

Mini Language - minilang - simple-method - Reference

Current Mini Language (minilang) Reference Guide (Version 3)

Grammar version 2 (<http://ofbiz.apache.org/dtds/simple-methods-v2.xsd>)



Older versions

There are 2 older versions that you can find respectively at

[Mini-Language Guide \(Version 1 - Deprecated\)](#)

[Mini-Language Guide \(Version 2 - Deprecated\)](#)

Attributes

Attribute Types

Type	Description	Example
constant	A string constant. The string may not include an expression.	name="fooMethod"
constant+expr	A string constant. The string may include an expression, but an expression-only string is not allowed.	default="\${foo}Method"
expression	A UEL expression without enclosing brackets.	from="item.quantity * item. priceEach"
`\${expression}`	A UEL expression with enclosing brackets.	default="\${parameters.itemId}"
script	A short script ("scriptlet").	script="groovy:foo.bar();"

The <simple-method> element

<simple-method>

Contains a block of code.

A simple method can be called in either an event context from the Control Servlet (or another event) or in a service context through the Service Engine, or any other component that has access to a service dispatcher.

```
<simple-method method-name="fooMethod">
  <!-- some mini-language code goes here -->
</simple-method>
```

Attributes

Name	Type	Requirements	Description	Note
method-name	constant	required	A name (preferably a legal Java identifier) for this method.	Each simple method in a file must have a unique name.
short-description	constant	optional	A short description of the method.	Used for documentation.
login-required	constant	optional	Require a user login to run this method.	Defaults to "true".
use-transaction	constant	optional	Create a transaction if none exists for this thread.	Defaults to "true".
default-error-code	constant	optional	The default error return code.	Defaults to "error".
default-success-code	constant	optional	The default success return code.	Defaults to "success".
event-request-object-name	constant	optional	The name of the field containing the <code>javax.servlet.ServletRequest</code> object.	Defaults to "request".
event-response-object-name	constant	optional	The name of the field containing the <code>javax.servlet.ServletResponse</code> object.	Defaults to "response".

event-session-object-name	constant	optional	The name of the field containing the <code>javax.servlet.http.HttpSession</code> object.	Defaults to "session".
event-response-code-name	constant	optional	The name of the field containing the event response code.	Defaults to "_response_code_".
event-error-message-name	constant	optional	The name of the field containing the event error message.	Defaults to "_error_message_".
event-error-message-list-name	constant	optional	The name of the field containing the event message list.	Defaults to "_error_message_list_".
event-event-message-name	constant	optional	The name of the field containing the event message.	Defaults to "_event_message_".
event-event-message-list-name	constant	optional	The name of the field containing the event message list.	Defaults to "_event_message_list_".
service-response-message-name	constant	optional	The name of the field containing the service response message.	Defaults to "responseMessage".
service-error-message-name	constant	optional	The name of the field containing the service error message.	Defaults to "errorMessage".
service-error-message-list-name	constant	optional	The name of the field containing the service error message list.	Defaults to "errorMessageList".
service-error-message-map-name	constant	optional	The name of the field containing the service error message map.	Defaults to "errorMessageMap".
service-success-message-name	constant	optional	The name of the field containing the service success message.	Defaults to "successMessage".
service-success-message-list-name	constant	optional	The name of the field containing the service success message list.	Defaults to "successMessageList".

Child Elements

Name	Requirements	
(Any block of code)	optional	

Assignment Operations

Assignment operations modify the script's state.

<add-error>

Adds a message to the error message list.

```
<add-error>
  <fail-message message="There was an error" />
</add-error>
```

Attributes

Name	Type	Requirements	Description	Note
error-list-name	constant	optional	The name of a list that will contain the message.	Defaults to "error_list".

Child Elements

One of the following child elements is required:

Name	
<fail-message>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}})
<fail-property>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}})

<alt-permission>

Declares an alternate permission (<check-permission> sub-element).

Alternate permissions are checked when the primary permission check fails.

Attributes

Name	Type	Requirements	Description	Note
permission	constant, \${expression}	required	The permission to check.	
action	constant, \${expression}	optional	The action to be performed (permission scope).	Examples: "_ADMIN", "_CREATE", "_UPDATE".

<assert>

Adds an error message to the error message list for each condition that evaluates to false.

Attributes

Name	Type	Requirements	Description	Note
title	constant, \${expression}	optional	The title of the assert operation.	The title is used in test reports.
error-list-name	constant	optional	The name of a list that will contain the message.	Defaults to "error_list".

Child Elements

Name	
Any <condition> child element	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<check-id>

Evaluates the specified field, and adds an error message to the error message list if the field's value is not a valid data source ID value.

Valid IDs can be any sequence of characters or digits, but they must not contain the following characters: space [], double quote ["], single quote ['], ampersand [&], question mark [?], less-than sign [<], greater-than sign [>], forward-slash [/], back-slash [\\].

Attributes

Name	Type	Requirements	Description	Note
field	expression	optional	The name of the field to test.	
error-list-name	constant	optional	The name of a list that will contain the message.	Defaults to "error_list".

Child Elements

One of the following child elements is required:

Name	
<fail-message>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<fail-property>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<check-permission>

Checks if the user has the specified permission, and adds an error message to the error message list if the user does not have the specified permission.

Note that this element must be followed by the <check-errors> element for it to do anything meaningful.

Attributes

Name	Type	Requirements	Description	Note
permission	constant, \${expression}	required	The permission to check.	
action	constant, \${expression}	optional	The action to be performed (permission scope).	Examples: "_ADMIN", "_CREATE", "_UPDATE".
error-list-name	constant	optional	The name of a list that will contain the message.	Defaults to "error_list".

Child Elements

One or more of the following child element is optional:

Name	
<alt-permission>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

One of the following child elements is required:

Name	
<fail-message>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<fail-property>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<clear-field>

Sets the value of the specified field to null.

```
<clear-field field="foo" />
```

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to remove.	

<create-object>

Creates an instance of the specified class.

The <string> and <field> sub-elements are passed to the constructor method as arguments in the order they are specified. A runtime exception will be thrown if the sub-elements do not match the constructor method arguments.

```
<create-object field="foo" class-name="com.acme.FooClass">
  <string value="bar" />
  <field field="foo" />
</create-object>
```

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field for the new object.	
class-name	constant	required	The Java class name.	

Child Elements

Name	Requirements	
<field>	optional	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<string>	optional	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})



Deprecated - use the <script> element.

<fail-message>

Declares an error message.

```
<add-error>
  <fail-message message="There was an error." />
</add-error>
```

Attributes

Name	Type	Requirements	Description	Note
message	constant+expr	required	The message text.	

<fail-property>

Declares an error message found in a properties file.

```
<add-error>
  <fail-property resource="ErrorMessages" property="FooError" />
</add-error>
```

Attributes

Name	Type	Requirements	Description	Note
resource	constant	required	The name of the resource where the message can be found.	
property	constant	required	The message property key.	

<field>

Declares an environment field to be passed as an argument to an object method call.

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the environment field to use.	
type	constant	optional	The Java class of the argument.	Defaults to "java.lang.String".

<field-to-list>

Appends an object to the specified list.

```
<field-to-list list="barList" field="foo" />
```

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The field to add to the list.	The operation does nothing if the field does not exist.
list	expression	required	Name of the list.	If the list does not exist, one is created.



Deprecated - use the <set> element.

<field-to-request>

Copies a field to a servlet request attribute.

Valid only when the simple-method is called as an event, it is ignored otherwise.

```
<field-to-request field="foo" request-name="fooAttribute" />
```

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to copy from. The source of the assignment.	The operation does nothing if the field does not exist.
request-name	constant, \${expression}	optional	The servlet request attribute name. The target of the assignment.	Defaults to the value of field attribute.

<field-to-result>

Copies a field to a service OUT attribute.

```
<field-to-result field="foo" result-name="fooAttribute" />
```

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to copy from. The source of the assignment.	The operation does nothing if the field does not exist.
result-name	expression	optional	The name of the result field to set. The target of the assignment.	Defaults to the value of field attribute.

<field-to-session>

Copies a field to a servlet session attribute.

Valid only when the simple-method is called as an event, it is ignored otherwise.

```
<field-to-session field="foo" session-name="fooAttribute" />
```

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to copy from. The source of the assignment.	The operation does nothing if the field does not exist.
session-name	constant, \${expression}	optional	The servlet session attribute name. The target of the assignment.	Defaults to the value of field attribute.

<first-from-list>

Assigns the first (lowest order) entry in a list to a field.

```
<first-from-list entry-name="foo" list="fooList" />
```

Attributes

Name	Type	Requirements	Description	Note
entry-name	expression	required	The name of the field to set.	
list	expression	required	The name of the list that contains the object to copy.	Defaults to a null value if the list does not exist or is empty.



Deprecated - use the <set> element.

<list-to-list>

Copies list elements to another list.

```
<list-to-list list="foo" to-list-name="fooList" />
```

Attributes

Name	Type	Requirements	Description	Note
list	expression	required	The name of the list to copy from.	Operation does nothing if the list does not exist.
to-list-name	expression	required	Name of the list to copy to.	If the list does not exist, one will be created.

<map-to-map>

Copies a map to another map.

```
<map-to-map map="oldMap" to-map="newMap" />
```

Attributes

Name	Type	Requirements	Description	Note
map	expression	required	The name of the map to copy from.	Operation does nothing if the map does not exist.
to-map	expression	optional	The name of the map to copy to.	If this attribute is empty, the source map will be copied to the current environment.

<now>

Sets a field to the current system time.

```
<now field="fooNow" />
```

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to set, the target of the assignment.	
type	constant	optional	The field data type.	Defaults to "java.sql.Timestamp".

<now-date-to-env>

Sets a field to a `java.sql.Date` object initialized to the current system time.

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to set, the target of the assignment.	



Deprecated - use the `<now>` element.

<now-timestamp>

Sets a field to a `java.sql.Timestamp` object initialized to the current system time.

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to set, the target of the assignment.	



Deprecated - use the `<now>` element.

<order-by>

Declares a field or map key to sort on.

```
<order-map-list list="fooList">
  <order-by field="fooKey" />
</order-map-list>
```

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field or map key to sort on.	

<order-map-list>

Sorts a list of maps.

Maps are sorted by the keys specified in the `<order-by>` sub-elements.

```
<order-map-list list="fooList">
  <order-by field="fooKey" />
</order-map-list>
```

Attributes

Name	Type	Requirements	Description	Note
list	expression	required	The name of the list to be sorted.	The operation does nothing if the list is not found.

Child Elements

One or more of the following child element is required:

Name	
<order-by>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})

<property-to-field>

Assigns a resource property value to a field.

```
<property-to-field field="foo" resource="fooResource.properties" property="fooKey" />
```

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to set, the target of the assignment.	
resource	constant, \${expression}	required	The name of a properties resource. Can be a file on the classpath or a resource defined in the SystemProperty entity.	
property	constant, \${expression}	required	The property key.	
default	constant, \${expression}	optional	The default value to use if the property value is null or empty.	
no-locale	constant	optional	Suppress property value localization. The user's/system locale will be ignored when retrieving the property value.	Default is "false".
arg-list	expression	optional	The name of an argument list to be used with a formatting string. The argument list is applied to the property value.	The attribute does nothing if the argument list is not found. See the java.text.MessageFormat class for more information.

<request-

parameters-to-list>

Appends a servlet request parameter list to a list.

Valid only when the simple-method is called as an event, it is ignored otherwise.

```
<request-parameters-to-list list="fooList" request-name="fooParameter" />
```

Attributes

Name	Type	Requirements	Description	Note
request-name	constant, \${expression}	required	The name of the servlet request parameter. All matching parameter values will be added to the list.	
list	expression	optional	The name of the list.	If the list does not exist, one will be created. Defaults to the request-name attribute value.

<request-to-field>

Copies a servlet request attribute to a field.

Valid only when the simple-method is called as an event, it is ignored otherwise.

```
<request-to-field field="foo" request-name="fooAttribute" />
```

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to set, the target of the assignment.	
request-name	constant, \${expression}	optional	The servlet request attribute name.	Defaults to the value of field attribute.
default	constant, \${expression}	optional	A default value to use if the request attribute does not exist.	

<session-to-field>

Copies a servlet session attribute to a field.

Valid only when the simple-method is called as an HTTP event, it is ignored otherwise.

```
<session-to-field field="foo" session-name="fooAttribute" />
```

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to set, the target of the assignment.	
session-name	constant, \${expression}	optional	The servlet session attribute name.	Defaults to the value of the field attribute.
default	constant, \${expression}	optional	A default value to use if the session attribute does not exist.	

<set>

Assigns a field from an expression or script, or from a constant value.

```
<set field="foo" value="bar" />
<set field="baz" from="foo" />
<!-- Field "baz" contains the string "bar" -->
```

Attributes

Name	Type	Requirements	Description	Notes
field	expression	required	The name of the field to set, the target of the assignment.	
from	expression, script	required if <i>value</i> is empty	An expression or script that returns an object or null.	The script must be prefixed with the script language followed by a colon (":").
value	constant+expr	required if <i>from</i> is empty	A constant value.	Default type = "java.lang.String".
default	constant, \${expression}	optional	A default value that is used when the from attribute evaluates to null or empty.	
type	constant	optional	The Java data type of <i>field</i> .	"NewList" will create a new java.util.List, "NewMap" will create a new java.util.Map. Otherwise, the attribute must contain a valid Java class name.
locale	constant, \${expression}	optional	If a locale dependent type is passed allows to i18n the value.	Defaults to environment locale.
set-if-null	constant	optional	Controls if <i>field</i> can be set to null.	Defaults to "false".
set-if-empty	constant	optional	Controls if <i>field</i> can be set to an empty value. The meaning of "empty" depends on the Java data type.	Defaults to "true".

<set-calendar>

Adjusts a Timestamp by a specified time duration.

```
<set-calendar field="tomorrowStamp" from-field="nowTimestamp" day="1" />
<set-calendar field="yesterdayStamp" from-field="nowTimestamp" day="-1" />
```

Attributes

Name	Type	Requirements	Description	Notes
field	expression	required	The name of the field to set, the target of the assignment.	
from	expression, script	required if <i>value</i> is empty	An expression or script that returns an object or null.	The script must be prefixed with the script language followed by a colon (":").
value	constant+expr	required if <i>from</i> is empty	A constant value.	Default type = "java.lang.String".
default	constant, \${expression}	optional	A default value that is used when the from attribute evaluates to null or empty.	
set-if-null	constant	optional	Controls if <i>field</i> can be set to null.	Defaults to "false".

years	constant, \${expression}	optional	Add (optionally using +) or subtract (using -) a number of year(s).	If an expression is used, it should evaluate to an integer.
months	constant, \${expression}	optional	Add (optionally using +) or subtract (using -) a number of month(s).	If an expression is used, it should evaluate to an integer.
days	constant, \${expression}	optional	Add (optionally using +) or subtract (using -) a number of days(s).	If an expression is used, it should evaluate to an integer.
hours	constant, \${expression}	optional	Add (optionally using +) or subtract (using -) a number of hour(s).	If an expression is used, it should evaluate to an integer.
minutes	constant, \${expression}	optional	Add (optionally using +) or subtract (using -) a number of minute(s).	If an expression is used, it should evaluate to an integer.
seconds	constant, \${expression}	optional	Add (optionally using +) or subtract (using -) a number of second(s).	If an expression is used, it should evaluate to an integer.
millis	constant, \${expression}	optional	Add (optionally using +) or subtract (using -) a number of milli-second(s).	If an expression is used, it should evaluate to an integer.
period-align-start	constant, \${expression}	optional	Align the adjusted date to the start of a period: "day", "week", "month", "year"	
period-align-end	constant, \${expression}	optional	Align the adjusted date to the end of a period: "day", "week", "month", "year"	
locale	constant, \${expression}	optional	The locale (calendar) to be used for the operation	Defaults to the environment locale.
time-zone	constant, \${expression}	optional	The time zone to be used for the operation	Defaults to the environment time zone.

<set-current-user-login>

Sets the UserLogin entity value to be used for authentication for the rest of the method.

Attributes

Name	Type	Requirements	Description	Notes
value-field	expression	required	The name of the field that contains the UserLogin entity value.	A runtime exception will be thrown if the entity value does not exist.



Deprecated - you can pass an alternate UserLogin entity value to the called service's IN attributes.

<set-service-fields>

Copies map elements that match a service's IN attributes from a source map to a target map.

```
<set-service-fields service-name="fooService" map="barMap" to-map="fooServiceAttributes" />
```

Attributes

Name	Type	Requirements	Description	Notes
service-name	constant, \${expression}	required	The name of the service.	
map	expression	required	The name of the map to copy the matching IN attributes from.	The operation does nothing if the map does not exist.
to-map	expression	required	The name of the map to copy the matching IN attributes to.	If the map does not exist, a new one is created.

<string>

Declares a `java.lang.String` to be passed as an argument to a method call.

The String can be contained in the value attribute and/or in the element body. The element body text is appended to the value attribute text.

Attributes

Name	Type	Requirements	Description	Note
value	constant, \${expression}	optional	The String text.	

<string-append>

Performs string concatenation and formatting.

The operation starts by applying an argument list (if found) to the string attribute value, the result is prepended by the prefix attribute value, and the suffix attribute value is appended to the result. If the string specified in the field attribute exists, the final result is appended to it, else the field is set to the final result.

Attributes

Name	Type	Requirements	Description	Notes
field	expression	required	The name of the field to set, the target of the assignment.	
string	constant, \${expression}	required	The string to append to the field named in the field attribute.	This can be a formatting string that is used with the argument list specified in the arg-list attribute.
arg-list	expression	optional	The name of an argument list to be used with a formatting string.	The argument list is applied to the string attribute value. The attribute does nothing if the argument list is not found. See the java.text.MessageFormat class for more information.
prefix	constant, \${expression}	optional	A string that will be prepended to the string attribute value.	
suffix	constant, \${expression}	optional	A string that will be appended to the string attribute value.	

<string-to-list>

Adds a string to a list of strings.

Attributes

Name	Type	Requirements	Description	Notes
string	constant, \${expression}	required	The string to add to the list specified in the list attribute.	
list	expression	required	List to add the string to.	If the list does not exist, one will be created.
arg-list	expression	optional	The name of an argument list to be used with a formatting string.	The argument list is applied to the string attribute value. The attribute does nothing if the argument list is not found. See the java.text.MessageFormat class for more information.
message-field	constant	optional	Inserts a message above a field (used in conjunction with @fieldErrors FTL macro).	



Deprecated - use the <set> element.

<to-string>

Converts an object to a string.

Attributes

Name	Type	Requirements	Description	Notes
field	expression	required	The name of the field containing the object to convert.	The operation does nothing if the object is not found.
format	constant	optional	The format to use for the conversion.	
numeric-padding	constant	optional	Left-pad the string with the specified number of zeroes.	



Deprecated - use the <set> element.

<webapp-property-to-field>

Copies a property value from a properties file in a ServletContext resource to a field.

Valid only when the simple-method is called as an event, it is ignored otherwise.

```
<webapp-property-to-field field="foo" resource="/WEB-INF/foo.properties" property="fooKey" />
```

Attributes

Name	Type	Requirements	Description	Notes
field	expression	required	The name of the field to set, the target of the assignment.	
resource	constant, \${expression}	required	The resource location of the properties file inside the webapp, and relative to the root of the webapp (can be inside a war file).	Example: "/WEB-INF /myprops.properties".
property	constant, \${expression}	required	The property key.	
default	constant, \${expression}	optional	A default value to use if the property value is null or empty.	

Conditional/Looping Statements

Conditional/looping statements control the script execution path.

<and>

Combines conditional elements using a boolean AND.

Evaluates to true if all sub-elements evaluate to true.

```
<if>
  <condition>
    <and>
      <if-compare field="colorModel" operator="equals" value="RYB" />
      <or>
        <if-compare field="color" operator="equals" value="red" />
        <if-compare field="color" operator="equals" value="yellow" />
        <if-compare field="color" operator="equals" value="blue" />
      </or>
    </and>
  </condition>
  <then>
    <set field="isPrimaryColor" value="true" type="Boolean" />
  </then>
  <else>
    <set field="isPrimaryColor" value="false" type="Boolean" />
  </else>
</if>
```

Child Elements

One or more of the following child elements is required:

Name	
<and>	
<if-validate-method>	(?) (<a "="" href="https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}">https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})
<if-compare>	(?) (<a "="" href="https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}">https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})
<if-compare-field>	(?) (<a "="" href="https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}">https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})
<if-regexp>	(?) (<a "="" href="https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}">https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})
<if-empty>	(?) (<a "="" href="https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}">https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})
<if-has-permission>	(?) (<a "="" href="https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}">https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})
<or>	(?) (<a "="" href="https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}">https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})
<not>	(?) (<a "="" href="https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}">https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})
<xor>	(?) (<a "="" href="https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}">https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})

<break>

Causes script execution to exit the nearest looping element.

The operation will throw an exception if there is no enclosing looping element.

Enclosing Looping Elements

Name	
<iterate>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<iterate-map>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<loop>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<while>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<check-errors>

Halts script execution if the error message list contains any messages.

The error message list is returned to the calling process.

Attributes

Name	Type	Requirements	Description	Note
error-code	constant, \${expression}	optional	The error code to return to the calling process.	Defaults to value of the containing <simple-method> element "default-error-code" attribute.
error-list-name	constant, \${expression}	optional	The name of a list that will contain the message.	Defaults to "error_list".

<condition>

Combines a group of conditional elements into a single logical (true/false) expression.

```
<if>
  <condition>
    <and>
      <if-compare field="colorModel" operator="equals" value="RYB" />
    <or>
      <if-compare field="color" operator="equals" value="red" />
      <if-compare field="color" operator="equals" value="yellow" />
      <if-compare field="color" operator="equals" value="blue" />
    </or>
  </and>
</condition>
<then>
  <set field="isPrimaryColor" value="true" type="Boolean" />
</then>
<else>
  <set field="isPrimaryColor" value="false" type="Boolean" />
</else>
</if>
```

Child Elements

One of the following child elements is required:

Name	
<and>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-validate-method>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-compare>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-compare-field>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-regexp>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-empty>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-has-permission>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<or>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<not>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<xor>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<continue>

Causes script execution to return to the beginning of the nearest enclosing loop element.

The operation will throw an exception if there is no enclosing looping element.

Enclosing Looping Elements

Name	
<iterate>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<iterate-map>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<loop>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<while>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<else>

Contains a block of code to be executed when a condition evaluates to false.

```
<if>
  <condition>
    <and>
      <if-compare field="colorModel" operator="equals" value="RYB" />
    <or>
      <if-compare field="color" operator="equals" value="red" />
      <if-compare field="color" operator="equals" value="yellow" />
      <if-compare field="color" operator="equals" value="blue" />
    </or>
  </and>
</condition>
<then>
  <set field="isPrimaryColor" value="true" type="Boolean" />
</then>
<else>
  <set field="isPrimaryColor" value="false" type="Boolean" />
</else>
</if>
```

Child Elements

Name	Requirements	
(Any block of code)	optional	

<else-if>

Contains a condition and a block of code to be evaluated/executed when a parent condition evaluates to false.

If the condition of the parent element evaluates to false, then this element's condition will be evaluated, and if that condition evaluates to true, then the operations under the then element will be run.

```

<if>
  <condition>
    <and>
      <if-compare field="colorModel" operator="equals" value="RYB" />
      <or>
        <if-compare field="color" operator="equals" value="red" />
        <if-compare field="color" operator="equals" value="yellow" />
        <if-compare field="color" operator="equals" value="blue" />
      </or>
    </and>
  </condition>
  <then>
    <set field="isPrimaryColor" value="true" type="Boolean" />
  </then>
  <else-if>
    <condition>
      <and>
        <if-compare field="colorModel" operator="equals" value="CYM" />
        <or>
          <if-compare field="color" operator="equals" value="cyan" />
          <if-compare field="color" operator="equals" value="yellow" />
          <if-compare field="color" operator="equals" value="magenta" />
        </or>
      </and>
    </condition>
    <then>
      <set field="isPrimaryColor" value="true" type="Boolean" />
    </then>
  </else-if>
  <else>
    <set field="isPrimaryColor" value="false" type="Boolean" />
  </else>
</if>

```

Child Elements

The following child elements are required:

Name	
<condition>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{
<then>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{

<if>

Performs conditional processing on blocks of code.

This operation specifies combinations of conditions, alternate conditions, and operations to run based on the evaluation of condition expressions.

```

<if>
  <condition>
    <and>
      <if-compare field="colorModel" operator="equals" value="RYB" />
    <or>
      <if-compare field="color" operator="equals" value="red" />
      <if-compare field="color" operator="equals" value="yellow" />
      <if-compare field="color" operator="equals" value="blue" />
    </or>
  </and>
</condition>
<then>
  <set field="isPrimaryColor" value="true" type="Boolean" />
</then>
<else-if>
  <condition>
    <and>
      <if-compare field="colorModel" operator="equals" value="CYM" />
    <or>
      <if-compare field="color" operator="equals" value="cyan" />
      <if-compare field="color" operator="equals" value="yellow" />
      <if-compare field="color" operator="equals" value="magenta" />
    </or>
  </and>
</condition>
<then>
  <set field="isPrimaryColor" value="true" type="Boolean" />
</then>
</else-if>
<else>
  <set field="isPrimaryColor" value="false" type="Boolean" />
</else>
</if>

```

Child Elements

Name	Requirements	
<condition>	required	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})
<then>	required	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})
<else-if>	optional (one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})
<else>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})

<if-compare>

Compares a field to a constant value.

There are two versions of this element - one that contains sub-elements and can be used by itself, and another that does not contain sub-elements and is a sub-element of the <condition> element. The stand-alone version will execute its contained block of code if the condition evaluates to true, otherwise it will execute the block of code in its <else> sub-element (if one exists).

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field that will be compared. The l-value.	
operator	constant	required	The comparison operator. The operator describes the l-value compared to the r-value.	Valid values are: "equals", "not-equals", "less", "less-equals", "greater", "greater-equals", "contains", "is-null", "is-not-null", "is-empty". The "contains" operator returns true if the l-value contains the r-value. That operator does not perform type conversions. The l-value must be a collection type or a String. A null l-value evaluates to false. The "is-null", "is-not-null", and "is-empty" operators do not require an r-value. Any r-values will be ignored.

value	constant+expr	ignored when operator attribute equals "is-null", "is-not-null", or "is-empty" - required otherwise	The value that the field will be compared to. The r-value.	
type	constant	optional	The Java data type. Both values will be converted to this type before comparison.	Attribute must contain a valid Java class name. Invalid when the operator attribute value equals "contains".
format	constant, \${expression}	optional	Format used for type conversions.	Valid only when the type attribute is not empty.

Child Elements (stand-alone version only)

Name	Requirements	
(Any block of code)	optional	
<else>	optional	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{}())

<if-compare-field>

Compares a field to another field.

There are two versions of this element - one that contains sub-elements and can be used by itself, and another that does not contain sub-elements and is a sub-element of the <condition> element. The stand-alone version will execute its contained block of code if the condition evaluates to true, otherwise it will execute the block of code in its <else> sub-element (if one exists).

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field that will be compared. The l-value.	
operator	constant	required	The comparison operator. The operator describes the l-value compared to the r-value.	Valid values are: "equals", "not-equals", "less", "less-equals", "greater", "greater-equals", "contains". The "contains" operator returns true if the l-value contains the r-value. That operator does not perform type conversions. The l-value must be a collection type or a String. A null l-value evaluates to false.
to-field	expression	required	The name of the field that <i>field</i> will be compared to. The r-value.	
type	constant	optional	The Java data type. Both values will be converted to this type before comparison.	Attribute must contain a valid Java class name. Invalid when the operator attribute value equals "contains".
format	constant, \${expression}	optional	Format used for type conversions.	Valid only when the type attribute is not empty.

Child Elements (stand-alone version only)

Name	Requirements	
(Any block of code)	optional	
<else>	optional	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{}())

<if-empty>

Tests if a field is null or empty.

There are two versions of this element - one that contains sub-elements and can be used by itself, and another that does not contain sub-elements and is a sub-element of the <condition> element. The stand-alone version will execute its contained block of code if the condition evaluates to true, otherwise it will execute the block of code in its <else> sub-element (if one exists).

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to test.	Object to test must be a collection, string, or a class that implements org.ofbiz.base.lang.IsEmpty.

Child Elements (stand-alone version only)

Name	Requirements	
(Any block of code)	optional	
<else>	optional	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})

<if-has-permission>

Tests if the current user has the specified permission.

There are two versions of this element - one that contains sub-elements and can be used by itself, and another that does not contain sub-elements and is a sub-element of the <condition> element. The stand-alone version will execute its contained block of code if the condition evaluates to true, otherwise it will execute the block of code in its <else> sub-element (if one exists).

Attributes

Name	Type	Requirements	Description	Note
permission	constant, \${expression}	required	The permission to check.	
action	constant, \${expression}	optional	The action to be performed (permission scope).	Examples: "_ADMIN", "_CREATE", "_UPDATE".

Child Elements (stand-alone version only)

Name	Requirements	
(Any block of code)	optional	
<else>	optional	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})

<if-instance-of>

Tests if a field is an instance of a Java class.

The operation will execute its contained block of code if the condition evaluates to true, otherwise it will execute the block of code in its <else> sub-element (if one exists).

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to test.	Test evaluates to false if the field does not exist.
class	constant	required	The name of the class to test for.	Attribute must contain a valid Java class name.

Child Elements

Name	Requirements	
(Any block of code)	optional	
<else>	optional	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})

<if-not-empty>

Tests if a field is not null and not empty.

The operation will execute its contained block of code if the condition evaluates to true, otherwise it will execute the block of code in its <else> sub-element (if one exists).

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to test.	Object to test must be a collection, string, or a class that implements org.ofbiz.base.lang.IsEmpty.

Child Elements

Name	Requirements	
(Any block of code)	optional	
<else>	optional	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})

<if-regex>

Tests if a field matches a regular expression.

There are two versions of this element - one that contains sub-elements and can be used by itself, and another that does not contain sub-elements and is a sub-element of the <condition> element. The stand-alone version will execute its contained block of code if the condition evaluates to true, otherwise it will execute the block of code in its <else> sub-element (if one exists).

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to test.	The field object will be converted to a string before the test. An empty string will be used if the field object is not found.
expr	constant, \${expression}	required	The regular expression to test for.	

Child Elements (stand-alone version only)

Name	Requirements	
(Any block of code)	optional	
<else>	optional	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})

<if-validate-method>

Invokes a Java class static method that returns a boolean value.

There are two versions of this element - one that contains sub-elements and can be used by itself, and another that does not contain sub-elements and is a sub-element of the <condition> element. The stand-alone version will execute its contained block of code if the static method returns true, otherwise it will execute the block of code in its <else> sub-element (if one exists).

Attributes

Name	Type	Requirements	Description	Note
field	expression	required	The name of the field to use as the method argument.	The field object will be converted to a string before the method call. An empty string will be used if the field object is not found.
method	constant	required	The name of the static method that will be called to validate the field.	Must be a static method that takes a single java.lang.String parameter and return a boolean.
class	constant	optional	The name of the Java class that contains the validation method.	Defaults to "org.ofbiz.base.util.UtilValidate".

Child Elements (stand-alone version only)

Name	Requirements	
(Any block of code)	optional	
<else>	optional	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}})

<iterate>

Contains a block of code that is executed once for each entry in a collection.

Attributes

Name	Type	Requirements	Description	Note
list	expression	required	The name of the field that contains the collection to iterate over.	The operation does nothing if the field is not found.
entry	expression	required	The name of the field that will contain the current collection entry.	

Child Elements

Name	Requirements	
(Any block of code)	optional	

<iterate-map>

Contains a block of code that is executed once for each entry in a map.

Attributes

Name	Type	Requirements	Description	Note
map	expression	required	The name of the field that contains the map to iterate over.	The operation does nothing if the field is not found.
key	expression	required	The name of the field that will contain the current map entry key.	
value	expression	required	The name of the field that will contain the current map entry value.	

Child Elements

Name	Requirements	
(Any block of code)	optional	

<loop>

Contains a block of code that is executed repeatedly until a maximum count is reached.

Attributes

Name	Type	Requirements	Description	Note
count	constant, \${expression}	required	A maximum count expression.	Must evaluate to a positive integer. Defaults to "0" (zero)
field	expression	optional	The name of the field that will contain the current count value.	

Child Elements

Name	Requirements	
(Any block of code)	optional	

<not>

Inverts a conditional element using a boolean NOT.

Evaluates to true if the sub-element evaluates to false.

Child Elements

One of the following child elements is required:

Name	
<and>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{0})
<if-validate-method>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{0})
<if-compare>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{0})
<if-compare-field>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{0})
<if-regexp>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{0})
<if-empty>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{0})
<if-has-permission>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{0})

<or>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<not>	
<xor>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<or>

Combines conditional elements using a boolean OR.

Evaluates to true if any sub-element evaluates to true.

```
<if>
  <condition>
    <and>
      <if-compare field="colorModel" operator="equals" value="RYB" />
      <or>
        <if-compare field="color" operator="equals" value="red" />
        <if-compare field="color" operator="equals" value="yellow" />
        <if-compare field="color" operator="equals" value="blue" />
      </or>
    </and>
  </condition>
  <then>
    <set field="isPrimaryColor" value="true" type="Boolean" />
  </then>
  <else>
    <set field="isPrimaryColor" value="false" type="Boolean" />
  </else>
</if>
```

Child Elements

One or more of the following child elements is required:

Name	
<and>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-validate-method>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-compare>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-compare-field>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-regexp>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-empty>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-has-permission>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<or>	
<not>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<xor>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<return>

Returns control (execution) to the calling process.

Attributes

Name	Type	Requirements	Description	Note
response-code	constant, \${expression}	optional	A response code to return to the calling process.	Defaults to the value of the containing <simple-method> element "default-success-code" attribute.

<then>

Contains a block of code to be executed when a <condition> element evaluates to true.

```

<if>
  <condition>
    <and>
      <if-compare field="colorModel" operator="equals" value="RYB" />
      <or>
        <if-compare field="color" operator="equals" value="red" />
        <if-compare field="color" operator="equals" value="yellow" />
        <if-compare field="color" operator="equals" value="blue" />
      </or>
    </and>
  </condition>
  <then>
    <set field="isPrimaryColor" value="true" type="Boolean" />
  </then>
  <else>
    <set field="isPrimaryColor" value="false" type="Boolean" />
  </else>
</if>

```

Child Elements

Name	Requirements	
(Any block of code)	optional	

<while>

Contains a block of code that is executed repeatedly while a <condition> element evaluates to true.

Child Elements

The following child elements are required:

Name	
<condition>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<then>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<xor>

Combines conditional elements using a boolean XOR.

Evaluates to true if only one sub-element evaluates to true.

Child Elements

One or more of the following child elements is required:

Name	
<and>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-validate-method>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-compare>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-compare-field>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-regexp>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-empty>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<if-has-permission>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<or>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<not>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<xor>	

Call Operations

Operations that transfer control to another process. Note that some of these operations behave like assignment operators - since they modify the script's state, but they are listed separately for clarity.

<call-bsh>

Executes a BSH script.

The script can be contained in a Java resource, and/or a short script can be included in the element body.

If a Java resource is specified, and the element body contains a script, the element body script will be executed after the Java resource script is executed.

Attributes

Name	Type	Requirements	Description	Note
resource	constant	required if the element body is empty	A Java resource that contains the script.	



Deprecated - use the <script> element.

<call-class-method>

Calls a Java class method using the given fields as parameters.

The <string> and <field> sub-elements are passed to the method as arguments in the order they are specified. If the method returns a value, the value will be put in the named field.

Attributes

Name	Type	Requirements	Description	Note
class-name	constant	required	The name of the class containing the static method.	
method-name	constant	required	The name of the static method to call.	
ret-field	expression	optional	The name of the field to put the method return value in.	

Child Elements

One or more of the following child elements is optional:

Name	
<string>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<field>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})



Deprecated - use the <script> element.

<call-map-processor>

Invokes a <simple-map-processor> element.

The map processor can be contained in another file, and/or a <simple-map-processor> element can be included in the element body.

If an XML resource is specified, and the element body contains a <simple-map-processor> element, the contained map processor will be executed after the external map processor is executed.

Attributes

Name	Type	Requirements	Description	Note
in-map-name	expression	required	The name of the field that contains the map to be processed.	If the map does not exist, one will be created.
out-map-name	expression	required	The name of the field that will receive the processed map.	If the map does not exist, one will be created.
xml-resource	constant	optional	The location of the map processor file.	
processor-name	constant	optional	The name of the <simple-map-processor> element.	

error-list-name	expression	optional	The name of the field that will contain a list of error messages generated by the processor.	Defaults to "error_list".
-----------------	------------	----------	--	---------------------------

Child Elements

The following child element is optional:

Name	
<simple-map-processor>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<call-object-method>

Calls a method on an existing Java object.

The <string> and <field> sub-elements are passed to the method as arguments in the order they are specified. If the method returns a value, the value will be put in the named field.

Attributes

Name	Type	Requirements	Description	Note
obj-field	expression	required	The name of the field containing the object to be called.	A runtime exception is thrown if the object is not found.
method-name	constant	required	The name of the method to call.	
ret-field	expression	optional	The name of the field to put the method return value in.	

Child Elements

One or more of the following child elements is optional:

Name	
<string>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<field>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})



Deprecated - use the <script> element.

<call-service>

Invokes a service through the Service Engine.

Attributes

Name	Type	Requirements	Description	Note
service-name	constant, \${expression}	required	The name of the service to invoke.	
in-map-name	expression	optional	The name of a field containing a map that will be used for the service's IN parameters.	
include-user-login	constant	optional	Include the current UserLogin entity value in the called service IN parameters.	Defaults to "true".
break-on-error	constant	optional	Halt script execution if the called service returns an error.	Defaults to "true".
error-code	constant	optional	The error code returned by the called service.	Defaults to the enclosing <simple-method> "default-error-code" attribute value.
success-code	constant	optional	The success code returned by the called service.	Defaults to the enclosing <simple-method> "default-success-code" attribute value.
require-new-transaction	constant	optional	Require a new transaction for the called service.	Defaults to "false".
transaction-timeout	constant	optional	The timeout for the new transaction, in seconds.	Defaults to the value set in the called service's definition.

Child Elements

Name	Requirements	
<results-to-map>	optional (one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{})
<result-to-field>	optional (one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{})
<result-to-request>	optional (one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{})
<result-to-session>	optional (one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{})
<result-to-result>	optional (one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{})
<error-prefix>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{})
<error-suffix>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{})
<success-prefix>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{})
<success-suffix>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{})
<message-prefix>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{})
<message-suffix>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{})
<default-message>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{})

<call-service-async>

Invokes an OFBiz service asynchronously.

Attributes

Name	Type	Requirements	Description	Note
service-name	constant, \${expression}	required	The name of the service to invoke.	
in-map-name	expression	optional	The name of a field containing a map that will be used for the service's IN parameters.	
include-user-login	constant	optional	Include the current UserLogin entity value in the called service IN parameters.	Defaults to "true".

<call-simple-method>

Invokes a Mini-language simple method.

Attributes

Name	Type	Requirements	Description	Notes
method-name	constant	required	The name of the <simple-method> element.	
xml-resource	constant	optional	The location of the <simple-method> file.	Defaults to current file.
scope	constant	optional	The memory scope to use: "inline" or "function".	When set to "inline", existing variables can be modified by the called script. When set to "function", existing variables are protected from modification, and the called script returns values via the <field-to-result> element. Defaults to "inline".

Child Elements

Name	Requirements	
<result-to-field>	optional (one or more), used when scope="function"	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{}})

<default-message>

Declares a message to be used when the called service does not return a message.

The message can be included in the element body or in a Java resource.

Attributes

Name	Type	Requirements	Description	Notes
resource	constant	optional	The name of a properties file on the classpath.	
property	constant	optional	The property key.	

<error-prefix>

Declares a message to prepend to the error message.

The message can be included in the element body or in a Java resource.

Attributes

Name	Type	Requirements	Description	Notes
resource	constant	optional	The name of a properties file on the classpath.	
property	constant	optional	The property key.	

<error-suffix>

Declares a message to append to the error message.

The message can be included in the element body or in a Java resource.

Attributes

Name	Type	Requirements	Description	Notes
resource	constant	optional	The name of a properties file on the classpath.	
property	constant	optional	The property key.	

<message-prefix>

Declares a message to prepend to any kind of message.

The message can be included in the element body or in a Java resource.

Attributes

Name	Type	Requirements	Description	Notes
resource	constant	optional	The name of a properties file on the classpath.	
property	constant	optional	The property key.	

<message-suffix>

Declares a message to append to any kind of message.

The message can be included in the element body or in a Java resource.

Attributes

Name	Type	Requirements	Description	Notes
resource	constant	optional	The name of a properties file on the classpath.	
property	constant	optional	The property key.	

<results-to-map>

Copies the called service's result map to the specified field.

Attributes

Name	Type	Requirements	Description	Notes
map-name	expression	required	The name of the target field.	

<result-to-field>

Copies the called service's OUT attribute to the specified field.

Attributes

Name	Type	Requirements	Description	Notes
result-name	expression	required	The name of the called service's OUT attribute.	
field	expression	optional	The name of target field.	Defaults to the result-name attribute value.

<result-to-request>

Copies the called service's OUT attribute to the specified request attribute.

Valid only when the script is run in an event.

Attributes

Name	Type	Requirements	Description	Notes
result-name	expression	required	The name of the called service's OUT attribute.	
request-name	expression	optional	The name of the target request attribute.	Defaults to the result-name attribute value.

<result-to-session>

Copies the called service's OUT attribute to the specified session attribute.

Valid only when the script is run in an event.

Attributes

Name	Type	Requirements	Description	Notes
result-name	expression	required	The name of the called service's OUT attribute.	
session-name	expression	optional	The name of the target session attribute.	Defaults to the result-name attribute value.

<result-to-result>

Copies service OUT attributes from a called service to the calling service's OUT attributes.

This element can also be used to copy the called service OUT attributes to the return result of a simple-method called as a function.

Attributes

Name	Type	Requirements	Description	Notes
result-name	expression	required	The name of the called service's OUT attribute.	
service-result-name	expression	optional	The name of the calling service's OUT attribute (or function return result).	Defaults to the result-name attribute value.

<script>

Executes a script.

The script can be contained in a Java resource, and/or a short inline script can be included in the element body or in the script attribute.

If a Java resource is specified in addition to an inline script, the inline script will be executed after the Java resource script is executed.

Attributes

Name	Type	Requirements	Description	Notes
location	constant	required if <i>script</i> is empty	The location of the script file.	Script functions/methods can be invoked by appending a hash (#) and the function/method name.
script	script	required if <i>location</i> and element body are empty	A short script (scriptlet). Can be used instead of a file.	

<success-prefix>

Declares a message to prepend to the success message.

The message can be included in the element body or in a Java resource.

Attributes

Name	Type	Requirements	Description	Notes
resource	constant	optional	The name of a properties file on the classpath.	
property	constant	optional	The property key.	

<success-suffix>

Declares a message to append to the success message.

The message can be included in the element body or in a Java resource.

Attributes

Name	Type	Requirements	Description	Notes
resource	constant	optional	The name of a properties file on the classpath.	
property	constant	optional	The property key.	

Entity Operations

Operations that interact with a data source. Note that many of these operations behave like assignment operators - since they modify the script's state, but they are listed separately for clarity.

<clear-cache-line>

Clears entity values from the cache.

```
<clear-cache-line entity-name="Party" />
```

Attributes

Name	Type	Requirements	Description	Notes
entity-name	constant, \${expression}	required	The name of the entity.	
map	expression	optional	The name of a field containing a map that will be used to find matching entity values.	If the fields in the map form the full primary key the entry will be removed from the byPrimaryKey cache. If the map exists but the fields do not include a full primary key the entry will be removed from the byAnd cache. If no map name is specified the entry will be removed from the all caches.

<clear-entity-caches>

Removes all entity values from the cache.

This element should be used sparingly because of the performance impact.

<clone-value>

Creates a copy of an entity value.

Attributes

Name	Type	Requirements	Description	Notes
value-field	expression	required	The name of the field that contains the entity value to be copied.	The operation does nothing if the entity value is not found.
new-value-field	expression	required	The name of the field that will contain the new entity value.	

<condition-expr>

Uses an SQL operator to compare an entity field to a variable or constant value.

Attributes

Name	Type	Requirements	Description	Notes
field-name	constant	required	The name of the entity field that will be compared. The l-value.	

operator	constant	required	The SQL operator.	If set to "between" then the r-value must be a collection that has two entries in it. If set to "in" or "not-in" then the r-value must be a collection. Defaults to "equals".
from-field	expression	required if the value attribute is empty	The name of an environment field that the entity field will be compared to. The r-value.	
value	constant, \${expression}	required if the from-field attribute is empty	A constant value that the entity field will be compared to. The r-value.	
ignore-if-null	constant	optional	Ignore the condition if the r-value is null.	Defaults to "false".
ignore-if-empty	constant	optional	Ignore the condition if the r-value is empty.	Defaults to "false".
ignore-case	constant	optional	Perform a case-insensitive comparison.	Defaults to "false".
ignore	constant, \${expression}	optional	Ignore the condition if the r-value evaluates to "true".	Defaults to "false".

<condition-list>

Uses an SQL operator to combine entity find conditions into a single expression.

Attributes

Name	Type	Requirements	Description	Notes
combine	constant	required	The SQL operator.	Defaults to "and".

Child Elements

One or more of the following child elements is required:

Name	
<condition-expr>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<condition-list>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<condition-object>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<condition-object>

Declares an environment field (variable) that contains an entity find condition expression object.

Attributes

Name	Type	Requirements	Description	Notes
field	expression	required	The name of the field that contains an entity find condition expression object	

<create-value>

Creates a new entity value in the data source.

Attributes

Name	Type	Requirements	Description	Notes
value-field	expression	required	The name of the field that contains the entity value to be created.	The operation generates an error and halts script execution if the entity value is not found.
do-cache-clear	constant	optional	Clear the cache.	Defaults to "true".
or-store	constant	optional	Store the entity value if it already exists.	Defaults to "false". If set to "false", an error will result if an instance of the entity value exists in the data source with the same primary key.

<entity-and>

Finds a group of entity values by using a set of entity field name/value pairs that are ANDed together.

The name/value pairs used for the search are defined in child elements.

Attributes

Name	Type	Requirements	Description	Notes
entity-name	constant, \${expression}	required	The name of the entity to search in.	A runtime exception will be thrown if the entity does not exist.
list	expression	required	The name of the field that will contain the result list.	
filter-by-date	constant, \${expression}	optional	Filter the result list by the current date.	The entity must have "fromDate" and "thruDate" fields. Defaults to "false".
distinct	constant, \${expression}	optional	Filter the results so that all of them are unique.	Defaults to "false".
use-cache	constant, \${expression}	optional	Use the entity cache.	Defaults to "false".
delegator-name	constant, \${expression}	optional	Overrides the current delegator by specifying a delegator name.	

Child Elements

Name	Requirements	
<field-map>	required (one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<select-field>	optional (one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<order-by>	optional (one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<limit-range>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<limit-view>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<use-iterator>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<entity-condition>

Finds a group of entity values by using a condition expression.

Attributes

Name	Type	Requirements	Description	Notes
entity-name	constant, \${expression}	required	The name of the entity to search in.	A runtime exception will be thrown if the entity does not exist.
list	expression	required	The name of the field that will contain the result list.	
filter-by-date	constant, \${expression}	optional	Filter the result list by the current date.	The entity must have "fromDate" and "thruDate" fields. Defaults to "false".
distinct	constant, \${expression}	optional	Filter the results so that all of them are unique.	Defaults to "false".
use-cache	constant, \${expression}	optional	Use the entity cache.	Defaults to "false".
delegator-name	constant, \${expression}	optional	Overrides the current delegator by specifying a delegator name.	

Child Elements

One of the following child elements is required:

Name	
<condition-expr>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<condition-list>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<condition-object>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

The following child elements are optional:

Name	Requirements
------	--------------

<having-condition-list>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<select-field>	optional (one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<order-by>	optional (one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<limit-range>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<limit-view>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<use-iterator>	optional (only one)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<entity-count>

Counts a group of entity values that match a condition expression.

Attributes

Name	Type	Requirements	Description	Notes
entity-name	constant, \${expression}	required	The name of the entity to count in.	A runtime exception will be thrown if the entity does not exist.
count-field	expression	required	Name of the field that will contain the count result.	
delegator-name	constant, \${expression}	optional	Overrides the current delegator by specifying a delegator name.	

Child Elements

One of the following child elements is required:

Name	
<condition-expr>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<condition-list>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<condition-object>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

The following child element is optional:

Name	
<having-condition-list>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<entity-data>

Load or assert an entity data XML file.

The operation can be used in two modes: "load" or "assert". The "load" mode will load the XML entity data into the data source. The "assert" mode will compare the XML entity data with the data source - any mismatched or missing data will be logged in the error message list.

Attributes

Name	Type	Requirements	Description	Notes
location	constant, \${expression}	required	The location of an XML file to load in data source mode or to verify in assert mode.	A message will be added to the error message list if the file is not found.
mode	constant	optional	Operation mode: "load" or "assert".	Defaults to "load".
timeout	constant	optional	Override the current transaction to start a new transaction and load the data with a different timeout.	Valid only when mode="load".
error-list-name	constant, \${expression}	optional	The name of the error message list.	Defaults to "error_list". It will be created if it does not exist.
delegator-name	constant, \${expression}	optional	Overrides the current delegator by specifying a delegator name.	

<entity-one>

Finds a single entity value by using the primary key.

The primary key can be specified in the <field-map> child elements, or if they are absent, the primary key will be constructed from matching environment fields.

Attributes

Name	Type	Requirements	Description	Notes
entity-name	constant, \${expression}	required	The name of the entity to search in.	A runtime exception will be thrown if the entity does not exist.
value-field	expression	required	The name of the field that will contain the entity value.	
use-cache	constant, \${expression}	optional	Use the entity cache.	Defaults to "false".
auto-field-map	constant, \${expression}	optional	Look for all primary key field names in the current context as well as in the parameters map.	Defaults to "true".

Child Elements

Name	Requirements	
<field-map>	optional(one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<select-field>	optional (one or more)	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<field-map>

Declares an entity field name/value pair.

Attributes

Name	Type	Requirements	Description	Note
field-name	expression	required	The name of the entity field.	Defaults to from-field attribute value.
from-field	expression	required if the value attribute is empty.	The name of an environment field that contains the entity field value.	
value	constant+expression	required if the from-field attribute is empty.	A constant entity field value.	

<filter-list-by-and>

Filters a list of entity values by the fields in the specified map.

Each entity value in the list will be compared to the specified map, and if the entity value's fields match the map, the entity value will be copied to the target list.

Attributes

Name	Type	Requirements	Description	Note
list	expression	required	The name of the field that contains the list of entity values.	
map	expression	required	The name of the field that contains the map that will be used for the entity field comparison.	
to-list	expression	optional	The name of the field the filtered list will be put into.	Defaults to the value of the list attribute (operation replaces the old list).

<filter-list-by-date>

Filters a list of entity values by their date-range fields.

Each entity value in the list will be compared to the specified date, and if the entity value's date-range fields include the date, the entity value will be copied to the target list.

Attributes

Name	Type	Requirements	Description	Note
list	expression	required	The name of the field that contains the list of entity values.	

to-list	expression	optional	The name of the field the filtered list will be put into.	Defaults to the value of the list attribute (operation replaces the old list).
valid-date	expression	optional	The name of a field that contains the date object to filter by.	The date object must be a java.sql.Timestamp data type. Defaults to now.
from-field-name	constant	optional	The name of the entity field to use as the beginning date.	Defaults to "fromDate".
thru-field-name	constant	optional	The name of the entity field to use as the ending date.	Defaults to "thruDate".

<find-by-and>

Finds a group of entity values by using a set of entity field name/value pairs that are ANDed together.

The name/value pairs used for the search are contained in a map.

Attributes

Name	Type	Requirements	Description	Notes
entity-name	constant, \${expression}	required	The name of the entity to search in.	A runtime exception will be thrown if the entity does not exist.
list	expression	required	The name of the field that will contain the result list.	
map	expression	required	The name of the field containing a map that will be used for the search.	The map name/value pairs will be combined using a boolean AND.
order-by-list	expression	optional	The name of the field containing a list that contains field names that you want the operation to order the results by.	Each entry in the list is a field name. The field name can be preceded by a plus or a minus sign to specify an ascending or descending sort for that field. The default is ascending sort.
fields-to-select-list	expression	optional	The name of a list or set in the method environment that contains field names to select.	
use-cache	constant, \${expression}	optional	Use the entity cache.	Defaults to "false".
use-iterator	constant, \${expression}	optional	Use the EntityListIterator when doing the query.	This is much more efficient for large data sets because the results are read incrementally instead of all at once. The EntityListIterator must be closed when you are finished, but this is done automatically by the iterate operation. Defaults to "false".
delegator-name	constant, \${expression}	optional	Overrides the current delegator by specifying a delegator name.	

<find-by-primary-key>

Returns a single entity value that is retrieved from the data source by using the primary key.

The primary key used for the search is contained in a map.

Attributes

Name	Type	Requirements	Description	Notes
entity-name	constant, \${expression}	optional if the entity name comes from the primary key map	The name of the entity to search in.	If the primary key map is an entity value, then the value's entity name can be used instead of this attribute. A runtime exception will be thrown if the entity does not exist.
value-field	expression	required	The name of the field that will contain the entity value.	
map	expression	required	The name of the field containing a map that will be used for the search.	
fields-to-select-list	expression	optional	The name of a list or set in the method environment that contains field names to select.	
use-cache	constant, \${expression}	optional	Use the entity cache.	Defaults to "false".

delegator-name	constant, \${expression}	optional	Overrides the current delegator by specifying a delegator name.	
----------------	--------------------------	----------	---	--

<get-related>

Finds entity values that are related to an existing entity value.

Attributes

Name	Type	Requirements	Description	Notes
value-field	expression	required	The name of the field containing the entity value.	A runtime exception will be thrown if the entity value does not exist.
relation-name	constant, \${expression}	required	The name of the entity value relation to use.	Script execution will halt if the relation does not exist.
list	expression	required	The name of the field that will contain the result list.	
map	expression	required	The name of the field containing a map that will be used for the search.	The map name/value pairs will be combined using a boolean AND. All entity values will be found if this attribute is empty.
order-by-list	expression	optional	The name of the field containing a list that contains field names that you want the operation to order the results by.	Each entry in the list is a field name. The field name can be preceded by a plus or a minus sign to specify an ascending or descending sort for that field. The default is ascending sort.
use-cache	constant, \${expression}	optional	Use the entity cache.	Defaults to "false".

<get-related-one>

Finds an entity value that is related to an existing entity value.

Attributes

Name	Type	Requirements	Description	Notes
value-field	expression	required	The name of the field containing the entity value.	A runtime exception will be thrown if the entity value does not exist.
relation-name	constant, \${expression}	required	The name of the entity value relation to use.	Script execution will halt if the relation does not exist.
to-value-field	expression	required	The name of the field that will contain the related entity value.	
use-cache	constant, \${expression}	optional	Use the entity cache.	Defaults to "false".

<having-condition-list>

A list of conditions that are combined with a boolean AND or OR that are applied to an entity value group.

This element implements the SQL HAVING clause.

Attributes

Name	Type	Requirements	Description	Notes
combine	constant, \${expression}	optional	The boolean operator used to combine conditions.	Defaults to "and".

Child Elements

One of the following child elements is required:

Name	
<condition-expr>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<condition-list>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<condition-object>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})

<limit-range>

Constrains an entity find result by specifying a starting index and the number of results to include.

Attributes

Name	Type	Requirements	Description	Notes
start	constant, \${expression}	required	A zero-based starting index.	The attribute value must resolve to a non-negative integer.
size	constant, \${expression}	required	The number of results to include.	The attribute value must resolve to a non-negative integer.

<limit-view>

Constrains an entity find result by specifying a view index and a view size.

Example: view-index="3" and view-size="10" returns results 21 to 30.

Attributes

Name	Type	Requirements	Description	Notes
view-index	constant, \${expression}	required	A one-based view index.	The attribute value must resolve to a non-negative integer.
size	constant, \${expression}	required	The number of results to view.	The attribute value must resolve to a non-negative integer.

<make-next-seq-id>

Generates a sequence string and uses it to update an entity value field.

The sequence string is generated by scanning all similar values in the entity and adding an increment amount to the highest sequence value found.

Attributes

Name	Type	Requirements	Description	Notes
value-field	expression	required	The name of the field containing the entity value.	A runtime exception will be thrown if the entity value does not exist.
seq-field-name	constant, \${expression}	required	The name of the entity value field that will receive the sequence string.	
numeric-padding	constant, \${expression}	optional	Left-pad the sequence string with the specified number of zeros.	Defaults to "5".
increment-by	constant, \${expression}	optional	Sequence increment amount.	Defaults to "1".

<make-value>

Creates a new entity value.

The new entity value exists only in memory - it is not persisted in the data source until it is stored.

Attributes

Name	Type	Requirements	Description	Notes
entity-name	constant, \${expression}	required	The name of the entity.	A runtime exception will be thrown if the entity does not exist.
value-field	expression	required	The name of the field that will contain the new entity value.	
map	expression	optional	The name of the field that contains a map that is used to initialize the entity value fields.	Map values whose key matches one of the entity value fields will be copied to the corresponding entity value field. Defaults to parameters.

<order-value-list>

Performs an in-memory sort of a list of entity values.

Attributes

Name	Type	Requirements	Description	Notes
list	expression	required	The name of the field that contains the list of entity values.	The target field will be set to null if the list is not found.
order-by-list	expression	required	The name of the field that contains the list of field names to sort on.	Field names can be preceded by a plus symbol  or a minus symbol  to specify an ascending or descending sort for that field. The default is ascending sort. If the order-by-list is not found the target field will be set to a copy of the source list.
to-list	expression	optional	The name of the field that will contain the sorted list.	Defaults to the field specified in the list attribute.

<refresh-value>

Refresh an entity value from the data source.

Attributes

Name	Type	Requirements	Description	Notes
value-field	expression	required	The name of the field containing the entity value.	A runtime exception will be thrown if the entity value does not exist.
do-cache-clear	constant, \${expression}	optional	Clear the cache.	Defaults to "true".

<remove-by-and>

Removes entity values from the data source.

Uses a map of name/value pairs that will be combined using a boolean AND to constrain the removal.

Attributes

Name	Type	Requirements	Description	Notes
entity-name	constant, \${expression}	required	The name of the entity to search in.	A runtime exception will be thrown if the entity does not exist.
map	expression	required	The name of the field containing a map that will be used for the constraint.	The map name/value pairs will be combined using a boolean AND. If the map is missing or empty, then all entity values will be removed.
do-cache-clear	constant, \${expression}	optional	Clear the cache.	Defaults to "true".

<remove-list>

Removes from the data source all entity values in a list.

If an entity value in the list is a complete primary key, then just that entity value will be removed from the data source.

If an entity value in the list is not a complete primary key, then all matching entity values will be removed from the data source.

Attributes

Name	Type	Requirements	Description	Notes
list	expression	required	The name of the field that contains the list of entity values.	A runtime exception will be thrown if the list does not exist.
do-cache-clear	constant, \${expression}	optional	Clear the cache.	Defaults to "true".

<remove-related>

Removes entity values from the data source that are related to an existing entity value.

For a one-to-one relationship it will remove a single entity value if it exists, and for a one-to-many relationship it will remove all related entity values.

Attributes

Name	Type	Requirements	Description	Notes
value-field	expression	required	The name of the field containing the entity value.	A runtime exception will be thrown if the entity value does not exist.
relation-name	constant, \${expression}	required	The name of the entity value relation to use.	Script execution will halt if the relation does not exist.
do-cache-clear	constant, \${expression}	optional	Clear the cache.	Defaults to "true".

<remove-value>

Removes an entity value from the data source.

Attributes

Name	Type	Requirements	Description	Notes
value-field	expression	required	The name of the field containing the entity value.	A runtime exception will be thrown if the entity value does not exist.
do-cache-clear	constant, \${expression}	optional	Clear the cache.	Defaults to "true".

<select-field>

Declares an entity field to include in the entity operation.

Attributes

Name	Type	Requirements	Description	Notes
field-name	constant, \${expression}	required	The entity field name.	

<sequenced-id>

Generates a unique numeric sequence value.

Sequence values are unique per sequence name.

Attributes

Name	Type	Requirements	Description	Notes
sequence-name	constant, \${expression}	required	The sequence name.	In most cases, this is the name of the entity the sequence value will be used in, but any name can be used.
field	expression	required	The name of the field to set, the target of the assignment.	
stagger-max	constant	optional	Sequence stagger maximum.	If set to "1" the sequence value will be incremented by 1, otherwise the sequence value will be incremented by an amount between 1 and the stagger maximum. Defaults to "1".
get-long-only	constant	optional	Preserve the sequence type - a numeric long.	If set to "false" the sequence value will be converted to a string. Defaults to "false".

<set-nonpk-fields>

Copies non-PK fields from a map to an entity value.

Attributes

Name	Type	Requirements	Description	Notes
value-field	expression	required	The name of the field containing the entity value.	A runtime exception will be thrown if the entity value does not exist.
map	expression	required	The name of the field that contains a map that is used to initialize the entity value non-pk fields.	Map values whose key matches one of the entity value fields will be copied to the entity value. A runtime exception will be thrown if the map does not exist.
set-if-null	constant, \${expression}	optional	Set entity value fields that are null or empty.	Defaults to "true".

<set-pk-fields>

Copies PK fields from a map to an entity value.

Attributes

Name	Type	Requirements	Description	Notes
value-field	expression	required	The name of the field containing the entity value.	A runtime exception will be thrown if the entity value does not exist.
map	expression	required	The name of the field that contains a map that is used to initialize the entity value pk fields.	Map values whose key matches one of the entity value fields will be copied to the entity value. A runtime exception will be thrown if the map does not exist.
set-if-null	constant, \${expression}	optional	Set entity value fields that are null or empty.	Defaults to "true".

<store-list>

Stores in the data source all entity values in a list.

Entity values that do not exist in the data source will be added to the data source.

Attributes

Name	Type	Requirements	Description	Notes
list	expression	required	The name of the field that contains the list of entity values.	A runtime exception will be thrown if the list does not exist.
do-cache-clear	constant, \${expression}	optional	Clear the cache.	Defaults to "true".

<store-value>

Stores an entity value in the data source.

Attributes

Name	Type	Requirements	Description	Notes
value-field	expression	required	The name of the field containing the entity value.	A runtime exception will be thrown if the entity value does not exist. Script execution will halt if the entity value does not exist in the data source.
do-cache-clear	constant, \${expression}	optional	Clear the cache.	Defaults to "true".

<transaction-begin>

Begins a transaction if one is not already in place.

Note that all simple-methods are executed inside a transaction unless the simple-method is flagged to not use a transaction.

Attributes

Name	Type	Requirements	Description	Notes
began-transaction-name	expression	optional	The name of the field that contains a Boolean value specifying whether or not a transaction was begun in the current transaction demarcation.	Defaults to "beganTransaction".

<transaction-commit>

Commits a transaction that was begun with the <transaction-begin> element.

Operation does nothing if no transaction was begun.

Attributes

Name	Type	Requirements	Description	Notes
began-transaction-name	expression	optional	The name of the field that contains a Boolean value specifying whether or not a transaction was begun in the current transaction demarcation.	Defaults to "beganTransaction".

<transaction-rollback>

Rolls back a transaction that was begun with the <transaction-begin> element.

If no transaction was begun with the transaction-begin element, a "set rollback only" operation will be performed - which will force the transaction to rollback regardless of which method or object is responsible for beginning and ending the transaction.

Attributes

Name	Type	Requirements	Description	Notes
began-transaction-name	expression	optional	The name of the field that contains a Boolean value specifying whether or not a transaction was begun in the current transaction demarcation.	Defaults to "beganTransaction".

<use-iterator>

Use the EntityListIterator for the entity find operation.

Logging

<log>

Outputs a log a message.

Attributes

Name	Type	Requirements	Description	Notes
level	constant	required	The logging level to use.	Must be one of the following: "verbose", "timing", "info", "important", "warning", "error", "fatal", "always". Defaults to "info".
message	constant+expression	required	The message to log.	

<trace>

Logs detailed information on the internal state of its sub-elements, using the specified log level.

Attributes

Name	Type	Requirements	Description	Notes
------	------	--------------	-------------	-------

level	constant	required	The logging level to use.	Must be one of the following: "verbose", "timing", "info", "important", "warning", "error", "fatal", "always". Defaults to "info".
-------	----------	----------	---------------------------	--

Child Elements

Name	Requirements	
(Any block of code)	optional	

Calculate Operations

`<calculate>`

Performs an arithmetic calculation and puts the result in the specified field.

Attributes

Name	Type	Requirements	Description	Notes
field	expression	required	The name of the field to set, the target of the assignment.	
rounding-mode	constant, \${expression}	optional	Rounding mode for the calculation, primarily for divide operation.	Defaults to "HalfEven".
decimal-scale	constant, \${expression}	optional	Initial scale (decimals) to use.	Defaults to "2".
decimal-format	constant, \${expression}	optional	Decimal format to use for conversion to string.	
type	constant	optional	Data type of the calculation result.	Defaults to "BigDecimal".

Child Elements

One or more of the following child elements is optional:

Name	
<code><calcop></code>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})
<code><number></code>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})



Deprecated - use the `<set>` element.

`<calcop>`

A basic arithmetic operation.

Attributes

Name	Type	Requirements	Description	Notes
operator	constant	required	The arithmetic operator.	Defaults to "add".
field	expression	optional	The name of the field to use.	

Child Elements

One or more of the following child elements is optional:

Name	
<code><calcop></code>	
<code><number></code>	(?) (https://cwiki.apache.org/OFBADMIN/mini-language-reference.html#Mini-languageReference-{{{}}}{}})



Deprecated - use the `<set>` element.

`<number>`

Declares a numeric constant to be used in the calculation.

Attributes

Name	Type	Requirements	Description	Notes
value	constant, <code>\${expression}</code>	required	A numeric constant.	



Deprecated - use the `<set>` element.

Map Processor

`<compare>`

`<compare-field>`

`<convert>`

`<copy>`

`<fail-message>`

`<fail-property>`

`<make-in-string>`

`<not-empty>`

`<process>`

`<regexp>`

`<simple-map-processor>`

`<validate-method>`

Mini-language Reference