

### 2.5.5 <config:config-item-map-named>

The <config:config-item-map-named> element is a container element for sequences, where each setting in the sequence is identified by its name.

The <config:config-item-map-named> element may have the following attribute: config:name 18.91.

The <config:config-item-map-named> element may have the following child element: <config:config-item-map-entry> 2.5.4.

## 2.6. Cursor Position Setting

To represent a text cursor position within a document, a processing instruction with PITarget opendocument (see §2.6 of [XML1.0]) **should** be used. The name of the cursor position processing instruction, cursor-position, **shall** follow the PITarget opendocument. The processing instruction may have arbitrary application specific attributes, for instance to connect the cursor position with a certain view of the document, where the views themselves are specified as application specific settings.

**Note 1:** A common view setting for editing applications is the position where the text cursor was while saving the document. For WYSIWYG applications, this usually will be a position within a paragraph only. For applications that provide an XML based view of the document, the cursor position could be also between arbitrary elements, or even within tags.

**Note 2:** Where a text cursor position is not sufficient to recreate a document view, applications may use arbitrary document specific settings in addition to the cursor position processing instruction. They may also use arbitrary document specific settings if the cursor position is not a text cursor position, but for instance a selection of drawing objects.

## 2.7. <office:scripts>

The <office:scripts> element contains all the <office:script> elements for a document. Each <office:script> element contains a script. A document may contain several scripts in different scripting languages.

**Note:** Scripts are not required to use XML document models. A script can operate a Document Object Model (DOM) composed from the XML representation of a document in OpenDocument format (see [DOM2]), or on an application specific API.

Scripts cannot modify a document while the document is loading. However, some events are called immediately after the document is loaded.

In addition to <office:script> elements, the <office:scripts> element may also contain <office:event-listeners> elements which contain the events assigned to the document itself. See 13.5.

The <office:scripts> element has no attributes.

The <office:scripts> element may have the following child elements: <office:event-listeners> 9.3.23 and <office:script> 2.7.1.

### 2.7.1 <office:script>

The <office:script> element contains a script.

The <office:script> element may have the following attribute: script:language 18.647.

The `<office:script>` element has no child elements.

Ed. Note: It seems to be reasonable to move some of the above text to this new section.

**Note:** In most situations, the element contains the source code of the script, but it may also contain a compiled version of the script or a link to external script code.

## 2.8. `<office:font-face-decls>`

A document in OpenDocument format may contain font face declarations. A font face declaration provides information about the fonts used by the author of a document. ref-style:font-face

The `<office:font-face-decls>` element has no attributes.

The `<office:font-face-decls>` element may have the following child element:  
`<style:font-face>` 15.9.

## 2.9. `<office:styles>`, `<office:automatic-styles>` and `<office:master-styles>`

The OpenDocument format supports the following types of *styles*:

- **Common styles**

A common style is a style chosen by a user for a document or some portion thereof.

- **Automatic styles**

An automatic style contains formatting properties that, in the user interface view of a document, are automatically assigned to an object such as a paragraph. The term *automatic* indicates that the style is generated automatically.

- **Master styles**

A master style contains formatting information and additional content that is displayed with the document content when the style is applied.

Note: Master pages are an example of a master style. The content that can be added by a master style depends on the type of document in question. A master page for a text document could include content for headers and footers. A master page for a presentation document could define the drawing shapes that appear in the background of a page.

Common and automatic styles have the same XML representation, but they are contained within two distinct container elements, as follows:

- common styles `<office:styles>`
- automatic styles `<office:automatic-styles>`

Master styles are contained within its own container element: `<office:master-styles>`.

See chapter 15.

The `<office:styles>` element has no attributes.

The `<office:styles>` element may have the following child elements: `<draw:fill-image>` 15.21.5, `<draw:gradient>` 15.21.1, `<draw:hatch>` 15.21.4, `<draw:marker>` 15.21.7, `<draw:opacity>` 15.21.6, `<draw:stroke-dash>` 15.21.8, `<number:boolean-style>` 15.14.6, `<number:currency-style>` 15.14.2, `<number:date-style>` 15.14.4, `<number:number-style>` 15.14.1, `<number:percentage-style>` 15.14.3,

<number:text-style> 15.14.7, <number:time-style> 15.14.5, <style:default-page-layout> 15.5, <style:default-style> 15.3, <style:presentation-page-layout> 15.22, <style:style> 15.2, <svg:linearGradient> Error: Reference source not found, <svg:radialGradient> Error: Reference source not found, <text:bibliography-configuration> 15.16.3, <text:linenumbering-configuration> 15.16.1, <text:list-style> 15.17, <text:notes-configuration> 15.16.2 and <text:outline-style> 15.18.

The <office:automatic-styles> element has no attributes.

The <office:automatic-styles> element may have the following child elements: <number:boolean-style> 15.14.6, <number:currency-style> 15.14.2, <number:date-style> 15.14.4, <number:number-style> 15.14.1, <number:percentage-style> 15.14.3, <number:text-style> 15.14.7, <number:time-style> 15.14.5, <style:page-layout> 15.4, <style:style> 15.2 and <text:list-style> 15.17.

The <office:master-styles> element has no attributes.

The <office:master-styles> element may have the following child elements: <draw:layer-set> 9.2.2, <style:handout-master> 9.2.1, <style:master-page> 15.6 and <table:table-template> 15.7.

## 2.10. Page Styles and Layout

The style and layout of the pages in a document are determined by:

- Page Layouts
- Master Pages

A *page layout* describes the physical properties or geometry of a page, for example, page size, margins, header height, and footer height.

A *master page* is a template for pages in a document. It contains a reference to a page layout which specifies the physical properties of the page and may also contain static content that is displayed on all pages in the document that use the master page. Examples of static content are headers, footers, or background graphics.

If a text or spreadsheet document is displayed in a paged layout, the master pages are instantiated to generate a sequence of pages containing the document content. When a master page is instantiated, an empty page is generated with the properties of the page master and the static content of the master page. The body of the page is then filled with content. If multiple pages in a document use the same master page, the master page can be instantiated several times within the document.

In text and spreadsheet documents, a master page can be assigned to paragraph and table styles using a `style:master-page-name` attribute. Each time the paragraph or table style is applied to text, a page break is inserted before the paragraph or table. A page that starts at the page break position uses the specified master page.

In drawings and presentations, master pages can be assigned to drawing pages using a `style:parent-style-name` attribute.

**Note:** The OpenDocument paging methodology differs significantly from the methodology used in [XSL]. In XSL, headers and footers are contained within page sequences that also contain the document content. In the OpenDocument format, headers and footers are contained in page styles. With either approach, the content of headers and footers can be changed or omitted without affecting the document content.

The `<table:last-column>` element may have the following attributes: `table:paragraph-style-name 18.1041` and `table:style-name 18.1076`.

The `<table:last-column>` element has no child elements.

#### 15.7.4 `<table:body>`, `<table:even-rows>`, `<table:odd-rows>`, `<table:even-columns>` and `<table:odd-columns>`

For the remaining cells, the cells styles can either be specified by the `<table:body>` element, or by the `<table:even-rows>/<table:odd-rows>` or `<table:even-columns>/<table:odd-columns>` element pairs if different cell styles should be applied to even and odd rows or columns.

The `<table:body>` element may have the following attributes: `table:paragraph-style-name 18.1041` and `table:style-name 18.1076`.

The `<table:body>` element has no child elements.

The `<table:even-rows>` element may have the following attributes: `table:paragraph-style-name 18.1041` and `table:style-name 18.1076`.

The `<table:even-rows>` element has no child elements.

The `<table:odd-rows>` element may have the following attributes: `table:paragraph-style-name 18.1041` and `table:style-name 18.1076`.

The `<table:odd-rows>` element has no child elements.

The `<table:even-columns>` element may have the following attributes: `table:paragraph-style-name 18.1041` and `table:style-name 18.1076`.

The `<table:even-columns>` element has no child elements.

The `<table:odd-columns>` element may have the following attributes: `table:paragraph-style-name 18.1041` and `table:style-name 18.1076`.

The `<table:odd-columns>` element has no child elements.

#### 15.8. `<table:background>`

The element `<table:background>` specifies the table style that shall be applied to the table itself.

The `<table:background>` element may have the following attribute: `table:style-name 18.1076`.

The `<table:background>` element has no child elements.

#### 15.9. `<style:font-face>`

OpenDocument font face declarations directly correspond to the `@font-face` font description of [CSS2] (see §15.3.1) and the `<font-face>` element of [SVG] (see §20.8.3), but have the following two extensions:

- OpenDocument font face declarations optionally may have a unique name. This name can be used inside styles (i.e., as attribute of `<style:text-properties>` element) as value of the `style:font-name` attribute to immediately select a font face declaration. If a font face

declaration is referenced this way, the steps described in §15.5 the [CSS2] font matching algorithms for selecting a font declaration based on the font-family, font-style, font-variant, font-weight and font-size descriptors will not take place, but the referenced font face declaration is used directly.

- Additional font descriptor attributes.

With the exception mentioned above, conforming applications should implement the CSS2 font matching algorithm as described in §15.5 the [CSS2], but they may also implement variants of it. They are especially allowed to implement a font matching based only on the font face declarations, that is, a font matching that is not applied to every character independently but only once for each font face declaration.

Font face declarations support the font descriptor attributes and elements described in §20.8.3 of [SVG].

The `<style:font-face>` element may have the following attributes: `style:font-adornments` 18.716, `style:font-charset` 18.717, `style:font-family-generic` 18.722, `style:font-pitch` 18.729, `style:name` 18.785, `svg:accent-height` 18.867, `svg:alphabetic` 18.868, `svg:ascent` 18.869, `svg:bbox` 18.870, `svg:cap-height` 18.871, `svg:descent` 18.873, `svg:font-family` 18.877, `svg:font-size` 18.878, `svg:font-stretch` 18.879, `svg:font-style` 18.880, `svg:font-variant` 18.881, `svg:font-weight` 18.882, `svg:hanging` 18.887, `svg:ideographic` 18.889, `svg:mathematical` 18.890, `svg:overline-position` 18.894, `svg:overline-thickness` 18.895, `svg:panose-1` 18.896, `svg:slope` 18.901, `svg:stemh` 18.903, `svg:stemv` 18.904, `svg:strikethrough-position` 18.907, `svg:strikethrough-thickness` 18.908, `svg:underline-position` 18.914, `svg:underline-thickness` 18.915, `svg:unicode-range` 18.916, `svg:units-per-em` 18.917, `svg:v-alphabetic` 18.918, `svg:v-hanging` 18.919, `svg:v-ideographic` 18.920, `svg:v-mathematical` 18.921, `svg:widths` 18.924 and `svg:x-height` 18.926.

The `<style:font-face>` element may have the following child elements: `<svg:definition-src>` 15.13 and `<svg:font-face-src>` 15.9.1.

### 15.9.1 `<svg:font-face-src>`

See §20.8.3 of [SVG].

The `<svg:font-face-src>` element has no attributes.

The `<svg:font-face-src>` element may have the following child elements: `<svg:font-face-name>` 15.12 and `<svg:font-face-uri>` 15.10.

### 15.10. `<svg:font-face-uri>`

See § 20.8.3 of [SVG].

The `<svg:font-face-uri>` element may have the following attributes: `xlink:actuate` 18.1240, `xlink:href` 18.1241 and `xlink:type` 18.1244.

The `<svg:font-face-uri>` element may have the following child element: `<svg:font-face-format>` 15.11.

### 15.11. `<svg:font-face-format>`

See §20.8.3 of [SVG].

The `<svg:font-face-format>` element may have the following attribute: `svg:string` 18.909.

The `<svg:font-face-format>` element has no child elements.

## 15.12. `<svg:font-face-name>`

See §X of [SVG].

The `<svg:font-face-name>` element may have the following attribute: `svg:name` 18.891.

The `<svg:font-face-name>` element has no child elements.

## 15.13. `<svg:definition-src>`

See § 20.8.3 of [SVG].

The `<svg:definition-src>` element may have the following attributes: `xlink:actuate` 18.1240, `xlink:href` 18.1241 and `xlink:type` 18.1244.

The `<svg:definition-src>` element has no child elements.

## 15.14. Data Styles

Data styles describe how to display different types of data, for example, a number or a date. The elements and attributes that are used to represent data styles are contained in the namespace `urn:oasis:names:tc:opendocument:xmlns:datastyle:1.0`. The prefix `number` denotes the data styles namespace.

This section describes the OpenDocument representation of the following data styles:

- Number style
- Currency style
- Percentage style
- Date style
- Boolean style
- Text style

### 15.14.1 `<number:number-style>`

The `<number:number-style>` element describes the style for decimal numbers.

These elements describe the display format of the number. The elements can be preceded or followed by `<number:text>` elements, which contain any additional text to be displayed before or after the number.

The `<number:number-style>` element may have the following attributes: `number:country` 18.548, `number:language` 18.555, `number:title` 18.563, `number:transliteration-country` 18.564, `number:transliteration-format` 18.565, `number:transliteration-language` 18.566, `number:transliteration-style` 18.567, `style:name` 18.785 and `style:volatile` 18.859.

## 18.408. fo:column-gap

The fo:column-gap attribute specifies the gap between columns for <style:columns> elements that do not contain individual <style:column> elements. If there are individual column elements, this attribute is ignored.

**Note:** This attribute has the same name as an [XSL] property but it is attached to a different element.

**Ed. Note** Same problem as with 18.396. Note that we omit percentage and inherit in the values.

The fo:column-gap attribute may be used with the following element: <style:columns> 16.3.2.

## 18.409. fo:country

See §7.9.1 of [XSL].

For styles, it is evaluated for any [UNICODE] characters that are not CJK or complex text layout (CTL) characters.

**Ed. Note** We omit “none” and “inherit.”

The fo:country attribute may be used with the following elements: <style:text-properties> 15.14.8.1, <text:alphabetical-index-source> 7.9.1 and <text:bibliography-configuration> 15.16.3.

## 18.410. fo:end-indent

See §7.10.8 of [XSL].

**Ed. Note** We omit “percentage” and “inherit.”

The fo:end-indent attribute may be used with the following element: <style:column> 16.8.2.

The default value for this attribute is 0cm.

## 18.411. fo:font-family

See §7.8.2 of [XSL].

The fo:font-family attribute is evaluated for any [UNICODE] character that is not a CJK or complex text layout (CTL) character.

**Ed. Note** XSL modifies the CSS definition to string. Does that mean font names are strings but still a list as per CSS?

The fo:font-family attribute may be used with the following element: <style:text-properties> 15.14.8.1.

## 18.412. fo:font-size

See §7.8.4 of [XSL].

This attribute is evaluated for any [UNICODE] character that is not a CJK or complex text layout (CTL) character.

Although it is recommended to use the `style:font-name` attribute, this attribute may be used to specify the properties of a font.

The `fo:font-size` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

Ed. Note We omit `absolute-size`, `relative-size` and `inherit`.

Ed. Note Deprecate?

Ed. Note I added the final paragraph based on model of other font property attributes.

## 18.413. `fo:font-style`

See §7.8.7 of [XSL].

This attribute is evaluated for any [UNICODE] character that is not a CJK or complex text layout (CTL) character.

Ed. Note We omit `backslant` and `inherit`.

The `fo:font-style` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

## 18.414. `fo:font-variant`

See §7.8.8 of [XSL].

Ed. Note In case we switch to XSL1.1, this must be 7.9.8.

Ed. Note We only omit `inherit`.

Ed. Note I would strike the following text because `fo:text-transform` was retained in CSS only for compatibility reasons and is not recommended for use in XSL. Suggest we deprecate `fo:text-transform` and say not to use it with `fo:font-variant`. If anyone does, that is clearly beyond the text of the standard. "For some implementations, the `fo:font-variant` and `fo:text-transform` attributes are mutually exclusive. If both attributes are used simultaneously and have different values than `normal` and `none`, then the result is undefined."

The `fo:font-variant` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

## 18.415. `fo:font-weight`

See §7.8.9 of [XSL].

This attribute is evaluated for any [UNICODE] character that is not a CJK or complex text layout (CTL) character.

The permitted values are `normal`, `bold`, and numeric values 100-900, in steps of 100. Unsupported numerical values are rounded off to the next supported value.

Ed. Note Why this doesn't fall under the `font:style-name`, etc. mechanism is unclear.

Ed. Note We omit `bolder`, `lighter`, `inherit`.

The `fo:font-weight` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.



The innermost enclosing area of the anchor defines the layout environment of the drawing shape.

If the value of the attribute is `true`, the following conditions must hold:

- The drawing shape flows with the text of its layout environment.
- The drawing shape does not leave its layout environment in the either text flow direction.

When a drawing shape leaves its layout environment due to its position and/or size in the forward text flow direction, it flows with the text flow to the next layout environment. If there is no next layout environment, the position of the layout environment is adjusted. If the drawing shape is larger than its layout environment, the size of the layout environment is adjusted accordingly.

If due to its position such a drawing shape would leave its layout environment in the backward text flow direction, the position is adjusted to the beginning of its layout environment.

In general such a drawing object is allowed to leave its layout environment in the other (non text flow) directions. Exceptions from this rule are the layout environments table cell area and frame area. For these layout environments it must be ensured that the drawing shape is laid out inside its layout environment by adjusting the position and (if necessary) the size.

If the value of the attribute is `false`, the drawing shape does not have to flow with the text of its layout environment. Thus, it can be laid out anywhere on the page its anchor is on.

This attribute is not applicable to drawing shapes that are anchored with `text:anchor-type="as-char"`, because such drawing shapes are treated as characters. For drawing shapes, which are anchored to a certain page with `text:anchor-type="page"`, the attribute is also not applicable, because such drawing shapes are to be laid out on the page they are anchored to.

Ed. Note I tried to adjust the language in the first paragraph following the second list. The non-text flow direction paragraph seems unclear to me but I don't have a suggested fix.

The `style:flow-with-text` attribute may be used with the following element:  
`<style:graphic-properties>` 16.14.

## 18.716. `style:font-adornments`

The `style:font-adornments` attributes specifies a characteristic of a font.

**Note:** A font characteristic includes bold or italic, which when used with a font family name, specifies some particular font.

The `style:font-adornments` attribute may be used with the following element:  
`<style:font-face>` 15.9.

## 18.717. `style:font-charset`

The `style:font-charset` attribute specifies the character set of a font.

The value of this attributes can be `x-symbol` or the character encoding in the notation described in the §4.3.3 of [XML1.0]. If the value is `x-symbol`, all characters that are displayed using this font must be contained in the [UNICODE] character range 0xf000 to 0xf0ff.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although the use of the `style:font-name` attribute is recommended, this attribute may be used to specify the properties of a font.

**Ed. Note** For this attribute as well as 18.693 and 18.694, do we mean to deprecate their use?

The `style:font-charset` attribute may be used with the following elements: `<style:font-face>` 15.9 and `<style:text-properties>` 15.14.8.1.

## 18.718. `style:font-charset-asian`

The `style:font-charset-asian` attribute specifies the character set of a font.

It is evaluated for [UNICODE] characters that are CJK characters.

The value of this attributes can be `x-symbol` or the character encoding in the notation described in the §4.3.3 of [XML1.0]. If the value is `x-symbol`, all characters that are displayed using this font must be contained in the [UNICODE] character range 0xf000 to 0xf0ff.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although the use of the `style:font-name-asian` attribute is recommended, this attribute may be used to specify the properties of a font.

**Ed. Note** For this attribute as well as 18.692 and 18.694, do we mean to deprecate their use?

The `style:font-charset-asian` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

## 18.719. `style:font-charset-complex`

The `style:font-charset`, `style:font-charset-asian` and `style:font-charset-complex` attributes specify the character set of a font.

The `style:font-charset-complex` attribute is evaluated for [UNICODE] characters that are complex text layout (CTL) characters.

The value of this attribute can be `x-symbol` or the character encoding in the notation described in the §4.3.3 of [XML1.0]. If the value is `x-symbol`, all characters that are displayed using this font must be contained in the [UNICODE] character range 0xf000 to 0xf0ff.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting properties element.

Although the use of the `style:font-name-complex` attribute is recommended, this attribute may be used to specify the properties of a font.

**Ed. Note** For this attribute as well as 18.692 and 18.693, do we mean to deprecate their use?

The `style:font-charset-complex` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

## 18.720. `style:font-family-asian`

The `style:font-family-asian` attribute specifies the font family for a text.

This is evaluated for [UNICODE] characters that are CJK characters.

Although the use of the `style:font-name-asian` attribute is recommended, this attribute may be used to specify the properties of a font. See §7.8.2 of [XSL].

**Ed. Note Shouldn't we deprecate this attribute?**

The `style:font-family-asian` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.721. style:font-family-complex

The `style:font-family-complex` attribute specifies the font family for a text.

This attribute is evaluated for [UNICODE] characters that are complex text layout (CTL) characters.

Although the use of the `style:font-name-complex` attribute is recommended, this attribute may be used to specify the properties of a font. See §7.8.2 of [XSL].

**Ed. Note Shouldn't we deprecate this attribute?**

The `style:font-family-complex` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.722. style:font-family-generic

The `style:font-family-generic` attribute specifies a generic font family name.

This attribute is evaluated for any [UNICODE] character other than CJK or complex text layout (CTL) characters.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although it is recommended to use the `style:font-name` attribute, this attribute may be used to specify the properties of a font.

**Ed. Note Deprecate?**

The `style:font-family-generic` attribute may be used with the following elements:  
<style:font-face> 15.9 and <style:text-properties> 15.14.8.1.

## 18.723. style:font-family-generic-asian

The `style:font-family-generic-asian` attribute specifies a generic asian font family name.

The `style:font-family-generic-asian` attribute is evaluated for [UNICODE] characters that are CJK characters.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although it is recommended to use the `style:font-name-asian` attribute, this attribute may be used to specify the properties of a font.

**Ed. Note Deprecate?**

The `style:font-family-generic-asian` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

## 18.724. `style:font-family-generic-complex`

The `style:font-family-generic-complex` attribute specifies a generic font family name.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although it is recommended to use the `style:font-name-complex` attribute, this attribute may be used to specify the properties of a font.

Ed. Note Deprecate?

The `style:font-family-generic-complex` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

## 18.725. `style:font-independent-line-spacing`

The `style:font-independent-line-spacing` attribute specifies if font independent line spacing is used. If the attribute value is `true`, then the line height is calculated only from the font height as specified by the font size attributes `fo:font-size`, `style:font-size-asian` and `style:font-size-complex`. If the value is `false`, the font metric of the actual font is taken into account.

The `style:font-independent-line-spacing` attribute may be used with the following element: `<style:paragraph-properties>` 16.6.

## 18.726. `style:font-name`

The `style:font-name` attribute uses the name of a font that has been declared by a `<style:font-face>` element within an `<office:font-face-decls>` element.

This attribute is evaluated for any [UNICODE] character other than CJK or complex text layout (CTL) characters.

Ed. Note I deleted: "The `style:font-name` attribute species the name of a font that is used to display a bullet character." which occurred under our treatment of list styles. Or should that be explicit?

The `style:font-name` attribute may be used with the following elements: `<style:list-level-properties>` 16.13 and `<style:text-properties>` 15.14.8.1.

## 18.727. `style:font-name-asian`

The `style:font-name-asian` attribute uses the name of the font that is declared by a `<style:font-face>` element within the `<office:font-face-decls>` element.

This attribute is evaluated for [UNICODE] characters that are CJK characters.

The `style:font-name-asian` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

## 18.728. style:font-name-complex

The `style:font-name-complex` attribute uses the name of the font that is declared by a `<style:font-face>` element within the `<office:font-face-decls>` element.

The `style:font-name-complex` attribute is evaluated for [UNICODE] characters that are complex text layout (CTL) characters.

**Ed. Note** I had to change the language for this attribute from “specifies” to “uses” as the font name is actually declared elsewhere and this attribute and the other font name attributes simply make use of that name for reference purposes.

The `style:font-name-complex` attribute may be used with the following element:  
`<style:text-properties>` 15.14.8.1.

## 18.729. style:font-pitch

The `style:font-pitch` attribute specifies whether a font has a fixed or variable width.

This attribute is evaluated for any [UNICODE] character other than CJK or complex text layout (CTL) characters.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although it is recommended to use the `style:font-name` attribute, this attribute may be used to specify the properties of a font.

**Ed. Note** Deprecate?

The `style:font-pitch` attribute may be used with the following elements: `<style:font-face>` 15.9 and `<style:text-properties>` 15.14.8.1.

## 18.730. style:font-pitch-asian

The `style:font-pitch-asian` attribute specifies whether a font has a fixed or variable width.

This attribute is evaluated for [UNICODE] characters that are CJK characters.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although it is recommended to use the `style:font-name-asian` attribute, this attribute may be used to specify the properties of a font.

**Ed. Note** Deprecate?

The `style:font-pitch-asian` attribute may be used with the following element:  
`<style:text-properties>` 15.14.8.1.

## 18.731. style:font-pitch-complex

The `style:font-pitch-complex` attribute specifies whether a font has a fixed or variable width.

This attribute is evaluated for [UNICODE] characters that are complex text layout (CTL) characters.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although it is recommended to use the `style:font-name-complex` attribute, this attribute may be used to specify the properties of a font.

Ed. Note Deprecate?

The `style:font-pitch-complex` attribute may be used with the following element:  
`<style:text-properties>` 15.14.8.1.

## 18.732. style:font-relief

The `style:font-relief` attribute specifies whether a font should be embossed, engraved, or neither.

Ed. Note This would be the perfect place to use the `style:font-name` mechanism but I can't because these properties are not listed under the definition used for `<style:font-face>` attributes. Consider adding these properties there, probably future release.

The `style:font-relief` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

## 18.733. style:font-size-asian

The `style:font-size-asian` attribute specifies the size of a font.

This attribute is evaluated for [UNICODE] characters that are CJK characters.

The value of this attribute is either an absolute length or a percentage as described in §7.8.4 of [XSL]. In contrast to XSL, percentage values can be used within common styles only and are based on the font height of the parent style rather than to the font height of the attributes neighborhood. Absolute font heights such as `medium`, `large`, `x-large`, and so on, and relative font heights such as `smaller`, and `larger` are not supported.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although it is recommended to use the `style:font-name-asian` attribute, this attribute may be used to specify the properties of a font.

Ed. Note Deprecate?

Ed. Note I added the two final paragraphs based on the model of other attributes with font properties.

The `style:font-size-asian` attribute may be used with the following element:  
`<style:text-properties>` 15.14.8.1.

## 18.734. style:font-size-complex

The `fo:font-size`, `style:font-size-asian` and `style:font-size-complex` attributes specify the size of a font.

This attribute is evaluated for [UNICODE] characters that are complex text layout (CTL) characters.

The value of this attribute is either an absolute length or a percentage as described in §7.8.4 of [XSL]. In contrast to XSL, percentage values can be used within common styles only and are based on the font height of the parent style rather than to the font height of the attributes neighborhood. Absolute font heights such as medium, large, x-large, and so on, and relative font heights such as smaller, and larger are not supported.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although it is recommended to use the `style:font-name-complex` attribute, this attribute may be used to specify the properties of a font.

Ed. Note In case we switch to XSL1.1, this must be 7.9.4.

Ed. Note Deprecate?

Ed. Note I added the two final paragraphs based on the model of other attributes with font properties.

The `style:font-size-complex` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.735. style:font-size-rel

The `style:font-size-rel` attribute specifies a relative font size change.

This attribute is evaluated for any [UNICODE] character other than CJK or complex text layout (CTL) characters.

This attribute specifies a relative font size change as a length. It cannot be used within automatic styles. This attribute changes the font size based on the font size of the parent style.

Ed. Note Consider removing the restriction and moving wholly into styles with `style:font-name`, etc.

The `style:font-size-rel` attribute may be used with the following element: <style:text-properties> 15.14.8.1.

## 18.736. style:font-size-rel-asian

The `style:font-size-rel-asian` attributes specifies a relative font size change.

This attribute is evaluated for [UNICODE] characters that are CJK characters.

This attribute specifies a relative font size change as a length. It cannot be used within automatic styles. This attribute changes the font size based on the font size of the parent style.

Ed. Note Consider removing the restriction and moving wholly into styles with `style:font-name`, etc.

The `style:font-size-rel-asian` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.737. style:font-size-rel-complex

The `style:font-size-rel`, `style:font-size-rel-asian` and `style:font-size-rel-complex` attributes specify a relative font size change.

The `style:font-size-rel-complex` attribute is evaluated for [UNICODE] characters that are complex text layout (CTL) characters.

This attribute specifies a relative font size change as a length. It cannot be used within automatic styles. This attribute changes the font size based on the font size of the parent style.

**Ed. Note** Consider removing the restriction and moving wholly into styles with `style:font-name`, etc.

The `style:font-size-rel-complex` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.738. style:font-style-asian

The `style:font-style-asian` attribute specifies whether to use normal or italic font face. See §7.8.7 of [XSL].

This attribute is evaluated for [UNICODE] characters that are CJK characters.

**Ed. Note** Why this doesn't fall under the `font:style-name`, etc. mechanism is unclear.

The `style:font-style-asian` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.739. style:font-style-complex

The `style:font-style-complex` attribute specifies whether to use normal or italic font face. See §7.8.7 of [XSL].

This attribute is evaluated for [UNICODE] characters that are complex text layout (CTL) characters.

The `fo:font-style` attribute is evaluated for any other [UNICODE] character.

**Ed. Note** Why this doesn't fall under the `font:style-name`, etc. mechanism is unclear.

The `style:font-style-complex` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.740. style:font-style-name

The `style:font-style-name` attribute specifies a font style name.

This attribute is evaluated for any [UNICODE] character that is not a CJK or complex text layout (CTL) character.

These attributes are ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting properties element.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although it is recommended to use the `style:font-name` attribute, this attribute may be used to specify the properties of a font.

**Ed. Note** Deprecate?

The `style:font-style-name` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.



## 18.741. style:font-style-name-asian

The `style:font-style-name-asian` attribute specifies a font style name.

The `style:font-style-name-asian` attribute is evaluated for [UNICODE] characters that are CJK characters.

These attributes are ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting properties element.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although it is recommended to use the `style:font-name-asian` attribute, this attribute may be used to specify the properties of a font.

**Ed. Note Deprecate?**

The `style:font-style-name-asian` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.742. style:font-style-name-complex

The `style:font-style-name-complex` attribute specifies a font style name.

The `style:font-style-name-complex` attribute is evaluated for [UNICODE] characters that are complex text layout (CTL) characters.

This attribute is ignored if there is no corresponding `fo:font-family` attribute attached to the same formatting property element.

Although it is recommended to use the `style:font-name-complex` attribute, this attribute may be used to specify the properties of a font.

**Ed. Note Deprecate?**

The `style:font-style-name-complex` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.743. style:font-weight-asian

The `style:font-weight-asian` attribute specifies the weight of a font. See §7.8.9 of [XSL].

This attribute is evaluated for [UNICODE] characters that are CJK characters.

The permitted values are `normal`, `bold`, and numeric values 100-900, in steps of 100. Unsupported numerical values are rounded off to the next supported value.

**Ed. Note Why this doesn't fall under the font:style-name, etc. mechanism is unclear.**

The `style:font-weight-asian` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.744. style:font-weight-complex

The `style:font-weight-complex` attributes specify the weight of a font. See §7.8.9 of [XSL].

This attribute is evaluated for [UNICODE] characters that are complex text layout (CTL) characters.

The `fo:font-weight` attribute is evaluated for any other [UNICODE] character.

The permitted values are `normal`, `bold`, and numeric values 100-900, in steps of 100. Unsupported numerical values are rounded off to the next supported value.

**Ed. Note** Why this doesn't fall under the `font:style-name`, etc. mechanism is unclear.

The `style:font-weight-complex` attribute may be used with the following element:  
`<style:text-properties>` 15.14.8.1.

## 18.745. `style:footnote-max-height`

The `style:footnote-max-height` attribute specifies the maximum amount of space on the page that a footnote can occupy. The value of the attribute is a length, which determines the maximum height of the footnote area.

If the value of this attribute is set to `0in`, there is no limit to the amount of space that the footnote can occupy.

The `style:footnote-max-height` attribute may be used with the following element:  
`<style:page-layout-properties>` 16.3.

## 18.746. `style:glyph-orientation-vertical`

The `style:glyph-orientation-vertical` attribute specifies the vertical glyph orientation. The attribute specifies an angle or automatic mode. The only possible angle is `0`, which disables this feature.

**Ed. Note** What does “auto” mean?

The `style:glyph-orientation-vertical` attribute may be used with the following element:  
`<style:table-cell-properties>` 16.12.

## 18.747. `style:height`

The `style:height` attribute specifies the height of a column separator line. The value of this attribute is a percentage of the height of the columned area.

**Ed. Note** Relies on signed percentage so `-50%` is a legal value. I did change the prose to say: percentage of the height of the columned area, but that doesn't fix the schema.

The `style:height` attribute may be used with the following element: `<style:column-sep>` 16.8.3.

The default value for this attribute is `100%`.

## 18.748. `style:horizontal-pos`

The `style:horizontal-pos` attribute specifies the horizontal alignment of a frame in relation to a specific area for documents with the element `<office:text>`.

This attribute can have one of the following values: `from-left`, `left`, `center`, `right`, `from-inside`, `inside`, or `outside`.

The area where a position appears to is specified by the `style:horizontal-rel` attribute.

The values `from-inside`, `inside` and `outside` are used on pages with an even number and `from-left`, `left`, and `right` are used on pages with an odd page number. The value `center` may appear on odd or even numbered pages.

If the attribute value is `from-left` or `from-inside`, the `svg:x` attribute associated with the frame element specifies the horizontal position of the frame. Otherwise the `svg:x` attribute is ignored for text documents.

An `svg:x` attribute can be used within a graphic style. In that case, the attribute specifies a default position for new frames that are created using this style.

The following tables display the combinations of values of the attributes `style:horizontal-pos` and `style:horizontal-rel`. The values of these alignment attributes are listed in the first column on the left, and an alignment attribute value/anchor type value match is indicated by an X.

*Table 12 - Horizontal position values*

<b>Value of <i>style:horizontal-pos</i></b>	<b>Value of text : anchor - type</b>				
	<b><i>page</i></b>	<b><i>frame</i></b>	<b><i>paragraph</i></b>	<b><i>char</i></b>	<b><i>as-char</i></b>
any	X	X	X	X	

*Table 13 - Horizontal relation values*

<b>Value of <i>style:horizontal-rel</i></b>	<b>Value of text : anchor - type</b>				
	<b><i>page</i></b>	<b><i>frame</i></b>	<b><i>paragraph</i></b>	<b><i>char</i></b>	<b><i>as-char</i></b>
page	X		X	X	
page-content	X		X	X	
page-start-margin	X		X	X	
page-end-margin	X		X	X	
frame		X			
frame-content		X			
frame-start-margin		X			
frame-end-margin		X			
paragraph			X	X	
paragraph-content			X	X	

<i>Value of style:horizontal- rel</i>	<i>Value of text : anchor - type</i>				
	<i>page</i>	<i>frame</i>	<i>paragraph</i>	<i>char</i>	<i>as-char</i>
paragraph-start-margin			X	X	
paragraph-end-margin			X	X	
char				X	

The style:horizontal-pos attribute may be used with the following element:  
<style:graphic-properties> 16.14.

## 18.749. style:horizontal-rel

The style:horizontal-rel attribute specifies the area to which the horizontal position of a frame relates.

The value of this attribute can be one of the following: page, page-content, page-start-margin, page-end-margin, frame, frame-content, frame-start-margin, frame-end-margin, paragraph, paragraph-content, paragraph-start-margin, paragraph-end-margin, or char.

The value start-margin determines the left margin, except when the horizontal position is from-inside, inside or outside and the anchor for the frame is on a page with an even page number, in which case it determines the right margin. The value end-margin determines the opposite margin to the start-margin values.

The style:horizontal-rel attribute may be used with the following element:  
<style:graphic-properties> 16.14.

## 18.750. style:justify-single-word

The style:justify-single-word attribute specifies whether or not a single word should be justified when the last line in a paragraph is justified.

Specifying a style:justify-single-word attribute without specifying a fo:text-align and fo:text-align-last attribute has no effect. Unspecified, both fo:text-align and fo:text-align-last have the value start.

The style:justify-single-word attribute may be used with the following element:  
<style:paragraph-properties> 16.6.

## 18.751. style:language-asian

The style:language-asian attribute specifies the language of a text. See §7.9.2 of [XSL].

This attribute is evaluated for [UNICODE] characters that are CJK characters.

This attribute may be ignored if it is not specified together with a corresponding fo:country attribute.

**Ed. Note** What does “corresponding” mean? Valid for CJK? Or simply any fo:country attribute value?

The `style:language-asian` attribute may be used with the following element:  
`<style:text-properties>` 15.14.8.1.

## 18.752. `style:language-complex`

The `style:language-complex` attribute specifies the language of a text. See §7.9.2 of [XSL].

The `style:language-asian` attribute is evaluated for [UNICODE] characters that are CJK characters.

This attribute is evaluated for [UNICODE] characters that are complex text layout (CTL) characters.

This attribute may be ignored if it is not specified together with a corresponding `fo:country` attribute.

**Ed. Note** What does “corresponding” mean? Valid for CJK? Or simply any `fo:country` attribute value?

The `style:language-complex` attribute may be used with the following element:  
`<style:text-properties>` 15.14.8.1.

## 18.753. `style:layout-grid-base-height`

The `style:layout-grid-base-height` attribute specifies the height reserved in layout grid lines for non-ruby text.

The `style:layout-grid-base-height` attribute may be used with the following element:  
`<style:page-layout-properties>` 16.3.

## 18.754. `style:layout-grid-base-width`

The `style:layout-grid-base-width` attribute specifies a grid's base width. This attribute is only evaluated if `style:layout-grid-standard-mode` attribute has the value `true`.

The `style:layout-grid-base-width` attribute may be used with the following element:  
`<style:page-layout-properties>` 16.3.

## 18.755. `style:layout-grid-color`

The `style:layout-grid-color` attribute specifies the color of layout grid border lines.

The `style:layout-grid-color` attribute may be used with the following element:  
`<style:page-layout-properties>` 16.3.

## 18.756. `style:layout-grid-display`

The `style:layout-grid-display` attribute specifies whether layout grid border lines are displayed.

The `style:layout-grid-display` attribute may be used with the following element:  
`<style:page-layout-properties>` 16.3.

## 18.757. style:layout-grid-lines

The `style:layout-grid-lines` attribute specifies the number of layout grid lines per page. The number of lines actually displayed may be smaller than specified if the page does not have enough space to display the specified number of lines with the specified line height.

Ed. Note: Removed "(i.e., the sum of the base and ruby height)" as it does not apply to standard grid mode.

The `style:layout-grid-lines` attribute may be used with the following element:  
<style:page-layout-properties> 16.3.

## 18.758. style:layout-grid-mode

The `style:layout-grid-mode` attribute enables Asian layout grids. It has the following values:

- `none`: Disables the layout grid.
- `lines`: Enables a line layout, this is, the page is divided in a fixed number of lines. The exact number of lines depends on the other layout grid formatting properties. There is no space between the layout grid lines. The layout grid itself is centered on the page.
- `both`: Like `lines`, except that the lines are divided into rectangular cells. The calculation of the cells depends on the `style:layout-grid-standard-mode` property. Within a layout cell, nor more than one Asian [UNICODE] character is displayed. Asian characters that do not fit into a single cell are displayed centered into as many cells as required. Non Asian text is centered within as many cells as required.

The `style:layout-grid-mode` attribute may be used with the following element:  
<style:page-layout-properties> 16.3.

## 18.759. style:layout-grid-print

The `style:layout-grid-print` attribute specifies whether layout grid border lines are printed.

The `style:layout-grid-print` attribute may be used with the following element:  
<style:page-layout-properties> 16.3.

## 18.760. style:layout-grid-ruby-below

The `style:layout-grid-ruby-below` attribute specifies whether ruby text is displayed above or below the base text.

The `style:layout-grid-ruby-below` attribute may be used with the following element:  
<style:page-layout-properties> 16.3.

## 18.761. style:layout-grid-ruby-height

The `style:layout-grid-ruby-height` attribute specifies the height reserved in layout grid lines for ruby text. This attribute is only evaluated if `style:layout-grid-standard-mode` attribute has the value `false`.

The `style:layout-grid-ruby-height` attribute may be used with the following element:  
<style:page-layout-properties> 16.3.

## 18.762. style:layout-grid-snap-to

The `style:layout-grid-snap-to` attribute specifies whether the text is snapped to the grid or not. It is evaluated only if the `style:layout-grid-mode` attribute has the value `both` and the `style:layout-grid-standard-mode` attribute the value `true`.

Ed. Note: The proposal had a default value "True", but default values cannot be specified for formatting properties because they conflict with the style inheritance.

The `style:layout-grid-snap-to` attribute may be used with the following element: `<style:page-layout-properties>` 16.3.

## 18.763. style:layout-grid-standard-mode

The `style:layout-grid-standard-mode` attribute specifies how the rectangular grid cells are calculated if the `style:layout-grid-mode` attribute has the value `both`.

Ed. Note: Is it correct that the attribute is only calculated for the mode "both"?

If the value of the attribute is `true`, then the page is divided into a fixed numbers of lines, and the lines are divided into rectangular cells. The line height is specified by the `style:layout-grid-base-height` attribute, and the cell width is specified by `style:layout-grid-base-width` attribute. The number of cells per line depends on the grid base width. This mode is also called *standard paper mode*. The `style:layout-grid-ruby-height` attribute is ignored in this mode.

If the value of the attribute is `false`, then the page is divided in a fixed number of lines, and the lines are divided into square cells. The number of cells per grid line depends on the line height, where the line height is the sum of the base height and the ruby height as specified by the `style:layout-grid-base-height` and `style:layout-grid-ruby-height` attributes.

The `style:layout-grid-standard-mode` property can only be used in the default page layout. If the `style:layout-grid-standard-mode` attribute appears inside a `<style:page-layout>` element, then the attribute **shall** be ignored.

**Note:** This ensures that a single mode is used for the whole document.

The `style:layout-grid-standard-mode` attribute may be used with the following element: `<style:page-layout-properties>` 16.3.

## 18.764. style:leader-char

The `style:leader-char` attribute specifies a leader character.

The `style:leader-char` attribute may be used with the following element: `<text:index-entry-tab-stop>` 7.14.6.

## 18.765. style:leader-color

The `style:leader-color` attribute specifies the color of a leader line. The value of this attribute is either `font-color` or a color. If the value is `font-color`, the current text color is used for the leader line.

The `style:leader-color` attribute may be used with the following element: `<style:tab-stop>` 16.6.2.

## 18.766. style:leader-style

The `style:leader-style` attribute specifies a style for a leader line.

Ed. Note I deleted: "if and how a leader line is drawn." and note that the style has "none" as one possible value so it really does determine "if" a leader line is drawn. Suggest separating out appear/doesn't appear from style questions in some future version.

The `style:leader-style` attribute may be used with the following element: `<style:tab-stop>` 16.6.2.

## 18.767. style:leader-text

The `style:leader-text` attribute specifies a single Unicode character for use as leader text for tab stops. If the value of this attribute is more than one Unicode character, only the first character is used.

The default value of this attribute is a whitespace.

If an application supports only textual leaders of a particular type, if this attribute has any value that is not supported by the application, the application should use its textual leader values.

Textual leaders have a higher priority than line leaders, even if the leader text that is specified has to be adapted to be usable by the application.

This attribute is associated with the `<style:tab-stop>` element and its value must be a single [UNICODE] character.

Ed. Note The schema does not enforce the single Unicode character restriction. I have re-cast this text as it seemed deeply confusing to me. How an application could not support "textual" and still use any of the line leaders, which are also Unicode characters is beyond me. I suspect we need to change the substantive requirements of this attribute to simply use the specified text leader if specified and the line leader if not.

The `style:leader-text` attribute may be used with the following element: `<style:tab-stop>` 16.6.2.

The default value for this attribute is .

## 18.768. style:leader-text-style

The `style:leader-text-style` specifies a text style that is applied to a textual leader. It is not applied to leader lines. If the attribute appears in an automatic style, it may reference either an automatic text style or a common style. If the attribute appears in a common style, it may reference a common style only.

Ed. Note Should we eliminate the distinction between the styles?

The `style:leader-text-style` attribute may be used with the following element: `<style:tab-stop>` 16.6.2.

## 18.769. style:leader-type

The `style:leader-type` attribute specifies whether a leader line should be drawn, and if so, whether a single or double line will be used.



The `style:leader-type` attribute may be used with the following element: `<style:tab-stop>` 16.6.2.

### 18.770. `style:leader-width`

The `style:leader-width` attribute specifies the width of a leader line.

The `style:leader-width` attribute may be used with the following element: `<style:tab-stop>` 16.6.2.

### 18.771. `style:legend-expansion`

The `style:legend-expansion` attribute specifies the direction in which a legend expands. Values of `wide` and `high` cause the legend to be expanded horizontally and vertically. A value of `balanced` causes expansion into both directions. A value of `custom` with a numeric `style:legend-expansion-aspect-ratio` causes the legend to be expanded according to the specified ratio.

The `style:legend-expansion` attribute may be used with the following element: `<chart:legend>` 10.5.

### 18.772. `style:legend-expansion-aspect-ratio`

The `style:legend-expansion-aspect-ratio` attribute specifies the ratio between width and height for a `style:legend-expansion` attribute with value `custom`.

The `style:legend-expansion-aspect-ratio` attribute may be used with the following element: `<chart:legend>` 10.5.

### 18.773. `style:length`

The `style:length` attribute specifies the number of characters that are displayed in a larger font.

The value of this attribute can be a number or `word`, which specifies that the first word should be displayed in a larger font.

Ed. Note While “word” is no doubt convenient, isn't that locale specific?

The `style:length` attribute may be used with the following element: `<style:drop-cap>` 16.6.3.

The default value for this attribute is `1`.

### 18.774. `style:line-break`

The `style:line-break` attribute specifies a set of line breaking rules to use with a text. If the value is `strict`, line breaks are forbidden between certain user and application configurable characters. If the value is `normal`, line breaks may occur between arbitrary characters.

Ed. Note A reference here to “certain user and application configurable characters” would be useful here.

The `style:line-break` attribute may be used with the following element:  
<style:paragraph-properties> 16.6.

### 18.775. style:line-style

The `style:line-style` attribute specifies the style of a footnote separator line.

The `style:line-style` attribute may be used with the following element:  
<style:footnote-sep> 16.3.3.

### 18.776. style:lines

The `style:lines` attribute specifies the number of text lines which a font spans. If the value of this attribute is 1, <style:drop-caps> is disabled.

Ed. Note The text “If the value of this attribute is 1 or 0, drop caps is disabled.” is incorrect. Datatype is `positiveInteger`. Do we mean to have a priority of `style:lines` over `style:drop-caps`? I don't have any feelings one way or the other but we should be explicit about that choice.

The `style:lines` attribute may be used with the following element: <style:drop-cap>  
16.6.3.

The default value for this attribute is 1.

### 18.777. style:letter-kerning

The `style:letter-kerning` attribute specifies whether kerning between characters is enabled or disabled kerning.

The `style:letter-kerning` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

### 18.778. style:line-height-at-least

The `style:line-height-at-least` attribute specifies a minimum line height. The value of this attribute is a length.

The effect of this attribute is disabled when `fo:line-height` has the value of `normal`.

The `style:line-height-at-least` attribute may be used with the following element:  
<style:paragraph-properties> 16.6.

### 18.779. style:line-spacing

The `style:line-spacing` attribute specifies a fixed distance between two lines.

The effect of this attribute is disabled when `fo:line-height` has the value of `normal`.

The `style:line-spacing` attribute may be used with the following element:  
<style:paragraph-properties> 16.6.

## 18.780. style:list-style-name

The `style:list-style-name` attribute specifies a list style for style families with paragraph formatting properties. This applies to automatic and common styles.

The list style specified by this attribute is applied to headings and paragraphs that are contained in a list, where the list does not specify a list style itself, and the list has no list style specification for any of its parents.

The `style:list-style-name` attribute value can be empty. If empty, this attribute does not inherit a list style value from a parent style.

Ed. Note Reconsider the automatic and common styles issue. I deleted: "Like formatting properties, list style information is inherited from parent styles. The `style:list-style-name` attribute value can be empty. In that case, an association with a list style that is inherited from a parent style will be removed." and replaced it with the text shown. I assume that is correct if the idea is that inheritance is interrupted by the empty value. The attribute should be inheriting a list style and not an association with a list style.

The `style:list-style-name` attribute may be used with the following element:  
<style:style> 15.2.

## 18.781. style:master-page-name

The `style:master-page-name` attribute defines a master page for a paragraph or table style. This applies to automatic and common styles.

If this attribute is associated with a style, a page break is inserted when the style is applied and the specified master page is applied to the resulting page.

This attribute is ignored if it is associated with a paragraph style that is applied to a paragraph within a table.

The `style:master-page-name` attribute may be used with the following element:  
<style:style> 15.2.

## 18.782. style:may-break-between-rows

The `style:may-break-between-rows` attribute specifies that a page break may occur inside a table.

The `style:may-break-between-rows` attribute may be used with the following element:  
<style:table-properties> 16.9.

## 18.783. style:min-row-height

The `style:min-row-height` attribute specifies a fixed minimum height for a row.

Ed. Note In the 2<sup>nd</sup> edition, at 15.10.1 we say: "The `style:row-height` property specifies a fixed row height, while the `style:min-row-height` property specifies a fixed height." Shouldn't the second clause say "fixed minimum height"?

The `style:min-row-height` attribute may be used with the following element:  
<style:table-row-properties> 16.11.

## 18.784. style:mirror

The `style:mirror` attribute specifies whether or not an image is mirrored before it is displayed. The mirroring can be vertical or horizontal or both.

The value of this attribute can be `none`, `vertical`, `horizontal`, `horizontal-on-odd`, or `horizontal-on-even`. The value `vertical` and the various horizontal values can be specified together, separating them by a white space.

The value `vertical` and the various horizontal values can be specified together, separating them by a white space.

Ed. Note I deleted: "Horizontal mirroring can be restricted to images that are only located on either odd or even pages." as it only partially explained one set of values. I am not certain what the use case is for vertical and horizontal mirroring at the same time but simply note it here in case that needs to be changed.

The `style:mirror` attribute may be used with the following element: `<style:graphic-properties>` 16.14.

## 18.785. style:name

The `style:name` attribute specifies the name of a:

- 
- data style - It can be used with all data style elements.
- font declaration – The unique name of a font declaration. This name can be used inside styles (i.e., as an attribute of the `<style:text-properties>` element) as value of the `style:font-name` attribute to immediately select a font face declaration
- list style
- master page - Each master page is referenced using the page name. The name specified must be unique to the document instance.
- outline style - The outline style as a list style can be referenced by the `style:list-style-name` attribute.
- page layout
- style – With the `style:family` attribute, the `style:name` attribute uniquely identifies a style. The `<office:styles>`, `<office:automatic-styles>` and `<office:master-styles>` elements each must not contain two styles with the same family and the same name.

For automatic styles, a name is generated during document export. If the document is exported several times, it cannot be assumed that the same name is generated each time.

In an XML document, the name of each style is a unique name that may be independent of the language selected for an office applications user interface.

The `style:name` attribute may be used with the following elements: `<number:boolean-style>` 15.14.6, `<number:currency-style>` 15.14.2, `<number:date-style>` 15.14.4, `<number:number-style>` 15.14.1, `<number:percentage-style>` 15.14.3, `<number:text-style>` 15.14.7, `<number:time-style>` 15.14.5, `<style:font-face>` 15.9, `<style:master-page>` 15.6, `<style:page-layout>` 15.4, `<style:presentation-`

`page-layout`> 15.22, `<style:style>` 15.2, `<text:list-style>` 15.17 and `<text:outline-style>` 15.18.

## 18.786. `style:next-style-name`

The `style:next-style-name` attribute specifies the style to be used with pages and paragraphs.

Within `<office:text>` and `<office:spreadsheet>` documents, the `style:next-style-name` attribute identifies the master page that is used for the next page if the current page is entirely filled. This attribute is optional. If the next style name is not specified, the current master page is used for the next page. The value of this attribute must be the name of another `<style:master-page>` element.

Within styles for paragraphs, `style:next-style-name` attribute specifies the style to be used for the next paragraph if a paragraph break is inserted in the user interface. By default, the current style is used as the next style.

The `style:next-style-name` attribute may be used with the following elements:  
`<style:master-page>` 15.6 and `<style:style>` 15.2.

## 18.787. `style:num-format`

The `style:num-format` attribute specifies a numbering sequence. The supported number sequences:

- Numeric: 1, 2, 3, ...
- Alphabetic: a, b, c, ... or A, B, C, ...
- Roman: i, ii, iii, iv, ... or I, II, III, IV,...

The value of this attribute can be "1", "a", "A", "i", or "I". For some elements, the attribute value also can be empty. In that case, no number is displayed.

If no value is given, no number sequence is displayed.

Note: Misnaming. What is illustrated here is not a format. Yes, XSLT has some unfortunate language but what is being described is a format token, not a format. I added the no value given part to cover the use for page numbers where no value is given.

The `style:num-format` attribute may be used with the following elements: `<style:page-layout-properties>` 16.3, `<text:character-count>` 6.5.17.4, `<text:database-row-number>` 6.6.5, `<text:image-count>` 6.5.17.6, `<text:linenumbers-configuration>` 15.16.1, `<text:list-level-style-number>` 15.17.2, `<text:notes-configuration>` 15.16.2, `<text:object-count>` 6.5.17.7, `<text:outline-level-style>` 15.18.1, `<text:page-count>` 6.5.17.1, `<text:page-number>` 6.3.3, `<text:page-variable-get>` 6.7.1.2, `<text:paragraph-count>` 6.5.17.2, `<text:sequence>` 6.4.12, `<text:table-count>` 6.5.17.5 and `<text:word-count>` 6.5.17.3.

## 18.788. `style:num-letter-sync`

The `style:num-letter-sync` attribute specifies whether letter synchronization shall take place. If letters are used in alphabetical order for numbering, there are two ways to process overflows within a digit, as follows:

- True: A new digit is inserted. Its start value is 'a' or 'A', and it is incremented every time an overflow occurs in the following digit. The numbering sequence (for lower case numberings) in that case is a,b,c, ..., z, aa, ab, ac, ...,az, ba, ..., and so on.
- False: A new digit is inserted that always has the same value as the following digit. The numbering sequence (for lower case numberings) in that case is a, b, c, ..., z, aa, bb, cc, ..., zz, aaa, ..., and so on.

Ed. Note I inserted the true/false values as the descriptions were in reverse order. After fixing that, it seemed clearer simply to say what was meant.

The style:num-letter-sync attribute may be used with the following elements: <style:page-layout-properties> 16.3, <text:character-count> 6.5.17.4, <text:database-row-number> 6.6.5, <text:image-count> 6.5.17.6, <text:linenumering-configuration> 15.16.1, <text:list-level-style-number> 15.17.2, <text:notes-configuration> 15.16.2, <text:object-count> 6.5.17.7, <text:outline-level-style> 15.18.1, <text:page-count> 6.5.17.1, <text:page-number> 6.3.3, <text:page-variable-get> 6.7.1.2, <text:paragraph-count> 6.5.17.2, <text:sequence> 6.4.12, <text:table-count> 6.5.17.5 and <text:word-count> 6.5.17.3.

## 18.789. style:num-prefix

The style:num-prefix attribute specifies what to display before a number.

If the style:num-prefix and style:num-suffix values do not contain any character that has a Unicode category of Nd, NI, No, Lu, Ll, Lt, Lm or Lo, an [XSLT] format attribute can be created from the OpenDocument attributes by concatenating the values of the style:num-prefix, style:num-format, and style:num-suffix attributes.

The style:num-prefix attribute can also specify a character before the value of a text:bullet-char attribute.

The style:num-prefix attribute may be used with the following elements: <style:page-layout-properties> 16.3, <text:list-level-style-bullet> 15.17.1, <text:list-level-style-number> 15.17.2, <text:notes-configuration> 15.16.2 and <text:outline-level-style> 15.18.1.

## 18.790. style:num-suffix

The style:num-prefix and style:num-suffix attributes specify what to display before and after the number.

If the style:num-prefix and style:num-suffix values do not contain any character that has a Unicode category of Nd, NI, No, Lu, Ll, Lt, Lm or Lo, an [XSLT] format attribute can be created from the OpenDocument attributes by concatenating the values of the style:num-prefix, style:num-format, and style:num-suffix attributes.

The style:num-suffix attribute can also specify a character after the value of a text:bullet-char attribute.

The style:num-suffix attribute may be used with the following elements: <style:page-layout-properties> 16.3, <text:list-level-style-bullet> 15.17.1, <text:list-level-style-number> 15.17.2, <text:notes-configuration> 15.16.2 and <text:outline-level-style> 15.18.1.

## 18.791. style:number-wrapped-paragraphs

The `style:number-wrapped-paragraphs` attribute specifies the number of paragraphs that can wrap around a frame if the anchor position of a frame or drawing shape is a paragraph or a character, and the wrap mode specified by the `style:wrap` attribute is `left`, `right`, `parallel`, or `dynamic`.

This attribute is only recognized in frames or styles that have a `style:wrap` attribute attached with a value of `left`, `right`, `parallel`, or `dynamic`.

If the value is `no-limit`, there is no limit on the number of paragraphs that are allowed to wrap around a frame.

The `style:number-wrapped-paragraphs` attribute may be used with the following element: `<style:graphic-properties>` 16.14.

## 18.792. style:overflow-behavior

The `style:overflow-behavior` attribute specifies the behavior of text boxes where the containing text does not fit into the text box, for text boxes contained in text documents. If the attribute value is `clip`, the text that does not fit into the text box is not displayed. If the attribute value is `auto-create-new-frame`, a new frame will be created on the next page, with the same position and dimensions of the original frame.

If the `style:overflow-behavior` attribute value is `auto-create-new-frame` and the text box has a minimum width or height specified, then the text box will grow until the page bounds are reached before a new frame is created.

The `style:overflow-behavior` attribute may be used with the following element: `<style:graphic-properties>` 16.14.

## 18.793. style:page-layout-name

The `style:page-layout-name` attribute specifies a page layout style that contains sizes, border and orientation attributes.

**Ed. Note** We did have this defined separately for master, notes and handout pages. I am not real sure how a “page-layout-name” differs from an ordinary style name.

The `style:page-layout-name` attribute may be used with the following elements: `<presentation:notes>` 9.2.5, `<style:handout-master>` 9.2.1 and `<style:master-page>` 15.6.

## 18.794. style:page-number

The `style:page-number` attribute specifies the page number that should be used for new page when either a paragraph or table style specifies a master page that should be applied beginning from the start of a paragraph or table.

The attribute value can be an integer value or the value `auto`. An integer value specifies the page number of the new page directly. The value `auto` specifies that the page gets the page number of the previous page, incremented by one.

The `style:page-number` attribute may be used with the following elements: `<style:paragraph-properties>` 16.6 and `<style:table-properties>` 16.9.

## 18.795. style:page-usage

The `style:page-usage` attribute specifies the type of pages that a page master should generate.

Ed. Note How do left and right differ from odd and even? Not for 1.2 but I think we need to become more consistent on terminology for formatting in particular. Easy for users to see differences in formatting as lack of interoperability. Not true but hard to cure that mis-impression.

The `style:page-usage` attribute may be used with the following element: `<style:page-layout>` 15.4.

The default value for this attribute is `all`.

## 18.796. style:paper-tray-name

The `style:paper-tray-name` attribute specifies the paper tray to use when printing the document. The names assigned to the printer trays depends upon the printer. If the value of this attribute is `default`, the default tray specified in the printer configuration settings is used.

The `style:paper-tray-name` attribute may be used with the following element: `<style:page-layout-properties>` 16.3.

## 18.797. style:parent-style-name

The `style:parent-style-name` attribute specifies the name of a parent style. The parent style cannot be an automatic style and has to exist.

If a parent style is not specified, the default style which has the same family as the current style is used.

Ed. Note We say in version 6: "Default styles exist for all style families that are represented by the `<style:style>` element specified in section 15.1" but that really isn't true, is it? That is we don't provide those default styles because we haven't declared any styles. Yes? So, we should say: Default styles must be defined for all style families that are represented by `<style:style>` elements. Yes?

The `style:parent-style-name` attribute may be used with the following element: `<style:style>` 15.2.

## 18.798. style:percentage-data-style-name

The `style:percentage-data-style-name` attribute specifies for a chart style, the name of a percentage data style. A percentage data style defines how percentage values are displayed when an object displays absolute numbers and percentage values at the same time.

The `style:percentage-data-style-name` attribute may be used with the following element: `<style:style>` 15.2.

## 18.799. style:position

The `style:position` attribute varies depending upon the element upon which it appears as follows:



- `<style:tab-stop>` - specifies the position of a tab stop. Depending on the value of the `text:relative-tab-stop-position` attribute in the `<text:table-of-content-source>`, `<text:illustration-index-source>`, `<text:object-index-source>`, `<text:user-index-source>` or `<text:alphabetical-index-source>` parent element, the position of the tab is interpreted as being relative to the left margin or the left indent.
- `<style:background-image>` - specifies the position of a background image in a paragraph. Its value can be a space separated combination of top, center or bottom for the vertical position and left, center or right for the horizontal position. The vertical and horizontal positions can be specified in any order. If one position is specified, the other position defaults to center.

Ed. Note We have several position related attributes. Consider for consolidation in a future version.

The `style:position` attribute may be used with the following elements: `<style:background-image>` 16.3.1, `<style:tab-stop>` 16.6.2 and `<text:index-entry-tab-stop>` 7.14.6.

For a `<style:background-image>` 16.3.1 element the default value for this attribute is center.

## 18.800. style:print

The `style:print` attribute specifies the components in a spreadsheet document to print.

The value of this attribute is a list of the following values separated by blanks: headers, grid, annotations, objects (including graphics), charts, drawings, formulas, zero-values.

The `style:print` attribute may be used with the following element: `<style:page-layout-properties>` 16.3.

## 18.801. style:print-content

The `style:print-content` attribute varies depending upon the element upon which it appears as follows:

- `<style:table-column-properties>` - specifies if cell content is printed
- `<style:graphic-properties>` - specifies if frame content is printed

Ed. Note Printing of content is closely related to display which is handled variously. Consider consolidation in a future version.

The `style:print-content` attribute may be used with the following elements: `<style:graphic-properties>` 16.14 and `<style:table-cell-properties>` 16.12.

## 18.802. style:print-page-order

The `style:print-page-order` attribute specifies the order in which data in a spreadsheet is numbered and printed when the data does not fit on one printed page.

The value of this attribute can be `ttb` or `ltr`. The value `ttb` prints the data vertically from the left column to the bottom row of the sheet. The value `ltr` prints the data horizontally from the top row to the right column of the sheet.

The `style:print-page-order` attribute may be used with the following element:  
`<style:page-layout-properties>` 16.3.

### 18.803. `style:print-orientation`

The `style:print-orientation` attribute specifies the orientation of the printed page. The value of this attribute can be `portrait` or `landscape`.

The `style:print-orientation` attribute may be used with the following element:  
`<style:page-layout-properties>` 16.3.

### 18.804. `style:protect`

The `style:protect` attribute varies depending upon the element upon which it appears as follows:

- `<style:graphic-properties>` - specifies whether the content, size, or position of a frame is protected. The value of this attribute can be either `none` or a space separated list that consists of any of the values `content`, `position`, or `size`.
- `<style:section-properties>` - specifies that sections should not be changed. The user interface should prevent the user from manually making any changes. The `style:protect` attribute should be set by default for linked sections or indexes. Removing the protection makes these sections accessible to the user, but any changes will be over-written by updates to those sections.

Ed. Note How do we mean “should” in the second item?

The `style:protect` attribute may be used with the following elements: `<style:graphic-properties>` 16.14 and `<style:section-properties>` 16.8.

### 18.805. `style:punctuation-wrap`

The `style:punctuation-wrap` attribute specifies whether or not a punctuation mark, if one is present, can be hanging, that is, whether it can be placed in the margin area at the end of a full line of text.

The `style:punctuation-wrap` attribute may be used with the following element:  
`<style:paragraph-properties>` 16.6.

### 18.806. `style:register-true`

The `style:register-true` attribute specifies whether the lines on both sides of a printed page align. The text baselines of text in page columns or text box columns also align.

Ed. Note Alignment is not a matter of “seem to run from one column to another.” Either they are aligned, the value of this attribute is `true`, or they don't.

The `style:register-true` attribute may be used with the following element:  
`<style:paragraph-properties>` 16.6.

## 18.807. style:register-truth-ref-style-name

The `style:register-truth-ref-style-name` attribute specifies a paragraph style. The line distance specified of the paragraph style is used as the reference line distance for all paragraphs that have the register-truth feature enabled.

The `style:register-truth-ref-style-name` attribute may be used with the following element: `<style:page-layout-properties>` 16.3.

## 18.808. style:rel-column-width

The `style:rel-column-width` attribute specifies a relative width of a column with a number value, followed by a '\*' character. If  $r_c$  is the relative width of the column,  $r_s$  the sum of all relative columns widths, and  $w_s$  the absolute width that is available for these columns, then the absolute width  $w_c$  of the column is  $w_c = r_c w_s / r_s$ .

Ed. Note I deleted: "Applications that support specifying the relative width of a column may specify widths in this way, but it is not required." We should handle this sort of issue under conformance so that all the required/must features are gathered together in one place.

The `style:rel-column-width` attribute may be used with the following element: `<style:table-column-properties>` 16.10.

## 18.809. style:rel-height

The `style:rel-height` attribute specifies height of a drawing object as a relative value within a frame. The relative value either is a percentage value, the special value `scale`, or the special value `scale-min`.

The interpretation of relative values depends on the anchor of the drawing object. If the anchor for the drawing object is in a table cell, the percentage value is relative to the surrounding table box. If the anchor for the drawing object is in a text box, the percentage value is relative to the surrounding text box. In other cases, the percentage values is relative to the width of the page or window.

The value `scale` for the height means that the height should be calculated depending on the width, so that the ratio of width and height of the original image or object size is preserved.

The value `scale-min` equals the value `scale`, except that the calculated width or height is a minimum height rather than an absolute one.

To support applications that don't support relative width and heights, applications that save the attribute `style:rel-height` should also provide the real height in `svg:height` and `fo:min-height` attributes.

The `style:rel-height` attribute may be used with the following elements: `<draw:frame>` 9.4.2 and `<style:graphic-properties>` 16.14.

## 18.810. style:rel-width

The `style:rel-width` attribute varies depending upon the element where it appears as follows:

For `<draw:frame>`, it specifies the width of a drawing object as a relative value within a frame. The relative value either is a percentage value, the special value `scale`, or the special value `scale-min`.

The interpretation of relative values depends on the anchor of the drawing object. If the anchor for the drawing object is in a table cell, the percentage value is relative to the surrounding table box. If the anchor for the drawing object is in a text box, the percentage value is relative to the surrounding text box. In other cases, the percentage values is relative to the width of the page or window.

The value scale for the width means that the width should be calculated depending on the height, so that the ratio of with and height of the original image or object size is preserved.

The value scale-min equals the value scale, except that the calculated width or height is a minimum height rather than an absolute one.

To support applications that don't support relative width, applications that save the attribute `style:rel-width` should also provide the real width and heights in the `svg:width` and `fo:min-width` attributes.

For `<style:column>`, it specifies the width of a column. The column widths are specified as number values instead of lengths. To get the absolute column width, the space that is available for a columned area is distributed among the columns proportional to these numbers.

The column width is not specified in a percentage length, but rather in terms of relative weights, that is, a number followed by a '\*' character. The total space available for the entire table is distributed among its columns according to its relative width.

For `<style:table-properties>`, it specifies the width of a table relative to the width of the area that the table is in. User agents that support specifying the relative width of a table can specify widths in this way, but it is not essential.

**Ed. Note I am deleted the excursion on why tables must have fixed widths. It is sufficient that we announce the rule.**

For `<style:footnote-sep>`, it specifies the length of the footnote separator line as a percentage of the body text area.

The `style:rel-width` attribute may be used with the following elements: `<draw:frame>` 9.4.2, `<style:column>` 16.8.2, `<style:footnote-sep>` 16.3.3, `<style:graphic-properties>` 16.14 and `<style:table-properties>` 16.9.

## 18.811. style:repeat

The `style:repeat` attribute specifies if an image can be repeated or stretched over a particular area. The value of the attribute can be `no-repeat`, `repeat`, or `stretch`.

The `style:repeat` attribute may be used with the following elements: `<style:background-image>` 16.3.1, `<style:drawing-page-properties>` 16.17 and `<style:graphic-properties>` 16.14.

For a `<style:background-image>` 16.3.1 element the default value for this attribute is `repeat`.

## 18.812. style:repeat-content

The `style:repeat-content` attribute specifies whether text content of a cell is displayed as many times as there is space left in the cell's writing direction. The attribute has no effect for cell content that contains a line break.

The `style:repeat-content` attribute may be used with the following element: `<style:table-cell-properties>` 16.12.

### 18.813. style:rotation-align

The style:rotation-align attribute specifies how the edge of the text in a cell is aligned after a rotation. There are four alignment options: "none", "bottom", "top", or "center".

Table 14 - Rotation align values

<b>Alignment</b>	<b>Text is...</b>	<b>Borders and background are...</b>
None.	Rotated.	Unchanged.
Bottom of the cell.	Rotated and may overlap with other cells if the text is longer than the length of the cell.	Positioned parallel to the text, whereby the upper or lower edge is drawn at the original position of the cell.
Top of the cell.		
Center of the cell.		

The style:rotation-align attribute may be used with the following element:  
<style:table-cell-properties> 16.12.

### 18.814. style:rotation-angle

The style:rotation-angle attribute specifies the rotation angle of cell content in degrees.

The style:rotation-angle attribute may be used with the following elements:  
<style:chart-properties> 16.16 and <style:table-cell-properties> 16.12.

#### 18.814.1 style:rotation-angle

The style:rotation-angle attribute specifies the value of a rotation angle in degrees.

### 18.815. style:row-height

The style:row-height attribute specifies a fixed row height.

The style:row-height attribute may be used with the following element: <style:table-row-properties> 16.11.

### 18.816. style:ruby-align

The style:ruby-align attribute specifies the horizontal alignment of the ruby text in relationship to the ruby base.

Ed. Note In the 2<sup>nd</sup> edition, we use the same language for 15.6.1 and 5.6.2, which are 18.789 and 18.790 here. One is horizontal alignment and the other is vertical.

The style:ruby-align attribute may be used with the following element: <style:ruby-properties> 16.7.

### 18.817. style:ruby-position

The style:ruby-position attribute specifies the vertical position of the ruby text in relationship to the ruby base.

Ed. Note The `style:row-height` property specifies a fixed row height, while the `style:min-row-height` property specifies a fixed height.

The `style:ruby-position` attribute may be used with the following element: `<style:ruby-properties>` 16.7.

## 18.818. `style:run-through`

The `style:run-through` attribute specifies whether content of a frame is displayed in the background or foreground.

The value of this attribute can be `foreground` or `background`. If the value is `foreground`, the frame content is displayed in front of the text. If the value is `background`, the frame content is displayed behind the text.

Ed. Note Does the use of this attribute depend upon `style:wrap` having the value “run-through?” Seems implied by: “If the value of the `style:wrap` attribute is `run-through`, it can be further specified whether the content of the frame should be displayed in the background or in the foreground.” Moreover, does `foreground/background` have any meaning with a transparent object? How would one tell the difference in a 2-dimensional representation?

The `style:run-through` attribute may be used with the following element: `<style:graphic-properties>` 16.14.

## 18.819. `style:scale-to`

The `style:scale-to` attribute specifies that a document is to be scaled to a percentage value, where 100% equals no scaling. When using this attribute, all pages are enlarged or reduced in size while printing.

If this attribute and `style:scale-to-pages` are absent, a document is not scaled.

The `style:scale-to` attribute may be used with the following element: `<style:page-layout-properties>` 16.3.

## 18.820. `style:scale-to-pages`

The `style:scale-to-pages` attribute specifies the number of pages on which a document should be printed. The document is then scaled to fit the defined number of pages.

If this attribute and `style:scale-to` are absent, a document is not scaled.

Ed. Note In the 2<sup>nd</sup> edition, the third paragraph of 15.2.16 has “...on which the the document...”

The `style:scale-to-pages` attribute may be used with the following element: `<style:page-layout-properties>` 16.3.

## 18.821. `style:script-type`

The `style:script-type` attribute specifies which script dependent attributes (like `font-family`, `style:font-family-asian`, `style:font-family-complex`) are currently active for some text. The attribute should be evaluated by applications that do not support script types to select the correct script dependent formatting properties. Application that support script types may also evaluate the attribute and overwrite the script type they would evaluate for a certain character, but that is not required.

The usage of this attribute simplifies for instance transformations from and to [CSS2]/[XSL] and other formats that don't have script-dependent attributes, and also can be used to assign script-types to weak [UNICODE] characters, where applications may choose different script types.

The values of this attribute are latin, asian, complex and ignore. The value ignore can be used only within default styles. If it is set, all script-dependent attributes are applied to all script types. This would mean for example that a `fo:font-family` would be applied to all script types as well as a `style:font-family-asian` or `style:font-family-complex`. This simplifies saving documents with applications that do not support a script type.

**Ed. Note** Why the limitation to default styles?

The `style:script-type` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

## 18.822. style:shadow

The shadow attribute `style:shadow` specifies a shadow effect.

Acceptable values are defined in defined in §7.16.5 of [XSL].

**Ed. Note** All of our shadow references ultimately refer back to a definition that cites the XSL standard so I collapsed them into one. BTW, the schema says "string" and does not repeat the values set forth in 7.16.5 of XSL.

The `style:shadow` attribute may be used with the following elements: `<style:graphic-properties>` 16.14, `<style:header-footer-properties>` 16.4, `<style:page-layout-properties>` 16.3, `<style:paragraph-properties>` 16.6, `<style:table-cell-properties>` 16.12 and `<style:table-properties>` 16.9.

## 18.823. style:shrink-to-fit

The `style:shrink-to-fit` attribute specifies whether the content of a cell, if necessary, gets shrunk to fit into a cell. Shrinking means that the cell's font size is decreased, so that the complete text fits into the cell. The attribute has no effect on cells where the cell content fits already into the cell.

The `style:shrink-to-fit` attribute may be used with the following element: `<style:table-cell-properties>` 16.12.

## 18.824. style:snap-to-layout-grid

The `style:snap-to-layout-grid` attribute specifies whether the layout of a paragraph should consider the layout grid settings of the page.

**Ed. Note** The original text has a paragraph considering the layout grid settings of the page. Rather unlikely event.

The `style:snap-to-layout-grid` attribute may be used with the following element: `<style:paragraph-properties>` 16.6.

## 18.825. style:style

The `style:style` attribute specifies the line style of a column separator line.

The `style:style` attribute may be used with the following element: `<style:column-sep>` 16.8.3.

The default value for this attribute is `solid`.

## 18.826. `style:style-name`

The `style:style-name` attribute specifies the text style to apply to characters specified by a `<style:drop-cap>` element.

Ed. Note I inserted the drop-cap element reference as to say “characters using the larger font” is unclear. Partially because of the new ordering of the attributes. Less likely to be unclear as organized originally.

The `style:style-name` attribute may be used with the following element: `<style:drop-cap>` 16.6.3.

## 18.827. `style:tab-stop-distance`

The attribute `style:tab-stop-distance` specifies the distance between default tab stops. A default tab stop is repeated automatically after the specified distance. Default tab stops usually are only evaluated if they are specified within a default style.

The `style:tab-stop-distance` attribute may be used with the following element: `<style:paragraph-properties>` 16.6.

## 18.828. `style:table-centering`

The `style:table-centering` attribute specifies whether tables are centered horizontally and/or vertically on the page. This attribute only applies to spreadsheet documents.

The value of this attribute can be `horizontal`, `vertical`, `both`, or `none`. The default is to align the table to the top-left or top-right corner of the page, depending of its writing direction.

The `style:table-centering` attribute may be used with the following element: `<style:page-layout-properties>` 16.3.

## 18.829. `style:text-align-source`

The `style:text-align-source` attribute specifies the source of a text-align attribute. If the value of this attribute is `fix`, the value of the `fo:text-align` attribute is used. If the value is `value-type`, the text alignment depends on the value-type of the cell.

The `style:text-align-source` attribute may be used with the following element: `<style:table-cell-properties>` 16.12.

## 18.830. `style:text-autospace`

The `style:text-autospace` attribute specifies whether to add space between Asian, Western, and complex texts.

The possible values are `none` and `ideograph-alpha`.

Ed. Note It isn't clear if this is horizontal or vertical space or how much space.



The `style:text-autospace` attribute may be used with the following element: `<style:paragraph-properties>` 16.6.

### 18.831. `style:text-blinking`

The `style:text-blinking` attribute specifies whether or not text blinks.

The `style:text-blinking` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.832. `style:text-combine`

The `style:text-combine` attribute specifies whether to combine characters so that they are displayed within two lines.

The value of this attribute can be `none`, `letters` or `lines`.

If the value is `lines`, all characters with this attribute value that immediately follow each other are displayed within two lines of approximately the same length. There can be a line break between any two characters to meet this constraint.

If the value of the attribute is `letters`, up to 5 characters are combined within two lines. Any additional character is displayed as normal text.

**Ed. Note** Why the value “letters” and not simply allow the specification of a length to be so displayed?

The `style:text-combine` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.833. `style:text-combine-start-char`

The `style:text-combine-start-char` attribute specifies the start character that is displayed before a portion of text whose `style:text-combine` attribute has a value of `lines`.

The `style:text-combine-start-char` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.834. `style:text-combine-end-char`

The `style:text-combine-end-char` attribute specifies the end character that is displayed after a portion of text whose `style:text-combine` attribute has a value of `lines`.

The `style:text-combine-end-char` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.835. `style:text-emphasize`

The `style:text-emphasize` attribute specifies emphasis in Asian documents.

The value of this attribute consists of two space-separated values.

The first value represents the style to use for emphasis and it can be `none`, `accent`, `dot`, `circle`, or `disc`.

The second value represents the position of the emphasis and it can be above or below. If the first value is none, the second value can be omitted.

**Ed. Note** We need to define the values *accent*, *dot*, *circle* and *disc*.

The `style:text-emphasize` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.836. style:text-line-through-color

The `style:text-line-through-color` attribute specifies the color that is used for line-through text. The value of this attribute is either `font-color` or a color. If the value is `font-color`, the current text color is used for underlining.

The `style:text-line-through-color` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.837. style:text-line-through-mode

The `style:text-line-through-mode` attribute specifies whether lining through is applied to words only or to portions of text. If lining through is applied to text portions, the spaces between words and the words are line-through.

**Ed. Note** All the CSS3 references are out-dated by later versions that are still not final. So I removed these and will be removing others.

The `style:text-line-through-mode` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.838. style:text-line-through-style

The `style:text-line-through-style` attribute specifies the style for rendering a line through text.

**Ed. Note** The text did say this attribute specifies whether text has a line through, but that is incorrect. The `style:text-line-through-type` attribute serves that function.

The `style:text-line-through-style` attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.839. style:text-line-through-text

The `style:text-line-through-text` attribute specifies a text that is used for line through. The attribute will be evaluated only if the value of `style:text-line-through-style` attribute is different than none.

If the attribute value is not empty, the attribute value string is used for line-through instead of the line style that has been specified by the `style:text-line-through-style` attribute. If the application does not support line-through with text, the attribute is ignored, and the line style specified by the `style:text-line-through-style` attribute will be used.

If the application supports line-through with single characters only, and the `style:text-line-through-text` attribute value as more than one character, the first character of the value should be used for line-through. If the applications supports line-through with with certain characters only

(like "x" or "/"), the application should use one of these characters if the attribute specifies characters that are not supported.

The `style:text-line-through-text` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.840. `style:text-line-through-text-style`

The `style:text-line-through-text-style` specifies a text style that is applied to text-line-through characters. It is not applied to line-through lines. If the attribute appears in an automatic style, it may reference either an automatic text style or a common style. If the attribute appears in a common style, it may reference a common style only.

Ed. Note Can we eliminate the differences in style references? That seems really problem prone.

The `style:text-line-through-text-style` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.841. `style:text-line-through-type`

The `style:text-line-through-type` attribute specifies whether text is lined through, and if so, whether a single or double line will be used.

Ed. Note Tricky. If its value is none and `style:text-line-through-style` is not none, then a single character can be used for line through. Otherwise, if it has the value single or double, then line through occurs. So, the only way to avoid line through is to have "" (empty string) or avoid this attribute altogether? Seems like a very difficult way to accomplish this goal.

The `style:text-line-through-type` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.842. `style:text-line-through-width`

The `style:text-line-through-width` attribute specifies the width of a line-through line. The value bold specifies a line width that is calculated from the font sizes like an auto width, but is wider than an auto width.

Ed. Note Ok, I'll bite. How much wider than auto width?

The `style:text-line-through-width` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.843. `style:text-outline`

The `style:text-outline` attribute specifies whether to display an outline of text or the text itself. This attribute can have a value of true or false.

The `style:text-outline` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.844. `style:text-position`

The `style:text-position` attribute specifies whether text is positioned above or below the baseline and to specify the relative font height that is used for this text.

This attribute can have one or two values.

The first value must be present and specifies the vertical text position as a percentage of the current font height or it takes one of the values `sub` or `super`. Negative percentages or the `sub` value place the text below the baseline. Positive percentages or the `super` value place the text above the baseline. If `sub` or `super` is specified, the application chooses an appropriate text position.

The second value is optional and specifies the font height as a percentage of the current font-height. If this value is not specified, an appropriate font height is used. Although this value may change the font height that is displayed, it never changes the current font height that is used for additional calculations.

The `style:text-position` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.845. `style:text-rotation-angle`

The `style:text-rotation-angle` attribute specifies an angle to which text is rotated. The value of this attribute can be 0, 90, or 270.

If more than one character is selected, the entire selection is rotated as a block.

**Ed. Note** Why is the value of rotation limited? Note change from: "If this attribute is specified for more than one character, all text containing these characters is rotated." to second paragraph as shown.

The `style:text-rotation-angle` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.846. `style:text-rotation-scale`

The `style:text-rotation-scale` attribute specifies whether for rotated text the width of the text should be scaled to fit into the current line height or the width of the text should remain fixed, therefore changing the current line height.

The `style:text-rotation-scale` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.847. `style:text-scale`

The `style:text-scale` attribute specifies whether to decrease or increase the width of text by scaling the font width.

The `style:text-scale` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

### 18.848. `style:text-underline-color`

The `style:text-underline-color` attribute specifies a color that is used to underline text. The value of this attribute is either `font-color` or a color. If the value is `font-color`, the current text color is used for underlining.

The `style:text-underline-color` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

## 18.849. style:text-underline-mode

The style:text-underline-mode attribute specifies whether underlining is applied to words only or to portions of text. If underlining is applied to text portions, the spaces between words and the words are underlined.

The style:text-underline-mode attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.850. style:text-underline-style

The style:text-underline-style attribute specifies if and how text is underlined. The value of this attribute is the underlining style for the text, for example, single, dotted, dash.

Ed. Note Both 18.823 and 18.824 control underlining. What happens if they conflict? Need to have one and only one in charge of underlining. Probably need to define the values as well.

The style:text-underline-style attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.851. style:text-underline-type

The style:text-underline-type attribute specifies whether text is underlined, and if so, whether a single or double line will be used for underlining.

Ed. Note Both 18.823 and 18.824 control underlining. What happens if they conflict? Need to have one and only one in charge of underlining.

The style:text-underline-type attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.852. style:text-underline-width

The style:text-underline-width attribute specifies the width of an underline. The value bold specifies a line width that is calculated from the font sizes like an auto width, but is wider than an auto width.

Ed. Note How wide is "wider" than auto width?

The style:text-underline-width attribute may be used with the following element:  
<style:text-properties> 15.14.8.1.

## 18.853. style:type

The style:type attribute specifies the tab stop type. The values taken by that attribute vary by the element on which it appears as follows:

- <text:index-entry-tab-stop> - left and right. If the value of this attribute is left, the style:position attribute must also be used. Otherwise, this attribute must be omitted.
- <style:tab-stop> - left, center, right or char

The style:type attribute may be used with the following elements: <style:tab-stop> 16.6.2 and <text:index-entry-tab-stop> 7.14.6.

For a `<style:tab-stop>` 16.6.2 element the default value for this attribute is `left`.

## 18.854. `style:use-optimal-column-width`

The `style:use-optimal-column-width` attribute specifies that a column width should be recalculated automatically if content in the column changes.

The `style:use-optimal-column-width` attribute may be used with the following element: `<style:table-column-properties>` 16.10.

## 18.855. `style:use-optimal-row-height`

The `style:use-optimal-row-height` attribute specifies that a row height should be recalculated automatically if content in the row changes.

The `style:use-optimal-row-height` attribute may be used with the following element: `<style:table-row-properties>` 16.11.

## 18.856. `style:use-window-font-color`

The `style:use-window-font-color` attribute specifies whether or not the window foreground color should be as used as the foreground color for a light background color and white for a dark background color.

Ed. Note Sorry, I am sure that is meaningful to a member of the graphics team but I have no idea what is being described.

The `style:use-window-font-color` attribute may be used with the following element: `<style:text-properties>` 15.14.8.1.

## 18.857. `style:vertical-align`

The `style:vertical-align` attribute varies depending upon the element where it appears as follows:

- `<style:paragraph-properties>` - specifies the vertical position of a character. By default characters are aligned according to their baseline. Characters may also be vertically aligned as follows:
  - `top` —To the top of the line.
  - `middle` —To the center of the line.
  - `bottom` — To the bottom of the line.
  - `auto` — Automatically, which sets the vertical alignment to suit the text rotation. Text that is rotated 0 or 90 degrees is aligned to the baseline, while text that is rotated 270 degrees is aligned to the center of the line.
  - `baseline`
- `<style:section-properties>` - specifies how to vertically align a line that is less than 100% of its height within the columned area. The value of this attribute can be either `top`, `middle`, or `bottom`.

- `<style:table-cell-properties>` - specifies the vertical alignment of text in a table cell. The options for the vertical alignment attribute are as follows:
  - `top` — Aligns text vertically with the top of the cell.
  - `middle` — Aligns text vertically with the middle of the cell.
  - `bottom` — Aligns text vertically with the bottom of the cell.
  - `automatic` – The application decide how to align the text.

**Ed. Note** We are inconsistent with `auto` vs. `automatic`, and note the disagreement on the meaning of `automatic`.

The `style:vertical-align` attribute may be used with the following elements:  
`<style:column-sep>` 16.8.3, `<style:paragraph-properties>` 16.6 and  
`<style:table-cell-properties>` 16.12.

For a `<style:column-sep>` 16.8.3 element the default value for this attribute is `top`.

## 18.857.1 style:vertical-pos

The `style:vertical-pos` attribute specifies the vertical alignment of a frame relative to a specific area.

The value of this attribute can be one of the following: `from-top`, `top`, `middle`, `below` or `bottom`. The area that the position is relative to is specified by the `style:vertical-rel` attribute. The values `top`, `middle` and `bottom` specify the corners of the frame and the reference area to be aligned. The value `below` specifies that the top corner of the frame is positioned below the reference area.

If the value of this attribute is `from-top`, the `svg:y` attribute associated with the frame element specifies the vertical position of the frame. Otherwise, the `svg:y` attribute is ignored for text documents.

Some values may be used in connection with certain frame anchor and relation types only.

The following tables display the possible values of the attributes `style:vertical-pos` and `style:vertical-rel`. The possible values of these alignment attributes are listed in the first column on the left, and an alignment attribute value/anchor type value match is indicated by an X.

Table 15 - Vertical Position Values

<b>Value of <i>style:vertical-pos</i></b>	<b>Value of text : anchor - type</b>				
	<b><i>page</i></b>	<b><i>frame</i></b>	<b><i>paragraph</i></b>	<b><i>char</i></b>	<b><i>as-char</i></b>
any	X	X	X	X	X

Table 16 - Vertical Relation Values

<b>Value of <i>style:vertical-rel</i></b>	<b>Value of text : anchor - type</b>				
	<b><i>page</i></b>	<b><i>frame</i></b>	<b><i>paragraph</i></b>	<b><i>char</i></b>	<b><i>as-char</i></b>
page	X				
page-content	X				

<b>Value of style:vertical- rel</b>	<b>Value of text : anchor - type</b>				
	<b>page</b>	<b>frame</b>	<b>paragraph</b>	<b>char</b>	<b>as-char</b>
frame		X			
frame-content		X			
paragraph			X	X	
paragraph-content			X	X	
char				X	X
line					X
baseline					X
text					X

Ed. Note Deleted “An `svg:y` attribute can be used within a graphic style. In that case, the attribute specifies a default position for new frames that are created using this style.” Frames are always specified with graphic styles so unnecessary.

The `style:vertical-pos` attribute may be used with the following elements:  
`<style:graphic-properties>` 16.14 and `<style:list-level-properties>` 16.13.

## 18.858. style:vertical-rel

The `style:vertical-rel` attribute specifies the area against which the vertical position of a frame is positioned.

The value of this attribute can be one of the following: `page`, `page-content`, `frame`, `frame-content`, `paragraph`, `paragraph-content`, `line`, `baseline`, `text` or `char`.

Some values can be used with only certain frame anchor types.

The `style:vertical-rel` attribute may be used with the following elements:  
`<style:graphic-properties>` 16.14 and `<style:list-level-properties>` 16.13.

## 18.859. style:volatile

The `style:volatile` attribute specifies whether or not unused style in a document are retained or discarded.

If the value of the attribute is `true`, the application keeps the style if possible. If the value is `false`, the application discards the unused styles.

The `style:volatile` attribute may be used with the following elements: `<number:boolean-style>` 15.14.6, `<number:currency-style>` 15.14.2, `<number:date-style>` 15.14.4, `<number:number-style>` 15.14.1, `<number:percentage-style>` 15.14.3, `<number:text-style>` 15.14.7 and `<number:time-style>` 15.14.5.



## 18.860. style:width

The `style:width` attribute varies depending on the element upon which it appears as follows:

- `<style:columns>` - specifies the width of a column separator line
- `<style:footnote-sep>` - width or thickness of a line
- `<style:table-properties>` - specifies the fixed width of a table. Every table must have a fixed width.

Ed. Note There are several “width” attributes and related properties. Consider for consolidation in later version.

The `style:width` attribute may be used with the following elements: `<style:column-sep>` 16.8.3, `<style:footnote-sep>` 16.3.3 and `<style:table-properties>` 16.9.

## 18.861. style:wrap

The `style:wrap` attribute specifies how text is displayed around a frame or graphic object. The possible values are:

- `none`: Text does not wrap around the shape.
- `left`: Text wraps around the left side of the shape.
- `right`: Text wraps around the right side of the shape.
- `parallel`: Text wraps around both sides of the shape.
- `dynamic`: Text may wrap around both sides of the shape, provided that there is sufficient space left.
- `biggest`: Text may wrap around the shape where the difference to the left or right page or column border is largest.
- `run-through`: Text runs through the shape.

Ed. Note In the 2<sup>nd</sup> edition, we say “wraps around the left side” when talking about the right setting. I corrected that here. BTW, what does “sufficient space left” mean? I am not sure I understand “biggest” either.

The `style:wrap` attribute may be used with the following element: `<style:graphic-properties>` 16.14.

## 18.862. style:wrap-contour

Within text documents, the `style:wrap-contour` attribute specifies for some frame types that the text should wrap around the shape of the object in the frame rather than around the frame itself. This is called contour wrapping.

Ed. Note Shouldn't we say what “some frame types” those are? Or are they frame types with this attribute?

The `style:wrap-contour` attribute may be used with the following element: `<style:graphic-properties>` 16.14.

## 18.863. style:wrap-contour-mode

The `style:wrap-contour-mode` attribute specifies the nature of the wrapping of text around the contour of a shape.

This attribute is recognized only by frames/drawing shapes or styles that already have the `style:wrap` and `style:wrap-contour` attributes attached.

The value of the attribute can be `outside` or `full`. If the value of the attribute is `outside`, the text wraps around the general area to the left and right of the shape. If the value of the attribute is `full`, the text wraps around the shape and fills any possible spaces and indentations in the shape.

Ed. Note I realize this isn't going to be easy but "general area" is rather vague. We really need to establish or reference some page geometry definitions (the ones from XSL or DSSSL?) and use those. Perhaps not in 1.2 but certainly in future editions.

The `style:wrap-contour-mode` attribute may be used with the following element:  
`<style:graphic-properties>` 16.14.

## 18.864. style:wrap-dynamic-threshold

The `style:wrap-dynamic-threshold` attribute specifies the minimum distance between the page or column border and the object for which wrapping will be enabled. It is evaluated only if the `style:wrap` attribute has a value of `dynamic`.

The `style:wrap-dynamic-threshold` attribute may be used with the following element:  
`<style:graphic-properties>` 16.14.

## 18.865. style:writing-mode

The `style:writing-mode` attribute varies depending on the element where it appears as follows:

- `<style:graphic-properties>` - §7.27.7 of [XSL] + page (text box)
- `<style:page-layout>` - - §7.27.7 of [XSL]
- `<style:paragraph-properties>` - §7.27.7 of [XSL] + page
- `<style:section-properties>` - §7.27.7 of [XSL] + page
- `<style:table-cell-properties>` - - §7.27.7 of [XSL] + page
- `<style:table-properties>` - §7.27.7 of [XSL] + page

Ed. Note The one place, `style:page-layout` where I would expect the page value, it is prohibited. Note that a normative reference to XSL gets us also:

lr-tb

Inline components and text within a line are written left-to-right. Lines and blocks are placed topto-bottom.

Establishes the following directions:

- `inline-progression-direction` to left-to-right

If any right-to-left reading characters are present in the text, the inline-progression-direction for glyph-areas may be further modified by the Unicode BIDI algorithm.

- block-progression-direction to top-to-bottom
- shift-direction to bottom-to-top

That is the advantage of normative referencing.

The `style:writing-mode` attribute may be used with the following elements:  
<style:graphic-properties> 16.14, <style:page-layout-properties> 16.3,  
<style:paragraph-properties> 16.6, <style:section-properties> 16.8,  
<style:table-cell-properties> 16.12 and <style:table-properties> 16.9.

## 18.866. style:writing-mode-automatic

The `style:writing-mode-automatic` attribute specifies if an application is allowed to recalculate the writing mode of a paragraph based on its content whenever the content changes. The actual value for the writing-mode should be contained in a `style:writing-mode` attribute, so that applications that do not support an automatic writing mode calculation or use a different algorithm always know the actual value.

By specifying a `fo:text-align` with value `start` additionally, the text alignment can be adapted to the writing mode simultaneously.

Ed. Note Well, its more complicated than simply saying “whenever the content changes” if we want to have meaningful conformance clauses. Since we are using Unicode, what layout characters or other properties of characters qualify as “content changes?” Serious question for which there are no easy answers. It has been a while since I have read the Unicode standard in detail but I think we should consider asking someone who works for it for some suggested language.

The `style:writing-mode-automatic` attribute may be used with the following element:  
<style:paragraph-properties> 16.6.

## 18.867. svg:accent-height

See §20.8.3 of [SVG].

The `svg:accent-height` attribute may be used with the following element: <style:font-face> 15.9.

## 18.868. svg:alphabetic

See §20.8.3 of [SVG].

The `svg:alphabetic` attribute may be used with the following element: <style:font-face> 15.9.

## 18.869. svg:ascent

See §20.8.3 of [SVG].

The `svg:ascent` attribute may be used with the following element: <style:font-face> 15.9.

## 18.870. svg:bbbox

See §20.8.3 of [SVG].

The `svg:bbbox` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.871. svg:cap-height

See §20.8.3 of [SVG].

The `svg:cap-height` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.872. svg:d

See §8 of [SVG].

**Note:** Implementations may only support a subset of the SVG path specification, for instance no mixtures of open and closed curves for one shape, or no elliptical arc command.

**Note:** Applications should export the connector's geometric path using the `svg:d` attribute.

**Note:** Exporting the path assists simple document viewers that may not be capable of computing the path from other attributes.

**Ed. Note** Is this a conformance issue? How much of SVG is supported? Do we require some minimal set.

**Ed. Note:** Wrong order of attributes

The `svg:d` attribute may be used with the following elements: `<dr3d:extrude>` 9.5.5, `<dr3d:rotate>` 9.5.6, `<draw:connector>` 9.3.12, `<draw:contour-path>` 9.4.10.3, `<draw:marker>` 15.21.7 and `<draw:path>` 9.3.9.

## 18.873. svg:descent

See §20.8.3 of [SVG].

The `svg:descent` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.874. svg:cx

See §9.3 of [SVG].

For the use of the attribute on `<svg:radialGradient>` elements, see §13.3.2 of [SVG].

The `svg:cx` attribute may be used with the following elements: `<draw:area-circle>` 9.4.12.4, `<draw:circle>` 9.3.10, `<draw:ellipse>` 9.3.11 and `<svg:radialGradient>` Error: Reference source not found.

For a `<svg:radialGradient>` Error: Reference source not found element the default value for this attribute is 50%.

## 18.875. svg:cy

See §9.3 of [SVG].

For the use of the attribute on <svg:radialGradient> elements, see §13.3.2 of [SVG].

The svg:cy attribute may be used with the following elements: <draw:area-circle> 9.4.12.4, <draw:circle> 9.3.10, <draw:ellipse> 9.3.11 and <svg:radialGradient> Error: Reference source not found.

For a <svg:radialGradient> Error: Reference source not found element the default value for this attribute is 50%.

## 18.876. svg:fill-rule

See §11.3 of [SVG].

OpenDocument does not support the value inherit.

Ed. Note I am systematically inserting normative references to the SVG standard and noting where we depart from those definitions. It saves space and is more accurate. (The “A” in CAP.)

The svg:fill-rule attribute may be used with the following elements: <style:drawing-page-properties> 16.17 and <style:graphic-properties> 16.14.

## 18.877. svg:font-family

See §20.8.3 of [SVG].

The svg:font-family attribute may be used with the following element: <style:font-face> 15.9.

## 18.878. svg:font-size

See §20.8.3 of [SVG].

The svg:font-size attribute may be used with the following element: <style:font-face> 15.9.

## 18.879. svg:font-stretch

See §20.8.3 of [SVG].

The svg:font-stretch attribute may be used with the following element: <style:font-face> 15.9.

## 18.880. svg:font-style

See §20.8.3 of [SVG].

The svg:font-style attribute may be used with the following element: <style:font-face> 15.9.

## 18.881. **svg:font-variant**

See §20.8.3 of [SVG].

The `svg:font-variant` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.882. **svg:font-weight**

See §20.8.3 of [SVG].

The `svg:font-weight` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.883. **svg:fx**

See §13.2.3 of [SVG].

The `svg:fx` attribute may be used with the following element: `<svg:radialGradient>` Error: Reference source not found.

## 18.884. **svg:fy**

See §13.2.3 of [SVG].

The `svg:fy` attribute may be used with the following element: `<svg:radialGradient>` Error: Reference source not found.

## 18.885. **svg:gradientTransform**

See §13.2.2 and §13.2.3 of [SVG].

The `svg:gradientTransform` attribute may be used with the following elements:  
`<svg:linearGradient>` Error: Reference source not found and `<svg:radialGradient>` Error: Reference source not found.

## 18.886. **svg:gradientUnits**

See §13.2.2 and §13.2.3 of [SVG].

The `svg:gradientUnits` attribute may be used with the following elements:  
`<svg:linearGradient>` Error: Reference source not found and `<svg:radialGradient>` Error: Reference source not found.

The default value for this attribute is `objectBoundingBox`.

## 18.887. **svg:hanging**

See §20.8.3 of [SVG].

The `svg:hanging` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.888. svg:height

See §5.1.2 of [SVG].

Ed. Note Have I ever mentioned that I like precision in standards? Referencing into SVG is a nightmare but this is as close as I can get. Do need to check with graphics to see if we really imply all the semantics here.

Ed. Note I have temporarily deleted: “These values are overridden by the physical size of the linked image resource. They can be used to get an assumption of the size of the fill image without loading the image data.” a reference to `svg:height` and `svg:width` together. I am not sure what it means to “override” the values of these attributes. Rather, an image in a non-SVG format specifies its own size, yes? Why mix the two together? Moreover, the concluding sentence probably means an “estimate of the size” but that sounds application specific since we don't provide a place to record that information.

The `svg:height` attribute may be used with the following elements: `<chart:chart>` 10.3, `<chart:plot-area>` 10.6, `<dr3d:scene>` 9.5.1, `<draw:area-polygon>` 9.4.12.5, `<draw:area-rectangle>` 9.4.12.3, `<draw:caption>` 9.3.13, `<draw:circle>` 9.3.10, `<draw:contour-path>` 9.4.10.3, `<draw:contour-polygon>` 9.4.10.2, `<draw:control>` 9.3.15, `<draw:custom-shape>` 9.6.1, `<draw:ellipse>` 9.3.11, `<draw:fill-image>` 15.21.5, `<draw:frame>` 9.4.2, `<draw:page-thumbnail>` 9.3.16, `<draw:path>` 9.3.9, `<draw:polygon>` 9.3.7, `<draw:polyline>` 9.3.6, `<draw:rect>` 9.3.4, `<draw:regular-polygon>` 9.3.8, `<office:annotation>` 13.1, `<presentation:placeholder>` 15.23, `<style:graphic-properties>` 16.14 and `<style:header-footer-properties>` 16.4.

## 18.889. svg:ideographic

See §20.8.3 of [SVG].

The `svg:ideographic` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.890. svg:mathematical

See §20.8.3 of [SVG].

The `svg:mathematical` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.891. svg:name

See §20.8.3 of [SVG].

The `svg:name` attribute may be used with the following element: `<svg:font-face-name>` 15.12.

## 18.892. svg:offset

See §13.2.4 of [SVG].

The `svg:offset` attribute may be used with the following element: `<svg:stop>` 15.21.3.

## 18.893. svg:origin

See §19.2.12 of [SVG].

**Note:** SVG relies upon the definition of origin in SMIL, see: <http://www.w3.org/TR/2001/REC-smil-animation-20010904/#MotionOriginAttribute>.

Ed. Note Ok, this gets complicated. Yes, 19.2.12 of SVG is the correct citation, however, that part of SVG, cites part of SMIL, by name, not section (4.3, The animateMotion element), which contains the definition of origin, at: <http://www.w3.org/TR/2001/REC-smil-animation-20010904/#MotionOriginAttribute>. To reduce the aggravation to readers, I have added the reference they will be looking for is curious about this attribute.

The `svg:origin` attribute may be used with the following element: `<anim:animateMotion>` 14.2.3.

## 18.894. svg:overline-position

See §20.8.3 of [SVG].

The `svg:overline-position` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.895. svg:overline-thickness

See §20.8.3 of [SVG].

The `svg:overline-thickness` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.896. svg:panose-1

See §20.8.3 of [SVG].

The `svg:panose-1` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.897. svg:path

See §19.2.12 of [SVG].

The `svg:path` attribute may be used with the following element: `<anim:animateMotion>` 14.2.3.

## 18.898. svg:r

See §19.2.12 of [SVG].

For the use of the attribute on `<svg:radialGradient>` elements, see §13.3.2 of [SVG].

Ed. Note I deleted "If this attribute is not set, the position and size attributes are used to create a circle." Yes, `svg:width` and `svg:height` are the "size" attributes, `svg:x`, `svg:y` are "position" attributes. What was not stated by the definition was that all of these were presumed to be in a `<draw:circle>` element. We should not rely on context in the standard to make our meaning clear. Since we are talking about authoring content, why would `svg:r` ever be missing?



Ed. Note Yes, this occurs in the 2<sup>nd</sup> edition of 1.0 as well.

The `svg:r` attribute may be used with the following elements: `<draw:area-circle>` 9.4.12.4, `<draw:circle>` 9.3.10 and `<svg:radialGradient>` Error: Reference source not found.

For a `<svg:radialGradient>` Error: Reference source not found element the default value for this attribute is 50%.

## 18.899. `svg:rx`

See §9.4 of [SVG].

The `svg:rx` and `svg:ry` attributes can be used to round off the corners of a rectangle. The `svg:rx` attribute specifies the x-axis radius of the ellipse used to round off the corners of a rectangle. The `svg:ry` attribute specifies the y-axis radius of that ellipse. If only the `svg:rx` attribute is present then its value will be used for `svg:ry`. If only a `svg:ry` attribute is present then its value will be used for `svg:rx`.

Ed. Note I deleted: "If these attributes are not set, the position and size attributes are used to create an ellipse." See 18.847.

The `svg:rx` attribute may be used with the following elements: `<draw:ellipse>` 9.3.11 and `<draw:rect>` 9.3.4.

## 18.900. `svg:ry`

See §9.4 of [SVG].

The `svg:rx` and `svg:ry` attributes can be used to round off the corners of a rectangle. The `svg:rx` attribute specifies the x-axis radius of the ellipse used to round off the corners of a rectangle. The `svg:ry` attribute specifies the y-axis radius of that ellipse. If only the `svg:rx` attribute is present then its value will be used for `svg:ry`. If only a `svg:ry` attribute is present then its value will be used for `svg:rx`.

Ed. Note I deleted: "If these attributes are not set, the position and size attributes are used to create an ellipse." See 18.847.

The `svg:ry` attribute may be used with the following elements: `<draw:ellipse>` 9.3.11 and `<draw:rect>` 9.3.4.

## 18.901. `svg:slope`

See §20.8.3 of [SVG].

The `svg:slope` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.902. `svg:spreadMethod`

See §13.2.2 and §13.2.3 of [SVG].

The `svg:spreadMethod` attribute may be used with the following elements: `<svg:linearGradient>` Error: Reference source not found and `<svg:radialGradient>` Error: Reference source not found.

The default value for this attribute is `pad`.

### 18.903. **svg:stemh**

See §20.8.3 of [SVG].

The `svg:stemh` attribute may be used with the following element: `<style:font-face>` 15.9.

### 18.904. **svg:stemv**

See §20.8.3 of [SVG].

The `svg:stemv` attribute may be used with the following element: `<style:font-face>` 15.9.

### 18.905. **svg:stop-color**

See §13.2.4 of [SVG].

The `svg:stop-color` attribute may be used with the following element: `<svg:stop>` 15.21.3.

### 18.906. **svg:stop-opacity**

See §13.2.4 of [SVG].

The `svg:stop-opacity` attribute may be used with the following element: `<svg:stop>` 15.21.3.

### 18.907. **svg:strikethrough-position**

See §20.8.3 of [SVG].

The `svg:strikethrough-position` attribute may be used with the following element: `<style:font-face>` 15.9.

### 18.908. **svg:strikethrough-thickness**

See §20.8.3 of [SVG].

The `svg:strikethrough-thickness` attribute may be used with the following element: `<style:font-face>` 15.9.

### 18.909. **svg:string**

See §20.8.3 of [SVG].

The `svg:string` attribute may be used with the following element: `<svg:font-face-format>` 15.11.

### 18.910. **svg:stroke-color**

The attribute `svg:stroke-color` specifies the color of a stroke.

Ed. Note We should not use the `svg` namespace here. SVG didn't have a `stroke-color` attribute. Mostly as an artifact of the syntax which allowed: `style="stroke: red; stroke-width: 5;"` Yes, this occurs in the 2<sup>nd</sup> edition of version 1.0.

The `svg:stroke-color` attribute may be used with the following element: `<style:graphic-properties>` 16.14.

### 18.911. `svg:stroke-opacity`

The attribute `svg:stroke-opacity` specifies the opacity of a stroke. The value of this attribute can be a number between 0 (fully transparent) and 1 (fully opaque) or a percentage.

**Ed. Note** Why did we add percentage to the SVG values? In this particular case, we run into the signed percentage issue, what does -50% opacity mean? SVG reference is 11.4.

The `svg:stroke-opacity` attribute may be used with the following element: `<style:graphic-properties>` 16.14.

### 18.912. `svg:stroke-width`

The attribute `svg:stroke-width` specifies the width of a stroke.

The `svg:stroke-width` attribute may be used with the following element: `<style:graphic-properties>` 16.14.

### 18.913. `svg:type`

See §19.2.14 of [SVG].

**Ed. Note** We follow the SVG definition here.

The `svg:type` attribute may be used with the following element: `<anim:animateTransform>` 9.9.4.

### 18.914. `svg:underline-position`

See §20.8.3 of [SVG].

The `svg:underline-position` attribute may be used with the following element: `<style:font-face>` 15.9.

### 18.915. `svg:underline-thickness`

See §20.8.3 of [SVG].

The `svg:underline-thickness` attribute may be used with the following element: `<style:font-face>` 15.9.

### 18.916. `svg:unicode-range`

See §20.8.3 of [SVG].

The `svg:unicode-range` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.917. **svg:units-per-em**

See §20.8.3 of [SVG].

The `svg:units-per-em` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.918. **svg:v-alphabetic**

See §20.8.3 of [SVG].

The `svg:v-alphabetic` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.919. **svg:v-hanging**

See §20.8.3 of [SVG].

The `svg:v-hanging` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.920. **svg:v-ideographic**

See §20.8.3 of [SVG].

The `svg:v-ideographic` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.921. **svg:v-mathematical**

See §20.8.3 of [SVG].

The `svg:v-mathematical` attribute may be used with the following element: `<style:font-face>` 15.9.

## 18.922. **svg:viewBox**

The `svg:viewBox` attribute establishes a user coordinate system inside the physical coordinate system of the shape specified by the position and size attributes. This user coordinate system is used by the `<draw:enhanced-geometry>` element.

The syntax for using this attribute is the same as the [SVG] syntax. The value of the attribute are four numbers separated by white spaces, which define the left, top, right, and bottom dimensions of the user coordinate system.

Implementations may ignore the view box attribute. The implied coordinate system then has its origin at the left, top corner of the shape, without any scaling relative to the shape.

**Ed. Note** I compared this to the definition of `viewBox` in SVG 7.7 and the use of the coordinate system language seems unnecessarily dense to me. Isn't it true that a `viewBox` is defining the the display of an image in a particular container? Can't we simply say that without the user coordinate system versus physical coordinate system language? Or is there something being said here that I am missing?