### iterate (Iterate)

This element specifies how the animation should be successively applied to sub elements of the target element for a repeated effect. It can be applied to contained [timing](timing.docx) and animation structures over the letters, words, or shapes within a target element.

[Example: Consider a text animation where the words appear letter by letter. The <iterate> element should be used as follows:

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

 <p:[cTn](cTn.docx) id="1" >

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

 <p:iterate type="lt">

 <p:[tmPct](tmPct.docx) [val](val.docx)="10000"/>

 </p:iterate>

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

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

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

End Example]

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

|  |  |
| --- | --- |
| Child Elements | Subclause |
| [tmAbs](tmAbs.docx) (Time Absolute) | § |
| [tmPct](tmPct.docx) (Time Percentage) | § |

|  |  |
| --- | --- |
| Attributes | Description |
| backwards (Backwards) | This attribute specifies whether to go backwards in the timeline to the previous node.The possible values for this attribute are defined by the XML Schema boolean datatype. |
| type (Iterate Type) | This attribute specifies the iteration behavior and applies it to each letter, word or shape within a container element. Valid values are by word, by letter, or by element. If there is no text or block elements such as shapes within the container or a single word, letter, or shape (depending on iterate type) then no iteration will happen and the behavior is applied to the element itself instead.The possible values for this attribute are defined by the [ST\_IterateType](ST_IterateType.docx) simple type (§). |

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

<complexType name="CT\_TLIterateData">

 <choice minOccurs="1" maxOccurs="1">

 <element name="[tmAbs](tmAbs.docx)" type="CT\_TLIterateIntervalTime"/>

 <element name="[tmPct](tmPct.docx)" type="CT\_TLIterateIntervalPercentage"/>

 </choice>

 <attribute name="type" type="[ST\_IterateType](ST_IterateType.docx)" use="optional" default="el"/>

 <attribute name="backwards" type="xsd:boolean" use="optional" default="false"/>

</complexType>