

Review M: DITAVAL elements

Table of contents

1 DITAVAL elements.....	3
1.1 <alt-text>.....	3
1.2 <endflag>.....	3
1.3 <prop>.....	4
1.4 <revprop>.....	7
1.5 <startflag>.....	9
1.6 <style-conflict>.....	10
1.7 <val>.....	11
A Aggregated RFC-2119 statements.....	13
Index.....	14

1 DITAVAL elements

A **DITAVAL document identifies content that is filtered and flagged at rendering time**. The **DITAVAL document** has an extension of `.ditaval`.

1.1 <alt-text>

The `<alt-text>` element in a DITAVAL document specifies alternate text for an image that is used to flag content. If an image is not specified, the text is used to mark the flagged content.

Rendering expectations

If no alternate text is specified, processors can provide default alternate text to indicate the start and end point of the flagged content.

Example

The following code sample shows a DITAVAL document that is used to render icons before content that is specific to particular audiences. The `<alt-text>` element provides alternate text for the icons:

```
<val>
  <prop action="flag" att="audience" val="novice">
    <startflag imageref="novice-icon.gif">
      <alt-text>Novice</alt-text>
    </startflag>
  </prop>
  <prop action="flag" att="audience" val="expert">
    <startflag imageref="expert-icon.gif">
      <alt-text>Expert</alt-text>
    </startflag>
  </prop>
</val>
```

1.2 <endflag>

The `<endflag>` element in a DITAVAL document specifies information that identifies the end of flagged content. The information can be an image, alternate text, or both.

Usage information

If the `<endflag>` element does not specify an image or provide alternate text, the element has no defined purpose.

Rendering expectations

Processors treat the information provided by the `<endflag>` element in the following way:

- If an image is specified, the image is used as a flag to identify the end of the flagged content. If the `<alt-text>` element contains content, the content is used as alternate text for the image.
- If alternate text is specified but the `<endflag>` element does not specify an image, the alternate text is used to indicate the end of the flagged content.

Attributes

The following attribute is available on this element:

@imageref

Specifies a URI reference to the image, using the same syntax as the @href attribute. See [The href attribute](#) for information on supported values and processing implications.

Example

The following code sample shows a DITAVAL document that is used to flag content that applies to administrators. The <startflag> and <endflag> elements provide text that is used to indicate the start and end point of the flagged content.

```
<val>
  <prop action="flag" att="audience" val="administrator">
    <startflag>
      <alt-text>Administrator content</alt-text>
    </startflag>
    <endflag>
      <alt-text>End of administrator content</alt-text>
    </endflag>
  </prop>
</val>
```

1.3 <prop>

The <prop> element in a DITAVAL document specifies filtering or flagging actions that occur when rendering. The actions target the @props attribute or specializations of @props, including @audience, @deliveryTarget, @otherprops, @platform, and @product.

Usage information

The following table lists the functions **that are performed** by the <prop> element in a DITAVAL document:

Markup	Result
A <prop> element that specifies both an @att and a @val attribute	Specifies an action (exclude, flag, include, or pass through) for the attribute or attribute group with the specified value
A <prop> element that specifies only an @att attribute	Sets a default action for the specified attribute or attribute group
A <prop> element without an @att and @val attribute	Sets a default action for all conditional-processing attributes not explicitly specified in the DITAVAL document

Rendering expectations

001 (13)

For the @color and @backcolor attributes on <prop> and <revprop>, processors **SHOULD** support **at least** the following values:

- The color names listed under the heading "<color>" in [the XSL version 1.1 specification](#)
- The associated hex code

For the @style attribute on <rev> and <revprop>, processors **SHOULD** support the following tokens:

- bold
- double-underline
- italics

- overline
- underline

In addition, processors **MAY** support proprietary tokens for the `@style` attribute. Such tokens **SHOULD** have a processor-specific prefix to identify them as proprietary. If a processor encounters an unsupported style token, it **MAY** issue a warning, and it **MAY** render content that is flagged with such a style token by using some default formatting.

Processing expectations

002 (13)

The following markup in a DITAVAL document is an error condition:

- More than one `<prop>` element with no `@att` attribute
- More than one `<prop>` element with the same `@att` attribute and no value
- More than one `<prop>` element with the same `@att` attribute and same `@value`

Processors **MAY** provide an error or warning message for these error conditions.

The following list outlines how processors apply `@outputclass` flags:

- If one or more DITAVAL properties apply `@outputclass` flags to the same element, and the element already specifies one or more values for the `@outputclass` attribute, processors treat the element as if **the tokens for the `@outputclass` attribute that were provided in the DITAVAL document** are specified first.
- If two or more DITAVAL properties apply `@outputclass` flags to the same element, processors treat the element as if each value was specified for the `@outputclass` attribute. The order of the **tokens for the `@outputclass` attribute that were provided in the DITAVAL document** is undefined.

Attributes

The following attributes are available on this element:

@action (REQUIRED)

Specifies the action to be taken. The following values are supported:

exclude

Indicates that the content is excluded from the output, if all values for the specified attribute are excluded.

flag

Indicates that the content is included in the output and flagged, if the content has not been excluded.

include

Indicates that the content is included in the output. This is the default behavior, unless otherwise set.

passthrough

Indicates that the content is included in the output and that the attribute value is preserved. This enables further processing by a runtime engine. The attribute value is preserved in the syntax that is required by the runtime engine.

@att

Specifies the conditional-processing attribute that is targeted. The value is the literal attribute name or the name of a group within one of those attributes, with the group name specified using the generalized attribute syntax. If the @att attribute is absent, then the <prop> element declares a default behavior for anything not explicitly specified in the DITAVAL document.

@backcolor

(If the @action attribute is set to "flag") Specifies the background color for flagged text. Colors can be entered by name or hex code. When images are flagged, the background color is rendered as a thick border. If the @action attribute is not set to "flag", this attribute is ignored.

@color

(If the @action attribute is set to "flag") Specifies the color for flagged text. Colors can be entered by name or hex code. When images are flagged, the color is rendered as a thin border. If the @action attribute is not set to "flag", this attribute is ignored.

@outputclass

(If the @action attribute is set to "flag") Specifies a value for the @outputclass attribute. The flagged element is treated as if the specified @outputclass value was specified on that element. **If the @action attribute is not set to "flag", this attribute is ignored.**

@style

(If the @action attribute is set to "flag") Specifies the formatting to use for flagged text. This attribute can contain multiple space-delimited tokens. If the @action attribute is not set to "flag" this attribute is ignored.

@val

Specifies the attribute value to be acted upon. If the @val attribute is absent, then the <prop> element declares a default behavior for any value in the specified attribute.

Example

The following code sample shows a DITAVAL document that contains three <prop> elements:

```
<?xml version="1.0" encoding="UTF-8"?>
<val>
  <prop action="exclude"/>
  <prop action="passthrough" att="otherprops"/>
  <prop action="include" att="product" val="base-product"/>
</val>
```

The following list outlines the actions that the DITAVAL document specifies:

1. Sets a default action of "exclude". With the exception of the other conditions **that are** specified in the **above** DITAVAL document, the content of any element that specifies a conditional-processing attribute is excluded from the rendered output.
2. Sets a default action of "passthrough" for the @otherprops attribute. The content of any element that specifies the @otherprops attribute is included in the output. In addition, the value for the @otherprops attribute is preserved in the rendered output, if supported by the output format.
3. Sets an action of "include" for any element that specifies a value of "base-product" for the @product attribute. The content of any element that specifies a value of "base-product" for the @product attribute is included in the rendered output.

When a DITA map is processed using the above DITAVAL document, the following DITA elements are excluded:

1. **Any element for which the @audience, @deliveryTarget, @platform, and @props attributes (or specializations of @props) specify a non-null value.**

2. Any element for which the `@product` attribute specifies a value that is not equal to "base-product".

All other content is included.

1.4 <revprop>

The `<revprop>` element in a DITAVAL document identifies a value of the `@rev` attribute for flagging. Unlike the conditional processing attributes, which can be used for both filtering and flagging, the `@rev` attribute can only be used for flagging.

Usage information

Neither the `<reprop>` element or the `@rev` attribute are designed to be used for version control.

Rendering expectations

If no alternate text is specified, processors can provide default alternate text to indicate the start and end point of the flagged content.

003 (13)

For the `@color` and `@backcolor` attributes on `<prop>` and `<revprop>`, processors **SHOULD** support **at least** the following values:

- The color names listed under the heading "`<color>`" in [the XSL version 1.1 specification](#)
- The associated hex code

For the `@style` attribute on `<rev>` and `<revprop>`, processors **SHOULD** support the following tokens:

- bold
- double-underline
- italics
- overline
- underline

In addition, processors **MAY** support proprietary tokens for the `@style` attribute. Such tokens **SHOULD** have a processor-specific prefix to identify them as proprietary. If a processor encounters an unsupported style token, it **MAY** issue a warning, and it **MAY** render content that is flagged with such a style token by using some default formatting.

Processing expectations

004 (13)

It is an error to include more than one `<revprop>` element with the same `@val` attribute. Recovery from this error is implementation dependent. In such cases processors **MAY** provide an error or warning message.

The following list outlines how processors apply `@outputclass` flags:

- If one or more DITAVAL properties apply `@outputclass` flags to the same element, and the element already specifies one or more values for the `@outputclass` attribute, processors treat the element as if **the tokens for the `@outputclass` attribute that were provided in the DITAVAL document** are specified first.

- If two or more DITAVAL properties apply `@outputclass` flags to the same element, processors treat the element as if each value was specified for the `@outputclass` attribute. The order of the **tokens for the `@outputclass` attribute that were provided in the DITAVAL document is undefined.**

Attributes

The following attributes are available on this element:

@action (REQUIRED)

Specifies the action to be taken. The following values are supported:

flag

Indicates that the content is included in the output and flagged, if the content has not been excluded.

include

Indicates that the content is included in the output and not flagged. This is the default behavior, unless otherwise set.

passthrough

Indicates that the content is included in the output and that the attribute value is preserved. This enables further processing by a runtime engine. The attribute value is preserved in the syntax that is required by the runtime engine.

@backcolor

(If the `@action` attribute is set to "flag") Specifies the background color for flagged text. Colors can be entered by name or **hex** code. When images are flagged, the background color is rendered as a thick border. If the `@action` attribute is not set to "flag", this attribute is ignored.

@changebar

(If the `@action` attribute is set to "flag") Specifies a color, style, or character to be used for rendering a change bar. If the `@action` attribute is not set to "flag", this attribute is ignored.

@color

(If the `@action` attribute is set to "flag") Specifies the color for flagged text. Colors can be entered by name or **hex** code. When images are flagged, the color is rendered as a thin border. If the `@action` attribute is not set to "flag", this attribute is ignored.

@outputclass

(If the `@action` attribute is set to "flag") Specifies a value for the `@outputclass` attribute. The flagged element is treated as if the specified `@outputclass` value was specified on that element. **If the `@action` attribute is not set to "flag", this attribute is ignored.**

@style

(If the `@action` attribute is set to "flag") Specifies the formatting to use for flagged text. This attribute can contain multiple space-delimited tokens. If the `@action` attribute is not set to "flag" this attribute is ignored.

@val

Specifies the value of the `@rev` attribute. If the `@val` attribute is not specified, then the `<revprop>` element declares a default behavior for any instance of the `@rev` attribute.

Example

The following code sample shows how the `<revprop>` element can be used to flag content that has been marked with the `@rev` attribute. Elements that specify `rev="edits"` are rendered in red text, and glyphs mark the start and end points of the flagged revision. Alternate text is provided.

```
<val>
  <revprop action="flag" color="red" val="edits">
    <startflag imageref="start-glyph.png">
      <alt-text>Start of revision</alt-text>
    </startflag>
    <endflag imageref="end-glyph.png">
      <alt-text>End of revision</alt-text>
    </endflag>
  </revprop>
</val>
```

Related concepts

[Flagging based on metadata attributes](#)

1.5 <startflag>

The `<startflag>` element in a DITAVAL document specifies information that identifies the beginning of flagged content. The information can be an image, alternate text, or both.

Usage information

If the `<startflag>` element does not specify an image or provide alternate text, the element has no defined purpose.

Rendering expectations

Processors treat the information provided by the `<startflag>` element in the following way:

- If an image is specified, the image is used as a flag to identify the beginning of the flagged content. If the `<alt-text>` element contains content, the content is used as alternate text for the image.
- If alternate text is specified but the `<startflag>` element does not specify an image, the alternate text is used to indicate the beginning of the flagged content.

Attributes

The following attribute is available on this element:

@imageref

Specifies a URI reference to the image, using the same syntax as the `@href` attribute. See [The href attribute](#) for information on supported values and processing implications.

Example

The following code sample shows a DITAVAL document that is used to render icons before content that is specific to a particular operating system. The `<startflag>` elements specify the icons, and the `<alt-text>` elements **specify** alternate text.

```
<val>
  <prop action="flag" att="platform" val="linux">
    <startflag imageref="linux-icon.gif">
      <alt-text>Linux</alt-text>
    </startflag>
  </prop>
</val>
```

```

    </startflag>
  </prop>
  <prop action="flag" att="platform" val="mac">
    <startflag imageref="mac-icon.gif">
      <alt-text>Macintosh</alt-text>
    </startflag>
  </prop>
  <prop action="flag" att="platform" val="windows">
    <startflag imageref="windows-icon.gif">
      <alt-text>Windows</alt-text>
    </startflag>
  </prop>
</val>

```

1.6 <style-conflict>

The <style-conflict> element in a DITAVAL document declares the behavior to be used when one or more flagging methods collide on **the same element**.

Rendering expectations

The following table details how conflicts are resolved when different flagging methods are specified for the same element:

Flagging method	Conflict behavior
background	Use the color specified by the @background-conflict-color attribute on the <style-conflict> element. If no background conflict color is specified, use a default color that is appropriate for the output format.
changebar	Add all change bars that apply.
color	Use the color specified by the @foreground-conflict-color attribute on the <style-conflict> element. If no foreground conflict color is specified, use a default color that is appropriate for the output format.
style	Add all font styles that apply. If two different kinds of underline are used, default to the heaviest (double underline) and use the color that is specified by the @foreground-conflict-color attribute. If no foreground conflict color is specified, use a default color that is appropriate for the output format.
<endflag>	Add all flags that apply.
<startflag>	Add all flags that apply.

Attributes

The following attributes are available on this element:

@background-conflict-color

Specifies the color to be used when more than one background color applies to a single element. Colors can be entered by name or code.

@foreground-conflict-color

Specifies the color to be used when more than one color applies to a single element. Colors can be entered by name or code.

Example

The following code sample shows a DITAVAL document that specifies that a background color of "#ffffb3" is used when there are conflicts:

```
<?xml version="1.0" encoding="UTF-8"?>
<val>
  <style-conflict background-conflict-color="#ffffb3"/>
  <prop action="flag" att="platform" val="dita" bgcolor="#ccffb3"/>
  <prop action="flag" att="platform" val="lwdita" bgcolor="#ffe6ff"/>
</val>
```

Any element that specifies a value of "dita lwdita" or "lwdita dita" is rendered with a light-yellow background color.

1.7 <val>

The <val> element is the root element of a DITAVAL document.

Processing expectations

For information about processing DITAVAL documents, including how to filter or flag elements with multiple property attributes or multiple properties within a single attribute, see [Conditional processing](#).

Example

This section contains examples of DITAVAL documents and how they can be used.

Figure 1: Sample DITAVAL document

The following code sample shows a DITAVAL document that excludes certain content, flags certain content, flags certain revisions, and provides a background color for when there are style conflicts:

```
<val>
  <style-conflict background-conflict-color="red"/>
  <prop action="exclude" att="audience" val="internal-test"/>
  <prop action="flag" att="product" val="YourProd" bgcolor="purple"/>
  <prop action="flag" att="product" bgcolor="blue"
    color="yellow" style="underline" val="MyProd">
    <startflag imageref="startflag.jpg">
      <alt-text>This is the start of my product info</alt-text>
    </startflag>
    <endflag imageref="endflag.jpg">
      <alt-text>This is the end of my product info</alt-text>
    </endflag>
  </prop>
  <revprop action="flag" val="1.2"/>
</val>
```

This sample DITAVAL document performs the following actions:

- Elements that specify audience="internal-test" are **excluded**.
- Elements that specify product="YourProd" are rendered with a background color of purple.
- Elements with product="MyProd" get the following actions:
 - The image startflag.jpg is placed at the start of the element.
 - The image endflag.jpg is placed at the end of the element.
 - The element is rendered with a background color of blue.
 - The text in the element is rendered in yellow, and the text is underlined.
- Elements marked with rev="1.2" are flagged with the default revision flags, which are implementation dependent.

- When there are conflicts, for example, if an element is marked with `product="MyProd YourProd"`, it will be flagged with a background color of red.

Figure 2: DITAVAL document that overrides the default "include" action

The following code sample shows a DITAVAL document that sets a default value of "exclude" for every conditional-processing attribute. That default value is then overridden by the `<prop>` elements with a value of "include."

```
<val>
  <prop action="exclude"/>
  <prop action="include" att="audience" val="everybody"/>
  <prop action="include" att="audience" val="novice"/>
  <prop action="include" att="product" val="productA"/>
  <prop action="include" att="product" val="productB"/>
</val>
```

This DITAVAL document performs the following actions:

- The first `<prop>` element does not specify an attribute, which sets a default action of "exclude" for every conditional-processing attribute. This means that, by default, any property value not otherwise defined in this document evaluates to "exclude". Note that this same behavior can be limited to a single attribute. The following `<prop>` element sets a default action of "exclude" for all properties specified on the `@platform` attribute: `<prop action="exclude" att="platform"/>`
- The second and third `<prop>` elements set an action of "include" for two values on the `@audience` attribute. All other values on the `@audience` attribute still evaluate to "exclude".
- The fourth and fifth `<prop>` elements set an action of "include" for two values on the `@product` attribute. All other values on the `@product` attribute still evaluate to "exclude".

Related concepts

[Filtering based on metadata attributes](#)

[Flagging based on metadata attributes](#)

[Examples of conditional processing](#)

A Aggregated RFC-2119 statements

This appendix contains all the normative statements from the DITA 2.0 specification. They are aggregated here for convenience in this non-normative appendix.

Item	Conformance statement
001 (4)	<p>For the <code>@color</code> and <code>@backcolor</code> attributes on <code><prop></code> and <code><revprop></code>, processors SHOULD support at least the following values:</p> <ul style="list-style-type: none">• The color names listed under the heading "<color>" in the XSL version 1.1 specification• The associated hex code <p>For the <code>@style</code> attribute on <code><rev></code> and <code><revprop></code>, processors SHOULD support the following tokens:</p> <ul style="list-style-type: none">• bold• double-underline• italics• overline• underline <p>In addition, processors MAY support proprietary tokens for the <code>@style</code> attribute. Such tokens SHOULD have a processor-specific prefix to identify them as proprietary. If a processor encounters an unsupported style token, it MAY issue a warning, and it MAY render content that is flagged with such a style token by using some default formatting.</p>
002 (5)	<p>The following markup in a DITAVAL document is an error condition:</p> <ul style="list-style-type: none">• More than one <code><prop></code> element with no <code>@att</code> attribute• More than one <code><prop></code> element with the same <code>@att</code> attribute and no value• More than one <code><prop></code> element with the same <code>@att</code> attribute and same <code>@value</code> <p>Processors MAY provide an error or warning message for these error conditions.</p>
003 (7)	<p>For the <code>@color</code> and <code>@backcolor</code> attributes on <code><prop></code> and <code><revprop></code>, processors SHOULD support at least the following values:</p> <ul style="list-style-type: none">• The color names listed under the heading "<color>" in the XSL version 1.1 specification• The associated hex code <p>For the <code>@style</code> attribute on <code><rev></code> and <code><revprop></code>, processors SHOULD support the following tokens:</p> <ul style="list-style-type: none">• bold• double-underline• italics• overline• underline <p>In addition, processors MAY support proprietary tokens for the <code>@style</code> attribute. Such tokens SHOULD have a processor-specific prefix to identify them as proprietary. If a processor encounters an unsupported style token, it MAY issue a warning, and it MAY render content that is flagged with such a style token by using some default formatting.</p>
004 (7)	<p>It is an error to include more than one <code><revprop></code> element with the same <code>@val</code> attribute. Recovery from this error is implementation dependent. In such cases processors MAY provide an error or warning message.</p>

Index

D

DITAVAL

elements

- `<alt-text>` [3](#)
- `<endflag>` [3](#)
- `<prop>` [4](#)
- `<revprop>` [7](#)
- `<startflag>` [9](#)
- `<style-conflict>` [10](#)
- `<val>` [11](#)

E

element groups

- DITAVAL [3](#)

elements

DITAVAL

- `<alt-text>` [3](#)
- `<endflag>` [3](#)
- `<prop>` [4](#)
- `<revprop>` [7](#)
- `<startflag>` [9](#)
- `<style-conflict>` [10](#)
- `<val>` [11](#)

F

filtering [3](#)

flagging [3](#), [7](#)

- alternate text [3](#)

R

revisions [3](#)