#### Colors

Given its own section within DrawingML [Basics](Basics.docx), colors are an integral part of the DrawingML framework. Colors are used in virtually every object to help describe it's appearance when it is rendered on the screen. Since not every generating application wishes to represent color in the same manner, it is possible to specify color in a number of different ways.

##### alpha (Alpha)

This element specifies its input color with the specific opacity, but with its color unchanged.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the opacity as expressed by a percentage value.  [Example: The following represents a green solid [fill](fill.docx) which is 50% opaque  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:alpha [val](val.docx)="50000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_PositiveFixedPercentage](ST_PositiveFixedPercentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_PositiveFixedPercentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_PositiveFixedPercentage](ST_PositiveFixedPercentage.docx)" use="required"/>

</complexType>

##### [alphaMod](alphaMod.docx) (Alpha Modulation)

This element specifies a more or less opaque version of its input color. An alpha modulate never increases the alpha beyond 100%. A 200% alpha modulate makes a input color twice as opaque as before. A 50% alpha modulate makes a input color half as opaque as before.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the opacity as expressed by a percentage relative to the input color.  [Example: The following represents a green solid [fill](fill.docx) which is 50% opaque  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:alphaMod [val](val.docx)="50000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_PositivePercentage](ST_PositivePercentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_PositivePercentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_PositivePercentage](ST_PositivePercentage.docx)" use="required"/>

</complexType>

##### alphaOff (Alpha Offset)

This element specifies a more or less opaque version of its input color. Increases or decreases the input alpha percentage by the specified percentage offset. A 10% alpha offset increases a 50% opacity to 60%. A -10% alpha offset decreases a 50% opacity to 40%. The transformed alpha values are limited to a range of 0 to 100%. A 10% alpha offset increase to a 100% opaque object still results in 100% opacity.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the opacity as expressed by a percentage offset increase or decrease relative to the input color. Increases will never increase the opacity beyond 100%, decreases will never decrease the opacity below 0%.  [Example: The following represents a green solid [fill](fill.docx) which is 90% opaque  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:alphaOff [val](val.docx)="-10000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_FixedPercentage](ST_FixedPercentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_FixedPercentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_FixedPercentage](ST_FixedPercentage.docx)" use="required"/>

</complexType>

##### blue (Blue)

This element specifies the input color with the specific blue component, but with the red and green color components unchanged.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### blueMod (Blue Modification)

This element specifies the input color with its blue component modulated by the given percentage. A 50% blue modulate will reduce the blue component by half. A 200% blue modulate will double the blue component.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### blueOff (Blue Offset)

This element specifies the input color with its blue component shifted, but with its red and green color components unchanged.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### comp (Complement)

This element specifies that the color rendered should be the complement of its input color with the complement being defined as such. Two colors are called complementary if, when mixed they produce a shade of grey. For instance, the complement of red which is RGB (255, 0, 0) is cyan which is RGB (0, 255, 255).

Primary colors and secondary colors are typically paired in this way:

* red and cyan (where cyan is the mixture of green and blue)
* green and magenta (where magenta is the mixture of red and blue)
* blue and yellow (where yellow is the mixture of red and green)

[Example:

The following represents the complement of red:

<a:solidFill>  
 <a:srgbClr [val](val.docx)="FF0000">  
 <a:comp/>  
 </a:srgbClr>  
</a:solidFill>

[end](end.docx) example]

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_ComplementTransform"/>

##### gamma (Gamma)

This element specifies that the output color rendered by the generating application should be the sRGB gamma shift of the input color.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_GammaTransform"/>

##### gray (Gray)

This element specifies a grayscale of its input color, taking into relative intensities of the red, green, and blue primaries.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_GrayscaleTransform"/>

##### green (Green)

This elements specifies the input color with the specified green component, but with its red and blue color components unchanged.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### greenMod (Green Modification)

This element specifies the input color with its green component modulated by the given percentage. A 50% green modulate will reduce the green component by half. A 200% green modulate will double the green component.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### greenOff (Green Offset)

This element specifies the input color with its green component shifted, but with its red and blue color components unchanged.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### hslClr (Hue, Saturation, Luminance Color Model)

This element specifies a color using the HSL color model. A perceptual gamma of 2.2 is assumed.

Hue refers to the dominant wavelength of color, saturation refers to the purity of its hue, and luminance refers to its lightness or darkness.

As with all colors, colors defined with the HSL color model may have color transforms applied to it.

[Example:

The color blue having RGB value RRGGBB = (00, 00, 80) is equivalent to

<a:solidFill>

<a:hslClr hue="14400000" sat="100000" [lum](lum.docx)="50000">

</a:solidFill>

[end](end.docx) example]

|  |
| --- |
| Parent Elements |
| accent1 (§); accent2 (§); accent3 (§); accent4 (§); accent5 (§); accent6 (§); [alphaInv](alphaInv.docx) (§); [bgClr](bgClr.docx) (§); bgRef (§); buClr (§); [clrFrom](clrFrom.docx) (§); clrMru (§); [clrRepl](clrRepl.docx) (§); [clrTo](clrTo.docx) (§); clrVal (§); [contourClr](contourClr.docx) (§); custClr (§); dk1 (§); dk2 (§); [duotone](duotone.docx) (§); [effectClrLst](effectClrLst.docx) (§); effectRef (§); [extrusionClr](extrusionClr.docx) (§); [fgClr](fgClr.docx) (§); [fillClrLst](fillClrLst.docx) (§); fillRef (§); folHlink (§); fontRef (§); from (§); [glow](glow.docx) (§); [gs](gs.docx) (§); highlight (§); hlink (§); [innerShdw](innerShdw.docx) (§); [linClrLst](linClrLst.docx) (§); lnRef (§); lt1 (§); lt2 (§); [outerShdw](outerShdw.docx) (§); penClr (§); [prstShdw](prstShdw.docx) (§); [solidFill](solidFill.docx) (§); tcTxStyle (§); to (§); [txEffectClrLst](txEffectClrLst.docx) (§); [txFillClrLst](txFillClrLst.docx) (§); [txLinClrLst](txLinClrLst.docx) (§) |

|  |  |
| --- | --- |
| Child Elements | Subclause |
| alpha (Alpha) | § |
| [alphaMod](alphaMod.docx) (Alpha Modulation) | § |
| alphaOff (Alpha Offset) | § |
| blue (Blue) | § |
| blueMod (Blue Modification) | § |
| blueOff (Blue Offset) | § |
| comp (Complement) | § |
| gamma (Gamma) | § |
| gray (Gray) | § |
| green (Green) | § |
| greenMod (Green Modification) | § |
| greenOff (Green Offset) | § |
| hue (Hue) | § |
| hueMod (Hue Modulate) | § |
| hueOff (Hue Offset) | § |
| inv (Inverse) | § |
| invGamma (Inverse Gamma) | § |
| [lum](lum.docx) (Luminance) | § |
| lumMod (Luminance Modulation) | § |
| lumOff (Luminance Offset) | § |
| red (Red) | § |
| redMod (Red Modulation) | § |
| redOff (Red Offset) | § |
| sat (Saturation) | § |
| satMod (Saturation Modulation) | § |
| satOff (Saturation Offset) | § |
| shade (Shade) | § |
| [tint](tint.docx) (Tint) | § |

|  |  |
| --- | --- |
| Attributes | Description |
| hue (Hue) | Specifies the angular value describing the wavelength. Expressed in 1/6000ths of a degree.  The possible values for this attribute are defined by the [ST\_PositiveFixedAngle](ST_PositiveFixedAngle.docx) simple type (§). |
| [lum](lum.docx) (Luminance) | Specifies the luminance referring to the lightness or darkness of the color. Expressed as a percentage with 0% referring to maximal dark (black) and 100% referring to maximal white.  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |
| sat (Saturation) | Specifies the saturation referring to the purity of the hue. Expressed as a percentage with 0% referring to grey, 100% referring to the purest form of the hue.  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_HslColor">

<sequence>

<group ref="EG\_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>

</sequence>

<attribute [name](name.docx)="hue" type="[ST\_PositiveFixedAngle](ST_PositiveFixedAngle.docx)" use="required"/>

<attribute [name](name.docx)="sat" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

<attribute [name](name.docx)="[lum](lum.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### hue (Hue)

This element specifies the input color with the specified hue, but with its saturation and luminance unchanged.

[Example: The following two solid fills are equivalent.

<a:solidFill>  
 <a:hslClr hue="14400000" sat="100000" [lum](lum.docx)="50000">  
</a:solidFill>

<a:solidFill>  
 <a:hslClr hue="0" sat="100000" [lum](lum.docx)="50000">  
 <a:hue [val](val.docx)="14400000"/>  
 <a:hslClr/>  
</a:solidFill>

[end](end.docx) example]

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the actual angle value to be used with the input color's hue component.  The possible values for this attribute are defined by the [ST\_PositiveFixedAngle](ST_PositiveFixedAngle.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_PositiveFixedAngle">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_PositiveFixedAngle](ST_PositiveFixedAngle.docx)" use="required"/>

</complexType>

##### hueMod (Hue Modulate)

This element specifies the input color with its hue modulated by the given percentage. A 50% hue modulate decreases the angular hue value by half. A 200% hue modulate doubles the angular hue value.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the opacity as expressed by a percentage relative to the input color.  [Example: The following represents a green solid [fill](fill.docx) which is 50% opaque  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:alphaMod [val](val.docx)="50000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_PositivePercentage](ST_PositivePercentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_PositivePercentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_PositivePercentage](ST_PositivePercentage.docx)" use="required"/>

</complexType>

##### hueOff (Hue Offset)

This element specifies the input color with its hue shifted, but with its saturation and luminance unchanged.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the actual angular value of the shift. The result of the shift must be between 0 and 360 degrees. Shifts resulting in angular values less than 0 will be treated as 0. Shifts resulting in angular values greater than 360 will be treated as 360.  [Example:  The following increases the hue angular value by 10 degrees.  <a:solidFill>  <a:hslClr hue="0" sat="100000" [lum](lum.docx)="50000"/>  <a:hueOff [val](val.docx)="600000"/> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Angle](ST_Angle.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Angle">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Angle](ST_Angle.docx)" use="required"/>

</complexType>

##### inv (Inverse)

This element specifies the inverse of its input color. For example, the inverse of red (1, 0, 0) is cyan (0, 1, 1 ).

[Example:

The following represents cyan, the inverse of red:

<a:solidFill>  
 <a:srgbClr [val](val.docx)="FF0000">  
 <a:inv/>  
 </a:srgbClr>  
</a:solidFill>

[end](end.docx) example]

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_InverseTransform"/>

##### invGamma (Inverse Gamma)

This element specifies that the output color rendered by the generating application should be the inverse sRGB gamma shift of the input color.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_InverseGammaTransform"/>

##### [lum](lum.docx) (Luminance)

This element specifies the input color with the specified luminance, but with its hue and saturation unchanged. Typically luminance values fall in the range [0%, 100%].

[Example:

The following two solid fills are equivalent:

<a:solidFill>  
 <a:hslClr hue="14400000" sat="100000" [lum](lum.docx)="50000">  
</a:solidFill>

<a:solidFill>  
 <a:hslClr hue="14400000" sat="100000" [lum](lum.docx)="0">  
 <a:lum [val](val.docx)="50000"/>  
 <a:hslClr/>  
</a:solidFill>

[end](end.docx) example]

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### lumMod (Luminance Modulation)

This element specifies the input color with its luminance modulated by the given percentage. A 50% luminance modulate will reduce the luminance by half. A 200% luminance modulate will double the luminance.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### lumOff (Luminance Offset)

This element specifies the input color with its luminance shifted, but with its hue and saturation unchanged.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### prstClr (Preset Color)

This element specifies a color which is bound to one of a predefined collection of colors.

[Example:

The following defines a solid [fill](fill.docx) bound to the "black" preset color.

<a:solidFill>  
 <a:prstClr [val](val.docx)="black">  
</a:solidFill>

[end](end.docx) example]

|  |
| --- |
| Parent Elements |
| accent1 (§); accent2 (§); accent3 (§); accent4 (§); accent5 (§); accent6 (§); [alphaInv](alphaInv.docx) (§); [bgClr](bgClr.docx) (§); bgRef (§); buClr (§); [clrFrom](clrFrom.docx) (§); clrMru (§); [clrRepl](clrRepl.docx) (§); [clrTo](clrTo.docx) (§); clrVal (§); [contourClr](contourClr.docx) (§); custClr (§); dk1 (§); dk2 (§); [duotone](duotone.docx) (§); [effectClrLst](effectClrLst.docx) (§); effectRef (§); [extrusionClr](extrusionClr.docx) (§); [fgClr](fgClr.docx) (§); [fillClrLst](fillClrLst.docx) (§); fillRef (§); folHlink (§); fontRef (§); from (§); [glow](glow.docx) (§); [gs](gs.docx) (§); highlight (§); hlink (§); [innerShdw](innerShdw.docx) (§); [linClrLst](linClrLst.docx) (§); lnRef (§); lt1 (§); lt2 (§); [outerShdw](outerShdw.docx) (§); penClr (§); [prstShdw](prstShdw.docx) (§); [solidFill](solidFill.docx) (§); tcTxStyle (§); to (§); [txEffectClrLst](txEffectClrLst.docx) (§); [txFillClrLst](txFillClrLst.docx) (§); [txLinClrLst](txLinClrLst.docx) (§) |

|  |  |
| --- | --- |
| Child Elements | Subclause |
| alpha (Alpha) | § |
| [alphaMod](alphaMod.docx) (Alpha Modulation) | § |
| alphaOff (Alpha Offset) | § |
| blue (Blue) | § |
| blueMod (Blue Modification) | § |
| blueOff (Blue Offset) | § |
| comp (Complement) | § |
| gamma (Gamma) | § |
| gray (Gray) | § |
| green (Green) | § |
| greenMod (Green Modification) | § |
| greenOff (Green Offset) | § |
| hue (Hue) | § |
| hueMod (Hue Modulate) | § |
| hueOff (Hue Offset) | § |
| inv (Inverse) | § |
| invGamma (Inverse Gamma) | § |
| [lum](lum.docx) (Luminance) | § |
| lumMod (Luminance Modulation) | § |
| lumOff (Luminance Offset) | § |
| red (Red) | § |
| redMod (Red Modulation) | § |
| redOff (Red Offset) | § |
| sat (Saturation) | § |
| satMod (Saturation Modulation) | § |
| satOff (Saturation Offset) | § |
| shade (Shade) | § |
| [tint](tint.docx) (Tint) | § |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the actual preset color value.  The possible values for this attribute are defined by the [ST\_PresetColorVal](ST_PresetColorVal.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_PresetColor">

<sequence>

<group ref="EG\_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>

</sequence>

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_PresetColorVal](ST_PresetColorVal.docx)"/>

</complexType>

##### red (Red)

This element specifies the input color with the specified red component, but with its green and blue color components unchanged.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### redMod (Red Modulation)

This element specifies the input color with its red component modulated by the given percentage. A 50% red modulate will reduce the red component by half. A 200% red modulate will double the red component.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### redOff (Red Offset)

This element specifies the input color with its red component shifted, but with its green and blue color components unchanged.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### sat (Saturation)

This element specifies the input color with the specified saturation, but with its hue and luminance unchanged. Typically saturation values fall in the range [0%, 100%].  
  
[Example:

The following two solid fills are equivalent:

<a:solidFill>  
 <a:hslClr hue="14400000" sat="100000" [lum](lum.docx)="50000">  
</a:solidFill>

<a:solidFill>  
 <a:hslClr hue="14400000" sat="0" [lum](lum.docx)="50000">  
 <a:sat [val](val.docx)="100000"/>  
 <a:hslClr/>  
</a:solidFill>

[end](end.docx) example]

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### satMod (Saturation Modulation)

This element specifies the input color with its saturation modulated by the given percentage. A 50% saturation modulate will reduce the saturation by half. A 200% saturation modulate will double the saturation.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### satOff (Saturation Offset)

This element specifies the input color with its saturation shifted, but with its hue and luminance unchanged. A 10% offset to 20% saturation yields 30% saturation.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the value of the blue component. The assigned value is specified as a percentage with 0% indicating minimal blue and 100% indicating maximum blue.  [Example: The following manipulates the [fill](fill.docx) from having RGB value RRGGBB = (00, FF, 00) to value RRGGBB= (00, FF, FF)  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:blue [val](val.docx)="100000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_Percentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### schemeClr (Scheme Color)

This element specifies a color bound to a user's theme. As with all elements which define a color, it is possible to apply a list of color transforms to the base color defined.

|  |
| --- |
| Parent Elements |
| accent1 (§); accent2 (§); accent3 (§); accent4 (§); accent5 (§); accent6 (§); [alphaInv](alphaInv.docx) (§); [bgClr](bgClr.docx) (§); bgRef (§); buClr (§); [clrFrom](clrFrom.docx) (§); clrMru (§); [clrRepl](clrRepl.docx) (§); [clrTo](clrTo.docx) (§); clrVal (§); [contourClr](contourClr.docx) (§); custClr (§); dk1 (§); dk2 (§); [duotone](duotone.docx) (§); [effectClrLst](effectClrLst.docx) (§); effectRef (§); [extrusionClr](extrusionClr.docx) (§); [fgClr](fgClr.docx) (§); [fillClrLst](fillClrLst.docx) (§); fillRef (§); folHlink (§); fontRef (§); from (§); [glow](glow.docx) (§); [gs](gs.docx) (§); highlight (§); hlink (§); [innerShdw](innerShdw.docx) (§); [linClrLst](linClrLst.docx) (§); lnRef (§); lt1 (§); lt2 (§); [outerShdw](outerShdw.docx) (§); penClr (§); [prstShdw](prstShdw.docx) (§); [solidFill](solidFill.docx) (§); tcTxStyle (§); to (§); [txEffectClrLst](txEffectClrLst.docx) (§); [txFillClrLst](txFillClrLst.docx) (§); [txLinClrLst](txLinClrLst.docx) (§) |

|  |  |
| --- | --- |
| Child Elements | Subclause |
| alpha (Alpha) | § |
| [alphaMod](alphaMod.docx) (Alpha Modulation) | § |
| alphaOff (Alpha Offset) | § |
| blue (Blue) | § |
| blueMod (Blue Modification) | § |
| blueOff (Blue Offset) | § |
| comp (Complement) | § |
| gamma (Gamma) | § |
| gray (Gray) | § |
| green (Green) | § |
| greenMod (Green Modification) | § |
| greenOff (Green Offset) | § |
| hue (Hue) | § |
| hueMod (Hue Modulate) | § |
| hueOff (Hue Offset) | § |
| inv (Inverse) | § |
| invGamma (Inverse Gamma) | § |
| [lum](lum.docx) (Luminance) | § |
| lumMod (Luminance Modulation) | § |
| lumOff (Luminance Offset) | § |
| red (Red) | § |
| redMod (Red Modulation) | § |
| redOff (Red Offset) | § |
| sat (Saturation) | § |
| satMod (Saturation Modulation) | § |
| satOff (Saturation Offset) | § |
| shade (Shade) | § |
| [tint](tint.docx) (Tint) | § |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the desired scheme.  [Example: The following represents a color bound to the "lt1" [theme](theme.docx) color  <a:solidFill>  <a:schemeClr [val](val.docx)="lt1"/> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_SchemeColorVal](ST_SchemeColorVal.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_SchemeColor">

<sequence>

<group ref="EG\_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>

</sequence>

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_SchemeColorVal](ST_SchemeColorVal.docx)" use="required"/>

</complexType>

##### scrgbClr (RGB Color Model - Percentage Variant)

This element specifies a color using the red, green, blue RGB color model. Each component, red, green, and blue is expressed as a percentage from 0% to 100%. A linear gamma of 1.0 is assumed.

Specifies the level of red as expressed by a percentage offset increase or decrease relative to the input color.

[Example: The following represent the same color

<a:solidFill>  
 <a:scrgbClr r="50000" g="50000" b="50000"/>  
</a:solidFill>  
<a:solidFill>  
 <a:srgbClr [val](val.docx)="BCBCBC"/>  
</a:solidFill>

[end](end.docx) example]

|  |
| --- |
| Parent Elements |
| accent1 (§); accent2 (§); accent3 (§); accent4 (§); accent5 (§); accent6 (§); [alphaInv](alphaInv.docx) (§); [bgClr](bgClr.docx) (§); bgRef (§); buClr (§); [clrFrom](clrFrom.docx) (§); clrMru (§); [clrRepl](clrRepl.docx) (§); [clrTo](clrTo.docx) (§); clrVal (§); [contourClr](contourClr.docx) (§); custClr (§); dk1 (§); dk2 (§); [duotone](duotone.docx) (§); [effectClrLst](effectClrLst.docx) (§); effectRef (§); [extrusionClr](extrusionClr.docx) (§); [fgClr](fgClr.docx) (§); [fillClrLst](fillClrLst.docx) (§); fillRef (§); folHlink (§); fontRef (§); from (§); [glow](glow.docx) (§); [gs](gs.docx) (§); highlight (§); hlink (§); [innerShdw](innerShdw.docx) (§); [linClrLst](linClrLst.docx) (§); lnRef (§); lt1 (§); lt2 (§); [outerShdw](outerShdw.docx) (§); penClr (§); [prstShdw](prstShdw.docx) (§); [solidFill](solidFill.docx) (§); tcTxStyle (§); to (§); [txEffectClrLst](txEffectClrLst.docx) (§); [txFillClrLst](txFillClrLst.docx) (§); [txLinClrLst](txLinClrLst.docx) (§) |

|  |  |
| --- | --- |
| Child Elements | Subclause |
| alpha (Alpha) | § |
| [alphaMod](alphaMod.docx) (Alpha Modulation) | § |
| alphaOff (Alpha Offset) | § |
| blue (Blue) | § |
| blueMod (Blue Modification) | § |
| blueOff (Blue Offset) | § |
| comp (Complement) | § |
| gamma (Gamma) | § |
| gray (Gray) | § |
| green (Green) | § |
| greenMod (Green Modification) | § |
| greenOff (Green Offset) | § |
| hue (Hue) | § |
| hueMod (Hue Modulate) | § |
| hueOff (Hue Offset) | § |
| inv (Inverse) | § |
| invGamma (Inverse Gamma) | § |
| [lum](lum.docx) (Luminance) | § |
| lumMod (Luminance Modulation) | § |
| lumOff (Luminance Offset) | § |
| red (Red) | § |
| redMod (Red Modulation) | § |
| redOff (Red Offset) | § |
| sat (Saturation) | § |
| satMod (Saturation Modulation) | § |
| satOff (Saturation Offset) | § |
| shade (Shade) | § |
| [tint](tint.docx) (Tint) | § |

|  |  |
| --- | --- |
| Attributes | Description |
| b (Blue) | Specifies the percentage of blue.  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |
| g (Green) | Specifies the percentage of green.  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |
| r (Red) | Specifies the percentage of red.  The possible values for this attribute are defined by the [ST\_Percentage](ST_Percentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_ScRgbColor">

<sequence>

<group ref="EG\_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>

</sequence>

<attribute [name](name.docx)="r" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

<attribute [name](name.docx)="g" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

<attribute [name](name.docx)="b" type="[ST\_Percentage](ST_Percentage.docx)" use="required"/>

</complexType>

##### shade (Shade)

This element specifies a darker version of its input color. A 10% shade is 10% of the input color combined with 90% black.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the opacity as expressed by a percentage value.  [Example: The following represents a green solid [fill](fill.docx) which is 50% opaque  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:alpha [val](val.docx)="50000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_PositiveFixedPercentage](ST_PositiveFixedPercentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_PositiveFixedPercentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_PositiveFixedPercentage](ST_PositiveFixedPercentage.docx)" use="required"/>

</complexType>

##### srgbClr (RGB Color Model - Hex Variant)

This element specifies a color using the red, green, blue RGB color model. Red, green, and blue is expressed as sequence of hex digits, RRGGBB. A perceptual gamma of 2.2 is used.

Specifies the level of red as expressed by a percentage offset increase or decrease relative to the input color.

[Example: The following represent the same color

<a:solidFill>  
 <a:scrgbClr r="50000" g="50000" b="50000"/>  
</a:solidFill>  
<a:solidFill>  
 <a:srgbClr [val](val.docx)="BCBCBC"/>  
</a:solidFill>

[end](end.docx) example]

|  |
| --- |
| Parent Elements |
| accent1 (§); accent2 (§); accent3 (§); accent4 (§); accent5 (§); accent6 (§); [alphaInv](alphaInv.docx) (§); [bgClr](bgClr.docx) (§); bgRef (§); buClr (§); [clrFrom](clrFrom.docx) (§); clrMru (§); [clrRepl](clrRepl.docx) (§); [clrTo](clrTo.docx) (§); clrVal (§); [contourClr](contourClr.docx) (§); custClr (§); dk1 (§); dk2 (§); [duotone](duotone.docx) (§); [effectClrLst](effectClrLst.docx) (§); effectRef (§); [extrusionClr](extrusionClr.docx) (§); [fgClr](fgClr.docx) (§); [fillClrLst](fillClrLst.docx) (§); fillRef (§); folHlink (§); fontRef (§); from (§); [glow](glow.docx) (§); [gs](gs.docx) (§); highlight (§); hlink (§); [innerShdw](innerShdw.docx) (§); [linClrLst](linClrLst.docx) (§); lnRef (§); lt1 (§); lt2 (§); [outerShdw](outerShdw.docx) (§); penClr (§); [prstShdw](prstShdw.docx) (§); [solidFill](solidFill.docx) (§); tcTxStyle (§); to (§); [txEffectClrLst](txEffectClrLst.docx) (§); [txFillClrLst](txFillClrLst.docx) (§); [txLinClrLst](txLinClrLst.docx) (§) |

|  |  |
| --- | --- |
| Child Elements | Subclause |
| alpha (Alpha) | § |
| [alphaMod](alphaMod.docx) (Alpha Modulation) | § |
| alphaOff (Alpha Offset) | § |
| blue (Blue) | § |
| blueMod (Blue Modification) | § |
| blueOff (Blue Offset) | § |
| comp (Complement) | § |
| gamma (Gamma) | § |
| gray (Gray) | § |
| green (Green) | § |
| greenMod (Green Modification) | § |
| greenOff (Green Offset) | § |
| hue (Hue) | § |
| hueMod (Hue Modulate) | § |
| hueOff (Hue Offset) | § |
| inv (Inverse) | § |
| invGamma (Inverse Gamma) | § |
| [lum](lum.docx) (Luminance) | § |
| lumMod (Luminance Modulation) | § |
| lumOff (Luminance Offset) | § |
| red (Red) | § |
| redMod (Red Modulation) | § |
| redOff (Red Offset) | § |
| sat (Saturation) | § |
| satMod (Saturation Modulation) | § |
| satOff (Saturation Offset) | § |
| shade (Shade) | § |
| [tint](tint.docx) (Tint) | § |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | The actual color value. Expressed as a sequence of hex digits RRGGBB.  The possible values for this attribute are defined by the [ST\_HexBinary3](ST_HexBinary3.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_SRgbColor">

<sequence>

<group ref="EG\_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>

</sequence>

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_HexBinary3](ST_HexBinary3.docx)" use="required"/>

</complexType>

##### sysClr (System Color)

This element specifies a color bound to predefined operating system elements.

[Example: The following represents the default color used for displaying text in a window.

<a:solidFill>  
 <a:sysClr [val](val.docx)="windowText"/>  
</a:solidFill>

[end](end.docx) example]

|  |
| --- |
| Parent Elements |
| accent1 (§); accent2 (§); accent3 (§); accent4 (§); accent5 (§); accent6 (§); [alphaInv](alphaInv.docx) (§); [bgClr](bgClr.docx) (§); bgRef (§); buClr (§); [clrFrom](clrFrom.docx) (§); clrMru (§); [clrRepl](clrRepl.docx) (§); [clrTo](clrTo.docx) (§); clrVal (§); [contourClr](contourClr.docx) (§); custClr (§); dk1 (§); dk2 (§); [duotone](duotone.docx) (§); [effectClrLst](effectClrLst.docx) (§); effectRef (§); [extrusionClr](extrusionClr.docx) (§); [fgClr](fgClr.docx) (§); [fillClrLst](fillClrLst.docx) (§); fillRef (§); folHlink (§); fontRef (§); from (§); [glow](glow.docx) (§); [gs](gs.docx) (§); highlight (§); hlink (§); [innerShdw](innerShdw.docx) (§); [linClrLst](linClrLst.docx) (§); lnRef (§); lt1 (§); lt2 (§); [outerShdw](outerShdw.docx) (§); penClr (§); [prstShdw](prstShdw.docx) (§); [solidFill](solidFill.docx) (§); tcTxStyle (§); to (§); [txEffectClrLst](txEffectClrLst.docx) (§); [txFillClrLst](txFillClrLst.docx) (§); [txLinClrLst](txLinClrLst.docx) (§) |

|  |  |
| --- | --- |
| Child Elements | Subclause |
| alpha (Alpha) | § |
| [alphaMod](alphaMod.docx) (Alpha Modulation) | § |
| alphaOff (Alpha Offset) | § |
| blue (Blue) | § |
| blueMod (Blue Modification) | § |
| blueOff (Blue Offset) | § |
| comp (Complement) | § |
| gamma (Gamma) | § |
| gray (Gray) | § |
| green (Green) | § |
| greenMod (Green Modification) | § |
| greenOff (Green Offset) | § |
| hue (Hue) | § |
| hueMod (Hue Modulate) | § |
| hueOff (Hue Offset) | § |
| inv (Inverse) | § |
| invGamma (Inverse Gamma) | § |
| [lum](lum.docx) (Luminance) | § |
| lumMod (Luminance Modulation) | § |
| lumOff (Luminance Offset) | § |
| red (Red) | § |
| redMod (Red Modulation) | § |
| redOff (Red Offset) | § |
| sat (Saturation) | § |
| satMod (Saturation Modulation) | § |
| satOff (Saturation Offset) | § |
| shade (Shade) | § |
| [tint](tint.docx) (Tint) | § |

|  |  |
| --- | --- |
| Attributes | Description |
| lastClr (Last Color) | Specifies the color value that was last computed by the generating application.  The possible values for this attribute are defined by the [ST\_HexBinary3](ST_HexBinary3.docx) simple type (§). |
| [val](val.docx) (Value) | Specifies the system color value.  The possible values for this attribute are defined by the [ST\_SystemColorVal](ST_SystemColorVal.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_SystemColor">

<sequence>

<group ref="EG\_ColorTransform" minOccurs="0" maxOccurs="unbounded"/>

</sequence>

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_SystemColorVal](ST_SystemColorVal.docx)" use="required"/>

<attribute [name](name.docx)="lastClr" type="[ST\_HexBinary3](ST_HexBinary3.docx)" use="optional"/>

</complexType>

##### [tint](tint.docx) (Tint)

This element specifies a lighter version of its input color. A 10% [tint](tint.docx) is 10% of the input color combined with 90% white.

|  |
| --- |
| Parent Elements |
| hslClr (§); prstClr (§); schemeClr (§); scrgbClr (§); srgbClr (§); sysClr (§) |

|  |  |
| --- | --- |
| Attributes | Description |
| [val](val.docx) (Value) | Specifies the opacity as expressed by a percentage value.  [Example: The following represents a green solid [fill](fill.docx) which is 50% opaque  <a:solidFill>  <a:srgbClr [val](val.docx)="00FF00">  <a:alpha [val](val.docx)="50000"/>  </a:srgbClr> </a:solidFill>  [end](end.docx) example]  The possible values for this attribute are defined by the [ST\_PositiveFixedPercentage](ST_PositiveFixedPercentage.docx) simple type (§). |

The following XML Schema fragment defines the contents of this element:

<complexType [name](name.docx)="CT\_PositiveFixedPercentage">

<attribute [name](name.docx)="[val](val.docx)" type="[ST\_PositiveFixedPercentage](ST_PositiveFixedPercentage.docx)" use="required"/>

</complexType>