Touchup processing instructions

Introduction

Arbortext Editor provides several Touchup processing instructions to “tweak” the formatting a document.

Each Touchup PI has a default behavior, and some have one or more attributes to customize their formatting, as described in the following table in **Table 97**.

Table 97 Touchup PIs

PIs	Tag	Attribute(s)	Type	Notes
<_deepcontentsplitting>	pair	state	on | off	Enables deep content splitting for a portion of a document; see **Deep content splitting Touchup PI** on page 716
		overfull	SIZE/DISTANCE	
		underfull	SIZE/DISTANCE	
<_eos-space>	singleton	—	—	Specifies an end-of-sentence space
<_font>	pair	†	—	Specifies font and highlighting properties; see **Touchup PIs with `e-i-cs`** below
<_hardspace>	singleton	—	—	Outputs an unbreakable word space
<_hyphen-point>	singleton	—	—	Marks an available hyphenation point
<_interword-space>	singleton	—	—	Outputs a word space
<_kern>	singleton	amount	SIZE/DISTANCE	Outputs the specified amount of horizontal space (the default is 12pt)
<_newline>	singleton	—	—	Forces a new line
<_newpage>	singleton	—	—	Forces a new page
<_newcolumn>	singleton	—	—	Forces a new column
<_nolinebreak>	pair	—	—	Surrounds content that should be kept together on one line

†Refer to *`Arbortext-path`*`\lib\_font-eic.fos`.
‡This PI is a user-defined tag (page 726) based on the <_touchup> PI. See **Keeps Touchup PIs** below.
§Refer to *`Arbortext-path`*`\lib\_touchup-eic.fos`.

continued . . .

TIP

Full Menus must be enabled for Touchup PIs to be displayed in the Format▸Touchup menu.

TIP

For best results, insert Touchup tag pairs around text rather than tags. For example:

XML fragment

```
<title><_font>PTC</_font></title>
```

instead of:

XML fragment

```
<_font><title>PTC</title></_font>
```

TIP

Other SGML/XML processing software may not recognize Touchup PIs. See **Processing instruction-related ACL** on page 738 for information on saving a document without processing instructions.

Table 97 Touchup PIs ***(cont'd)***

PIs	Tag	Attribute(s)	Type	Notes
<_nocolumnbreak>‡	pair	Same as <_touchup>	—	Surrounds content that should be kept together on one page
<_nopagebreak>‡	pair	Same as <_touchup>	—	Surrounds content that should be kept together on one page
<_touchup>	pair	§	—	Surrounds content to be formatted as specified in the tag's attributes; see **Touchup PIs with `e-i-cs`** below

†Refer to `*Arbortext-path*\lib\_font-eic.fos`.
‡This PI is a user-defined tag (page 726) based on the <_touchup> PI. See **`Keeps` Touchup PIs** below.
§Refer to `*Arbortext-path*\lib\_touchup-eic.fos`.

NOTE: Touchup PIs related to printed pages, such as <_nopagebreak>, do not apply to Edit window display.

NOTE: Touchup processing instructions can be dis-allowed in the doctype configuration file with the *allowTouchup* attribute on <Options>.

Touch PIs in generated text

As detailed in **Table 98**, some Touchup PIs that are saved to a string variable as part of #CONTENT are removed when the string is output from a `usetext source`. Other PIs s are treated differently when output from a `usetext source`. The rest function as usual. See **Figure 392** and **Figure 393** for examples.

Table 98 Touchup PIs in generated text

Touchup PIs saved with #CONTENT	PIs in `usetext source`
<_eos-space>	not removed
<_font>	removed
<_hardspace>	softened†
<_hyphen-point>	not removed‡
<_kern>	not removed
<_interword-space>	not removed
<_newcolumn>	removed
<_newline>	removed
<_newpage>	removed
<_nolinebreak>	removed
<_nopagebreak>	not removed
<_texmac>	not removed
<_texmacpair>	not removed
<_touchup>	not removed

†<_hardspace> is converted into a regular space in generated text.
‡<_hyphen-point> takes effect whether hyphenation is enabled in the FOSI or not.

Touchup PIs with `e-i-cs`

Two Touchup PIs may have one or more `e-i-cs` with `gitype="pi"` (which must be coded in the tagged FOSI editor). These `e-i-cs` may disable, replace, or add to default formatting for that PI.

- ▷ <_font> PI, page 704
- ▷ <_touchup> PI, page 708

<_font> PI

Default formatting is in *Arbortext-path*\lib_font-eic.fos, which is shown below.

Figure 387 Default formatting for <_font> PI

```
<!-- $Id: $ FOSI fragment supporting _font ATI tags-->

<!--NOTE: fillval used for direct mapping of choices; specval for mapping
of atitag.cf choice values to FOSI choice values-->

<e-i-c gi="font" gitype="pi">
<charlist inherit="1"></charlist>

<!-- _FONT TAG specval -->

<!--_font Posture choice mapping-->
<att>
<specval attname="Posture" attval="backslant-italic">
<charsubset><font posture="bsital"></charsubset></att>
<att>
<specval attname="Posture" attval="italic">
<charsubset><font posture="italic"></charsubset></att>
<att>
<specval attname="Posture" attval="backslant-oblique">
<charsubset><font posture="bsobl"></charsubset></att>
<att>
<specval attname="Posture" attval="oblique">
<charsubset><font posture="oblique"></charsubset></att>
<att>
<specval attname="Posture" attval="upright">
<charsubset><font posture=""></charsubset></att>

<!--_font Weight choice mapping-->
<att>
<specval attname="Weight" attval="ultrabold">
<charsubset><font weight="ultbold"></charsubset></att>
<att>
<specval attname="Weight" attval="extrabold">
<charsubset><font weight="exbold"></charsubset></att>
<att>
<specval attname="Weight" attval="bold">
<charsubset><font weight="bold"></charsubset></att>
<att>
<specval attname="Weight" attval="semibold">
<charsubset><font weight="sembold"></charsubset></att>
<att>
<specval attname="Weight" attval="medium">
<charsubset><font weight="medium"></charsubset></att>
<att>
```

```
<specval attname="Weight" attval="semilight">
<charsubset><font weight="semlight"></charsubset></att>
<att>
<specval attname="Weight" attval="light">
<charsubset><font weight="light"></charsubset></att>
<att>
<specval attname="Weight" attval="extralight">
<charsubset><font weight="exlight"></charsubset></att>
<att>
<specval attname="Weight" attval="ultralight">
<charsubset><font weight="ultlight"></charsubset></att>

<!--_font FontStyle choice mapping-->
<att>
<specval attname="FontStyle" attval="monospaced-sans-serif">
<charsubset><font style="monosans"></charsubset></att>
<att>
<specval attname="FontStyle" attval="monospaced-serifed">
<charsubset><font style="monoser"></charsubset></att>
<att>
<specval attname="FontStyle" attval="sans-serif">
<charsubset><font style="sanserif"></charsubset></att>
<att>
<specval attname="FontStyle" attval="serifed">
<charsubset><font style="serif"></charsubset></att>

<!--_font Width choice mapping-->
<att>
<specval attname="Width" ="ultracond">
<charsubset><font width="ultcond"></charsubset></att>
<att>
<specval attname="Width" attval="extracond">
<charsubset><font width="excond"></charsubset></att>
<att>
<specval attname="Width" attval="cond">
<charsubset><font width="cond"></charsubset></att>
<att>
<specval attname="Width" attval="semicond">
<charsubset><font width="semcond"></charsubset></att>
<att>
<specval attname="Width" attval="regular">
<charsubset><font width="regular"></charsubset></att>
<att>
<specval attname="Width" attval="semiexp">
<charsubset><font width="semexp"></charsubset></att>
<att>
<specval attname="Width" attval="exp">
<charsubset><font width="exp"></charsubset></att>
<att>
<specval attname="Width" attval="extraexp">
<charsubset><font width="exexp"></charsubset></att>
```

```
<att>
<specval attname="Width" attval="ultraexp">
<charsubset><font width="ultexp"></charsubset></att>

<!--_font Underline choice mapping-->
<att>
<specval attname="Underline" attval="no">
<charsubset><highlt scoring="0"></charsubset></att>
<! Note:  yes is here for backward compatility
   As of olive the touchup->font dialog doesn't support this choice. -->
<att>
<specval attname="Underline" attval="yes">
<charsubset><highlt scoring="1" scorestyle="single"></charsubset></att>
<att logic="and">
<specval attname="Underline" attval="#NE#\no\">
<specval attname="Underline" attval="#NE#\yes\">
<specval attname="Underline" attval="#NE#\inherit\">
<fillval attname="Underline" fillcat="highlt" fillchar="scorestyle">
<charsubset><highlt scoring="1"></charsubset></att>
<att>
<specval attname="ScoreSpace" attval="yes">
<charsubset><highlt scorespc="1"></charsubset></att>
<att>
<specval attname="ScoreSpace" attval="no">
<charsubset><highlt scorespc="0"></charsubset></att>

<!--ScoreCharOn is no longer supported in touchup->font dialog,
    but it is retained to support old _font PIs.
    Formerly it specified strikethrough

    The offset value (xscoreoff) should be coordinated with the value in
    edit/src/reseic.c for _cellfont. -->
<att logic="and">
<specval attname="ScoreCharOn" attval="yes">
<specval attname="ScoreSpace" attval="#NE#\no\">
<charsubset><highlt xscoring="1" xscorespc="1" xscoreoff="-0.2em"
xscorestyle="single"></charsubset></att>
<att logic="and">
<specval attname="ScoreCharOn" attval="yes">
<specval attname="ScoreSpace" attval="no">
<charsubset><highlt xscoring="1" xscorespc="0" xscoreoff="-0.2em"
xscorestyle="single"></charsubset></att>
<att logic="and">
<specval attname="ScoreCharOn" attval="no">
<specval attname="Strikethru" attval="#NE#\yes\">
<charsubset><highlt xscoring="0"></charsubset></att>

<!--_font strikethru mapping-->
<att logic="and">
<specval attname="Strikethru" attval="no">
<specval attname="ScoreCharOn" attval="#NE#\yes\">
```

```
<charsubset><highlt xscoring="0"></charsubset></att>
<att>
<specval attname="Strikethru" attval="yes">
<charsubset><highlt xscoring="1" xscoreoff="-0.2em" xscorestyle="single">
</charsubset></att>
<att logic="and">
<specval attname="Strikethru" attval="#NE#\no\">
<specval attname="Strikethru" attval="#NE#\yes\">
<specval attname="Strikethru" attval="#NE#\inherit\">
<specval attname="ScoreCharOn" attval="#NE#\yes\">
<fillval attname="Strikethru" fillcat="highlt" fillchar="xscorestyle">
<charsubset><highlt xscoring="1" ></charsubset></att>
<att>
<specval attname="XScoreSpace" attval="yes">
<charsubset><highlt xscorespc="1" ></charsubset></att>
<att>
<specval attname="XScoreSpace" attval="no">
<charsubset><highlt xscorespc="0" ></charsubset></att>
<att>
<specval attname="AllCap" attval="yes">
<charsubset><highlt allcap="1"></charsubset></att>
<att>
<specval attname="AllCap" attval="no">
<charsubset><highlt allcap="0"></charsubset></att>
<att>
<specval attname="SmallCap" attval="yes">
<charsubset><font smallcap="1"></charsubset></att>
<att>
<specval attname="SmallCap" attval="no">
<charsubset><font smallcap="0"></charsubset></att>

<!-- NOTE: no need to turn suppress or keeps line off -->
<att>
<specval attname="Suppress" attval="yes">
<charsubset><suppress sup="1"></charsubset></att>

<att>
<specval attname="KeepLine" attval="yes">
<charsubset><keeps keep="7" scope="line"></charsubset></att>

<!-- _FONT TAG fillval -->

<att logic="or">
<!--Fillval-able attributes on _font tag-->
<fillval attname="TypeSize"  fillcat="font"    fillchar="size">
<fillval attname="FamName"   fillcat="font"    fillchar="famname">
<fillval attname="Offset"    fillcat="font"    fillchar="offset">
<fillval attname="Leading"   fillcat="leading" fillchar="lead">
<fillval attname="ScoreChar"   fillcat="highlt" fillchar="scorechr">
<charsubset></charsubset></att>
<att logic="and">
```

```
<specval attname="FontColor" attval="#NE#\inherit\">
<specval attname="FontColor" attval="#NE#\default\">
<fillval attname="FontColor" fillcat="highlt" fillchar="fontclr">
<charsubset></charsubset></att>
<att logic="and">
<specval attname="BackColor" attval="#NE#\inherit\">
<specval attname="BackColor" attval="#NE#\default\">
<fillval attname="BackColor" fillcat="highlt" fillchar="bckclr">
<charsubset></charsubset></att>
<att logic="and">
<specval attname="XScoreColor" attval="#NE#\inherit\">
<specval attname="XScoreColor" attval="#NE#\default\">
<fillval attname="XScoreColor" fillcat="highlt" fillchar="xscoreclr">
<charsubset></charsubset></att>
<att logic="and">
<specval attname="ScoreColor" attval="#NE#\inherit\">
<specval attname="ScoreColor" attval="#NE#\default\">
<fillval attname="ScoreColor" fillcat="highlt" fillchar="scoreclr">
<charsubset></charsubset></att>
</e-i-c>
```

The ACL command `Modify_tag` opens a standard Windows Change Font dialog. The settings are mapped to FOSI as specified in **Figure 387** above.

<_touchup> PI

Default formatting is in *`Arbortext-path`*`\lib\_touchup-eic.fos`, which is shown next.

Figure 388 Default formatting for <_touchup> PI

```
<!-- $Id: $ FOSI fragment supporting _touchup ATI tag-->

<!--NOTE: fillval used for direct mapping of choices; specval for mapping
of atitag.cf choice values to FOSI choice values-->

<e-i-c gi="touchup" gitype="pi">
<charlist inherit="1"></charlist>

<!-- _TOUCHUP TAG specval -->

<!--_touchup FontStyle choice mapping-->
<att>
<specval attname="FontStyle" attval="monospaced-sans-serif">
<charsubset><font style="monosans"></charsubset></att>
<att>
<specval attname="FontStyle" attval="monospaced-serifed">
<charsubset><font style="monoser"></charsubset></att>
<att>
<specval attname="FontStyle" attval="sans-serif">
```

```
<charsubset><font style="sanserif"></charsubset></att>
<att>
<specval attname="FontStyle" attval="serifed">
<charsubset><font style="serif"></charsubset></att>

<!--_touchup FontPosture choice mapping-->
<att>
<specval attname="FontPosture" attval="backslant-italic">
<charsubset><font posture="bsital"></charsubset></att>
<att>
<specval attname="FontPosture" attval="italic">
<charsubset><font posture="italic"></charsubset></att>
<att>
<specval attname="FontPosture" attval="backslant-oblique">
<charsubset><font posture="bsobl"></charsubset></att>
<att>
<specval attname="FontPosture" attval="oblique">
<charsubset><font posture="oblique"></charsubset></att>
<att>
<specval attname="FontPosture" attval="upright">
<charsubset><font posture="upright"></charsubset></att>

<!--_touchup FontWeight choice mapping-->
<att>
<specval attname="FontWeight" attval="ultrabold">
<charsubset><font weight="ultbold"></charsubset></att>
<att>
<specval attname="FontWeight" attval="extrabold">
<charsubset><font weight="exbold"></charsubset></att>
<att>
<specval attname="FontWeight" attval="bold">
<charsubset><font weight="bold"></charsubset></att>
<att>
<specval attname="FontWeight" attval="semibold">
<charsubset><font weight="sembold"></charsubset></att>
<att>
<specval attname="FontWeight" attval="medium">
<charsubset><font weight="medium"></charsubset></att>
<att>
<specval attname="FontWeight" attval="semilight">
<charsubset><font weight="semlight"></charsubset></att>
<att>
<specval attname="FontWeight" attval="light">
<charsubset><font weight="light"></charsubset></att>
<att>
<specval attname="FontWeight" attval="extralight">
<charsubset><font weight="exlight"></charsubset></att>
<att>
<specval attname="FontWeight" attval="ultralight">
<charsubset><font weight="ultlight"></charsubset></att>
```

```
<!--_touchup HighltScoring choice mapping-->
<!-- off and on are obsolete as of olive;
     they are retained only for backward compatibility -->
<att>
<specval attname="HighltScoring" attval="on">
<fillval attname="HighltScoreColor" fillcat="highlt" fillchar="scoreclr">
<charsubset><highlt scoring="1" scorestyle="single"></charsubset></att>
<att>
<specval attname="HighltScoring" attval="off">
<charsubset><highlt scoring="0"></charsubset></att>

<att logic="and">
<specval attname="HighltScoring" attval="#NE#\on\">
<specval attname="HighltScoring" attval="#NE#\off\">
<specval attname="HighltScoring" attval="#NE#\inherit\">
<fillval attname="HighltScoring" fillcat="highlt" fillchar="scorestyle">
<fillval attname="HighltScoreColor" fillcat="highlt" fillchar="scoreclr">
<charsubset><highlt scoring="1"></charsubset></att>

<!--_touchup HighltAllcap choice mapping-->
<att>
<specval attname="HighltAllcap" attval="on">
<charsubset><highlt allcap="1"></charsubset></att>
<att>
<specval attname="HighltAllcap" attval="off">
<charsubset><highlt allcap="0"></charsubset></att>

<!--_touchup HighltScorespc choice mapping-->
<att>
<specval attname="HighltScorespc" attval="on">
<charsubset><highlt scorespc="1"></charsubset></att>
<att>
<specval attname="HighltScorespc" attval="off">
<charsubset><highlt scorespc="0"></charsubset></att>

<!--_touchup AlgroupRefpoint choice mapping-->
<att>
<specval attname="AlgroupRefpoint" attval="bottom">
<charsubset><algroup refpoint="bottom"></charsubset></att>
<att>
<specval attname="AlgroupRefpoint" attval="last">
<charsubset><algroup refpoint="last"></charsubset></att>
<att>
<specval attname="AlgroupRefpoint" attval="first">
<charsubset><algroup refpoint="first"></charsubset></att>
<att>
<specval attname="AlgroupRefpoint" attval="top">
<charsubset><algroup refpoint="top"></charsubset></att>

<!--_touchup SuppressSup choice mapping-->
<att>
```

```
<specval attname="SuppressSup" attval="yes">
<charsubset><suppress sup="1"></charsubset></att>
<att>
<specval attname="SuppressSup" attval="no">
<charsubset><suppress sup="0"></charsubset></att>

<!--_touchup TextbrkStartln choice mapping-->
<att>
<specval attname="TextbrkStartln" attval="yes">
<charsubset><textbrk startln="1"></charsubset></att>
<att>
<specval attname="TextbrkStartln" attval="no">
<charsubset><textbrk startln="0"></charsubset></att>

<!--_touchup TextbrkEndln choice mapping-->
<att>
<specval attname="TextbrkEndln" attval="yes">
<charsubset><textbrk endln="1"></charsubset></att>
<att>
<specval attname="TextbrkEndln" attval="no">
<charsubset><textbrk endln="0"></charsubset></att>
<!--_touchup TextbrkStartcol choice mapping-->
<att>
<specval attname="TextbrkStartcol" attval="yes">
<charsubset><textbrk startcol="1"></charsubset></att>
<att>
<specval attname="TextbrkStartcol" attval="no">
<charsubset><textbrk startcol="0"></charsubset></att>

<!--_touchup PrespCondit choice mapping-->
<att>
<specval attname="PrespCondit" attval="discard">
<charsubset><presp condit="discard"></charsubset></att>
<att>
<specval attname="PrespCondit" attval="keep">
<charsubset><presp condit="keep"></charsubset></att>

<!--_touchup PostspCondit choice mapping-->
<att>
<specval attname="PostspCondit" attval="discard">
<charsubset><postsp condit="discard"></charsubset></att>
<att>
<specval attname="PostspCondit" attval="keep">
<charsubset><postsp condit="keep"></charsubset></att>

<!--_touchup QuaddingQuad choice mapping-->
<att>
<specval attname="QuaddingQuad" attval="asis">
<charsubset><quadding quad="asis"></charsubset></att>
<att>
<specval attname="QuaddingQuad" attval="justify">
```

```
<charsubset><quadding quad="justify"></charsubset></att>
<att>
<specval attname="QuaddingQuad" attval="out">
<charsubset><quadding quad="out"></charsubset></att>
<att>
<specval attname="QuaddingQuad" attval="in">
<charsubset><quadding quad="in"></charsubset></att>
<att>
<specval attname="QuaddingQuad" attval="center">
<charsubset><quadding quad="center"></charsubset></att>
<att>
<specval attname="QuaddingQuad" attval="left">
<charsubset><quadding quad="left"></charsubset></att>
<att>
<specval attname="QuaddingQuad" attval="right">
<charsubset><quadding quad="right"></charsubset></att>

<!--_touchup QuaddingLastquad choice mapping-->
<att>
<specval attname="QuaddingLastquad" attval="relative">
<charsubset><quadding lastquad="relative"></charsubset></att>
<att>
<specval attname="QuaddingLastquad" attval="justify">
<charsubset><quadding lastquad="ljustify"></charsubset></att>
<att>
<specval attname="QuaddingLastquad" attval="out">
<charsubset><quadding lastquad="lout"></charsubset></att>
<att>
<specval attname="QuaddingLastquad" attval="in">
<charsubset><quadding lastquad="lin"></charsubset></att>
<att>
<specval attname="QuaddingLastquad" attval="center">
<charsubset><quadding lastquad="lcenter"></charsubset></att>
<att>
<specval attname="QuaddingLastquad" attval="left">
<charsubset><quadding lastquad="lleft"></charsubset></att>
<att>
<specval attname="QuaddingLastquad" attval="right">
<charsubset><quadding lastquad="lright"></charsubset></att>

<!--_touchup HyphenHyph choice mapping-->

<att>
<specval attname="HyphenHyph" attval="yes">
<charsubset><hyphen hyph="1"></charsubset></att>
<att>
<specval attname="HyphenHyph" attval="no">
<charsubset><hyphen hyph="0"></charsubset></att>

<!--_touchup KeepsKeep choice mapping-->
<att>
```

```
<specval attname="KeepsKeep" attval="yes">
<charsubset><keeps keep="7"></charsubset></att>
<att>
<specval attname="KeepsKeep" attval="no">
<charsubset><keeps keep="0"></charsubset></att>

<!--_touchup KeepsPrev choice mapping-->
<att>
<specval attname="KeepsPrev" attval="yes">
<charsubset><keeps prev="7"></charsubset></att>
<att>
<specval attname="KeepsPrev" attval="no">
<charsubset><keeps prev="0"></charsubset></att>

<!--_touchup KeepsNext choice mapping-->
<att>
<specval attname="KeepsNext" attval="yes">
<charsubset><keeps next="7"></charsubset></att>
<att>
<specval attname="KeepsNext" attval="no">
<charsubset><keeps next="0"></charsubset></att>

<!--_touchup KeepsBoundary choice mapping-->
<att>
<specval attname="KeepsBoundary" attval="line">
<charsubset><keeps scope="line"></charsubset></att>
<att>
<specval attname="KeepsBoundary" attval="page">
<charsubset><keeps scope="page"></charsubset></att>
<att>
<specval attname="KeepsBoundary" attval="column">
<charsubset><keeps scope="col"></charsubset></att>

<!--_touchup FontWidth choice mapping-->
<att>
<specval attname="FontWidth" attval="ultracond">
<charsubset><font width="ultcond"></charsubset></att>
<att>
<specval attname="FontWidth" attval="extracond">
<charsubset><font width="excond"></charsubset></att>
<att>
<specval attname="FontWidth" attval="cond">
<charsubset><font width="cond"></charsubset></att>
<att>
<specval attname="FontWidth" attval="semicond">
<charsubset><font width="semcond"></charsubset></att>
<att>
<specval attname="FontWidth" attval="regular">
<charsubset><font width="regular"></charsubset></att>
<att>
<specval attname="FontWidth" attval="semiexp">
```

```
<charsubset><font width="semexp"></charsubset></att>
<att>
<specval attname="FontWidth" attval="exp">
<charsubset><font width="exp"></charsubset></att>
<att>
<specval attname="FontWidth" attval="extraexp">
<charsubset><font width="exexp"></charsubset></att>
<att>
<specval attname="FontWidth" attval="ultraexp">
<charsubset><font width="ultexp"></charsubset></att>

<att logic="or">
<specval attname="PrespNominal" attval="#ANY">
<specval attname="PrespMinimum" attval="#ANY">
<specval attname="PrespMaximum" attval="#ANY">
<charsubset><presp priority="force"><textbrk startln="1"></charsubset>
</att>

<att logic="or">
<specval attname="PostspNominal" attval="#ANY">
<specval attname="PostspMinimum" attval="#ANY">
<specval attname="PostspMaximum" attval="#ANY">
<charsubset><postsp priority="force"><textbrk endln="1"></charsubset>
</att>

<!-- _TOUCHUP TAG fillval -->

<att logic="or">
<!--Fillval-able attributes on _touchup tag-->
<fillval attname="FontSize"        fillcat="font"    fillchar="size">
<fillval attname="FontFamname"     fillcat="font"    fillchar="famname">
<fillval attname="FontOffset"      fillcat="font"    fillchar="offset">
<fillval attname="LeadingLead"     fillcat="leading" fillchar="lead">
<fillval attname="HighltScorewt"   fillcat="highlt"  fillchar="scorewt">
<fillval attname="HighltScoreoff"  fillcat="highlt"  fillchar="scoreoff">
<fillval attname="HighltForpct"    fillcat="highlt"  fillchar="forpct">
<fillval attname="TextbrkStartpg"  fillcat="textbrk" fillchar="startpg">
<fillval attname="TextbrkNewpgmdl" fillcat="textbrk" fillchar="newpgmdl">
<fillval attname="PrespNominal"    fillcat="presp"   fillchar="nominal">
<fillval attname="PrespMinimum"    fillcat="presp"   fillchar="minimum">
<fillval attname="PrespMaximum"    fillcat="presp"   fillchar="maximum">
<fillval attname="PrespPriority"   fillcat="presp"   fillchar="priority">
<fillval attname="PostspNominal"   fillcat="postsp"  fillchar="nominal">
<fillval attname="PostspMinimum"   fillcat="postsp"  fillchar="minimum">
<fillval attname="PostspMaximum"   fillcat="postsp"  fillchar="maximum">
<fillval attname="PostspPriority"  fillcat="postsp"  fillchar="priority">
<fillval attname="IndentLeftind"   fillcat="indent"  fillchar="leftind">
<fillval attname="IndentRightind"  fillcat="indent"  fillchar="rightind">
<fillval attname="IndentFirstln"   fillcat="indent"  fillchar="firstln">
<fillval attname="SpanSpan"        fillcat="span"    fillchar="span">
<fillval attname="WordspNominal"   fillcat="wordsp"  fillchar="nominal">
```

```
<fillval attname="WordspMinimum"   fillcat="wordsp"  fillchar="minimum">
<fillval attname="WordspMaximum"   fillcat="wordsp"  fillchar="maximum">
<fillval attname="BoxingToffset"   fillcat="boxing"  fillchar="toffset">
<fillval attname="BoxingBoffset"   fillcat="boxing"  fillchar="boffset">
<fillval attname="BoxingLoffset"   fillcat="boxing"  fillchar="loffset">
<fillval attname="BoxingRoffset"   fillcat="boxing"  fillchar="roffset">
<fillval attname="BoxingTrel"      fillcat="boxing"  fillchar="trel">
<fillval attname="BoxingBrel"      fillcat="boxing"  fillchar="brel">
<fillval attname="BoxingSiderel"   fillcat="boxing"  fillchar="siderel">
<fillval attname="BoxingTthick"    fillcat="boxing"  fillchar="tthick">
<fillval attname="BoxingBthick"    fillcat="boxing"  fillchar="bthick">
<fillval attname="BoxingLthick"    fillcat="boxing"  fillchar="lthick">
<fillval attname="BoxingRthick"    fillcat="boxing"  fillchar="rthick">
<fillval attname="BoxingInpct"     fillcat="boxing"  fillchar="inpct">
<fillval attname="BoxingRotation"  fillcat="boxing"  fillchar="rotation">
<fillval attname="BoxingOrigin"    fillcat="boxing"  fillchar="origin">
<fillval attname="BoxingProjection" fillcat="boxing" fillchar="projection">
<fillval attname="BoxingFixedwidth" fillcat="boxing" fillchar="fixedwidth">
<charsubset></charsubset>
</att>
<att logic="and">
<specval attname="HighltFontclr" attval="#NE#\inherit\">
<specval attname="HighltFontclr" attval="#NE#\default\">
<fillval attname="HighltFontclr" fillcat="highlt" fillchar="fontclr">
<charsubset></charsubset>
</att>
<att logic="and">
<specval attname="HighltBckclr" attval="#NE#\inherit\">
<specval attname="HighltBckclr" attval="#NE#\default\">
<fillval attname="HighltBckclr" fillcat="highlt" fillchar="bckclr">
<charsubset></charsubset>
</att>
</e-i-c>
```

The ACL command `modify_tag` opens a custom Touchup dialog. Any settings are mapped to FOSI as specified in **Figure 388** above.

`Keeps` *Touchup PIs*

The <_nocolumnbreak> PI is a UDT of <_touchup>. In an ASCII file, <_nocolumnbreak> appears as <?Pub UDT _nocolumnbreak _touchup KeepsKeep="yes" KeepsPrev="no" KeepsNext="no" KeepsBoundary="column"?>.

The <_nopagebreak> PI is likewise a UDT of <_touchup>. In an ASCII file, <_nopagebreak> appears as <?Pub UDT _nopagebreak _touchup KeepsKeep="yes" KeepsPrev="no" KeepsNext="no" KeepsBoundary="page"?>.

NOTE: These `keeps` settings are not available in the <_touchup> dialog.

Deep content splitting Touchup PI

Arbortext Editor supports **deep content splitting**, which allows page breaks within structures that are too deep for the page, including table rows, boxed areas and `algroup`-ed columns. When deep content is too deep, a column/page overset results, with the deep content flowing off the bottom of the page so it does not appear when the document is printed.

Deep content splitting can be enabled just for a particular section of a document by selecting it and inserting a <_deepcontentsplitting> tag pair. Deep content splitting can also be enabled for all documents; see **Deep content splitting-related ACL** below for details.

NOTE: The <_deepcontentsplitting> PI does not apply to Edit window display.

Deep content splitting does not work with:

- Tables with long vertical spans
- Tables with repeating column headings and repeating table footers
- Pages with change bars
- Line numbering
- Table footnotes
- Gentext

Deep content splitting-related ACL

`Set deepcontentsplitting={on | off | never}` controls deep content splitting. The default is `on`. An `off` setting Is ignored when a <_deepcontentsplitting> tag pair is inserted in a document. To cause <_deepcontentsplitting> PIs to be ignored, use `set deepcontentsplitting=never`.

Deep content splitting example

In this example, deep content splitting is enabled for a portion of a document.

Figure 389 Deep content splitting for part of a document

XML fragment

```
<?Pub _deepcontentsplitting?><reftables>...</reftables>
<?Pub /_deepcontentsplitting?>
```

Touchup examples

Figure 390 shows Touchup PIs in an ASCII file.

Figure 390 Touchup PIs in XML

XML fragment

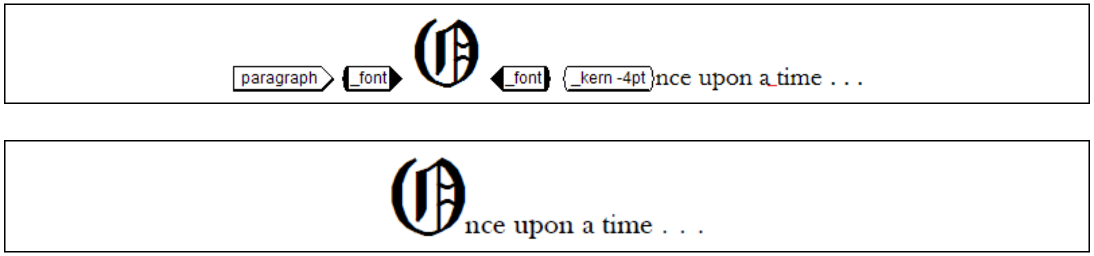
```
<?Pub _deepcontentsplitting?>...<?Pub /_deepcontentsplitting?>

<?Pub _font Posture="italic" Weight="bold" FontColor="navy"
BackColor="yellow" Underline="single" AllCap="yes" ScoreSpace="yes"?>
Blue foreground yellow background underlined<?Pub /_font?>

as follows:<?Pub _hardspace?>...

T.S.<?Pub _interword-space?>Eliot
To Whom It May Concern--<?Pub _eos-space?>This report...

<?Pub _font TypeSize="48pt" FamName="Arial"?>T
<?Pub _kern Amount="-6pt"?>otal...

<title>The Loch Ness Monster:<?Pub _newline?>Fact or Fiction?</title>

<paragraph>America's Independence Day is also known as Fourth of July
because <?Pub _nolinebreak?>July 4, 1776<?Pub /_nolinebreak?> ...

Theo<?Pub _hyphen-point?>dore Roo<?Pub _hyphen-point?>
se<?Pub _hyphen-point?>velt was President...

<para>The page breaks in this document are set manually
<?Pub _newpage?>in order to...

...</short-description><?Pub _newcolumn?><specifications>...

<?Pub _nopagebreak?><sponsors>...</sponsors><?Pub _nopagebreak?>

<?Pub _nocolumnbreak?><step><note>Be prepared to perform the next
step as soon as you complete this step.</note>...</step><step>...</step>
<?Pub /_nocolumnbreak?>

<?Pub _touchup SuppressSup="yes"?><confidential>...</confidential>
<?Pub /_touchup?>

<?Pub _touchup SpanSpan="2" AlgroupRefpoint="unaligned"?>...<?Pub
/_touchup?>
```

Figure 391 shows the use of two Touchup tags: <_font> and <_kern>.

Figure 391 <_font> and <_kern> in Edit window and print/PDF

XML fragment

```
<paragraph><?Pub _font TypeSize="48pt" FamName="Hershey Gothic English"?>O
<?Pub /_font?><?Pub _kern Amount="-4pt"?>nce upon a time . . .
```



Once upon a time . . .

Once upon a time . . .

In the next example, a <_newline> PI causes a line break in <title> but has no effect when saved as part of #CONTENT and output in the Table of Contents. Similarly, the <_font> PI changes the font in <title> but is not output in the TOC.

Figure 392 Saved <_newline> removed from TOC

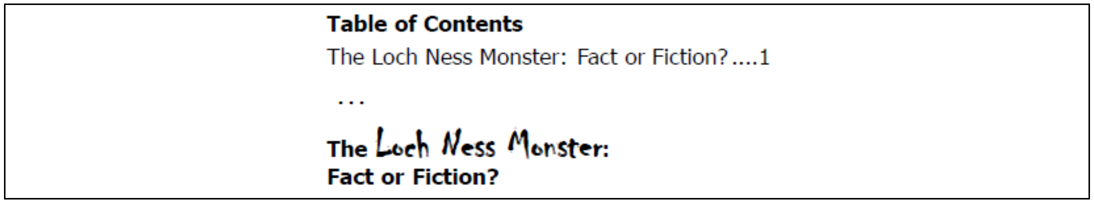


Table of Contents

The Loch Ness Monster: Fact or Fiction? 1

. . .

The Loch Ness Monster:
Fact or Fiction?

XML fragment

```
<title>The <?Pub _font TypeSize="20pt" FamName="Chiller"?>Loch Ness
Monster<?Pub /_font?>: <?Pub _newline?>Fact or Fiction?</title>
```

FOSI fragment

```
<stringdecl textid="toc.app" literal="" scope="1">
...
<e-i-c gi="title">
<charlist inherit="1">
<savetext textid="toc.app" append="1"
conrule="<toc.fmt>,#CONTENT,dotfill,folioct.txt[BO],</toc.fmt>">
...
<e-i-c gi="toc">
<charlist inherit="1">
<usetext source="<toc-title.fmt>,\Table of Contents\,</toc-title.fmt>,
toc.app">
<subchars>
<font inherit="1" famname="Tahoma" size="11pt">
<leading inherit="1" lead="14pt">
<indent inherit="1" leftind="0pt" rightind="*+18pi" firstln="*">
```

```
...
<e-i-c gi="toc.fmt">
<charlist inherit="1" charsubsetref="block"></charlist>
</e-i-c>
<e-i-c gi="toc-title.fmt">
<charlist inherit="1" charsubsetref="title"></charlist>
</e-i-c>
```

The following example is the same as the previous one, except that a <_touchup> PI is used in <title> instead of <_newline>. In this case, the line break is output in the Table of Contents.

Figure 393 Saved <_touchup> PI is output in TOC

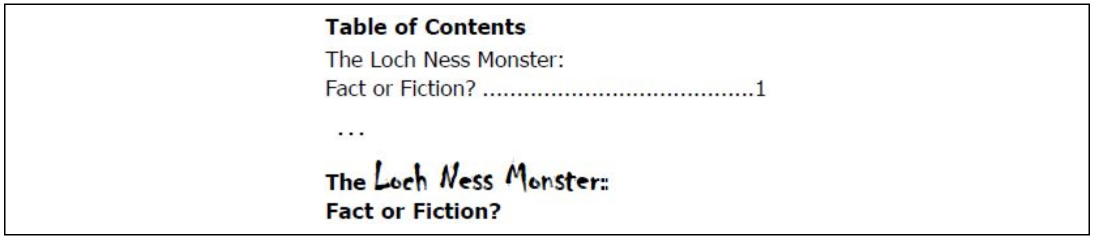


Table of Contents

The Loch Ness Monster:
Fact or Fiction? 1

...

The Loch Ness Monster:
Fact or Fiction?

XML fragment

```
<title>The <?Pub _font TypeSize="20pt" FamName="Chiller"?>Loch Ness
Monster<?Pub /_font?>: <?Pub _touchup TextbrkStartln="yes"?>Fact or
Fiction?<?Pub /_touchup?></title>
```

This example continues from **Figure 393**. The default formatting for <_font> and <_touchup> is disabled by coding an `e-i-c` for the PIs with an empty `charlist`, as shown in **Figure 394**.

Figure 394 <_touchup> and <_font> PIs disabled by e-i-cs

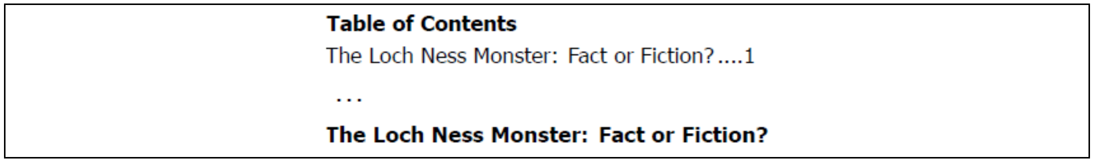


Table of Contents

The Loch Ness Monster: Fact or Fiction?....1

...

The Loch Ness Monster: Fact or Fiction?

```
<e-i-c gi="touchup font" gitype="pi">
<charlist inherit="1"></charlist>
</e-i-c>
```

This example continues from **Figure 394**, except that here the <_font> PIs specify different fonts. However, an `e-i-c` for <_font> specifies `famname="Chiller"`, which overrides the font settings on the PIs.

Figure 395 Formatting coded for <_font> PI `e-i-c` (Edit window view)

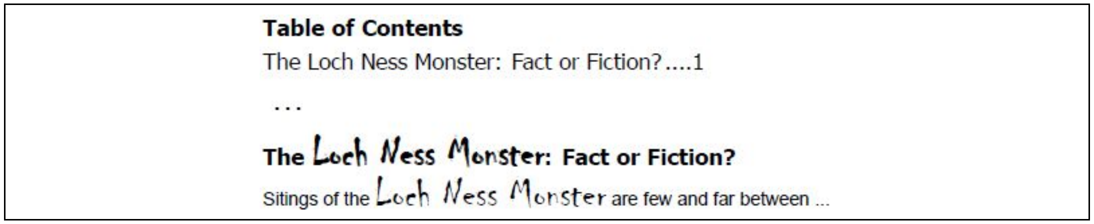


Table of Contents

The Loch Ness Monster: Fact or Fiction?....1

...

The Loch Ness Monster: Fact or Fiction?

Sitings of the Loch Ness Monster are few and far between ...

XML fragment

```
<title>The <?Pub _font TypeSize="20pt" FamName="Castellar"?>Loch Ness
Monster<?Pub /_font?>: <?Pub _touchup TextbrkStartln="yes"?>Fact or
Fiction?<?Pub /_touchup?></title><para>Sitings of the <?Pub _font
FamName="Arial"?>Loch Ness Monster<?Pub /_font?> are few and far between...
```

FOSI fragment

```
<e-i-c gi="touchup" gitype="pi">
<charlist inherit="1"></charlist>
</e-i-c>
<e-i-c gi="font" gitype="pi">
<charlist inherit="1">
<font famname="Chiller" size="20pt">
...
```

In the next example, <_touchup> is used to indicate the content changed.

Figure 396 <_touchup> outputs change bar in repeating float

XML fragment

```
<chapter>
<title><?Pub _touchup?>Introduction<?Pub /_touchup?></title>
```

FOSI fragment

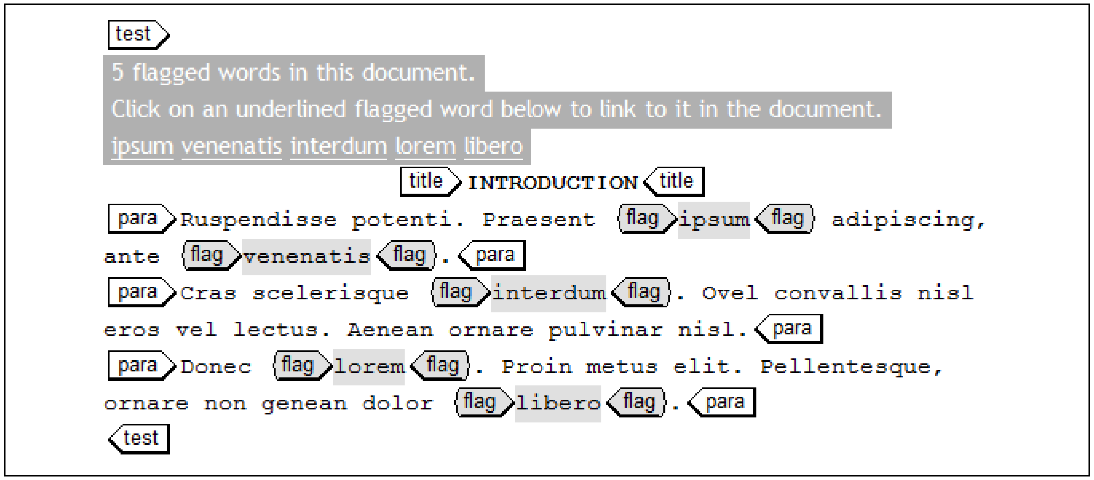
```
<floatloc floatid="continued.top" floattyp="atcolbrk">
...
<flowtext numcols="1" topfloat="continued.top">
<column ...>
</flowtext>
...
<e-i-c gi="title" context="chapter">
<charlist inherit="1" charsubsetref="title">
<savetext textid="chapter-title.txt" conrule="#CONTENT">
<usetext source="chapter-title.txt,\, continued\">
<subchars>
<float flidref="continued.top" width="page" scope="chapter">
...
<e-i-c gi="touchup" gitype="pi">
<charlist inherit="1">
```

```
<chgmark ...></chgmark>
```

The last example utilizes a user-defined tag named <flag>, which is based on the <_touchup> PI. (User-defined tags are discussed on page 726.)

The image below shows the Edit window. The <_touchup> e-i-c counts each <flag> and saves its contents to an appended string. The total number and appended list are output from time-independent variables at the beginning of the document in the Edit window.

Figure 397 Repeating categories in <_touchup> e-i-c



test
5 flagged words in this document.
Click on an underlined flagged word below to link to it in the document.
ipsum venenatis interdum lorem libero
title INTRODUCTION title
para Ruspendisse potenti. Praesent flag ipsum flag adipiscing, ante flag venenatis flag. para
para Cras scelerisque flag interdum flag. Ovel convallis nisl eros vel lectus. Aenean ornare pulvinar nisl. para
para Donec flag lorem flag. Proin metus elit. Pellentesque, ornare non genean dolor flag libero flag. para
test

FOSI fragment

```
<counter initial="0" style="arabic" enumid="flagct">
<stringdecl textid="flag.txt" hotlink="1">
<stringdecl textid="flagct.tiv" status="1">
<stringdecl textid="flags.app" status="1">
...
<e-i-c gi="test">
<charlist inherit="1" charsubsetref="block">
<textbrk startpg="recto" pageid="test.page" newpgmdl="global">
<savetext textid="flagct.tiv" conrule="flagct" placemnt="after">
</charlist>
<att>
<specval attname="editor-only" attloc="SYSTEM-VAR" attval="#ANY">
<charsubset>
<usetext source="\ \,flagct.tiv,\ flagged words in this document. \"
placemnt="before">
<subchars charsubsetref="block">
<font inherit="1" famname="Trebuchet MS">
<highlt inherit="1" bckclr="gray5" fontclr="#FFFFFF">
```

```
</subchars>
</usetext>
<usetext
source="\ Click on an underlined flagged word below to link to it
in the document. \" placemnt="before">
<subchars charsubsetref="block">
<font inherit="1" famname="Trebuchet MS">
<highlt inherit="1" bckclr="gray5" fontclr="#FFFFFF">
...
<usetext source="\ \,flags.app" placemnt="before">
<subchars>
<font inherit="1" famname="Trebuchet MS">
<highlt inherit="1" bckclr="gray5" fontclr="#FFFFFF">
...
<e-i-c gi="touchup" gitype="pi">
<charlist inherit="1">
<highlt bckclr="gray2">
<enumerat increm="1" enumid="flagct">
<savetext textid="flag.txt" conrule="#CONTENT">
<savetext textid="flags.app" conrule="flag.txt,\ \" append="1">
...
```