

FOSI TIP 

In the **All FOSI Components** panel, select **Query→All Non-DTD (Pseudo-elements) Elements** to see a list of e-i-c's for elements that are not defined in the DTD.

FOSI pseudo-elements

A **pseudo-element** is an element-in-context for an element that is not in the DTD. Pseudo-elements can be used to provide context, formatting, or programmatic capabilities, as described in the following sections.

See **Naming conventions** on page 717 for guidelines on naming pseudo-elements.

NOTE: The name of a pseudo-element is limited to 71 characters.

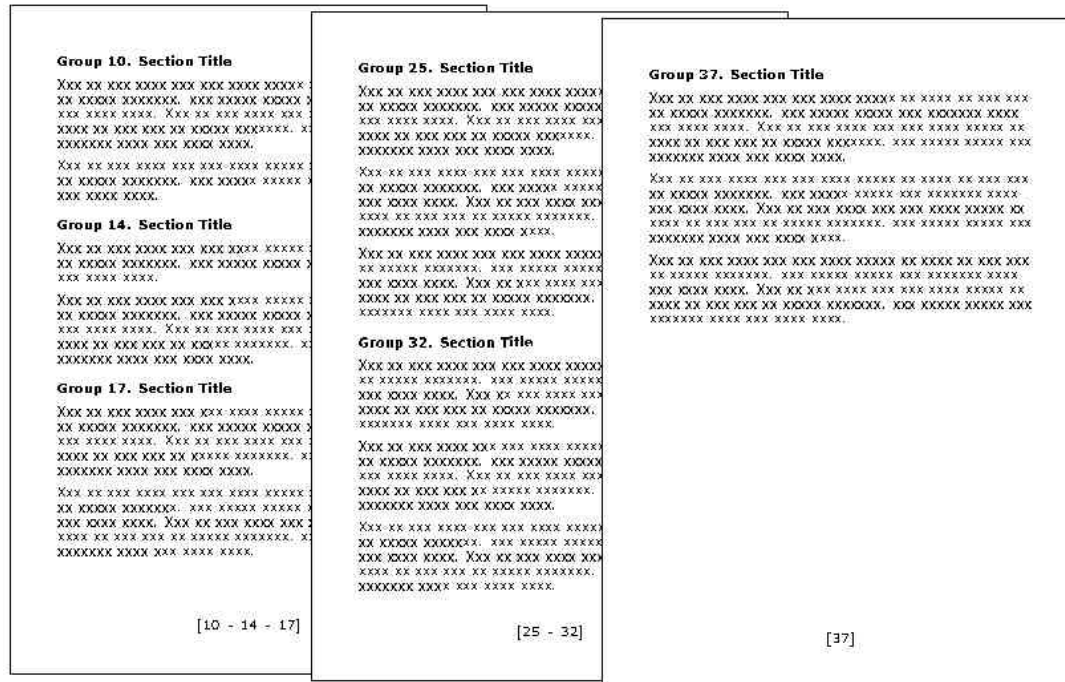
As with an element e-i-c, a pseudo-element e-i-c may have a context string and an occurrence setting. This is illustrated in **Figure 46 Floated <note> with different title formatting** on page 115.

Since a pseudo-element is not declared in the DTD, it has no attributes. However, it can use and test attribute values on ancestor elements. This is illustrated in **Figure 49** on page 120.

Pseudo-elements and occurrence

Occurrence applies to pseudo-elements as illustrated in the following example. Note that an e-i-c is not needed for `group.psu`.

Figure 42 Occurrence with pseudo-elements



DTD fragment

```
<!ELEMENT doc (section)+ >
<!ELEMENT section (title,para+) >
<!ATTLIST section group NUMBER #REQUIRED >
<!ELEMENT (title|para) (#PCDATA) >
```

XML fragment

```
<section group="10">
<title>...</title>
<para>...</para>
</section>
<section group="14">
<title>...</title>
<para>...</para>
</section>
<section group="17">
```

```

<title>...</title>
<para>...</para>
</section>
<section group="25">
<title>...</title>
<para>...</para>
</section>
<section group="32">
<title>...</title>
<para>...</para>
</section>
<section group="37">
<title>...</title>
<para>...</para>
</section>

```

FOSI fragment

```

<footer nomdepth="12pt" spaflow="12pt">
<usetext source="<group.psu>,groups-on-page.txt[FB],</group.psu>"
placemnt="before">
<subchars charsubsetref="block center">
<font size="12pt">
<leading lead="12pt">
</subchars>
</usetext>
</footer>

```

```

<e-i-c gi="group.fmt" context="group.psu" occur="only">
<charlist inherit="1" charsubsetref="inline">
<usetext source="\[" placemnt="before"></usetext>
<usetext source="\]" placemnt="after"></usetext>
</charlist>
</e-i-c>

```

```

<e-i-c gi="group.fmt" context="group.psu" occur="first">
<charlist inherit="1" charsubsetref="inline">
<usetext source="\[" placemnt="before"></usetext>
</charlist>
</e-i-c>

```

```

<e-i-c gi="group.fmt" context="group.psu" occur="middle">
<charlist inherit="1" charsubsetref="inline">
<usetext source="\ - \" placemnt="before"></usetext>
</charlist>
</e-i-c>

```

```

<e-i-c gi="group.fmt" context="group.psu" occur="last">
<charlist inherit="1" charsubsetref="inline">
<usetext source="\ - \" placemnt="before"></usetext>
<usetext source="\]" placemnt="after"></usetext>
</charlist>

```

```
</e-i-c>  
  
<e-i-c gi="section">  
<charlist inherit="1" charsubsetref="block keep-together">  
</charlist>  
<att>  
<fillval attname="group" attloc="section" fillcat="savetext"  
fillchar="conrule">  
<charsubset>  
<savetext textid="group.txt">  
<savetext textid="groups-on-page.txt"  
conrule="<group.fmt>,group.txt,</group.fmt>" append="0">  
</charsubset>  
</att>  
</e-i-c>
```

FOSI TIP 

When possible, consolidate all formatting in one formatting pseudo-element to avoid confusion by those who maintain the FOSI in the future.

Formatting pseudo-elements

Formatting pseudo-elements format text in a `usetext` source or a `savetext` `conrule`, as shown in the following FOSI fragment.

FOSI fragment

```
<savetext textid="figure-title.txt"
conrule="<fontsize14pt>,tablect.txt,</fontsize14pt>,\ \,table-title.txt"

<usetext source="<fontsize16pt.fmt>,\&sq;\,</fontsize16pt.fmt>,4pt,
<fontsize14p.fmt>,stepct.txt,</fontsize14pt.fmt>,">
...
<e-i-c gi="fontsize14pt.fmt">
<charlist inherit="1">
<font inherit="1" size="14pt">
</charlist>
</e-i-c>
...
<e-i-c gi="fontsize16pt.fmt">
<charlist inherit="1">
<font inherit="1" size="16pt">
</charlist>
</e-i-c>
```

Formatting pseudo-elements can be nested, as shown in **Figure 43 Suppressed and saved end notes** on page 113. Other examples of formatting pseudo-elements are included in **Pseudo-element examples** on page 113.

Context pseudo-elements

Sometimes a pseudo-element is needed just to provide a different context for an e-i-c, as shown in the following FOSI fragment.

FOSI fragment

```
<e-i-c gi="emphasis" context="para.ctx">
<charlist inherit="1" charsubsetref="semi-bold">
...
<e-i-c gi="emphasis" context="title.ctx">
<charlist inherit="1" charsubsetref="bold">
...
```

NOTE: A pseudo-element whose only purpose is to provide context does not need to be coded in the FOSI.

Examples of context pseudo-elements are included in **Pseudo-element examples** on page 113.

FOSI TIP 

Best practice is to name programming pseudo-elements according to their functionality, with .psu appended. For example: <highlight-every-other.psu>. When a string variable is used with a programming pseudo-element, append .psu to the textid. For example: highlight-every-other.psu.

Programming pseudo-elements

Programming pseudo-elements are used for programmatic processing rather than formatting document content or providing context. **Pseudo-element examples** on page 113 illustrate various uses of programming pseudo-elements.

Pseudo-element examples

The first example suppresses and saves <endnote> elements throughout a <chapter> for output at the end of each <chapter>. The formatting pseudo-elements <chapter-endnotes.fmt> and <bold.fmt are nested>. Notice that <chapter-endnotes.fmt> provides context for <bold.fmt>, which has two e-i-cs with different context strings and formatting for each context.

Figure 43 Suppressed and saved end notes

FOSI fragment

```
<counter enumid="endnotect" style="arabic" initial="0">
<stringdecl textid="chapter-endnotes.app" literal="">
...
<e-i-c gi="chapter-endnote" context="chapter">
<charlist inherit="1" charsubsetref="block SUPPRESS">
<enumerat enumid="chapter-endnotect" incremnt="1">
<savetext textid="chapter-endnotes.app"
conrule="<chapter-endnote.fmt>, <bold.fmt>, chapter-endnotect, </bold.fmt>,
@1.5pi, #CONTENT, </chapter-endnote.fmt>" append="1">
</charlist>
</e-i-c>

<e-i-c gi="chapter">
<charlist inherit="1">
<usetext source="chapter-endnotes.app" placemnt="after"></usetext>
...
<e-i-c gi="chapter-endnote.fmt">
<charlist inherit="1" charsubsetref="block endnotes"></charlist>
</e-i-c>

<e-i-c gi="bold.fmt" context="chapter-endnote.fmt">
<charlist inherit="1" charsubsetref="semi-bold">
</charlist>
</e-i-c>

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

The next example shows the use of formatting pseudo-elements to output boxing with double lines.

Arbortext does not support boxing a rule type of `double`. However, you can use pseudo-elements to output double-boxing, as the code below demonstrates. The boxing characteristic values for each box must be coordinated to achieve the desired appearance.

Figure 44 Double-line boxing

FOSI fragment

```
<e-i-c gi="sidebar">
<charlist inherit="1" charsubsetref="block SUPPRESS">
<usetext source="<outer-box.fmt>,<inner-box.fmt>,#CONTENT,</inner-box.fmt>,</outer-box.fmt></usetext>
...
<e-i-c gi="inner-box.fmt">
<charlist inherit="1" charsubsetref="block">
<boxing ...>
...
<e-i-c gi="outer-box.fmt">
<charlist inherit="1" charsubsetref="block">
<boxing ...>
...
```

Sometimes the boxed content includes gentext, as shown in the following figure.

Figure 45 Double-line boxing with gentext

```
<e-i-c gi="danger">
<charlist inherit="1" charsubsetref="block SUPPRESS">
<usetext source="<outer-box.fmt>,<inner-box.fmt>,<danger.fmt>,\DANGER!\,</danger.fmt>,#CONTENT,</inner-box.fmt>,</outer-box.fmt></usetext>
</charlist>
</e-i-c>

<e-i-c gi="inner-box.fmt">
<charlist inherit="1" charsubsetref="block">
<boxing ...>
</charlist>
</e-i-c>

<e-i-c gi="outer-box.fmt">
<charlist inherit="1" charsubsetref="block">
<boxing ...>
</charlist>
</e-i-c>
```

FOSI TIP

Formatting specs you need to know in order to code `boxing`: What is the distance between the content and all four sides of the inner box? Is the content indented, or do the sides of the boxes extend into the left and right margins? How much vertical space is desired between the preceding content and the top line of the box? How much vertical space is desired from the end of the content to the next element? For boxes with double rules: What is the distance between all four sides of the inner and outer boxes?

The following example uses a pseudo-element to provide context for the formatting pseudo-element, `note-title.fmt`. When the `float` attribute on `<note>` is set to `yes`, `<note>` is floated to the bottom of the page. Notice that an `e-i-c` for the `<floated-note.ctx>` pseudo-element is not needed since it does nothing except provide context.

Figure 46 Floated `<note>` with different title formatting

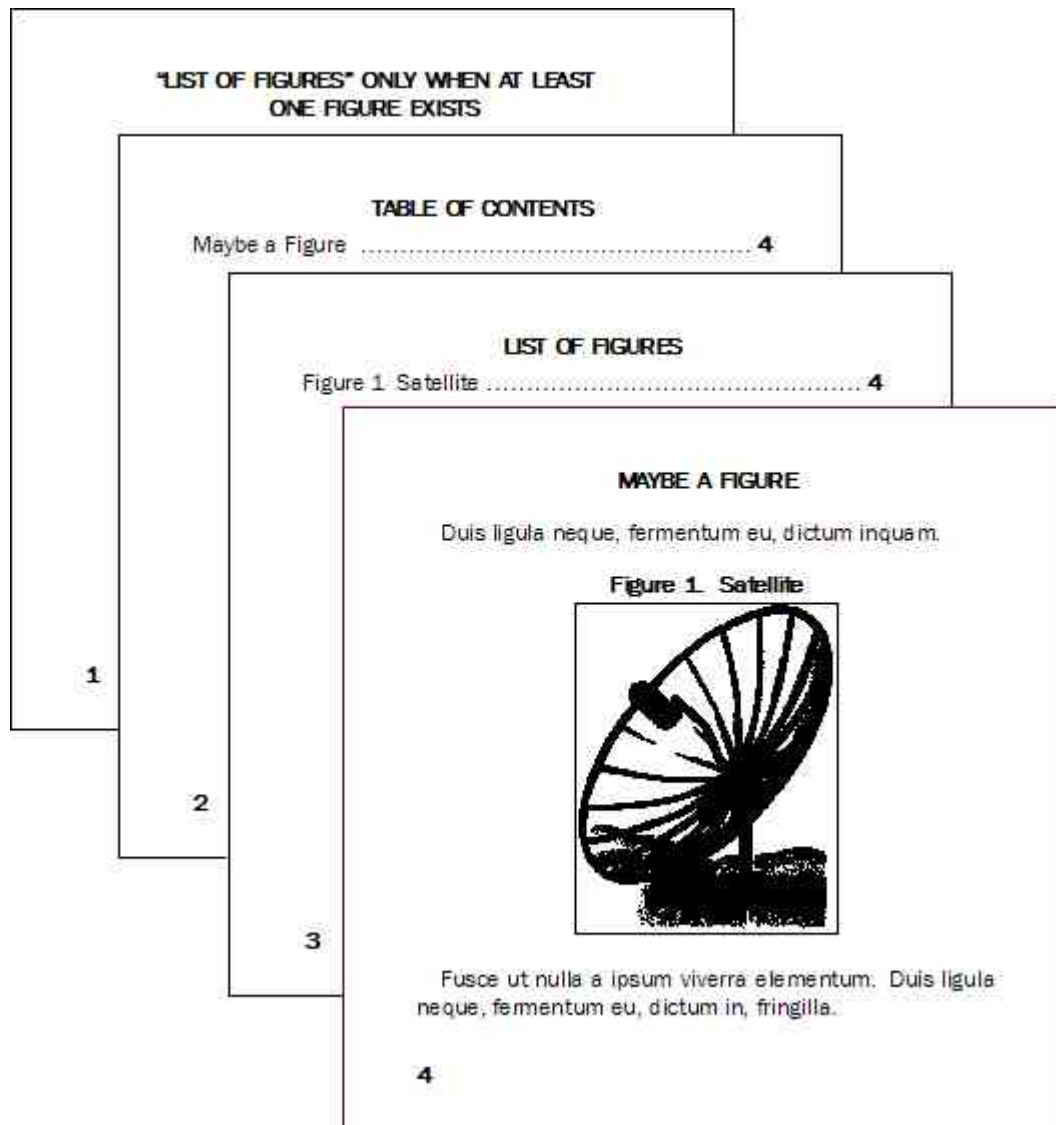
FOSI fragment

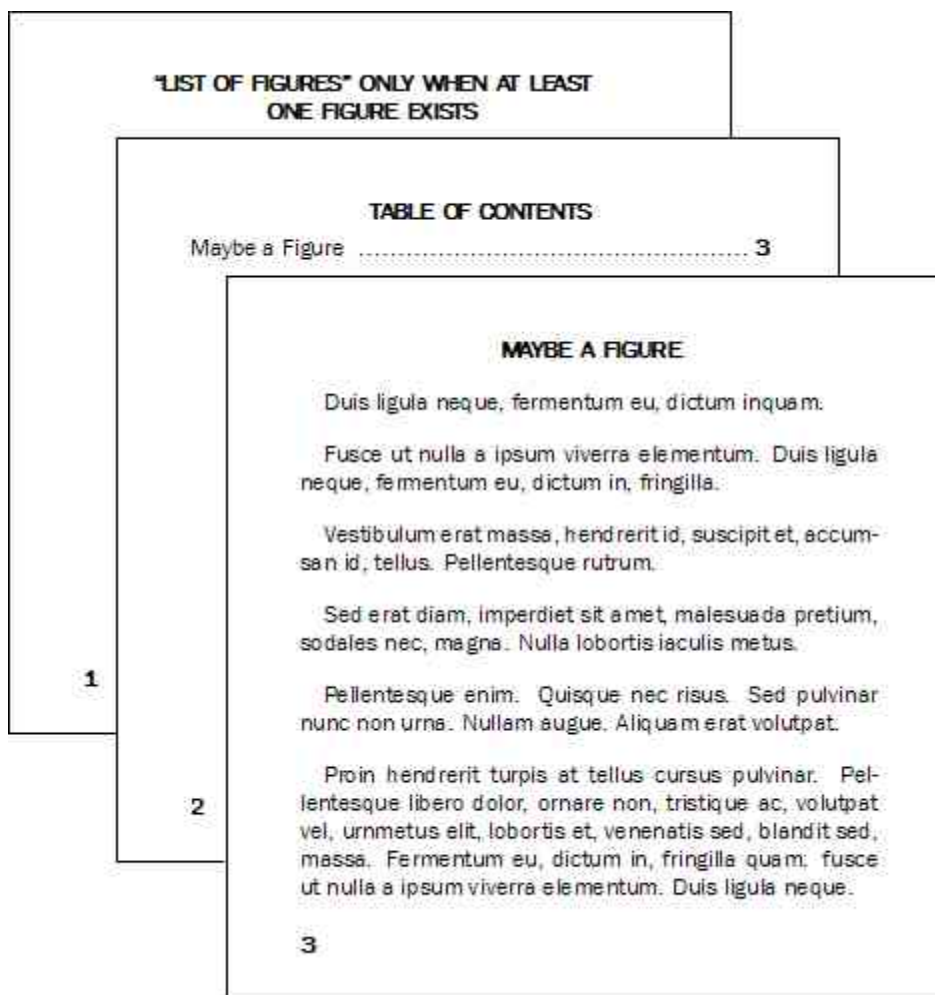
```
<floatloc
floatloc="note.bot" floattype="once" maxdepth="24pi" minspace="10pt"
nomspace="12pt" maxspace="14pt"> ...
<flowtext numcols="1" botfloat="note.bot">
...
<e-i-c gi="note">
<charlist inherit="1" charsubsetref="block">...</charlist>
<att>
<specval attloc="note" attname="float" attval="yes">
<charsubset charsubsetref="SUPPRESS">
<savetext placemnt="after" textid="floated-note.txt"
conrule="<floated-note.ctx>,
<floated-note-title.fmt>,\NOTE\,</floated-note-title.fmt>
,#CONTENT,</floated-note.ctx>">
<usetext placemnt="after" source="floated-note.txt">
<subchars charsubsetref='floated-note">
<float flidref="note.bot" width="page" pagetype="afterref"
scope="chapter">
</subchars>
</usetext>
</charsubset>
</att>
...
<e-i-c gi="figure-note-title.fmt" context="floated-note.ctx">
<charlist inherit="1" charsubsetref="title">
<font inherit="1" famname="Impact">
</charlist>
</e-i-c>
```

The following example uses the "do it once" programming pseudo-element technique to output "List of Tables" before the list of tables in the front matter, but only if the document has at least one table, with its required title. The first graphic shows the output when there is one `<figure>` in the document, as shown in the XML fragment. The second graphic shows the output when the document does not contain a `<figure>`, that is, when the highlighted portion of the XML file is deleted.

NOTE: The FOSI fragment shows two examples of one e-i-c with two element names (which are actually pseudo-elements).

Figure 47 Optional List of Figures



**XML fragment**

```

<document>
<front>
<titlepage>
<title>"List of Figures" only when at least one figure exists</title>
</titlepage>
<toc/>
</front>
<body>
<chapter>
<title>Maybe a figure</title>
<paragraph>Duis ligula neque, fermentum eu, dictum inquam.</paragraph>
<figure>

```

```

<title>Satellite</title>
<graphic name="satellite.tif"/>
</figure>
<paragraph>Fusce ut nulla ... </paragraph>
<paragraph>Vestibulum erat massa, ... </paragraph>
<paragraph>Sed erat diam ... </paragraph>
<paragraph>Pellentesque enim...</paragraph>
<paragraph> Proin hendrerit turpis ...</paragraph>
</chapter>
</body>
</document>

```

FOSI fragment

```

<stringdecl textid="maybe-figure-title.psu" literal=">
<stringdecl textid="lof.app" status="1">
<stringdecl textid="toc.app" status="1">

<e-i-c gi="document">
<charlist inherit="1">
<savetext textid="maybe-figure-title.psu" placemnt="before"
conrule="<maybe-figure-title.psu>,</maybe-figure-title.psu>">
<savetext textid="toc.app" conrule="<toc-title.fmt>,\Table of Contents\,
</toc-title.fmt>">
<savetext textid="toc.app" conrule="<toc.fmt>,\Maybe a Figure\,dotfill,\1\,
</toc.fmt>"
append="1">
</charlist>
</e-i-c>

<e-i-c gi="figure">
<charlist inherit="1" charsubsetref="block center">
<enumerat increm="1" enumid="figurect">
<savetext textid="figurect.txt" conrule="figurect">
<usetext source="maybe-figure-title.psu" placemnt="before"></usetext>
</charlist>
</e-i-c>

<e-i-c gi="title" context="figure">
<charlist inherit="1" charsubsetref="block prespace postspace bold">
<savetext textid="lof.app" append="1"
conrule="<lof.fmt>,\Figure \,figurect.txt,0.5em,#CONTENT,dotfill,
folioct.txt[B0],</lof.fmt>">
<usetext source="\Figure \,figurect.txt,\. \\" placemnt="before"></usetext>
</charlist>
</e-i-c>

<e-i-c gi="toc">
<charlist inherit="1" charsubsetref="block next-page">
<usetext source="toc.app"></usetext>
<usetext source="lof.app">
<subchars charsubsetref="next-page"></subchars>

```

```

</usetext>
</charlist>
</e-i-c>

<e-i-c gi="maybe-figure-title.psu">
<charlist inherit="1">
<savetext textid="lof.app" conrule="<lof-title.fmt>,\List of Figures\,
</lof-title.fmt>"
placemnt="before">
<savetext textid="maybe-figure-title.psu"
conrule="\" placemnt="after" append="0">
</charlist>
</e-i-c>

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

<e-i-c gi="lof-title.fmt toc-title.fmt">
<charlist inherit="1" charsubsetref="block center postspace allcaps bold">
</charlist>
</e-i-c>

```

The “Every nth” technique is an extension of the “do it once” programming pseudo-element technique. In the following example, outputting “NTH TIME” represents the formatting that would occur when the nth <counted- element> is counted.

Figure 48 Every nth

```

<stringdecl ="every-nth.psu">

<e-i-c gi="toptag">
<charlist>
<savetext textid="every-nth.psu" conrule="<count1.psu>,</count1.psu>"
placemnt="before">
</charlist>
</e-i-c>

<e-i-c gi="count1.psu">
charlist>
<savetext textid="every-nth.psu" conrule="<count2.psu>,</count2.psu>"
</charlist>
</e-i-c>
<e-i-c gi="count2.psu">
<charlist>
<savetext textid="every-nth.psu" conrule="<countn-1.psu>,</countn-1.psu>"
</charlist>
</e-i-c>

```

```

<e-i-c gi="countn-1.psu">
<charlist>
<savetext textid="every-nth.psu" conrule="<nth.psu>,</nth.psu>">
</charlist>
</e-i-c>

<e-i-c gi="nth.psu">
<charlist>
<usetext source="\NTH TIME\" placemnt="before"></usetext>
<savetext textid="every-nth.psu" conrule="<count1.psu>,</count1.psu>"
placemnt="after"></charlist>
</e-i-c>

<e-i-c gi="counted-element">
<charlist>
<usetext source="every-nth.psu"></usetext>
</charlist>
</e-i-c>

```

The next example shows how the attributes on ancestors of a pseudo-element can be used and tested by the pseudo-element `e-i-c`.

Figure 49

A charsubset may not contain attribute rules. However, a pseudo-element can be used for this purpose as shown in the next example.

Figure 50 Pseudo-element for shared attribute

FOSI fragment

```

<e-i-c gi="a">
<charlist inherit="1">
<usetext source="<.psu>,</.psu>"></usetext>
</charlist>
</e-i-c>

<e-i-c gi="b">
<charlist inherit="1">
<usetext source="<.psu>,</.psu>"></usetext>
</charlist>
</e-i-c>

<e-i-c gi=".psu">
<charlist inherit="1">
</charlist>
<att>
<specval>
<charsubset>
</charsubset>
</att>

```

```
<att>  
<fillval>  
See putgraph.xml/tutorial1-16-completed.fos  
<charsubset>  
</charsubset>  
</att>  
</e-i-c>
```