#### forEach (For Each)

A looping structure, similar to a for loop in a programming language, which defines what data model points will use this [layout](layout.docx) node.

[Example: Consider the following example of a forEach being used within a DrawingML diagram:

<forEach name="Name5" ref="" axis="ch" ptType="node">

 <[layoutNode](layoutNode.docx) [name](name.docx)="node1" [styleLbl](styleLbl.docx)="" moveWith="">

 <[alg](alg.docx) type="[sp](sp.docx)"/>

 <[shape](shape.docx) xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships" r:blip="">

 <[adjLst](adjLst.docx)/>

 </[shape](shape.docx)>

 <[constrLst](constrLst.docx)/>

 </[layoutNode](layoutNode.docx)>

</forEach>

In this example, the forEach element will create a [layout](layout.docx) node, referenced by the name node1, for every associated data model point in the diagram. In this particular instance the forEach will create the [layout](layout.docx) node for every child of the current point node. [end](end.docx) example]

|  |
| --- |
| Parent Elements |
| [else](else.docx) (§); forEach (§); if (§); [layoutNode](layoutNode.docx) (§) |

|  |  |
| --- | --- |
| Child Elements | Subclause |
| [alg](alg.docx) (Algorithm) | § |
| [choose](choose.docx) (Choose Element) | § |
| [constrLst](constrLst.docx) (Constraint List) | § |
| [extLst](extLst.docx) (Extension List) | § |
| forEach (For Each) | § |
| [layoutNode](layoutNode.docx) (Layout Node) | § |
| [presOf](presOf.docx) (Presentation Of) | § |
| [ruleLst](ruleLst.docx) (Rule List) | § |
| [shape](shape.docx) (Shape) | § |

|  |  |
| --- | --- |
| Attributes | Description |
| axis (Axis) | Specifies the axis on which to select data from the data model. [Example: For example, axis="ch" will select children of the current point node and axis="des" will select all descendants. [end](end.docx) example]The possible values for this attribute are defined by the [ST\_AxisTypes](ST_AxisTypes.docx) simple type (§). |
| cnt (Count) | Specifies the count of items to use in a data set.[Example: Consider the following example of a forEach in a DrawingML diagram:<forEach name="Name5" ref="" axis="ch" ptType="node" cnt="2">...</forEach>In this example, [up](up.docx) to two children will be obtained through this forEach. [end](end.docx) example]The possible values for this attribute are defined by the [ST\_UnsignedInts](ST_UnsignedInts.docx) simple type (§). |
| hideLastTrans (Hide Last Transition) | In algorithms that support transitions, this attribute specifies that the last transition will not be rendered.  This allows for diagrams that [start](start.docx) and [end](end.docx) with a node.The possible values for this attribute are defined by the [ST\_Booleans](ST_Booleans.docx) simple type (§). |
| [name](name.docx) (Name) | A unique identifier for the [layout](layout.docx) node.The possible values for this attribute are defined by the XML Schema string datatype. |
| ptType (Data Point Type) | Specifies the type of data point to select.[Example: Consider the following example of a forEach in a DrawingML diagram:<forEach name="Name5" ref="" axis="ch" ptType="node" cnt="2">...</forEach>In this example, the forEach will select all node type points in the set. [end](end.docx) example]The possible values for this attribute are defined by the [ST\_ElementTypes](ST_ElementTypes.docx) simple type (§). |
| ref (Reference) | When used on a for-each element, causes the specified for-each element to be used instead.The possible values for this attribute are defined by the XML Schema string datatype. |
| [st](st.docx) (Start) | Specifies where to [start](start.docx) in a data set.[Example: Consider the following example of a forEach in a DrawingML diagram:<[presOf](presOf.docx) axis="desOrSelf" ptType="node" [st](st.docx)="2"/>In this example, the second element in the set will be the first point returned. [end](end.docx) example]The possible values for this attribute are defined by the [ST\_Ints](ST_Ints.docx) simple type (§). |
| step (Step) | Specifies the step to use in a data set. A step with a value of 2 will return every other item in the set.The possible values for this attribute are defined by the [ST\_Ints](ST_Ints.docx) simple type (§). |

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

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

 <choice minOccurs="0" maxOccurs="unbounded">

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

 <element name="shape" type="CT\_Shape" minOccurs="0" maxOccurs="1"/>

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

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

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

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

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

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

 <element name="[extLst](extLst.docx)" type="a:CT\_OfficeArtExtensionList" minOccurs="0" maxOccurs="1"/>

 </choice>

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

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

 <attributeGroup ref="AG\_IteratorAttributes"/>

</complexType>