

规范化规则处理

要使用规范化规则，您必须具有以下角色：

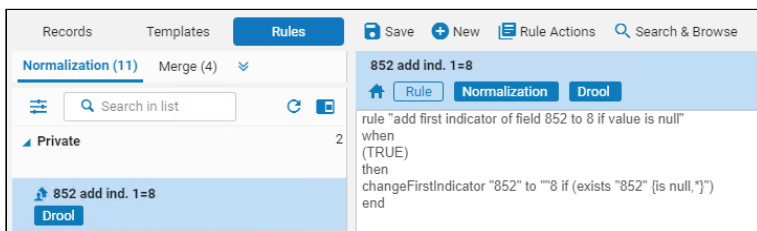
- 编目管理员
- 编目经理
- 编目员

规范化规则在多个阶段用于更改或更新书目元数据，例如，当在元数据编辑器中保存记录，使用导入配置文件导入，从外部检索资源导入，或在元数据编辑器中使用“完善记录”菜单编辑时。规范化规则按照特定语法写成，允许您创建对记录做出特定修改的规则。例如，您可以编辑或删除字段，也可以根据在书目记录中发生的条件指定操作。

规范化规则可在元数据编辑器中应用到单个记录或一组记录：

- 要应用规范化规则到单个记录，使用完善记录选项（见[元数据编辑器菜单和工具栏选项](#)部分）或在预览规范化规则时应用更改到单个记录（见[要预览规则的结果](#)）。
- 要对一组记录应用规范化规则，您需要使用MarcDroolNormalization、MarcXSLNormalization或DcDroolNormalization任务（见[使用规范化进程](#)）创建进程，并指定使用元数据编辑器创建的规范化规则（见[创建新规范化规则文件](#)）。创建该进程后，您可以使用该进程运行作业（见[在既定集合上运行手动作业](#)）。此外，您可以创建在元数据编辑器中选择保存时启用应用规范化规则的进程（见[使用规范化进程](#)）。

规范化规则是通过遵循特定的编程语法和使用元数据编辑器中规则选项卡下提供的编辑窗口创建的。



规则选项卡编辑窗口

除了创建您自己的规范化规则程序之外，还可以将现有规则复制/粘贴到编辑窗口中，或者使用出厂示例（规则选项卡）来开发您的规范化规则。有关规范化规则的语法和可以复制到编辑窗口的示例的更多信息，见[规范化规则语法](#)。

使用元数据编辑器中的预览功能，您可以：

- 并行查看规范化规则和元数据记录
- 在元数据记录运行时预览规则结果
- 在规则和预览更改之间切换
- 编辑规则并立即测试
- 有关规范化规则的简介，观看[规范化规则](#)（6:30）。
- 有关规范化规则示例的详情，观看[规范化规则](#)（1小时）。

创建规范化规则

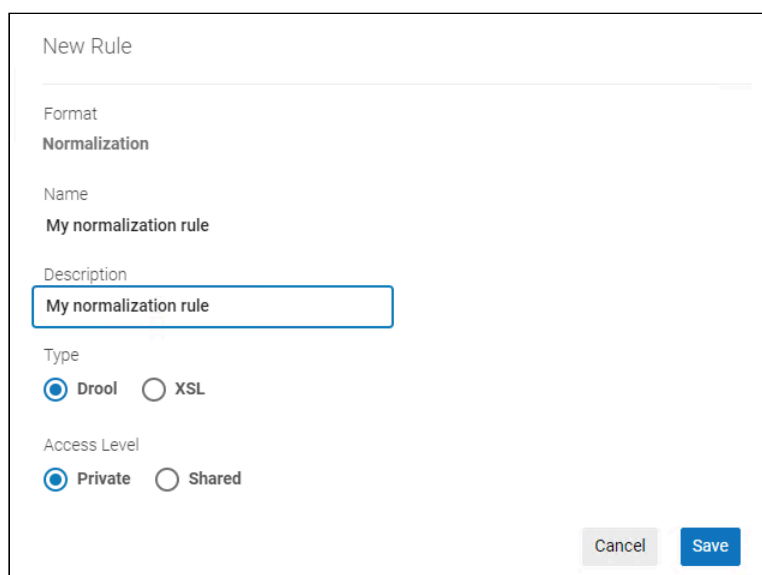
本节介绍如何创建规范化规则，使用先前创建的规则，以及在单个元数据记录运行时预览规则的结果。有关更多信息，见[该页面](#)的规范化规则区域。

请注意，链接到网络区的机构区记录的规范化规则只能应用于本地字段。要对此类记录应用规范化规则，仅在网络区机构中运行规范化规则和规范化进程。

对于网络区联盟成员，用户可以控制在本地或在网络机构中保存新规则。要完成该选项，打开记录或模板区域，导航至**新建 > 提交**选项。请注意，该选项根据用户生效。

要创建新规范化规则：

1. 在元数据编辑器页面（**资源 > 编目 > 打开元数据编辑器**）打开规则区域。
2. 要对MARC记录或都柏林核心记录创建规范化规则，选择**新建 > 规范化**。规范化规则属性（新规则）对话框打开。



规范化规则属性对话框

3. 输入规范化规则名称和描述。

Note

- 如果规则名称中使用单词**rule**，则必须不能大写。如果采用大写，则规则不起作用。
- 不要在规则名称中使用反斜杠（\）！否则，规则将无法用于过滤结果集功能。
- 规则名称/描述的字符限制为255个字符。

4. 要对MARC记录创建规范化规则，选择**Drool**。要对都柏林核心元数据记录创建规范化记录，选择**XSL**。

Note

MARC记录规范化规则也支持XSL。

5. 选择访问选项，**专用** 或 **共享**。如果选择**专用**，只有您可以处理该规则，并且规则不能包含在规范化进程中。如果选择

共享，您的规则在编目员之间共享。在这种情况下，多个用户可以同时查看规则，如果两个或更多的人打开规则进行编辑，当您尝试保存更改时，会显示一条警告消息。（您可以选择保留更改或允许其他用户进行更改并保存更改。）

6. 选择保存。元数据编辑器的编辑窗格打开。
您可以包括现有的规则语法（编辑 > 添加规则 > {type of rule}）或定义一个规则（详情参见[规范化规则语法](#)）。
7. 选择保存。规则将添加到规范化规则选项卡下的规则文件列表中。

要在共享区复制/提交规范化规则：

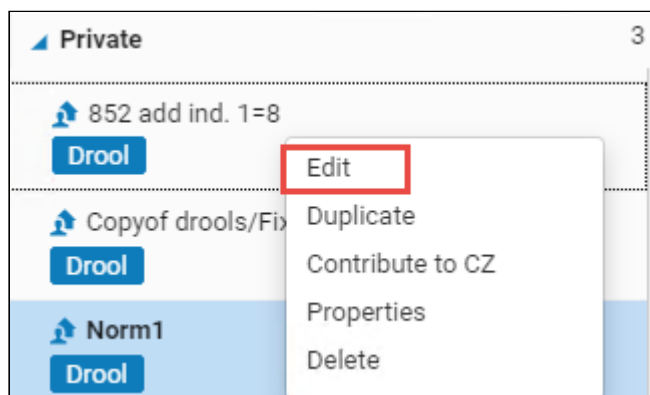
1. 在元数据编辑器页面（资源 > 编目 > 打开元数据编辑器），选择规则选项卡并展开规范化文件夹。
2. 要复制，请打开社区文件夹，右键单击要复制的规则并选择复制。复制规则对话框打开。输入名称和描述并指示是否保存为专有（仅对您可用）或共享规则（对您的机构中的所有用户可用）。
该对话框显示向共享区提交规则的用户的Alma用户名和联系电子邮件。如果您有任何疑问，可以联系此用户。
3. 要将您自己的规则提交到共享区，请右键单击规则并选择提交到共享区。
规则共享对话框打开。提供描述性名称和描述并选择保存在共享区中保存规则。

要使用现有的规范化规则：

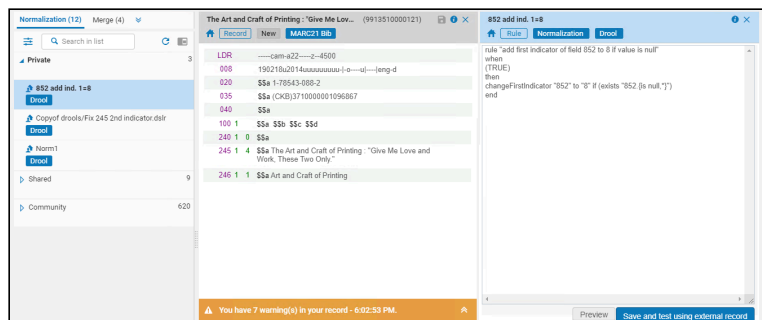
1. 在元数据编辑器页面（资源 > 编目 > 打开元数据编辑器），选择规则选项卡并展开规范化文件夹显示保存的规则。
2. 选择要使用的规则，然后选择下列选项：
 - 编辑 – 打开规则语法文本框，修改语法（详细信息见[规范化规则语法](#)）。
 - 删除 – 点击是确认删除规则文件。
 - 复制 – 复制所选的规则文件，修改并将其另存为新规则，而不影响原始文件。
 - 属性 – 打开规范化规则属性对话框，修改规则文件的属性。
3. 要重命名规则，复制规则，给复制的规则命名，然后删除旧规则。

要预览并保存和测试规则结果：

1. 找到要处理的书目记录（使用[仓储检索](#)或元数据编辑器 > 记录选项卡），并在元数据编辑器中打开它。
2. 按F6或选择拆分编辑器图标。
3. 在左面板选择规则选项卡并展开规范化文件夹。
4. 在专用或共享文件夹中（不是共享区）右击想要预览或测试的规则并选择编辑。



该规则显示在元数据编辑器的右窗格中。



5. 选择预览。规则将应用于记录，并显示结果。

Note

要更新网络区记录，规范化必须由网络区机构运行（不是成员）。机构无法更新网络区记录；因此，规范化仅应用于本地字段（成员的本地扩展）。预览网络区记录规则时，显示以下消息：请注意，在规范化进程运行时规范仅应用于本地字段。

- 点击应用更改保存对记录的修改或点击返回到规范化规则以继续编辑。
- 当您对规范化规则进行最后更改时，选择保存并使用外部记录测试以保存最终版本的规范化规则。有关详情，见[测试外部数据源的规范化规则](#)。

都柏林核心和MODS记录的规范化规则

在元数据编辑器中，有以下类型的规则与都柏林核心和MODS相关：

- XSL指示规则
- XSL规范化规则

Note

XSL规范化规则不支持discovery:local。

无法按照常规的规范化规则编写都柏林核心和MODS规范化规则，只能写为XSL。这意味您需要在元数据编辑器（如上文所述）中直接写XSL，或者可以在记事本（或其他外部应用）中写好然后复制到元数据编辑器。

您可以在元数据编辑器中查看共享区中都柏林核心的以下XSL规范化规则示例：

- EXL – 更改dc:language的值en为English
- EXL – 替换都柏林核心记录中的字符串的通用规则
- EXL_Add_field_accessRights

要写都柏林核心和MODS的XSL：

1. 在外部应用中写好XSL。
2. 打开元数据编辑器 > 规则部分。
3. 复制并粘贴XSL到元数据编辑器然后保存规则。

Primo VE和Esploro的规范化规则

除了Alma规范化规则，您可以对Primo VE和Esploro创建规范化规则。要创建这些规则，请选择**新建**，然后选择：

- **规范化（发现）** - 该选项仅在Primo VE在系统中定义时显示。如果对加载到Primo VE且不在Alma中管理的书目记录创建DC或XML规范化规则时选定。有关这些规则的语法，见下文的[DC和XML格式的规范化规则语法](#)。
- **规范化（研究）** - 该选项仅在Esploro在系统中定义时显示。有关详情，见[管理资料规范化规则（Esploro）](#)。

规范化规则语法

规则可以包含一个或多个其他规则。每个规则含有一个条件，且一个或多个应用于记录的操作。如果记录满足条件，操作将应用于记录。每个操作可以对记录中的单个字段执行。按照规则中出现的顺序执行操作。

定义含有多个规则的规则时，必须使用优先级因素。优先级更高的操作首先执行。例如，规则中优先级为2的操作先于优先级为1的操作。有关规范化规则中可用的优先级示例，见[Alma规范化规则示例](#)。请注意，仅在规则的字段一样时需要优先级。否则，当每个规则用于不同字段时，会从高到低执行多个规则。

在[规范化规则](#)视频（5分42秒）中了解有关创建规范化规则的更多信息。

在[删除或内容更改的规范化例程语法](#)视频（9:57分钟）中了解如何创建从记录中删除指定字段的规范化规则，或者更改这些字段的内容。

```
When
(<conditions on MARC record>) then
Action
End
```

<conditions on MARC record> 包含一个或多个适用于该记录的布尔子句。如果<conditions on MARC record> 返回 TRUE，则规则应用于记录；否则，不应用规则，并且不处理该记录。

Note

- When 必须是第一行中唯一的字。条件必须放在单独的一行上。
- 您仅可以在“When”条件中对整个记录使用多个条件。
- 尽管在规则中允许使用多个布尔运算符，但当选择大量布尔运算符时会降低运行速度。因此每条规则应包含不超过200个布尔运算符。
- 如果未指定，则条件将在记录级别起作用。如果您希望条件分别作用于每个MARC21字段，则应为每个字段指定条件。例如，当有多个带相同标签的MARC21字段时。

理解规范化规则中的转义字符

编写规范化规则时，您可能需要包含特殊字符，例如引号、制表符或反斜杠。要正确执行此操作，您需要使用转义字符，这些特殊组合可以告诉系统如何解释这些符号。

有效的转义序列

Alma可以识别以下转义字符：

- \b - 退格
- \t - 选项卡
- \n - 换行
- \f - 换页
- \r - 回车
- \" - 双引号
- \' - 单引号
- \\ - 反斜杠

这些序列必须以单个反斜杠开头，后跟一个有效字符。

避免常见错误

使用错误数量的反斜杠会导致错误。例如：

- \z或\\z无效，会触发错误。
- 没有有效转义字符的单独反斜杠(\)也是无效的。

如何修复错误

如果您看到类似这样的错误：

无效的转义序列（有效的转义序列为\b \t \n \f \r ""）"

您可以通过调整反斜杠来解决此问题。以下是一些示例：

- replacing \\\ with \\
- replacing \. with .
- 将\ (替换为 (, 并将\) 替换为)
- replacing \\| with \|

这些更改既保留规则的预期行为，又避免语法错误。

记录元素

条件和操作适用于记录元素，如MARC记录、字段（一个或多个）、指示符、子字段（一个或多个）和字段/子字段内容。

要测试条件或将操作应用于记录元素，该元素必须符合以下语法：

句法

表达	意义
"<tag>", "<new tag>"	表示一个字段标签，例如001、245等。

表达	意义
"<oldCode>", "<newCode>"	表示子字段代码，例如 a, b, c。
"<element>" 用于数据 字段	以下是数据字段的可能值： <ul style="list-style-type: none"> • FIELD – 例如：245 • FIELD_VALUE – 例如：245.value* • FIELD_INDICATOR – 例如：245.{1,2} • FIELD_SUBFIELD_CODE – 例如：245.a • FIELD_INDICATOR_SUBFIELD_CODE – 例如：245.{1,2}.a • FIELD_SUBFIELD_CODE_VALUE – 例如：245.a.value* • FIELD_INDICATOR_SUBFIELD_CODE_VALUE – 例如：245.{1,2}.a.value*
"<element>" 用于控制 字段	以下是控制字段的可能值： <ul style="list-style-type: none"> • FIELD_POSITION_LENGTH – 例如：LDR.{17,3} • FIELD_POSITION_LENGTH_VALUE – 例如：LDR.{17,3}.eng
固定位置字段 的"<element>"	以下为固定位置的UNIMARC/CNMARC 1XX字段的可能值： <ul style="list-style-type: none"> • FIELD – 例如：100 • FIELD_VALUE – 例如：100.{*,*}.{*,*}.value* • FIELD_INDICATOR – 例如：100.{1,2} • FIELD_SUBFIELD_CODE – 例如：100.a • FIELD_INDICATOR_SUBFIELD_CODE – 例如：100.{1,2}.a • FIELD_SUBFIELD_CODE_VALUE – 例如：100.a.value* • FIELD_INDICATOR_SUBFIELD_CODE_VALUE – 例如：100.{1,2}.a.{*,*}.value* • FIELD_POSITION_LENGTH – 例如：100.{*,*}.a.{17,3} • FIELD_POSITION_LENGTH_VALUE – 例如：100.{*,*}.a.{17,3}.* 仅与UNIMARC/CNMARC 1XX字段相关。
记录级别的CONDITION	以下是可能的条件选项。见下个部分的重要信息 (条件)。 <ul style="list-style-type: none"> • TRUE – 始终为true • not exists "{element}" – 如果元素不存在，则为true (数据字段) • not existsControl "{element}" – 如果元素不存在，则为true (控制字段) • not exists "{element}" – 如果元素至少存在一次，则为true (数据字段) • not existsControl "{element}" – 如果元素至少存在一次，则为true (控制字段) • existsMoreThanOnce "{element}" – 如果元素存在多次，则为true有关使用该条件的示例，见MARC记录的指示规则 - 语法示例。 • contains – 如果元素还有值则为true由合并规则使用。 以下为仅与UNIMARC/CNMARC 1XX字段相关的可能条件选项：

表达	意义
	<ul style="list-style-type: none"> • existsFixedField "{element}" - 如果 (1XX固定位置字段的数据) 不存在元素, 则为true • not existsFixedField "{element}" - 如果 (1XX固定位置字段的数据) 不存在元素, 则为true

条件

条件可以在整个规则级别 (WHEN) 或特定操作级别 (IF)。相同的条件将根据定义的级别而有所不同。

- WHEN子句 - 整个记录必须满足的条件, 以确定规则是否应用于记录
- IF (一个操作) - 适用于单个字段的条件, 以确定是否对该字段执行特定操作

条件为:

- containsScript - 使用该条件检测特定语言。containsScript条件使用如下可检查的语言列表: Arabic, Armenian, Bengali, Bopomofo, Braille, Buhid, Canadian_Aboriginal, Cherokee, Cyrillic, Devanagari, Ethiopic, Georgian, Greek, Gujarati, Gurmukhi, Han, Hangul, Hanunoo, Hebrew, Hiragana, Inherited, Kannada, Katakana, Khmer, Lao, Latin, Limbu, Malayalam, Mongolian, Myanmar, Ogham, Oriya, Runic, Sinhala, Syriac, Tagalog, Tagbanwa, TaiLe, Tamil, Telugu, Thaana, Thai, Tibetan 和 Yi。见以下语法示例:

```
rule "Is CJK in Authority"
when
  containsScript "Han" "1**"
then
  set indication."true"
end
```

- exists <element> - 找到至少一个匹配
 - exists <element> - 应用于数据字段在IF子句中使用, 操作元素和由条件测试的元素必须在同一 (数据) 字段。
 - existsControl <element> - 应用于控制字段在IF子句中使用, 操作元素和由条件测试的元素必须在同一 (控制) 字段。
- existsMoreThanOnce "{element}" - 找到多个匹配项应用于数据字段。在IF子句中使用, 操作元素和由条件测试的元素必须在同一 (数据) 字段。
- not exists <element> - 无匹配项
 - not exists <element> - 应用于数据字段在IF子句中使用, 操作元素和由条件测试的元素必须在同一 (数据) 字段。
 - not existsControl <element> - 应用于控制字段在IF子句中使用, 操作元素和由条件测试的元素必须在同一 (控制) 字段。
- recordHasDuplicateSubfields (有关指示规则, 见[使用指示规则](#)) - 如果根据字段、子字段找到当前记录的重复子字段 (子字段和内容), 返回true, 且按以下格式处理忽略字符 (charsToIgnore) 字符串:

```
recordHasDuplicateSubfields "<tag>" "<code>" "<charsToIgnore>"
```

可以指定由逗号分隔的多个标签 (字段)。可指定多个代码 (子字段), 并不需要空格分隔。可以指定没有空格分隔的

一个或多个字符在用于评估重复的子字段的内容的结果为忽略字符。有关更多信息，见[示例6](#)。

对于满足recordHasDuplicateSubfields条件（返回true）的记录，创建一组记录。

每个IF子句操作可以具有以下格式之一：

- 如果某个特定条件不正确，则应用，例如：`addControlField "{element}" if(not exists "{condition}")`
- 如果某个特定条件为真，则应用，例如：`addControlField "{element}" if(exists "{condition}")`
- 无条件应用，例如：`addControlField "{element}"`

后续声明中使用的逻辑或操作符为 (|) 符号，例如：`removeField "866" if (not exists "866.8.0|99.如果 (|) 符号是值的一部分，使用四个反斜杠 (\\) 避开，例如：removeField "866" if (exists "866.8.0\\\\|99")`

Note

如果条件中的值出现在记录中的某个位置，则满足if (exists...)条件。无法使用此规则复制/删除具有特定值的子字段。要删除或复制具有特定值的子字段，请使用XSL转换规则。

Note

这是一个已知问题，其中"|"无法转义。这将在以后的版本中修复。

操作列表

下表提供了可用操作的列表。

操作列表

操作	格式/示例	评论
将字段和子字段替换为其它字段和子字段。	<code>changeControlField "<tag>" to "<new tag>"</code> 例如： <code>changeControlField "007" to "008"</code>	更改控制字段的标记标识符；不修改内容。
	<code>changeField "<tag>" to "<new tag>"</code> 例如： <code>changeField "245" to "246"</code>	改变标签标识符；不修改指示符或子字段。
	<code>changeSubField "<tag>.<code>" to "<new code>"</code> <code>changeSubFieldOnlyFirst "<tag>.<code>" to "<new code>"</code> <code>changeSubFieldExceptFirst "<tag>.<code>" to "<new code>"</code> 例如： <code>changeSubField "035.b" to "a"</code>	在"<tag>"字段中更改子字段（或仅第一个子字段，或除第一个子字段之外的所有子字段）"<code>"为子字段"new code"。
	<code>changeFirstIndicator "<tag>" to "<value>"</code>	在<tag>中设置指定的指示符值。

操作	格式/示例	评论
	<p>changeSecondIndicator "<tag>" to "<value>"</p> <p>例如：changeFirstIndicator "245" to "3"</p>	
	<p>combineFields "<tag>" excluding "<comma-separated subfield list>"</p> <p>例如：combineFields "852" excluding "a,b"</p>	合并指定数字的所有字段。将所有子字段从第二行和后续行复制到第一行，不包括命名子字段；只有第一次排除的子字段被复制，并且只有在第一行不存在的情况下。
添加字段和子字段	<p>addField "<tag>.<code>.<value>"</p> <p>addField "<tag>.{<ind1>,<ind2>}.<code>.<value>"</p> <p>例如：addField "999.a.RESTRICTED"</p>	将字段添加到MARC记录。将子字段的值设置为指定值。
	<p>addControlField "<tag>.<value>"</p> <p>例如：addControlField "008.820305s1991#####nyu#####001#0#eng##"</p>	将控制字段添加到MARC记录。
	<p>addSubField "<tag>.<code>.<value>"</p> <p>addSubField "<tag>.{<ind1>,<ind2>}.<code>.<value>"</p> <p>例如：addSubField "245.h.[Journal]"</p>	添加子字段 <code> with value <value> to field <tag>.如果该字段不存在，则不添加。
	<p>addSystemNumber "<element>" from "<tag>" prefixed by "<prefix tag>"</p> <p>例如：addSystemNumber "035.a" from "001" prefixed by "003"</p>	使数据字段<element>等于括号中的第二个控制字段<prefix tag>的内容，其后为第一个控制字段<tag>的内容。 例如，如果001的值为9945110100121，003的值为DAV，左侧的示例条件将产生035，值为‡ (DAV) 9945110100121。
复制字段	<p>copyField "<tag>" to "<new tag>"</p> <p>copyField "<tag>.<code>" to "<new tag>.<new code>"</p> <p>copyField "<tag>" to "<new tag>.{<ind1>,<ind2>}"</p> <p>例如：copyField "971.a" to "100.u"</p>	<p>将字段复制到另一个字段。在第一版中，未指定子字段 (<code>和<new code>)，新字段含有所有旧字段中相同的子字段。在第二版中，如果只是未指定<new code>，新子字段与<code>中指定的相同。</p> <p>copyField 创建一个单独的字段，而不是将其添加到任何现有的字段。您可能希望将新字段与现有字段组合 (见 组合字段)。</p>
删除字段和子字段	<p>removeControlField "<tag>"</p> <p>例如：removeControlField "009"</p>	<p>删除所有出现的控制字段。</p> <p>请注意，如果移除控制字段008，如果您没有重新创建它Alma会立刻重新创建。考虑在移除后重新添加字段，例如：</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <pre>rule "remove 008" when</pre> </div>

操作	格式/示例	评论
		<pre>(TRUE) then removeControlField "008" addControlField "008.#####s2013####xx#####r#####000#0#eng#d" end</pre>
	<pre>removeField "<tag>" 例如: removeField "880"</pre>	删除所有出现的字段 <tag>.
	<pre>removeSubField "<tag>.<code>" 例如: removeSubField "245.h"</pre>	从指定字段删除所有出现的子字段 <code>。
替换字段或子字段中的文本	<pre>replaceControlContents "<tag>.{<position>,<length>}. <value>" with "<new value>" 示例: replaceControlContents "LDR.{7,1}.s" with "m"</pre>	Replaces <value> with "<new value>" in starting position <position> to <position>+<length> of control field <tag>. 仅替换匹配的文本 <value>。
	<pre>replaceContents "<tag>.<code>.<value>" with "<new value>" replaceContentsOnlyFirst "<tag>.<code>.<value>" with "<new value>" replaceContentsExceptFirst "<tag>.<code>.<value>" with "<new value>" 例如: replaceContents "245.h.[Journal]" with "[Book]"</pre>	使用“new value”替换字段“<tag>”的子字段<code>中匹配的字符串（或仅第一个匹配的字符串或除第一个匹配字符串以外的所有字符串）<value>。无匹配<value>的字符串或部分字符串未被修改。
	<pre>replaceSubFieldContents "<tag>.<code>" with "<tag>.<code>" 例如: replaceSubFieldContents "245.b" with "100.a"</pre>	用另一个子字段的内容替换子字段的内容。
	<pre>replaceFixedContents "<tag>.{<1_ind>,<2_ ind>}.<code>.{<position>,<length>}.<value>" with "<new value>" Example: replaceFixedContents "100.{1,2}.a.{0,8}.20150226" with "20220724"</pre>	替换UNIMARC和CNMARC 1XX固定位置字段中的<value>为<new value>。 仅与UNIMARC/CNMARC 1XX字段相关。
在子字段中添加文本	<pre>prefix "<tag>.<code>" with "<value>" 例如: prefix "035.b" with "(OCoLC)"</pre>	向“<tag>”字段的“<code>”子字段的值添加前缀。 新值为<code>后带有旧值。

操作	格式/示例	评论
	<p>prefixSubField "<tag>.<code>" with "<source tag>.<source code>"</p> <p>例如：prefixSubField "910.a" with "906.a"</p>	<p>向"<source tag>"字段中添加"<source code>"字字段的值，并作为子字段"<code>"的前缀添加到"<tag>"。</p> <p>新值将为"<source tag>"字段中子字段"<source code>"的值并跟有旧值。</p>
	<p>suffix "<tag>.<code>" with "<value>"</p> <p>例如：suffix "035.b" with "(OCoLC)"</p>	<p>向"<tag>"字段的"<code>"字字段的值添加后缀。</p> <p>新值为旧值后带有<value>。</p>
	<p>suffixSubField "<tag>.<code>" with "<source tag>.<source code>"</p> <p>例如：suffixSubField "910.a" with "907.c"</p>	<p>向"<source tag>"字段中添加"<source code>"字字段的值，并作为子字段"<code>"的后缀添加到"<tag>"。</p> <p>新值将为旧值后带有"<source tag>"字段中子字段"<source code>"的值。</p>
<p>在书目和规范记录中维护代理信息</p>	<p>addCreatingAgency "<tag>.<code>"</p> <p>例如：addCreatingAgency "040.a"</p>	<p>在"<tag>"字段的"<code>"子字段中添加创建代理商ISIL代码。</p>
<p>例如，该语法可用于在MARC 21书目元数据配置任务列表中选择规范化规则，以便在保存时规范化网络区书目记录。</p>	<p>addModifyingAgency "<tag>.<code>"</p> <p>例如：addModifyingAgency "040.d"</p>	<p>在"<tag>"字段的"<code>"子字段中添加修改代理ISIL代码。如果"<tag>.<code>"中已经有一个修改代理机构，那么会增加另一个代理商的ISIL代码。</p>
<p>Note</p> <p>此功能正在建设中。要启用此语法，请联系Ex Libris支持。</p>	<p>replaceModifyingAgency "<tag>.<code>"</p> <p>例如：replaceModifyingAgency "040.d"</p>	<p>在"<tag>"字段的"<code>"子字段中添加修改代理ISIL代码。存在于"<tag>.<code>"中的修改代理都将被替换。</p>

操作	格式/示例	评论
拆分子字段	<pre>splitSubField "<tag>.{ind1,ind2}.<code>.<delimiter>" to "<tag>.{<ind1>,<ind2>}.<code>" addSeq "<code>"</pre> <p>示例1: <code>splitSubField "866.a.;" to "555.{0,0}.a" addSeq "8"</code></p> <p>示例2: <code>splitSubField "555.a.- " to "859.{0,0}.a" addSeq "8"</code></p> <p>示例3: <code>splitSubField "859.a.\\\\".</code></p> <p>示例4: <code>splitSubField "999.a.;" to "555.a" addSeq "8"</code></p>	<p>标签为必备。</p> <p>指示符可选。</p> <p>因为拆分在子字段层级，代码为必备。</p> <p>分隔符可为任意字符串。如果分隔符不存在，完整的子字段复制为第一段，并添加后续。</p> <p>to组件可选。如果指定，创建多个to tag.code，每个都含有数据直到分隔符。见示例1和2。如果未指定to组件，子字段在相同字段中拆分为其他的相同子字段，如示例3。</p> <p>addSeq组件可选。如果to组件未指定则无关。指定addSeq时，如示例1添加含有序列的子字段；如果子字段已在初始字段中存在，序列添加到字段中如示例2。</p>
移除重复子字段	<pre>correctDuplicateSubfields "<tag>" "<code>"</pre> <p>例如：</p> <pre>rule "Remove duplicates" priority 1 when (TRUE) then correctDuplicateSubfields "610,630" "xyz" end</pre>	<p>根据作为参数传递的字段和子字段保留当前记录中第一次出现的内容并移除其他内容，以此更正重复子字段（例如，带有相同代码和相同值的子字段）。</p> <p>您可能想要使用recordHasDuplicateSubfields创建提供给使用correctDuplicateSubfields的规范化规则的结果集。有关更多信息，见示例6。</p> <p>对于带有不同值的子字段的重复数据删除，请见：如何在元数据编辑器中的同一字段内删除其中一个相似子字段之外的所有子字段。</p>
移动子字段	<pre>moveSubfieldsToEndOfField "<tag>" "<code>"</pre> <p>例如：</p> <pre>rule "Move subfields to end of field" priority 1 when (TRUE) then moveSubfieldsToEndOfField "650" "92" end</pre>	<p>移动每个子字段的第一个内容到字段末尾并移除相同子字段的其他内容。</p> <p>如果定义了多个子字段，根据规则将其按顺序放置在末尾。该例子中，子字段9放置在末尾，子字段2紧随其后。</p> <p>请注意moveSubfieldsToEndOfField操作不支持if声明。</p>
更正当前记录的重复字段	<pre>correctDuplicateFields "{fields}"</pre> <p>例如：</p> <pre>correctDuplicateFields "610,630,650"</pre>	<p>该操作使用一个参数，含有按逗号分隔字段值的字段，例如610,630,650。</p> <p>该操作根据由参数通过的字段更正当前记录的重复字段。</p> <hr/> <p>Note</p> <p>此操作仅可用于识别和纠正正在同一字段的多个实例之间发现的重复数据。</p> <hr/>

操作	格式/示例	评论
查找重复字段 (指示规则; 见 使用指示规则)	<pre>recordHasDuplicateFields "{fields}"</pre> <p>例如:</p> <pre>recordHasDuplicateFields "610,630,650"</pre>	<p>该操作使用一个参数, 含有按逗号分隔字段值的字段, 例如 610,630,650。</p> <p>该操作可以为true或false。如果根据由参数通过的字段如果在当前记录中找到了重复字段, 返回true。</p> <hr/> <p>Note</p> <p>此操作仅可用于识别和纠正同一字段的多个实例之间发现的重复数据。</p> <hr/>

通配符和特殊字符

Note

虽然目前不完全支持在Alma规范化规则中使用正则表达式, 但可以使用一些正则表达式 - 有关示例, 请见[Alma规范化规则示例](#)。

子字段分隔符 (\$\$) 不能在规则中出现, 包括规则名称。

#号在行的开头标识剩余内容为一条命令, 并在处理规则时忽略。

Note

规范化规则函数`replaceControlContents`不会将井号(#)视为空格。井号(#)不是通配符, 因此不受此功能支持。

星号 (*) 用于匹配任何字符串, 包括长度为0的字符串。例如, "`<tag>.<*>.<value>`"适用于标签`<tag>`中具有`<value>`值的所有子字段。*是“贪心”, 所以它在字符串中匹配尽可能多的字符。例如: 如果您有一个字符串“a b c d b e”, 模式“`b*b`”匹配“b c b d b”, 而不只是“b c b”。

空指示符 (但不是字段或子字段) 由破折号 (-) 指示。例如, "`<tag> {-,<ind2>}`"返回MARC标签为`<tag>`的所有字段, 第一个指示符未定义, 第二个指示符为`<ind2>`。

如果子字段的文本包含句点作为最后的字符, 则使用四个反斜杠来匹配句点。例如:

```
rule "Replace 'l v.' to 'Leaves' in $a (unconditional)"
when
  (TRUE)
then
  replaceContents "300.a.1 v\\\\" with "Leaves"
end
```

双引号 (") 可用于条件。要使用双引号作为条件的一部分, 请使用单引号将规则 (') 中的文本括起来, 而不是双引号

(“)。这样，您可以在双斜杠 (//) 之后的文本中使用双引号。

```
rule "populate 008 7-10 2016"
when
  (exists '245.{*, }.c.\"')
then
  replaceControlContents "008.{7,4}" with "2016"
end
```

希伯来语日期 (仅) 可用于条件。(希伯来语从右到左读取，因此下例中的双斜杠在双引号之前。)

```
rule "populate 008 7-10 2016"
when
  ((exists '260.{*, }.c.י"\\ֿׁׂ׃') OR (exists '264.{*, }.c.י"\\ֿׁׂ׃'))
then
  replaceControlContents "008.{7,4}" with "2016"
end
```

Note

- 不能在条件或值的第一个字符使用通配符。
- 要使用反斜杠 (\)，其后附上另一个反斜杠：\\。
- 当句点为字符串的最后字符时，使用四个反斜杠 (\\) 代表句点。当句点后面是另一个字符时，它不需要前面的四个反斜杠 (比如 `addField "907.a.F.L.T\\."`)。但是，最佳做法是在规范化规则中始终使用四个反斜杠，以确保最一致的预期结果。见下例。
- 如上文所述，如果在用单引号引用的条件中使用双引号或单引号，必须在其后附上双反斜杠。
- 如果 (|) 符号是值的一部分，使用四个反斜杠 (\\) 避开，例如：`removeField "866" if (exists "866.8.0\\|99")` 仅当条件中使用 (|) 符号时需要这样处理。

示例：在使用 `replaceContents` 的规范化规则中使用句点。

具有句点的示例记录：

```
245 00 $$a Feminist literary theory.: $$b a reader / $$c edited by Mary Eagleton.
```

```
246 0# $$a F.L.T.
```

上述示例记录的规范化规则：

```
rule "remove the periods in 245 and 246 subfield a (and replace periods with nothing);
precede period with four backslashes"
when
  (TRUE)
then
  replaceContents "245.a.\\\\" with ""
  replaceContents "246.a.\\\\" with ""
end
```

前后示例见下图。

040		\$\$a DLC \$\$c DLC	035	0	\$\$a AEW2509
049		\$\$a BXMM	040		\$\$a DLC \$\$c DLC
050	0	0 \$\$a PN98.W64 \$\$b F44 1996	049		\$\$a BXMM
082	0	0 \$\$a 801/95/082 \$\$2 20	050	0	0 \$\$a PN98.W64 \$\$b F44 1996
245	0	0 \$\$a Feminist literary theory : \$\$b a reader / \$\$c edited by Mary Eagleton.	082	0	0 \$\$a 801/95/082 \$\$2 20
246	0	0 \$\$a F.L.T.	245	0	0 \$\$a Feminist literary theory : \$\$b a reader / \$\$c edited by Mary Eagleton.
250		\$\$a 2nd ed.	246	0	\$\$a FLT
260		\$\$a Cambridge, Mass. : \$\$b Blackwell, \$\$c 1996.	250		\$\$a 2nd ed.
263		\$\$a 9511	260		\$\$a Cambridge, Mass. : \$\$b Blackwell, \$\$c 1996.
			263		\$\$a 9511

前后示例

示例：在使用addField的规范化规则中使用句点

以下是需要添加句点的记录示例：

906 \$\$a Architecture.

907 \$\$a F.L.T.

以下是使用始终包括斜杠的最佳做法的上述示例记录的规范化规则：

```
rule "Add field 906 with text Architecture and period at end and also add field 907 with F.L.T."
  salience 100
  when
    TRUE
  then
    addField "906.a.Architecture\\\\"
    addField "907.a.F\\\\".L\\\\".T\\\\"
  end
```

前后记录示例见下图。

504		\$\$a includes bibliographical references (p. [421]-428) and index.	index.
650	0	\$\$a Feminist literary criticism.	650 0 \$\$a Feminist literary criticism.
700	1	\$\$a Eagleton, Mary.	700 1 \$\$a Eagleton, Mary.
948		\$\$a LTI 08/08/2000	948 \$\$a LTI 08/08/2000
950		\$\$a OL	950 \$\$a OL
			906 \$\$a Architecture.
			907 \$\$a F.L.T.

前后示例

规范化规则示例

有关含有超过100个规范化规则示例的列表，见[Alma规范化规则示例](#)。

有关为XSL格式的书目记录创建规范化规则的更多信息，见[Alma XSL规范化规则示例](#)。

Note

Alma最高支持XSL版本3.0。