### tav (Time Animate Value)

This element defines a "keypoint" in animation interpolation.

[Example: Consider a shape with a "fly-in" animation. The <tav> element should be used as follows:

<p:[anim](anim.docx) calcmode="lin" valueType="num">

<p:[cBhvr](cBhvr.docx) additive="base">...

<p:[tavLst](tavLst.docx)>

<p:tav tm="0">

<p:[val](val.docx)>

<p:[strVal](strVal.docx) [val](val.docx)="1+#ppt\_h/2"/>

</p:[val](val.docx)>

</p:tav>

<p:tav tm="100000">

<p:[val](val.docx)>

<p:[strVal](strVal.docx) [val](val.docx)="#ppt\_y"/>

</p:[val](val.docx)>

</p:tav>

</p:[tavLst](tavLst.docx)>

</p:[anim](anim.docx)>

End Example]

|  |
| --- |
| Parent Elements |
| [tavLst](tavLst.docx) (§) |

|  |  |
| --- | --- |
| Child Elements | Subclause |
| [val](val.docx) (Value) | § |

|  |  |
| --- | --- |
| Attributes | Description |
| fmla (Formula) | This attribute allows for a specific formula to be used during the animation. This is specified in a semicolon-separated list of formulas. The value generated by interpolating the values list will be fed into the formula as an input. The resulting value is used to set the property being animated. There should be n-1 number of formulas for n values in the values list. The formula is applied starting at the corresponding time in the keyTimes list and up to the next time in the list.    Formulas can only support a calcMode (Calculation Mode) of linear or discrete. If another calcMode is specified or no calcMode is specified then a calcMode of linear will be assumed.  Formulas must be written using the elements defined below. When a formula is used in an attribute that takes more than one parameter (such as for animateScale or animateMotion), a space will be taken as a delimiter between parameter values. To work around this, either don't put spaces in the formula or include parentheses around the formula.    Formulas within values, from, to, by attributes can be made up of these:   * Standard arithmetic operators: ‘+’, ‘-‘, ‘\*’, ‘/’, ‘^’, ‘%’ (mod) * Constants: ‘pi’ ‘e’ * Conditional operators: ‘abs’, ‘min’, ‘max’, ‘?’ (if) * Comparison operators: '==', '>=', '<=', '!=', '!' * Trigonometric operators: ‘sin()’, ‘cos()’, ‘tan()’, ‘asin()’, ‘acos()’, ‘atan()’ * Natural logarithm ‘ln()’ * Property references (host supported properties)     [Note: The above formula elements are different than those used in javascript expressions. For example "cos()" used in formulas would be "Math.cos()" in javascript. end note]  The possible values for this attribute are defined by the XML Schema string datatype. |
| tm (Time) | This attribute specifies the time at which the attribute being animated will take on the value.  The possible values for this attribute are defined by the [ST\_TLTimeAnimateValueTime](ST_TLTimeAnimateValueTime.docx) simple type (§). |

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

<complexType name="CT\_TLTimeAnimateValue">

<sequence>

<element name="[val](val.docx)" type="CT\_TLAnimVariant" minOccurs="0" maxOccurs="1"/>

</sequence>

<attribute name="tm" type="[ST\_TLTimeAnimateValueTime](ST_TLTimeAnimateValueTime.docx)" use="optional" default="indefinite"/>

<attribute name="fmla" type="xsd:string" use="optional" default=""/>

</complexType>