### Style Inheritance

In order to compile the complete set of paragraph and character properties specified by any given style (as appropriate), a consumer must follow the rule of style inheritance to determine each property in that set.

Style inheritance states that [styles](styles.docx) of any given [type](type.docx) may inherit from other [styles](styles.docx) of that [type](type.docx), and therefore a consumer must ‘build up’ the style information by following the inheritance tree. This inheritance is defined via the [basedOn](basedOn.docx) element, which specifies the styleId of the parent style.

[Example: The “Tristan Test” paragraph style can inherit properties from the “Heading 1” paragraph style, which itself can inherit properties from the “Normal” paragraph style. end example]

To build up the resulting style, a consumer must trace the hierarchy (following each [basedOn](basedOn.docx) value) back to a style which has no [basedOn](basedOn.docx) element (is not based on another style). The resulting style is then constructed by following each level in the tree, applying the specified paragraph and/or character properties as appropriate. When properties conflict, they are overridden by each subsequent level (this includes turning OFF a property set at an earlier level). Properties which are not specified simply do not change those specified at earlier levels.

[Example: Consider a character style Green which specifies only that the text [color](color.docx) is green, but inherits from another character style Base which defines a font face of Arial, as well as bold:

<w:[style](style.docx) w:[type](type.docx)="character" w:styleId="Green">

 <w:[name](name.docx) w:val="Green" />

 <w:[basedOn](basedOn.docx) w:val="Base" />

 <w:[rPr](rPr.docx)>

 <w:[color](color.docx) w:val="22B14C" />

 </w:[rPr](rPr.docx)>

</w:[style](style.docx)>

...

../Local [Settings](Settings.docx)/Temp/styles.xml<w:[style](style.docx) w:[type](type.docx)="character" w:styleId="Base">

 <w:[name](name.docx) w:val="Base" />

 <w:[rPr](rPr.docx)>

 <w:[rFonts](rFonts.docx) w:ascii="Arial" w:hAnsi="Arial" />

 <w:[b](b.docx) />

 </w:[rPr](rPr.docx)>

</w:[style](style.docx)>

The definition of the Green character style has a [basedOn](basedOn.docx) element which specifies the Base style. This means that any use of the Green style is defined as bold, green, Arial text. end example]

Conversely, a producer should not output any property on a style which has already been set by a previous level of the style hierarchy, as well as those which match the document defaults. This means that if the document defaults or any previous level in a style’s hierarchy specify a property which is unchanged at this level, that property should not be part of the style definition in the resulting WordprocessingML.

[Example: If the document default font is Bauhaus 93 and the Heading 1 style also specifies the Bauhaus 93 font, then a producer should not output any [rFonts](rFonts.docx) element for the Heading 1 style definition, because that formatting is inherited from the document defaults. end example]