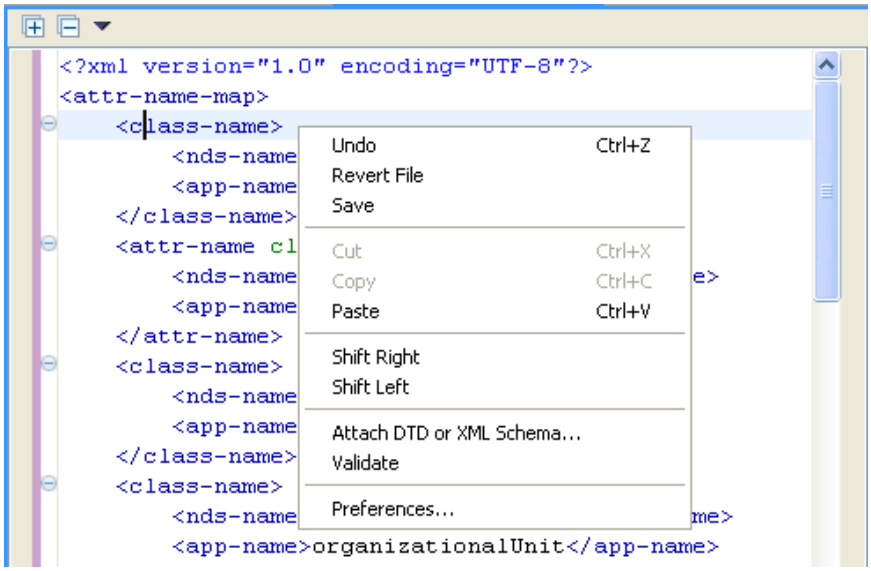


要查看整个树视图，请展开列出的每一项。

编辑 **XML** 源

可以通过 XML 编辑器来编辑 XML。可以在此处进行更改，也可以通过 GUI 界面来进行更改。

图 ***7-6*** 编辑纲要映射策略的 *XML* 源



加载的默认编辑器与 .xml 文件类型相关联。如果找不到默认编辑器，则装载系统文本编辑器。 XML 源视图的功能取决于所加载的编辑器。

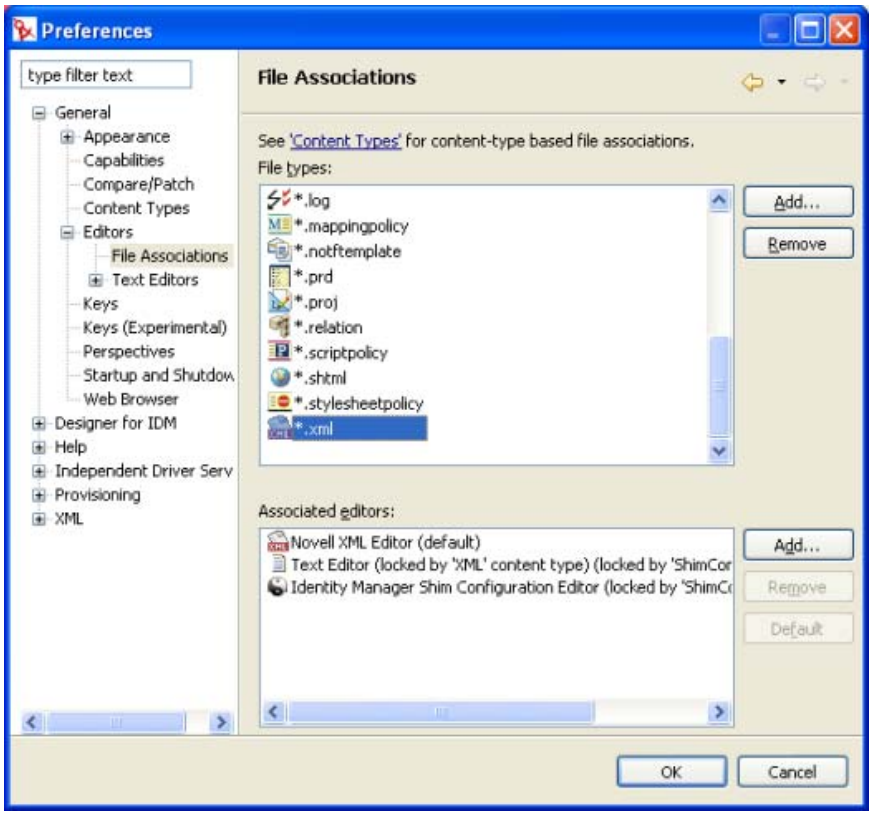
右击以显示 XML 编辑器所包含的功能列表。

- 复原：复原上一操作。
- 还原文件 将文件还原为所保存的上一版本。
- 保存：保存文件。
- 剪切：剪切所选信息。
- 粘贴：将信息粘贴到文档中。
- 右移：将本行向右缩排。
- 左移：将本行向左缩排。
- 挂接 **DTD** 或 **XML** 纲要：挂接 DTD 或 XML 纲要文件以验证策略。
- 验证：验证 XML 代码。
- 自选设置：设置 XML 编辑器的自选设置。

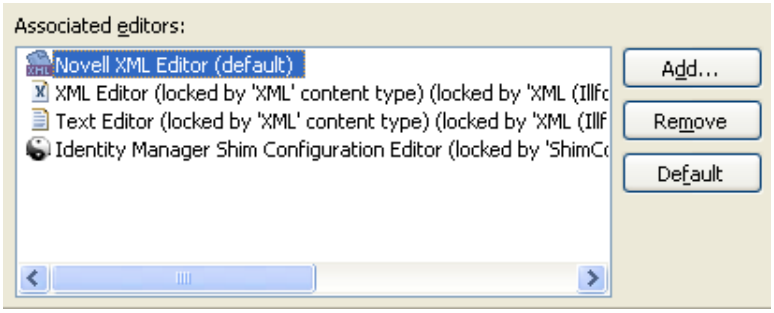
要为您的源视图选择其它 XML 编辑器，请执行以下操作：

1. 单击主菜单中的 " 窗口 ">" 自选设置 "。
2. 单击 " 常规 ">" 编辑器 ">" 文件关联 "。

3 从文件类型列表中选择 **xml*。



4 从 " 关联的编辑器 " 中，选择希望使用的编辑器 （例如， Novell XML 编辑器）。如果列表中没有您需要的编辑器，可以单击 " 添加 "，然后将其添加到列表中。



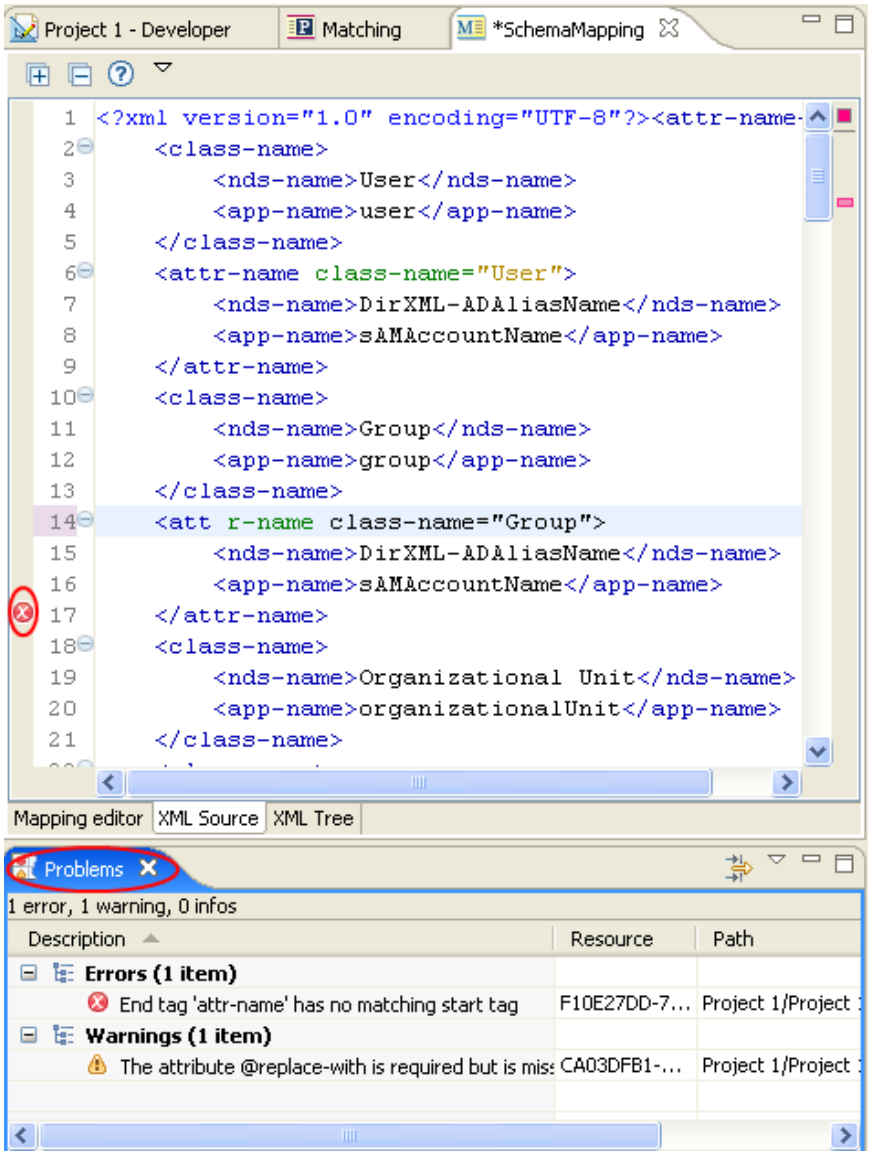
5 单击 " 确定 "。

6 关闭并重新打开纲要映射编辑器。将在 "XML 源 " 视图中加载默认编辑器。

验证 **XML** 源

XML 编辑器可以验证 XML 代码。右击并选择 " 验证 "。如果存在错误，则出现错误的行将显示一个红色的 x。窗口底部的解释将提供有关该问题的详细信息。

图 **7-7** 验证纲要映射策略



在本示例中， <attr-name> 的结束标签与其开始标签不匹配。

7.1.5 纲要映射策略的附加选项

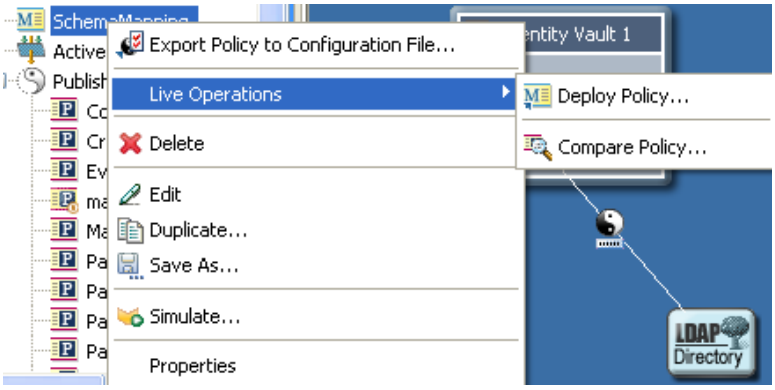
如果在 " 大纲 " 视图、 " 策略流程 " 视图或 " 策略集 " 视图中右击 " 纲要映射 " 策略，则将显示多个选项。

- “" 大纲 " 视图的附加选项”在第 405 页
- “" 策略流程 " 视图的附加选项”在第 405 页

- “" 策略集 " 视图的附加选项：” 在第 407 页

" 大纲 " 视图的附加选项

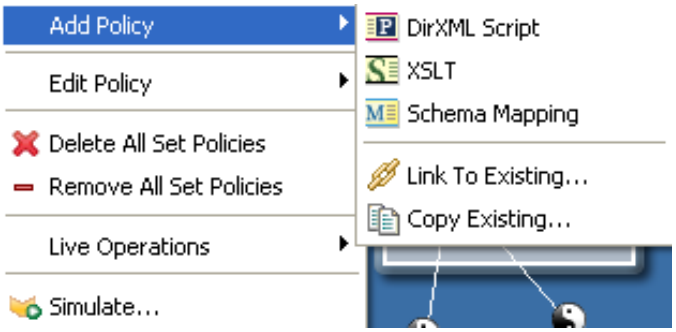
1 在 " 大纲 " 视图中，右击纲要映射策略。



- 将策略导出到配置文件中：将纲要映射策略另存为 .xml 文件。
- " 实时操作 "> **Deploy Policy** （部署策略）：将纲要映射策略部署到 Identity Vault。
- " 实时操作 "> **Compare Policy** （比较策略）：比较 Designer 和 Identity Vault 中的纲要映射策略。
- 删除：删除纲要映射策略。
- 编辑：起动纲要映射编辑器。有关详细信息，请参见 “编辑纲要映射策略” 在第 389 页 。
- 复制：创建纲要映射策略的副本。
- 另存为：将纲要映射策略另存为 .xml 文件。
- 模拟：测试纲要映射策略。有关详细信息，请参见 “测试纲要映射策略” 在第 392 页 。
- 属性：允许重命名纲要映射策略。

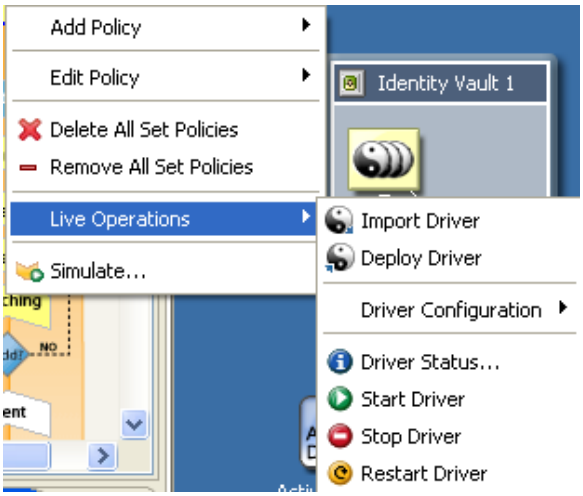
" 策略流程 " 视图的附加选项

1 在 " 策略流程 " 视图中，右击 " 纲要映射 " 策略。



- " 添加策略 "> **DirXML Script** （**DirXML** 底稿）：使用 DirXML® 底稿添加新的纲要映射策略。

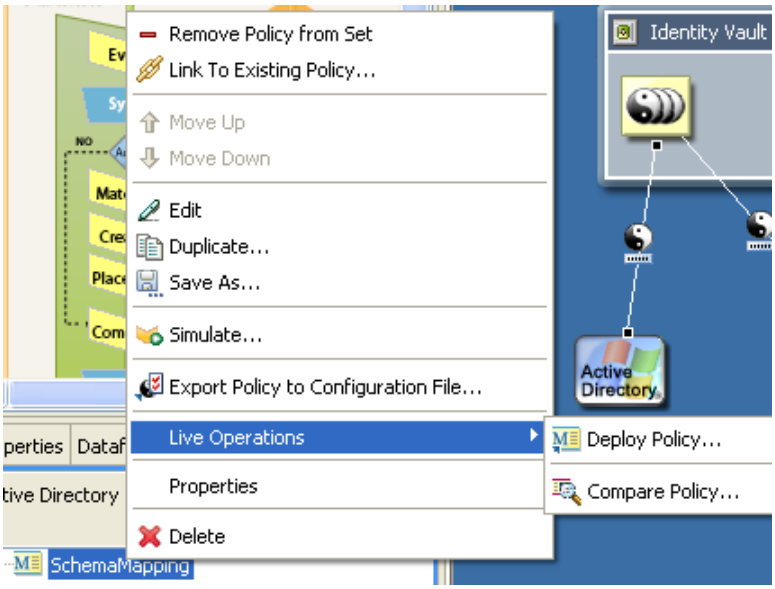
- " 添加策略 "> **XSLT**：使用 XSLT 添加新的纲要映射策略。
- " 添加策略 ">" 纲要映射 "：添加不包含任何信息的新纲要映射策略。
- " 添加策略 "> **Link to Existing** （链接到现有）：允许您浏览并选择一个现有纲要映射策略，然后将其链接到当前的纲要映射策略。
- " 添加策略 "> **Copy Existing** （复制现有）：允许您浏览至一个现有纲要映射策略并选择该策略，然后将其复制到当前的纲要映射策略。
- " 编辑策略 ">" 纲要映射 "：起动纲要映射编辑器。有关详细信息，请参见 “编辑纲要映射策略”在第 408 页。
- **Delete All Set Policies** （删除集中的所有策略）：删除选定策略集中的所有策略。
- **Remove All Set Policies** （去除集中的所有策略）：去除选定策略集中的所有策略，但不删除现有的策略。



- " 实时操作 ">" 导入驱动程序 "：从 Identity Vault 中导入现有驱动程序。
- " 实时操作 "> **Deploy Driver** （部署驱动程序）：将现有驱动程序部署到 Identity Vault。
- " 实时操作 ">" 驱动程序配置 "> **Import Attributes** （导入特性）：允许您从 Identity Vault 中导入特性，并将这些特性与 Designer 中的特性进行比较。
- " 实时操作 ">" 驱动程序配置 "> **Deploy Attributes** （部署特性）：允许您将 Designer 中的特性部署到 Identity Vault 中，并将这些特性与 Identity Vault 中的特性进行比较。
- " 实时操作 "> **Driver Status** （驱动程序状态）：显示驱动程序的状态。
- " 实时操作 ">" 启动驱动程序 "：启动驱动程序。
- " 实时操作 ">" 停止驱动程序 "：停止驱动程序。
- " 实时操作 "> **Restart Driver** （重启动驱动程序）：重启动驱动程序。
- 模拟：测试纲要映射策略。有关详细信息，请参见 “测试纲要映射策略”在第 392 页。

" 策略集 " 视图的附加选项：

1 在 " 策略集 " 视图中，右键单击纲要映射策略。



- **Remove Policy from Set** （去除集中的策略）：从策略集中去除纲要映射策略，但并不将其删除。
- **Link to Existing Policy** （链接到现有策略）：允许您浏览至其它纲要映射策略，并将其链接到现有的策略。
- 上移：按照策略的执行顺序，向上移动纲要映射策略。
- 下移：按照策略的执行顺序，向下移动纲要映射策略。
- 编辑：起动纲要映射编辑器。有关详细信息，请参见 “编辑纲要映射策略”在第 408 页 。
- 复制：创建纲要映射策略的副本。
- 另存为：将纲要映射策略另存为 .xml 文件。
- 模拟：测试纲要映射策略。有关详细信息，请参见 “测试纲要映射策略”在第 392 页 。
- 将策略导出到配置文件中：将纲要映射策略另存为 .xml 文件。
- " 实时操作 ">" 部署策略 "：将纲要映射策略部署到 Identity Vault。
- " 实时操作 ">" 比较策略 "：比较 Designer 和 Identity Vault 中的纲要映射策略。
- 属性：允许重命名纲要映射策略。
- 删除：删除纲要映射策略。

7.2 iManager 中的纲要映射策略任务

本节包含在 iManager 中执行与纲要映射策略有关的常见任务的说明：

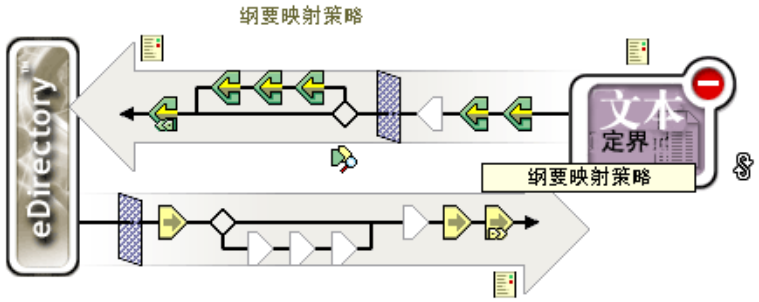
- “访问纲要映射策略”在第 408 页
- “编辑纲要映射策略”在第 408 页

7.2.1 访问纲要映射策略

1 在 iManager 中，展开 " 身份管理 " 职能，然后单击 "Identity Manager 概述 "。

2 为驱动程序集选择 " 搜索整个树 " 或 *Search in container*（在树枝中搜索），然后单击 " 搜索 "。

3 单击希望管理纲要映射策略的驱动程序。将打开 "Identity Manager 驱动程序概述 " 页。



4 单击 " 纲要映射策略 "。

5 单击 " 编辑 "。

7.2.2 编辑纲要映射策略

编辑纲要映射策略时分为两个部分。其一，编辑策略集中的策略布局。其二，使用纲要映射编辑器编辑策略本身。

策略布局

单击纲要映射策略后，将打开一个附带选项的窗口。

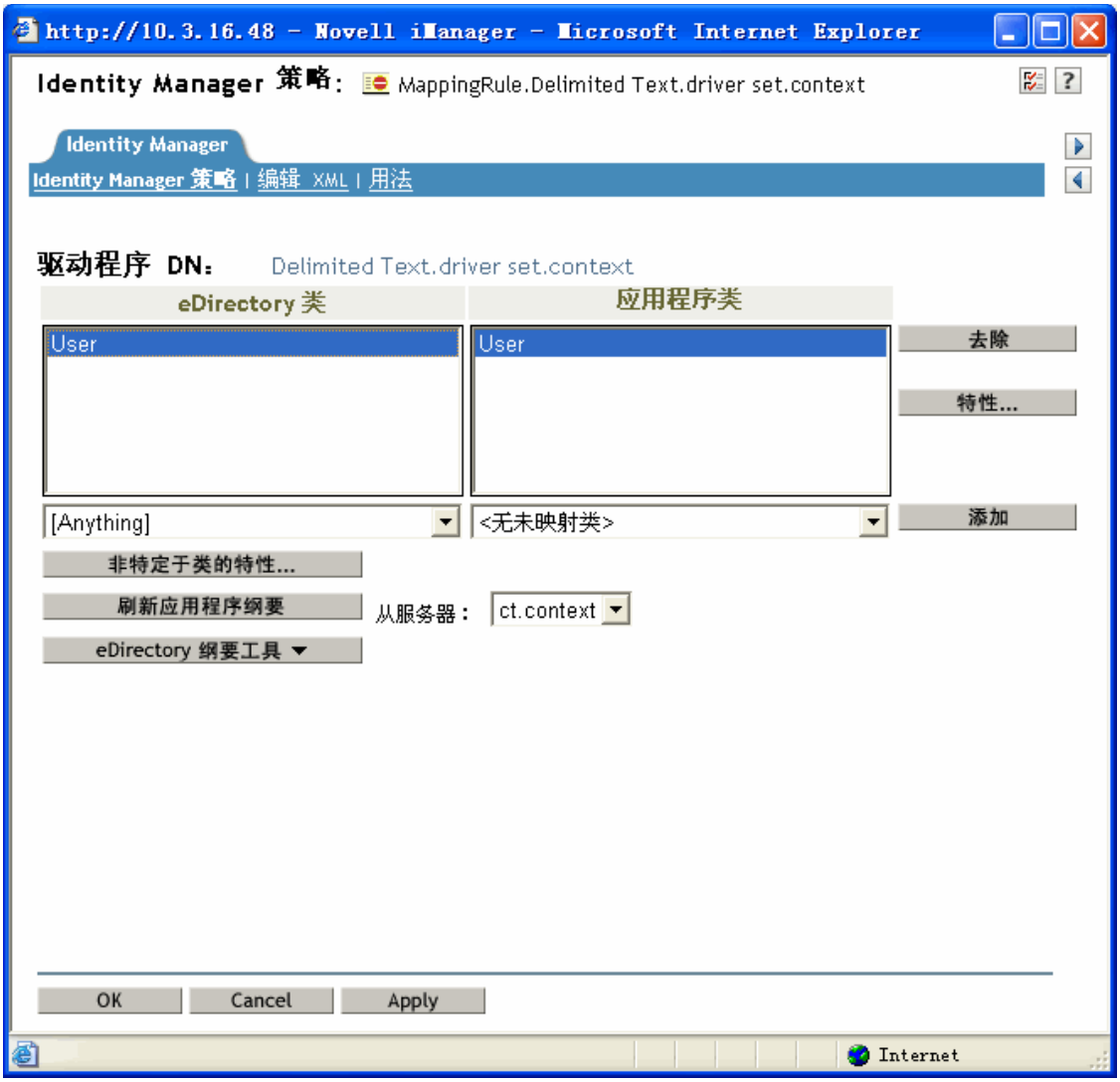
使用这些选项可以定位当前正在使用的策略。下表解释了其所有的选项。

选项	说明
上移策略	如果策略不止一个，则向上移动所选策略。
下移策略	如果策略不止一个，则向下移动所选策略。

选项	说明
插入	将新的或现有的策略插入到列出的策略中。
去除	去除选定的策略但不将其从策略集中删除。
编辑	起动纲要映射编辑器。
重命名	重命名选定的策略。
删除	删除选定的策略。

纲要映射编辑器

纲要映射编辑器是一个完整的图形界面，用于创建和管理纲要映射策略。纲要映射编辑器使用 XML 创建策略。



纲要映射编辑器包含以下三个选项卡：

- “Identity Manager 策略”在第 410 页

- “编辑 XML”在第 410 页
- “用法”在第 410 页

Identity Manager 策略

包含大部分信息，可以通过 GUI 界面在该选项卡中编辑策略。在纲要映射编辑器中，可以执行以下任务：

去除类和特性	选择希望去除的类或特性，然后单击 " 去除 "。
添加类	从下拉列表中选择 "eDirectory 类 "，然后再从下拉列表中选择 " 应用程序类 "。选择这两项后，单击 " 添加 "，然后单击 " 应用 " 保存更改。
添加特性	选择希望添加的特性的类，然后单击 " 特性 "。从下拉列表中选择 "eDirectory 类 "，然后再从下拉列表中选择 " 应用程序特性 "。选择这两项后，单击 " 添加 "，然后单击 " 确定 " 保存更改。
列出非特定于类的特性	如果存在不与类关联的特性，单击 " 非特定于类的特性 " 图标即可列出所有这些特性。
刷新应用程序纲要	如果应用程序的纲要已经更改，请单击 " 刷新应用程序纲要 " 图标。向导将联系已连接系统的服务器，检索新的纲要。更新纲要后，将在下拉列表中列出该纲要。
使用 eDirectory 纲要工具	• 添加特性 - 向选定的类中添加现有特性。 • Create Attribute （创建特性） - 创建新特性。 • Create Class （创建类） - 创建新类。 • 删除特性 - 删除选定的特性。 • 删除类 - 删除选定的类。 • 刷新 eDirectory 纲要 - 对 eDirectory 纲要进行更改后，如果单击 " 刷新 eDirectory 纲要 "，新的信息将更新到下拉列表中。

警告 **:** 请勿删除 Identity Vault 中正在使用的任何类或特性。这将导致对象成为未知对象。

编辑 XML

单击 *Enable XML editing* （启用 XML 编辑）可以编辑 DirXML 底稿策略。首先按照需要更改 DirXML 底稿，然后单击 " 应用 " 保存更改。

用法

将向您显示一个列表，其中包含当前正在参考此策略的驱动程序。此列表仅参考此策略驱动程序集中的策略。如果参考的策略属于其它驱动程序集，则这些策略将不会出现在此列表中。