

**Data dictionary for QIF Library Primitives.xsd (normative)**

schema location: **..\QIFLibrary\Primitives.xsd**  
 attributeFormDefault: **unqualified**  
 elementFormDefault: **qualified**  
 targetNamespace: **http://qifstandards.org/xsd/qif2**

## Groups

[LineSegment2dGroup](#)  
[LineSegmentGroup](#)

## Complex types

[ActualAxisType](#)  
[ActualPlaneType](#)  
[ActualPointType](#)  
[ActualUnitVectorType](#)  
[AngleRangeType](#)  
[ArrayBinaryType](#)  
[ArrayDoubleType](#)  
[ArrayI2Type](#)  
[ArrayI3Type](#)  
[ArrayIntType](#)  
[ArrayNaturalType](#)  
[ArrayPoint2dType](#)  
[ArrayPointType](#)  
[ArrayReferenceFullType](#)  
[ArrayReferenceType](#)  
[ArrayUnitVectorType](#)  
[ArrayUnsignedByteType](#)  
[AttributeBaseType](#)  
[AttributeBoolType](#)  
[AttributeD1Type](#)  
[AttributeD3Type](#)  
[AttributeI1Type](#)  
[AttributeI2Type](#)  
[AttributeQPidType](#)  
[AttributeStrType](#)  
[AttributesType](#)  
[AttributeUserType](#)  
[AxisType](#)  
[BinaryDataType](#)  
[CoordinateSystemCoreType](#)  
[ElementReferenceFullType](#)  
[ElementReferenceType](#)  
[LatitudeLongitudeSweepType](#)  
[LineSegment2dType](#)  
[LineSegmentType](#)  
[OrientedLatitudeLongitudeSweepType](#)  
[PlaneType](#)  
[PlaneXType](#)  
[PointType](#)  
[PolyLineType](#)  
[QIFReferenceFullType](#)  
[QIFReferenceType](#)  
[QPIdFullReferenceType](#)  
[SweepType](#)  
[TransformMatrixType](#)  
[TransformRotationType](#)  
[UnitVectorType](#)  
[UserDataXMLType](#)  
[VectorType](#)

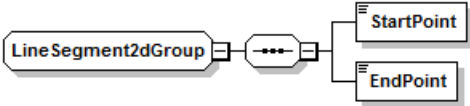
## Simple types

[D2Type](#)  
[D3Type](#)  
[D4Type](#)  
[DoublePositiveType](#)  
[I2Type](#)  
[ListBoolType](#)  
[ListDoubleType](#)  
[ListIntType](#)  
[ListNaturalType](#)  
[ListUnsignedByteType](#)  
[NaturalType](#)  
[ParameterRangeType](#)  
[Point2dSimpleType](#)  
[PointSimpleType](#)  
[QIFIdType](#)  
[QPIdReferenceType](#)  
[QPIdType](#)  
[UnitVector2dSimpleType](#)  
[UnitVectorSimpleType](#)  
[ValidityEnumType](#)  
[VectorSimpleType](#)


## Attr. groups

[AttrActualPoint](#)  
[AttrPoint](#)


group **LineSegment2dGroup**

diagram	
children	<a href="#">StartPoint</a> <a href="#">EndPoint</a>
used by	complexType <a href="#">LineSegment2dType</a>
annotation	documentation The LineSegmentGroup is a group which contains start point and an end point that define portion of 2d line.

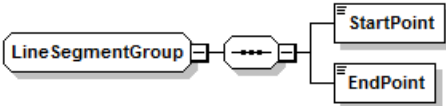
element **LineSegment2dGroup/StartPoint**

diagram	
type	<a href="#">Point2dSimpleType</a>
properties	content simple
facets	Kind Value Annotation length 2
annotation	documentation The StartPoint element is the beginning point of the 2d line segment.

element **LineSegment2dGroup/EndPoint**

diagram	
type	<a href="#">Point2dSimpleType</a>
properties	content simple
facets	Kind Value Annotation length 2
annotation	documentation The EndPoint element is the ending point of the 2d line segment.

group **LineSegmentGroup**


diagram	
children	<a href="#">StartPoint</a> <a href="#">EndPoint</a>
used by	complexType <a href="#">LineSegmentType</a>
annotation	documentation The LineSegmentGroup is a group which contains start point and an end point that define portion of line.

element **LineSegmentGroup/StartPoint**

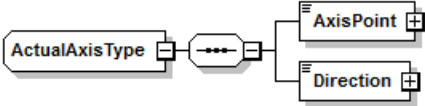
diagram	
---------	---

type	<a href="#">PointSimpleType</a>
properties	content simple
facets	Kind Value Annotation length 3
annotation	documentation The StartPoint element is the beginning point of the line segment.

#### element **LineSegmentGroup/EndPoint**

diagram	
type	<a href="#">PointSimpleType</a>
properties	content simple
facets	Kind Value Annotation length 3
annotation	documentation The EndPoint element is the ending point of the line segment.

#### complexType **ActualAxisType**

diagram	
children	<a href="#">AxisPoint</a> <a href="#">Direction</a>
annotation	documentation The ActualAxisType defines an actual feature axis.

element **ActualAxisType/AxisPoint**

diagram						
type	<a href="#">ActualPointType</a>					
properties	content complex					
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	<b>xs:token</b>				
	<a href="#">decimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">significantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">validity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">xDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">xSignificantFigures</a>	<b>xs:nonNegativeInteger</b>				

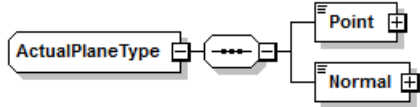
	<a href="#">xValidity</a> <a href="#">yDecimalPlaces</a> <a href="#">ySignificantFigures</a> <a href="#">yValidity</a> <a href="#">zDecimalPlaces</a> <a href="#">zSignificantFigures</a> <a href="#">zValidity</a> <a href="#">combinedUncertainty</a> <a href="#">meanError</a> <a href="#">xCombinedUncertainty</a> <a href="#">xMeanError</a> <a href="#">yCombinedUncertainty</a> <a href="#">yMeanError</a> <a href="#">zCombinedUncertainty</a> <a href="#">zMeanError</a>	<b><a href="#">ValidityEnumType</a></b> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <b><a href="#">ValidityEnumType</a></b> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <b><a href="#">ValidityEnumType</a></b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b>
annotation	documentation The AxisPoint element is the actual location of a point on the axis.	

element **ActualAxisType/Direction**

diagram						
type	<a href="#">ActualUnitVectorType</a>					
properties	content complex					
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	<b>xs:token</b>				
	<a href="#">decimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">significantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">validity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">xDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">xSignificantFigures</a>	<b>xs:nonNegativeInteger</b>				

	<a href="#">xValidity</a> <a href="#">yDecimalPlaces</a> <a href="#">ySignificantFigures</a> <a href="#">yValidity</a> <a href="#">zDecimalPlaces</a> <a href="#">zSignificantFigures</a> <a href="#">zValidity</a> <a href="#">combinedUncertainty</a> <a href="#">meanError</a> <a href="#">xCombinedUncertainty</a> <a href="#">xMeanError</a> <a href="#">yCombinedUncertainty</a> <a href="#">yMeanError</a> <a href="#">zCombinedUncertainty</a> <a href="#">zMeanError</a>	<a href="#">ValidityEnumType</a> xs:nonNegativeInteger xs:nonNegativeInteger <a href="#">ValidityEnumType</a> xs:nonNegativeInteger xs:nonNegativeInteger <a href="#">ValidityEnumType</a> xs:decimal xs:decimal xs:decimal xs:decimal xs:decimal xs:decimal xs:decimal xs:decimal xs:decimal
annotation	documentation The Direction element is the actual unit vector direction of the axis.	

### complexType **ActualPlaneType**

diagram		
children	<a href="#">Point</a> <a href="#">Normal</a>	
annotation	documentation The ActualPlaneType defines an actual plane by means of the plane unit normal vector and a point on the plane.	

element **ActualPlaneType/Point**

diagram						
type	<a href="#">ActualPointType</a>					
properties	content	complex				
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	<b>xs:token</b>				
	<a href="#">decimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">significantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">validity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">xDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">xSignificantFigures</a>	<b>xs:nonNegativeInteger</b>				



	<a href="#">xValidity</a> <a href="#">yDecimalPlaces</a> <a href="#">ySignificantFigures</a> <a href="#">yValidity</a> <a href="#">zDecimalPlaces</a> <a href="#">zSignificantFigures</a> <a href="#">zValidity</a> <a href="#">combinedUncertainty</a> <a href="#">meanError</a> <a href="#">xCombinedUncertainty</a> <a href="#">xMeanError</a> <a href="#">yCombinedUncertainty</a> <a href="#">yMeanError</a> <a href="#">zCombinedUncertainty</a> <a href="#">zMeanError</a>	<b><a href="#">ValidityEnumType</a></b> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <b><a href="#">ValidityEnumType</a></b> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <b><a href="#">ValidityEnumType</a></b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b>
annotation	documentation The Point element is an actual point on the plane.	

element **ActualPlaneType/Normal**

diagram						
type	<a href="#">ActualUnitVectorType</a>					
properties	content complex					
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	<b>xs:token</b>				
	<a href="#">decimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">significantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">validity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">xDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">xSignificantFigures</a>	<b>xs:nonNegativeInteger</b>				

	<a href="#">xValidity</a> <a href="#">yDecimalPlaces</a> <a href="#">ySignificantFigures</a> <a href="#">yValidity</a> <a href="#">zDecimalPlaces</a> <a href="#">zSignificantFigures</a> <a href="#">zValidity</a> <a href="#">combinedUncertainty</a> <a href="#">meanError</a> <a href="#">xCombinedUncertainty</a> <a href="#">xMeanError</a> <a href="#">yCombinedUncertainty</a> <a href="#">yMeanError</a> <a href="#">zCombinedUncertainty</a> <a href="#">zMeanError</a>	<b><a href="#">ValidityEnumType</a></b> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <b><a href="#">ValidityEnumType</a></b> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <b><a href="#">ValidityEnumType</a></b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b>
annotation	documentation The Normal element is the actual unit normal vector of the plane.	

complexType **ActualPointType**

diagram						
type	extension of <a href="#">PointType</a>					
properties	base <a href="#">PointType</a>					
used by	elements <a href="#">ActualAxisType/AxisPoint</a> <a href="#">ActualPlaneType/Point</a>					
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	<b>xs:token</b>				
	<a href="#">decimalPlaces</a>	<b>xs:nonNegativeInteger</b>				

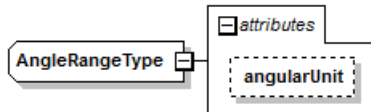
	<a href="#">significantFigures</a> <a href="#">validity</a> <a href="#">xDecimalPlaces</a> <a href="#">xSignificantFigures</a> <a href="#">xValidity</a> <a href="#">yDecimalPlaces</a> <a href="#">ySignificantFigures</a> <a href="#">yValidity</a> <a href="#">zDecimalPlaces</a> <a href="#">zSignificantFigures</a> <a href="#">zValidity</a> <a href="#">combinedUncertainty</a> <a href="#">meanError</a> <a href="#">xCombinedUncertainty</a> <a href="#">xMeanError</a> <a href="#">yCombinedUncertainty</a> <a href="#">yMeanError</a> <a href="#">zCombinedUncertainty</a> <a href="#">zMeanError</a>	<b>xs:nonNegativeInteger</b> <b><a href="#">ValidityEnumType</a></b> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <b><a href="#">ValidityEnumType</a></b> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <b><a href="#">ValidityEnumType</a></b> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <b><a href="#">ValidityEnumType</a></b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b> <b>xs:decimal</b>
annotation	documentation The ActualPointType defines an XYZ location in which the length units, accuracy, uncertainty and error of the coordinates may be specified.	

complexType **ActualUnitVectorType**

diagram						
type	extension of <a href="#">UnitVectorType</a>					
properties	base <a href="#">UnitVectorType</a>					
used by	elements <a href="#">ActualAxisType/Direction</a> <a href="#">ActualPlaneType/Normal</a>					
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	<b>xs:token</b>				
	<a href="#">decimalPlaces</a>	<b>xs:nonNegativeInteger</b>				

	<a href="#">significantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">validity</a> <b>ValidityEnumType</b> <a href="#">xDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">xSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">xValidity</a> <b>ValidityEnumType</b> <a href="#">yDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">ySignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">yValidity</a> <b>ValidityEnumType</b> <a href="#">zDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">zSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">zValidity</a> <b>ValidityEnumType</b> <a href="#">combinedUncertainty</a> <b>xs:decimal</b> <a href="#">meanError</a> <b>xs:decimal</b> <a href="#">xCombinedUncertainty</a> <b>xs:decimal</b> <a href="#">xMeanError</a> <b>xs:decimal</b> <a href="#">yCombinedUncertainty</a> <b>xs:decimal</b> <a href="#">yMeanError</a> <b>xs:decimal</b> <a href="#">zCombinedUncertainty</a> <b>xs:decimal</b> <a href="#">zMeanError</a> <b>xs:decimal</b>
annotation	documentation The ActualUnitVectorType is a unit vector in which the accuracy, uncertainty and error of the components may be specified. The linearUnit attribute included in AttrPoint must not be used since unit vectors do not have length units.

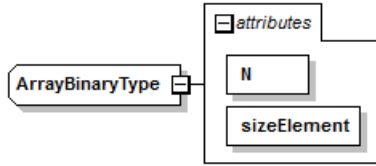
### complexType **AngleRangeType**

diagram						
type	extension of <a href="#">D2Type</a>					
properties	base <a href="#">D2Type</a>					
used by	elements <a href="#">SweepType/DomainAngle</a> <a href="#">LatitudeLongitudeSweepType/DomainLatitude</a> <a href="#">LatitudeLongitudeSweepType/DomainLongitude</a>					
facets	Kind	Value	Annotation			
	length	2				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">angularUnit</a>	<b>xs:token</b>				documentation The optional angularUnit attribute defines the UnitName for the AngleRangeType.
annotation	documentation The AngleRangeType defines an angle range.					

### attribute **AngleRangeType/@angularUnit**

type	<b>xs:token</b>
annotation	documentation The optional angularUnit attribute defines the UnitName for the AngleRangeType.

**complexType ArrayBinaryType**

diagram						
type	extension of <b>xs:base64Binary</b>					
properties	base <b>xs:base64Binary</b>					
attributes	Name <a href="#">N</a>	Type <b>xs:unsignedInt</b>	Use required	Default	Fixed	Annotation documentation The required N attribute shows how many elements are present in this array.
	<a href="#">sizeElement</a>	<b>xs:unsignedInt</b>	required			documentation The required sizeElement attribute shows the size (in bytes) of one element stored in the array. The total size of the binary array can be calculated as: N*sizeElement.
annotation	documentation The ArrayBinaryType represents an array of Base64-encoded binary elements. For base64Binary data the entire binary stream is encoded using the Base64 Alphabet in RFC 2045.					

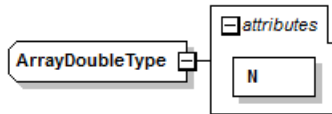
**attribute ArrayBinaryType/@N**

type	<b>xs:unsignedInt</b>
properties	use required
annotation	documentation The required N attribute shows how many elements are present in this array.

**attribute ArrayBinaryType/@sizeElement**

type	<b>xs:unsignedInt</b>
properties	use required
annotation	documentation The required sizeElement attribute shows the size (in bytes) of one element stored in the array. The total size of the binary array can be calculated as: N*sizeElement.

**complexType ArrayDoubleType**

diagram						
type	extension of <a href="#">ListDoubleType</a>					
properties	base <a href="#">ListDoubleType</a>					
attributes	Name <a href="#">N</a>	Type <b>xs:unsignedInt</b>	Use required	Default	Fixed	Annotation documentation The required N attribute shows how many objects are present in this array.
annotation	documentation					

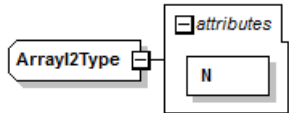


	The ArrayDoubleType is an array of double values.
--	---

**attribute ArrayDoubleType/@N**

type	<b>xs:unsignedInt</b>
properties	use required
annotation	documentation The required N attribute shows how many objects are present in this array.

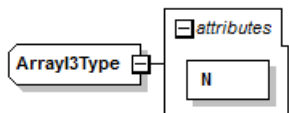
**complexType ArrayI2Type**

diagram						
type	extension of <a href="#">ListIntType</a>					
properties	base ListIntType					
attributes	Name <a href="#">N</a>	Type <b>xs:positiveInteger</b>	Use required	Default	Fixed	Annotation documentation The required N attribute gives the number of integer pairs represented in the array. The number of entries in the array must be 2*N.
annotation	documentation The ArrayI2Type is an array of integer numbers representing a specific number of ordered pairs of integers. Each successive group of two entries in the array represents the first and second components of a pair.					

**attribute ArrayI2Type/@N**

type	<b>xs:positiveInteger</b>
properties	use required
annotation	documentation The required N attribute gives the number of integer pairs represented in the array. The number of entries in the array must be 2*N.

**complexType ArrayI3Type**

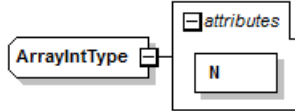
diagram						
type	extension of <a href="#">ListIntType</a>					
properties	base ListIntType					
attributes	Name <a href="#">N</a>	Type <b>xs:positiveInteger</b>	Use required	Default	Fixed	Annotation documentation The required N attribute gives the number of integer triplets represented in the array. The number of entries in the array must be 3*N.
annotation	documentation The ArrayI3Type is an array of integer numbers representing a specific number of ordered triplets of integers. Each					

	successive group of three entries in the array represents the first, second and third components of a triplet.
--	--

attribute **ArrayI3Type/@N**

type	<b>xs:positiveInteger</b>
properties	use required
annotation	documentation The required N attribute gives the number of integer triplets represented in the array. The number of entries in the array must be 3*N.

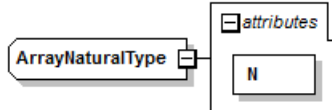
complexType **ArrayIntType**

diagram						
type	extension of <a href="#">ListIntType</a>					
properties	base ListIntType					
attributes	Name <a href="#">N</a>	Type <b>xs:unsignedInt</b>	Use required	Default	Fixed	Annotation documentation The required N attribute shows how many objects are present in this array.
annotation	documentation The ArrayIntType is an array of integer numbers.					

attribute **ArrayIntType/@N**

type	<b>xs:unsignedInt</b>
properties	use required
annotation	documentation The required N attribute shows how many objects are present in this array.

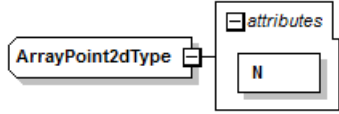
complexType **ArrayNaturalType**

diagram						
type	extension of <a href="#">ListNaturalType</a>					
properties	base ListNaturalType					
attributes	Name <a href="#">N</a>	Type <b>xs:unsignedInt</b>	Use required	Default	Fixed	Annotation documentation The required N attribute shows how many objects are present in this array.
annotation	documentation The ArrayNaturalType is an array of Natural numbers.					

attribute **ArrayNaturalType/@N**

type	<b>xs:unsignedInt</b>
properties	use required
annotation	documentation The required N attribute shows how many objects are present in this array.

complexType **ArrayPoint2dType**

diagram						
type	extension of <a href="#">ListDoubleType</a>					
properties	base ListDoubleType					
attributes	Name <a href="#">N</a>	Type <b>xs:positiveInteger</b>	Use required	Default	Fixed	Annotation documentation The required N attribute gives the number of points represented by the array. The number of entries in the array must be 2N.
annotation	documentation The ArrayPointType is an array of doubles representing a specific number of ordered 2D points. Each successive group of two entries in the array represents the X, and Y coordinates of a point.					

attribute **ArrayPoint2dType/@N**

type	<b>xs:positiveInteger</b>
properties	use required
annotation	documentation The required N attribute gives the number of points represented by the array. The number of entries in the array must be 2N.

complexType **ArrayPointType**

diagram						
type	extension of <a href="#">ListDoubleType</a>					
properties	base <a href="#">ListDoubleType</a>					
used by	complexType <a href="#">PolyLineType</a>					
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">N</a>	xs:positiveInteger	required			documentation The required N attribute gives the number of points represented by the array. The number of entries in the array must be 3N.
	<a href="#">linearUnit</a>	xs:token				
	<a href="#">decimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">significantFigures</a>	xs:nonNegativeInteger				
	<a href="#">validity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">xDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">xSignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">xValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">yDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">ySignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">yValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">zDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">zSignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">zValidity</a>	<a href="#">ValidityEnumType</a>				

annotation	documentation The ArrayPointType is an array of doubles representing a specific number of ordered 3D points. Each successive group of three entries in the array represents the X, Y, and Z coordinates of a point.
------------	--

**attribute ArrayPointType/@N**

type	<b>xs:positiveInteger</b>
properties	use required
annotation	documentation The required N attribute gives the number of points represented by the array. The number of entries in the array must be 3N.

**complexType ArrayReferenceFullType**

diagram						
children	<a href="#">Id</a>					
attributes	Name <a href="#">N</a>	Type <a href="#">NaturalType</a>	Use required	Default	Fixed	Annotation documentation The required N attribute shows how many Id elements are present in this array.
annotation	documentation The ArrayReferenceFullType is an array of object IDs.					

**attribute ArrayReferenceFullType/@N**

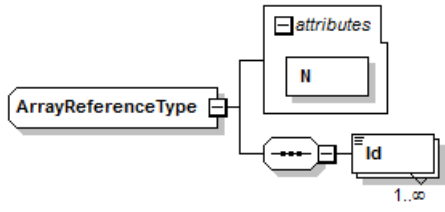
type	<a href="#">NaturalType</a>		
properties	use required		
facets	Kind minInclusive	Value 1	Annotation
annotation	documentation The required N attribute shows how many Id elements are present in this array.		

**element ArrayReferenceFullType/Id**

diagram						
type	<a href="#">QIFReferenceFullType</a>					
properties	minOcc maxOcc content	1 unbounded complex				

attributes	Name <a href="#">asmPath</a>	Type <a href="#">QIFIdType</a>	Use	Default	Fixed	Annotation documentation The optional asmPath attribute is an id which must be used for locating of the assembly path within the AsmPaths. The assembly path (instantiation chain) unambiguously identifies a model entity within an assembly.
annotation	documentation The Id element is reference to a QIF id that may include an AsmPath.					

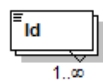
### complexType **ArrayReferenceType**

diagram						
children	<a href="#">Id</a>					
attributes	Name <a href="#">N</a>	Type <a href="#">NaturalType</a>	Use required	Default	Fixed	Annotation documentation The required N attribute shows how many Id elements are present in this array.
annotation	documentation The ArrayReferenceType is an array of object IDs.					

### attribute **ArrayReferenceType/@N**

type	<a href="#">NaturalType</a>		
properties	use required		
facets	Kind minInclusive	Value 1	Annotation
annotation	documentation The required N attribute shows how many Id elements are present in this array.		

### element **ArrayReferenceType/Id**

diagram							
type	<a href="#">QIFReferenceType</a>						
properties	<table><tr><td>minOcc</td><td>1</td></tr><tr><td>maxOcc</td><td>unbounded</td></tr><tr><td>content</td><td>complex</td></tr></table>	minOcc	1	maxOcc	unbounded	content	complex
minOcc	1						
maxOcc	unbounded						
content	complex						
annotation	<p>documentation</p> <p>Each Id element is the QIF id of an object.</p>						

complexType **ArrayUnitVectorType**

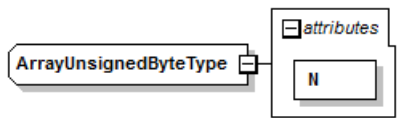
diagram						
type	extension of <a href="#">ListDoubleType</a>					
properties	base ListDoubleType					
attributes	<p>Name <a href="#">N</a></p> <p><a href="#">linearUnit</a></p> <p><a href="#">decimalPlaces</a></p> <p><a href="#">significantFigures</a></p> <p><a href="#">validity</a></p> <p><a href="#">xDecimalPlaces</a></p> <p><a href="#">xSignificantFigures</a></p> <p><a href="#">xValidity</a></p> <p><a href="#">yDecimalPlaces</a></p> <p><a href="#">ySignificantFigures</a></p> <p><a href="#">yValidity</a></p> <p><a href="#">zDecimalPlaces</a></p> <p><a href="#">zSignificantFigures</a></p> <p><a href="#">zValidity</a></p>	<p>Type <b>xs:positiveInteger</b></p> <p><b>xs:token</b></p> <p><b>xs:nonNegativeInteger</b></p> <p><b>xs:nonNegativeInteger</b></p> <p><b>ValidityEnumType</b></p> <p><b>xs:nonNegativeInteger</b></p> <p><b>xs:nonNegativeInteger</b></p> <p><b>ValidityEnumType</b></p> <p><b>xs:nonNegativeInteger</b></p> <p><b>xs:nonNegativeInteger</b></p> <p><b>ValidityEnumType</b></p> <p><b>xs:nonNegativeInteger</b></p> <p><b>xs:nonNegativeInteger</b></p> <p><b>ValidityEnumType</b></p>	<p>Use required</p>	<p>Default</p>	<p>Fixed</p>	<p>Annotation documentation The required N attribute gives the number of unit vectors represented by the array. The number of entries in the array must be 3N.</p>
annotation	<p>documentation The ArrayUnitVectorType is an array of doubles representing a specific number of ordered unit vectors. Each successive group of three entries in the array represents the X, Y, and Z components of a unit vector. The linearUnit attribute</p>					

	included in AttrPoint must not be used since unit vectors do not have length units.
--	---

**attribute ArrayUnitVectorType/@N**

type	<b>xs:positiveInteger</b>
properties	use required
annotation	documentation The required N attribute gives the number of unit vectors represented by the array. The number of entries in the array must be 3N.

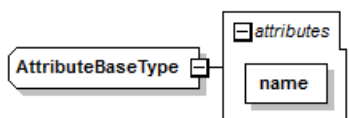
**complexType ArrayUnsignedByteType**

diagram						
type	extension of <a href="#">ListUnsignedByteType</a>					
properties	base ListUnsignedByteType					
attributes	Name <a href="#">N</a>	Type <b>xs:unsignedInt</b>	Use required	Default	Fixed	Annotation documentation The required N attribute shows how many objects are present in this array.
annotation	documentation The ArrayUnsignedByteType is an array of unsigned byte values.					

**attribute ArrayUnsignedByteType/@N**

type	<b>xs:unsignedInt</b>
properties	use required
annotation	documentation The required N attribute shows how many objects are present in this array.

**complexType AttributeBaseType**

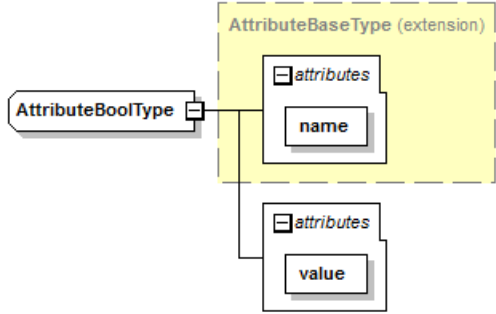
diagram						
properties	abstract true					
used by	element complexType	<b>Attribute</b> <a href="#">AttributeBoolType</a> <a href="#">AttributeD1Type</a> <a href="#">AttributeD3Type</a> <a href="#">AttributeI1Type</a> <a href="#">AttributeI2Type</a> <a href="#">AttributeQPidType</a> <a href="#">AttributeStrType</a> <a href="#">AttributeUserType</a>				
attributes	Name <a href="#">name</a>	Type <b>xs:string</b>	Use required	Default	Fixed	Annotation documentation The required name attribute is the name of the entity attribute. This name is a unique identifier of an attribute within the entity.
annotation	documentation The AttributeBaseType is the abstract base type for all entity attributes present in the CAD scene.					



attribute **AttributeBaseType/@name**

type	<b>xs:string</b>
properties	use required
annotation	documentation The required name attribute is the name of the entity attribute. This name is a unique identifier of an attribute within the entity.

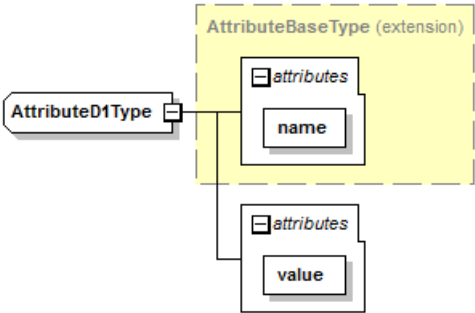
complexType **AttributeBoolType**

diagram						
type	extension of <a href="#">AttributeBaseType</a>					
properties	base <a href="#">AttributeBaseType</a>					
used by	element <b>AttributeBool</b>					
attributes	Name <a href="#">name</a>	Type <b>xs:string</b>	Use required	Default	Fixed	Annotation documentation The required name attribute is the name of the entity attribute. This name is a unique identifier of an attribute within the entity.
	<a href="#">value</a>	<b>xs:boolean</b>	required			documentation The required value attribute specifies the Boolean value of this entity attribute.
annotation	documentation The AttributeBoolType defines an entity attribute of Boolean type.					

attribute **AttributeBoolType/@value**

type	<b>xs:boolean</b>
properties	use required
annotation	documentation The required value attribute specifies the Boolean value of this entity attribute.

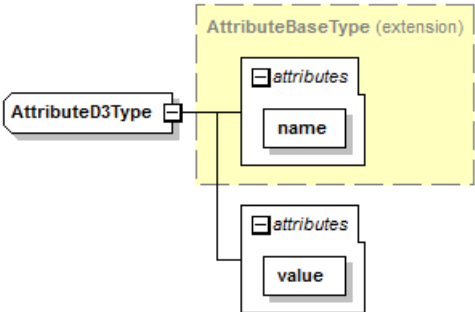
**complexType AttributeD1Type**

diagram						
type	extension of <a href="#">AttributeBaseType</a>					
properties	base <a href="#">AttributeBaseType</a>					
used by	element <b>AttributeD1</b>					
attributes	Name <a href="#">name</a>	Type <b>xs:string</b>	Use required	Default	Fixed	Annotation documentation The required name attribute is the name of the entity attribute. This name is a unique identifier of an attribute within the entity.
	<a href="#">value</a>	<b>xs:double</b>	required			documentation The required value attribute specifies the double value of this entity attribute.
annotation	documentation The AttributeD1Type defines an entity attribute of double type.					

**attribute AttributeD1Type/@value**

type	<b>xs:double</b>
properties	use required
annotation	documentation The required value attribute specifies the double value of this entity attribute.

**complexType AttributeD3Type**

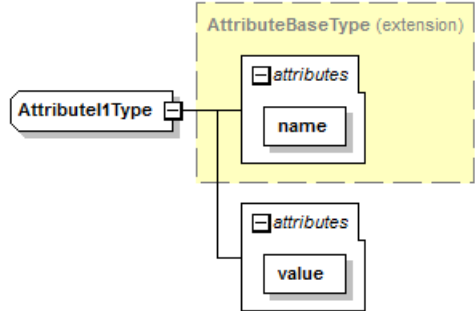
diagram						
type	extension of <a href="#">AttributeBaseType</a>					
properties	base <a href="#">AttributeBaseType</a>					
used by	element <b>AttributeD3</b>					

attributes	Name <a href="#">name</a>	Type <b>xs:string</b>	Use required	Default	Fixed	Annotation documentation The required name attribute is the name of the entity attribute. This name is a unique identifier of an attribute within the entity.
	<a href="#">value</a>	<a href="#">D3Type</a>	required			documentation The required value attribute specifies the integer triple value of this entity attribute.
annotation	documentation The AttributeD3Type defines an entity attribute of 'three doubles' type.					

attribute **AttributeD3Type/@value**

type	<a href="#">D3Type</a>
properties	use required
facets	Kind Value Annotation length 3
annotation	documentation The required value attribute specifies the integer triple value of this entity attribute.

complexType **Attributel1Type**

diagram						
type	extension of <a href="#">AttributeBaseType</a>					
properties	base <a href="#">AttributeBaseType</a>					
used by	element <b>Attributel1</b>					
attributes	Name <a href="#">name</a>	Type <b>xs:string</b>	Use required	Default	Fixed	Annotation documentation The required name attribute is the name of the entity attribute. This name is a unique identifier of an attribute within the entity. documentation The required value attribute specifies the integer value of this entity attribute.
	<a href="#">value</a>	<b>xs:integer</b>	required			
annotation	documentation The Attributel1Type defines an entity attribute of integer type.					

attribute **Attributel1Type/@value**

type	<b>xs:integer</b>
------	-------------------

properties	use required
annotation	documentation The required value attribute specifies the integer value of this entity attribute.

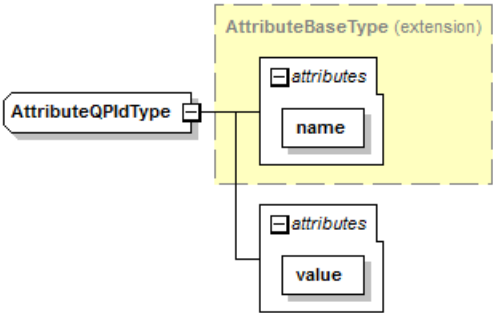
**complexType AttributeI2Type**

diagram						
type	extension of <a href="#">AttributeBaseType</a>					
properties	base AttributeBaseType					
used by	element <b>Attributel2</b>					
attributes	Name <a href="#">name</a>  <a href="#">value</a>	Type <b>xs:string</b>  <a href="#">I2Type</a>	Use required  required	Default    	Fixed    	Annotation documentation The required name attribute is the name of the entity attribute. This name is a unique identifier of an attribute within the entity. documentation The required value attribute specifies the integer pair value of this entity attribute.
annotation	documentation The AttributeI2Type defines an entity attribute of 'pair of integers' type.					

**attribute AttributeI2Type/@value**

type	<a href="#">I2Type</a>		
properties	use required		
facets	Kind length	Value 2	Annotation
annotation	documentation The required value attribute specifies the integer pair value of this entity attribute.		

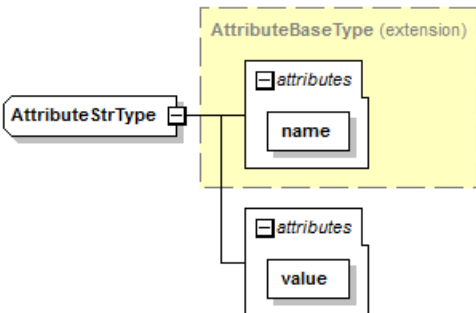
complexType **AttributeQPidType**

diagram						
type	extension of <a href="#">AttributeBaseType</a>					
properties	base <a href="#">AttributeBaseType</a>					
used by	element <b>AttributeQPId</b>					
attributes	Name <a href="#">name</a>	Type <b>xs:string</b>	Use required	Default	Fixed	Annotation documentation The required name attribute is the name of the entity attribute. This name is a unique identifier of an attribute within the entity.
	<a href="#">value</a>	<a href="#">QPIdType</a>	required			documentation The required value attribute specifies the QPid value of this entity attribute.
annotation	documentation The AttributeQPIdType defines an entity attribute of QPid type.					

attribute **AttributeQPIdType/@value**

type	<a href="#">QPIdType</a>
properties	use required
annotation	documentation The required value attribute specifies the QPid value of this entity attribute.

complexType **AttributeStrType**

diagram						
type	extension of <a href="#">AttributeBaseType</a>					
properties	base <a href="#">AttributeBaseType</a>					
used by	element <b>AttributeStr</b>					

attributes	Name <a href="#">name</a>	Type <b>xs:string</b>	Use required	Default	Fixed	Annotation documentation The required name attribute is the name of the entity attribute. This name is a unique identifier of an attribute within the entity.
	<a href="#">value</a>	<b>xs:string</b>	required			documentation The required value attribute specifies the string value of this entity attribute.
annotation	documentation The AttributeStrType defines an entity attribute of string type.					

### attribute **AttributeStrType/@value**

type	<b>xs:string</b>
properties	use required
annotation	documentation The required value attribute specifies the string value of this entity attribute.

### complexType **AttributesType**

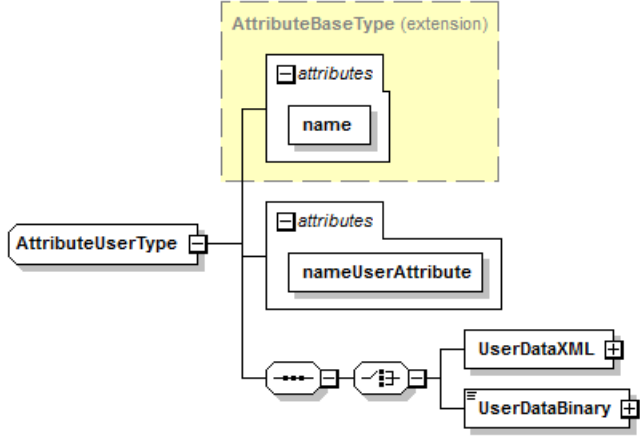
diagram	<pre> classDiagram     class AttributesType {         +Attribute 1..∞         +AttributeBool 1..∞         +AttributeD1 1..∞         +AttributeD3 1..∞         +AttributeI1 1..∞         +AttributeI2 1..∞         +AttributeQPid 1..∞         +AttributeStr 1..∞         +AttributeUser 1..∞     }     class attributes {         +N     } </pre>					
children	<b>Attribute</b>					
used by	element	<b>Attributes</b>				

attributes	Name <a href="#">N</a>	Type <a href="#">NaturalType</a>	Use required	Default	Fixed	Annotation documentation The required N attribute is the number of objects in this list.
annotation	documentation The AttributesType defines a list of entity attributes.					

attribute **AttributesType/@N**

type	<a href="#">NaturalType</a>		
properties	use required		
facets	Kind minInclusive	Value 1	Annotation
annotation	documentation The required N attribute is the number of objects in this list.		

complexType **AttributeUserType**

diagram						
type	extension of <a href="#">AttributeBaseType</a>					
properties	base <a href="#">AttributeBaseType</a>					
children	<a href="#">UserDataXML</a> <a href="#">UserDataBinary</a>					
used by	element <b>AttributeUser</b>					
attributes	Name <a href="#">name</a>	Type <b>xs:string</b>	Use required	Default	Fixed	Annotation documentation The required name attribute is the name of the entity attribute. This name is a unique identifier of an attribute within the entity.
	<a href="#">nameUserAttribute</a>	<b>xs:string</b>	required			documentation The required nameUserAttribute attribute is the name of user-defined attribute type. The structure of the user-defined attribute can be identified by this name.
annotation	documentation The AttributeUserType defines a user-defined entity attribute which contains a binary array or any user-defined structured XML data.					

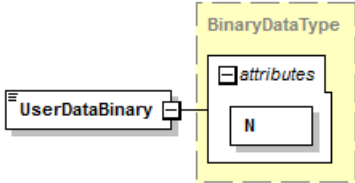
attribute **AttributeUserType/@nameUserAttribute**

type	<b>xs:string</b>
properties	use required
annotation	documentation The required nameUserAttribute attribute is the name of user-defined attribute type. The structure of the user-defined attribute can be identified by this name.

element **AttributeUserType/UserDataXML**

diagram	
type	<a href="#">UserDataXMLType</a>
properties	content complex
annotation	documentation The UserDataXML element is the user-defined data represented as an XML structure.

element **AttributeUserType/UserDataBinary**

diagram						
type	<a href="#">BinaryDataType</a>					
properties	content complex					
attributes	Name <a href="#">N</a>	Type <b>xs:unsignedInt</b>	Use required	Default	Fixed	Annotation documentation The required N attribute shows the size of the binary block in bytes.
annotation	documentation The UserDataBinary element is a binary block of user data.					

complexType **AxisType**

diagram	
children	<a href="#">AxisPoint</a> <a href="#">Direction</a>
annotation	documentation The AxisType defines a feature axis.



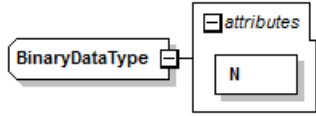
element **AxisType/AxisPoint**

diagram						
type	<a href="#">PointType</a>					
properties	content	complex				
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	xs:token				
	<a href="#">decimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">significantFigures</a>	xs:nonNegativeInteger				
	<a href="#">validity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">xDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">xSignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">xValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">yDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">ySignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">yValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">zDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">zSignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">zValidity</a>	<a href="#">ValidityEnumType</a>				
annotation	documentation The AxisPoint element is the location of a point on the axis.					

element **AxisType/Direction**

diagram						
type	<a href="#">UnitVectorType</a>					
properties	content	complex				
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	xs:token				
	<a href="#">decimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">significantFigures</a>	xs:nonNegativeInteger				
	<a href="#">validity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">xDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">xSignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">xValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">yDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">ySignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">yValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">zDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">zSignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">zValidity</a>	<a href="#">ValidityEnumType</a>				
annotation	documentation The Direction element is the unit vector direction of the axis.					

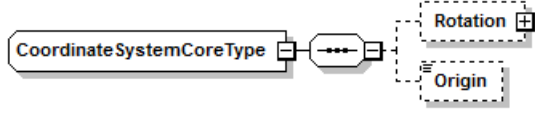
**complexType BinaryDataType**

diagram						
type	extension of <b>xs:base64Binary</b>					
properties	base <b>xs:base64Binary</b>					
used by	element <a href="#">AttributeUserType/UserDataBinary</a>					
attributes	Name <a href="#">N</a>	Type <b>xs:unsignedInt</b>	Use required	Default	Fixed	Annotation documentation The required N attribute shows the size of the binary block in bytes.
annotation	documentation The BinaryDataType represents a Base64-encoded arbitrary binary data block. For base64Binary data the entire binary stream is encoded using the Base64 Alphabet in RFC 2045.					

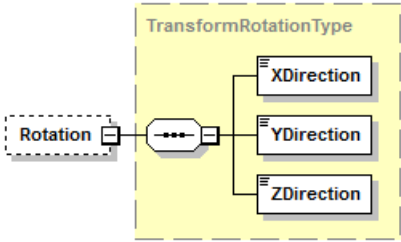
**attribute BinaryDataType/@N**

type	<b>xs:unsignedInt</b>
properties	use required
annotation	documentation The required N attribute shows the size of the binary block in bytes.

**complexType CoordinateSystemCoreType**


diagram						
children	<a href="#">Rotation</a> <a href="#">Origin</a>					
used by	complexType <a href="#">TransformMatrixType</a>					
annotation	documentation The CoordinateSystemCoreType defines the mathematical core of a Cartesian right-handed 3D coordinate system. It is also the base of the TransformMatrixType					

**element CoordinateSystemCoreType/Rotation**


diagram	
type	<a href="#">TransformRotationType</a>
properties	minOcc 0 maxOcc 1 content complex
children	<a href="#">XDirection</a> <a href="#">YDirection</a> <a href="#">ZDirection</a>

annotation	documentation The Rotation element element is an orthogonal basis of the Cartesian coordinate system.
------------	--

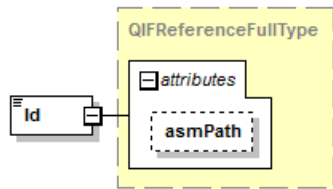
### element **CoordinateSystemCoreType/Origin**

diagram	
type	<a href="#">PointSimpleType</a>
properties	minOcc 0 maxOcc 1 content simple
facets	Kind Value Annotation length 3
annotation	documentation The Origin element is an origin of the coordinate system.

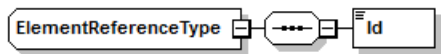
### complexType **ElementReferenceFullType**

diagram	
children	<a href="#">Id</a>
annotation	documentation The ElementReferenceFullType is an entity ID.


### element **ElementReferenceFullType/Id**

diagram						
type	<a href="#">QIFReferenceFullType</a>					
properties	content    complex					
attributes	Name <a href="#">asmPath</a>	Type <a href="#">QIFIdType</a>	Use	Default	Fixed	Annotation documentation The optional asmPath attribute is an id which must be used for locating of the assembly path within the AsmPaths. The assembly path (instantiation chain) unambiguously identifies a model entity within an assembly.
annotation	documentation The Id element is a reference to a QIF id that may include an AsmPath.					

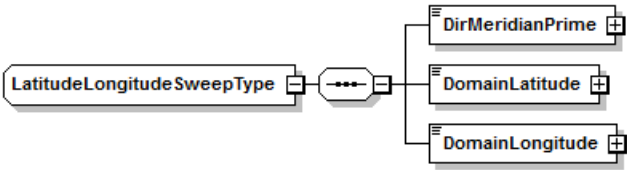
### complexType **ElementReferenceType**

diagram	
children	<a href="#">Id</a>
annotation	documentation The ElementReferenceType is an entity ID.

element **ElementReferenceType/Id**

diagram	
type	<a href="#">QIFReferenceType</a>
properties	content    complex
annotation	documentation The Id element is a reference to a QIF id that may not include an AsmPath.

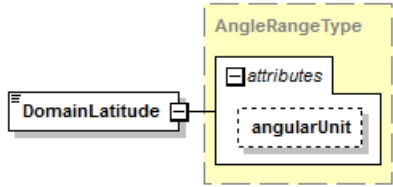
complexType **LatitudeLongitudeSweepType**

diagram	
children	<a href="#">DirMeridianPrime</a> <a href="#">DomainLatitude</a> <a href="#">DomainLongitude</a>
used by	complexType <a href="#">OrientedLatitudeLongitudeSweepType</a>
annotation	documentation The LatitudeLongitudeSweepType defines an angular sweep in two directions analogous to terrestrial latitude and longitude. For a sphere, latitude angles are +90 degrees at the north pole to -90 degrees at the south pole with 0 degrees at the equator. For a torus, latitude angles are 0 degrees at the equator farthest from the feature center and 180 or -180 degrees at the equator closest to the feature center, and the positive direction is towards the north pole vector from the equator farthest from the feature center. The positive direction in the longitude direction is counterclockwise (east) as viewed from the direction in which the north pole vector points. To use the LatitudeLongitudeSweepType, a north pole must be identified. For a torus, that is the north pole of the torus. For a sphere, the derived type OrientedLatitudeLongitudeSweepType must be used; it includes a north pole.

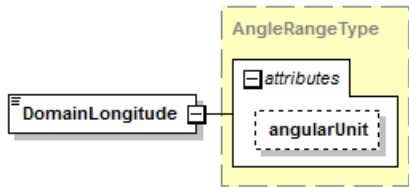
element **LatitudeLongitudeSweepType/DirMeridianPrime**

diagram						
type	<a href="#">UnitVectorType</a>					
properties	content	complex				
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	<b>xs:token</b>				
	<a href="#">decimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">significantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">validity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">xDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">xSignificantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">xValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">yDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">ySignificantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">yValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">zDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">zSignificantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">zValidity</a>	<a href="#">ValidityEnumType</a>				
annotation	documentation The DirMeridianPrime element is the direction of the prime meridian vector. The longitude is 0 on the PrimeMeridianVector. This vector must be perpendicular to the north pole vector.					

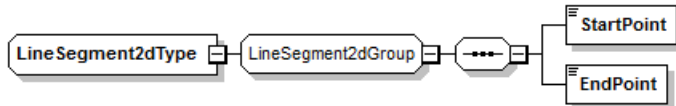
element **LatitudeLongitudeSweepType/DomainLatitude**

diagram						
type	<a href="#">AngleRangeType</a>					
properties	content complex					
facets	Kind	Value	Annotation			
	length	2				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">angularUnit</a>	<b>xs:token</b>				documentation The optional angularUnit attribute defines the UnitName for the AngleRangeType.
annotation	documentation The DomainLatitude element is the latitude domain. The unit type is angular unit. For a sphere, the latitude end angle must be greater than the latitude start angle. For a torus, either angle may be greater, but the sweep is always in the positive direction.					

element **LatitudeLongitudeSweepType/DomainLongitude**

diagram						
type	<a href="#">AngleRangeType</a>					
properties	content complex					
facets	Kind	Value	Annotation			
	length	2				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">angularUnit</a>	<b>xs:token</b>				documentation The optional angularUnit attribute defines the UnitName for the AngleRangeType.
annotation	documentation The DomainLongitude element is the longitude domain. The unit type is angular unit. Regardless of the values of the DomainLongitude, the longitude sweep is in the positive direction.					

complexType **LineSegment2dType**

diagram						
children	<a href="#">StartPoint</a> <a href="#">EndPoint</a>					
annotation	documentation The LineSegment2dType defines a portion of a 2d line defined and bounded by a start point and an end point.					

## complexType LineSegmentType

diagram						
children	<a href="#">StartPoint</a> <a href="#">EndPoint</a>					
attributes	Name <a href="#">linearUnit</a> <a href="#">decimalPlaces</a> <a href="#">significantFigures</a> <a href="#">validity</a> <a href="#">xDecimalPlaces</a> <a href="#">xSignificantFigures</a> <a href="#">xValidity</a> <a href="#">yDecimalPlaces</a> <a href="#">ySignificantFigures</a> <a href="#">yValidity</a> <a href="#">zDecimalPlaces</a> <a href="#">zSignificantFigures</a> <a href="#">zValidity</a>	Type <b>xs:token</b> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <a href="#">ValidityEnumType</a> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <a href="#">ValidityEnumType</a> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <a href="#">ValidityEnumType</a> <b>xs:nonNegativeInteger</b> <b>xs:nonNegativeInteger</b> <a href="#">ValidityEnumType</a>	Use	Default	Fixed	Annotation
annotation	documentation The LineSegmentType defines a portion of a line defined and bounded by a start point and an end point.					



complexType **OrientedLatitudeLongitudeSweepType**

diagram	<p>The diagram shows a dashed box labeled "LatitudeLongitudeSweepType (extension)". Inside this box are three elements: "DirMeridianPrime", "DomainLatitude", and "DomainLongitude". Outside the box, connected to the extension line, is the element "DirNorthPole". The base type "LatitudeLongitudeSweepType" is represented by a box with a dashed line extending from it to the extension box.</p>
type	extension of <a href="#">LatitudeLongitudeSweepType</a>
properties	base <code>LatitudeLongitudeSweepType</code>
children	<a href="#">DirMeridianPrime</a> <a href="#">DomainLatitude</a> <a href="#">DomainLongitude</a> <a href="#">DirNorthPole</a>
annotation	documentation The OrientedLatitudeLongitudeSweepType defines a LatitudeLongitudeSweepType with a north pole added.

element **OrientedLatitudeLongitudeSweepType/DirNorthPole**

diagram	<p>The diagram illustrates a UML class structure. A class named <b>DirNorthPole</b> is shown on the left, connected by a solid line to a larger class named <b>UnitVectorType</b> on the right. The <b>UnitVectorType</b> class is highlighted with a yellow background and a dashed border. Inside the <b>UnitVectorType</b> class, there is a compartment labeled <b>attributes</b> which contains a list of attributes: <b>linearUnit</b>, <b>decimalPlaces</b>, <b>significantFigures</b>, <b>validity</b>, <b>xDecimalPlaces</b>, <b>xSignificantFigures</b>, <b>xValidity</b>, <b>yDecimalPlaces</b>, <b>ySignificantFigures</b>, <b>yValidity</b>, <b>zDecimalPlaces</b>, <b>zSignificantFigures</b>, and <b>zValidity</b>. Each attribute is represented by a dashed box.</p>												
type	<a href="#">UnitVectorType</a>												
properties	content    complex												
facets	<table><tr><td>Kind</td><td>Value</td><td>Annotation</td></tr><tr><td>length</td><td>3</td><td></td></tr></table>	Kind	Value	Annotation	length	3							
Kind	Value	Annotation											
length	3												
attributes	<table><tr><th>Name</th><th>Type</th><th>Use</th><th>Default</th><th>Fixed</th><th>Annotation</th></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>	Name	Type	Use	Default	Fixed	Annotation						
Name	Type	Use	Default	Fixed	Annotation								

	<a href="#">linearUnit</a> <b>xs:token</b> <a href="#">decimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">significantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">validity</a> <b><a href="#">ValidityEnumType</a></b> <a href="#">xDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">xSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">xValidity</a> <b><a href="#">ValidityEnumType</a></b> <a href="#">yDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">ySignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">yValidity</a> <b><a href="#">ValidityEnumType</a></b> <a href="#">zDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">zSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">zValidity</a> <b><a href="#">ValidityEnumType</a></b>
annotation	documentation The DirNorthPole element is the direction of the north pole vector. For a torus this vector must be parallel or anti-parallel to the feature axis.

### complexType **PlaneType**

diagram	
children	<a href="#">Point</a> <a href="#">Normal</a>
used by	complexType <a href="#">PlaneXType</a>
annotation	documentation The PlaneType defines a plane by means of the plane unit normal vector and a point on the plane.

element **PlaneType/Point**

diagram						
type	<a href="#"><b>PointType</b></a>					
properties	content	complex				
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	<b>xs:token</b>				
	<a href="#">decimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">significantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">validity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">xDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">xSignificantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">xValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">yDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">ySignificantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">yValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">zDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">zSignificantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">zValidity</a>	<a href="#">ValidityEnumType</a>				
annotation	documentation The Point element is a point on the plane.					

element **PlaneType/Normal**

diagram						
type	<a href="#">UnitVectorType</a>					
properties	content	complex				
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	<b>xs:token</b>				
	<a href="#">decimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">significantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">validity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">xDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">xSignificantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">xValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">yDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">ySignificantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">yValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">zDecimalPlaces</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">zSignificantFigures</a>	<b>xs:nonNegativeInteger</b>				
	<a href="#">zValidity</a>	<a href="#">ValidityEnumType</a>				
annotation	documentation The Normal element is the unit normal vector of the plane.					

complexType **PlaneXType**

diagram	
type	extension of <a href="#">PlaneType</a>
properties	base <a href="#">PlaneType</a>
children	<a href="#">Point</a> <a href="#">Normal</a> <a href="#">Direction</a>
annotation	documentation The PlaneXType defines a plane with a given x-direction.

element **PlaneXType/Direction**

diagram	<pre>graph LR     Direction[Direction] --- UnitVectorType[UnitVectorType]     subgraph UnitVectorType         linearUnit[linearUnit]         decimalPlaces[decimalPlaces]         significantFigures[significantFigures]         validity[validity]         xDecimalPlaces[xDecimalPlaces]         xSignificantFigures[xSignificantFigures]         xValidity[xValidity]         yDecimalPlaces[yDecimalPlaces]         ySignificantFigures[ySignificantFigures]         yValidity[yValidity]         zDecimalPlaces[zDecimalPlaces]         zSignificantFigures[zSignificantFigures]         zValidity[zValidity]     end</pre>																		
type	<a href="#">UnitVectorType</a>																		
properties	content complex																		
facets	<table><tr><td>Kind</td><td>Value</td><td>Annotation</td></tr><tr><td>length</td><td>3</td><td></td></tr></table>	Kind	Value	Annotation	length	3													
Kind	Value	Annotation																	
length	3																		
attributes	<table><tr><td>Name</td><td>Type</td><td>Use</td><td>Default</td><td>Fixed</td><td>Annotation</td></tr><tr><td><a href="#">linearUnit</a></td><td>xs:token</td><td></td><td></td><td></td><td></td></tr><tr><td><a href="#">decimalPlaces</a></td><td>xs:nonNegativeInteger</td><td></td><td></td><td></td><td></td></tr></table>	Name	Type	Use	Default	Fixed	Annotation	<a href="#">linearUnit</a>	xs:token					<a href="#">decimalPlaces</a>	xs:nonNegativeInteger				
Name	Type	Use	Default	Fixed	Annotation														
<a href="#">linearUnit</a>	xs:token																		
<a href="#">decimalPlaces</a>	xs:nonNegativeInteger																		

	<a href="#">significantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">validity</a> <b>ValidityEnumType</b> <a href="#">xDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">xSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">xValidity</a> <b>ValidityEnumType</b> <a href="#">yDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">ySignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">yValidity</a> <b>ValidityEnumType</b> <a href="#">zDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">zSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">zValidity</a> <b>ValidityEnumType</b>
annotation	documentation The Direction element gives the direction of the positive X axis.

### complexType **PointType**

diagram						
type	extension of <a href="#">PointSimpleType</a>					
properties	base    PointSimpleType					
used by	elements <a href="#">AxisType/AxisPoint</a> <a href="#">PlaneType/Point</a> complexType <a href="#">ActualPointType</a>					
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	<b>xs:token</b>				
	<a href="#">decimalPlaces</a>	<b>xs:nonNegativeInteger</b>				

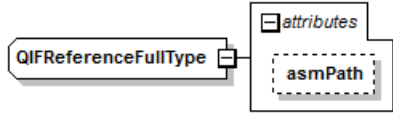
	<a href="#">significantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">validity</a> <b>ValidityEnumType</b> <a href="#">xDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">xSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">xValidity</a> <b>ValidityEnumType</b> <a href="#">yDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">ySignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">yValidity</a> <b>ValidityEnumType</b> <a href="#">zDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">zSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">zValidity</a> <b>ValidityEnumType</b>
annotation	documentation The PointType defines an XYZ location in which the length units and accuracy of the coordinates may be specified.

### complexType PolyLineType

diagram						
type	extension of <a href="#">ArrayPointType</a>					
properties	base    ArrayPointType					
attributes	Name <a href="#">N</a>	Type <b>xs:positiveInteger</b>	Use required	Default	Fixed	Annotation documentation The required N attribute gives the number of

	<p>points represented by the array. The number of entries in the array must be 3N.</p> <p> <a href="#">linearUnit</a>      <b>xs:token</b>  <a href="#">decimalPlaces</a>    <b>xs:nonNegativeInteger</b>  <a href="#">significantFigures</a>   <b>xs:nonNegativeInteger</b>  <a href="#">validity</a>            <b><a href="#">ValidityEnumType</a></b>  <a href="#">xDecimalPlaces</a>    <b>xs:nonNegativeInteger</b>  <a href="#">xSignificantFigures</a> <b>xs:nonNegativeInteger</b>  <a href="#">xValidity</a>            <b><a href="#">ValidityEnumType</a></b>  <a href="#">yDecimalPlaces</a>    <b>xs:nonNegativeInteger</b>  <a href="#">ySignificantFigures</a> <b>xs:nonNegativeInteger</b>  <a href="#">yValidity</a>            <b><a href="#">ValidityEnumType</a></b>  <a href="#">zDecimalPlaces</a>    <b>xs:nonNegativeInteger</b>  <a href="#">zSignificantFigures</a> <b>xs:nonNegativeInteger</b>  <a href="#">zValidity</a>            <b><a href="#">ValidityEnumType</a></b> </p>
annotation	<p>documentation</p> <p>The PolyLineType defines a polyline as an ordered list of points. The polyline is a continuous path consisting of line segments from point 1 to point 2, point 2 to point 3, etc. A polyline is not necessarily closed. To make a closed polyline, the last point should be the same as the first.</p>

### complexType QIFReferenceFullType

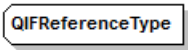
diagram						
type	extension of <a href="#">QIFReferenceType</a>					
properties	base    QIFReferenceType					
used by	elements <a href="#">ElementReferenceFullType/Id</a> <a href="#">ArrayReferenceFullType/Id</a>					
attributes	Name <a href="#">asmPath</a>	Type <a href="#">QIFIdType</a>	Use	Default	Fixed	Annotation documentation The optional asmPath attribute is an id which must be used for locating of the assembly path within the AsmPaths. The assembly path (instantiation chain) unambiguously identifies a model entity within an assembly.
annotation	<p>documentation</p> <p>The QIFReferenceFullType defines a 'full' reference which is used for locating of an entity within an assembly.</p>					



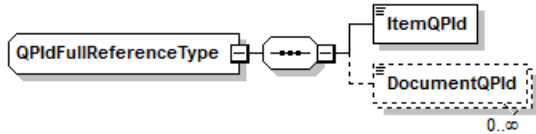
attribute **QIFReferenceFullType/@asmPath**

type	<a href="#">QIFIdType</a>
annotation	documentation The optional asmPath attribute is an id which must be used for locating of the assembly path within the AsmPaths. The assembly path (instantiation chain) unambiguously identifies a model entity within an assembly.

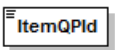
complexType **QIFReferenceType**

diagram	
type	extension of <a href="#">QIFIdType</a>
properties	base <a href="#">QIFIdType</a>
used by	elements <a href="#">ElementReferenceType/id</a> <a href="#">ArrayReferenceType/id</a> complexType <a href="#">QIFReferenceFullType</a>
annotation	documentation The QIFReferenceType defines an unsigned integer with no leading zeros. All references to objects in all QIF schema files are of QIFReferenceType and refer to the id attribute of the referenced object. In order to ensure that each reference refers to an object of the correct type, each reference must be covered by an appropriate key/keyref constraint.

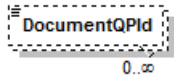
complexType **QPIdFullReferenceType**

diagram	
children	<a href="#">ItemQPId</a> <a href="#">DocumentQPId</a>
annotation	documentation The QPIdFullReferenceType identifies an item by using QPIDs. The item with the specified ItemQPId may exist in one or more QIF documents.

element **QPIdFullReferenceType/ItemQPId**

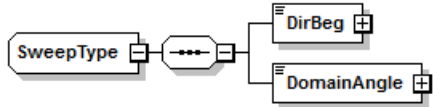
diagram	
type	<a href="#">QPIdReferenceType</a>
properties	content simple
annotation	documentation The ItemQPId is the QPId of an item.

element **QPIdFullReferenceType/DocumentQPId**

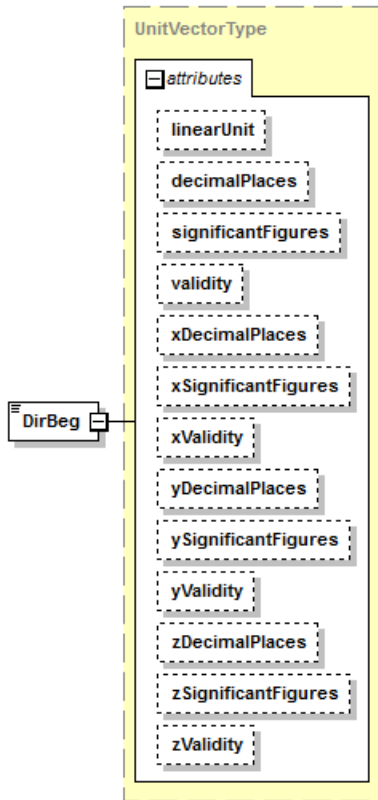
diagram	
type	<a href="#">QPIdReferenceType</a>
properties	minOcc 0 maxOcc unbounded content simple
annotation	documentation

	Each optional DocumentQPId is the QPId of a QIFDocument in which the item is found.
--	---

### complexType **SweepType**

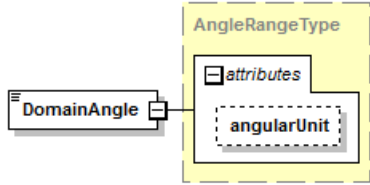
diagram	
children	<a href="#">DirBeg</a> <a href="#">DomainAngle</a>
annotation	documentation The SweepType defines a partial feature using a vector and an angle.

### element **SweepType/DirBeg**

diagram						
type	<a href="#">UnitVectorType</a>					
properties	content	complex				
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	xs:token				
	<a href="#">decimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">significantFigures</a>	xs:nonNegativeInteger				
	<a href="#">validity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">xDecimalPlaces</a>	xs:nonNegativeInteger				

	<a href="#">xSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">xValidity</a> <b><a href="#">ValidityEnumType</a></b> <a href="#">yDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">ySignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">yValidity</a> <b><a href="#">ValidityEnumType</a></b> <a href="#">zDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">zSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">zValidity</a> <b><a href="#">ValidityEnumType</a></b>
annotation	documentation The DirBeg element is a unit vector representing the beginning extent of the feature.

### element **SweepType/DomainAngle**

diagram						
type	<b><a href="#">AngleRangeType</a></b>					
properties	content	complex				
facets	Kind	Value	Annotation			
	length	2				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">angularUnit</a>	<b>xs:token</b>				documentation The optional angularUnit attribute defines the UnitName for the AngleRangeType.
annotation	documentation The DomainAngle element is the angle limits in angular units from the start vector to the start extent of the feature.					

complexType **TransformMatrixType**

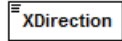
diagram						
type	extension of <a href="#">CoordinateSystemCoreType</a>					
properties	base <a href="#">CoordinateSystemCoreType</a>					
children	<a href="#">Rotation</a> <a href="#">Origin</a>					
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	xs:token				
	<a href="#">decimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">significantFigures</a>	xs:nonNegativeInteger				
	<a href="#">validity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">xDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">xSignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">xValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">yDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">ySignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">yValidity</a>	<a href="#">ValidityEnumType</a>				
	<a href="#">zDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">zSignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">zValidity</a>	<a href="#">ValidityEnumType</a>				

annotation	<p>documentation</p> <p>The TransformMatrixType defines a three dimensional transformation that may include rotation and translation, but not scaling. The vectors of the Rotation are unit vectors.</p> <p>documentation</p> <p>For any point, if:</p> <ol style="list-style-type: none"> <li>1. The coordinates of the point in the "before" coordinate system are x, y, and z.</li> <li>2. The coordinates of the point in the "after" coordinate system are X, Y, and Z.</li> <li>3. The components of the XDirection are Xi, Xj, and Xk.</li> <li>4. The components of the YDirection are Yi, Yj, and Yk.</li> <li>5. The components of the ZDirection are Zi, Zj, and Zk.</li> <li>6. The Cartesian coordinates of the Origin are Ox, Oy, and Oz.</li> </ol> <p>Then the following transformation equations hold.</p> $X = (Xi)x + (Yi)y + (Zi)z + Ox$ $Y = (Xj)x + (Yj)y + (Zj)z + Oy$ $Z = (Xk)x + (Yk)y + (Zk)z + Oz$
------------	--

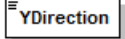
### complexType TransformRotationType

diagram	
children	<a href="#">XDirection</a> <a href="#">YDirection</a> <a href="#">ZDirection</a>
used by	element <a href="#">CoordinateSystemCoreType/Rotation</a>

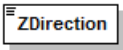
### element TransformRotationType/XDirection

diagram							
type	<a href="#">UnitVectorSimpleType</a>						
properties	content    simple						
facets	<table><tr><td>Kind</td><td>Value</td><td>Annotation</td></tr><tr><td>length</td><td>3</td><td></td></tr></table>	Kind	Value	Annotation	length	3	
Kind	Value	Annotation					
length	3						
annotation	<p>documentation</p> <p>The XDirection element is a unit vector that specifies the x rotation. When the origin is (0,0,0), the components of the X direction are the coordinates in the 'after' coordinate system of the point (1,0,0) in the 'before' system.</p>						

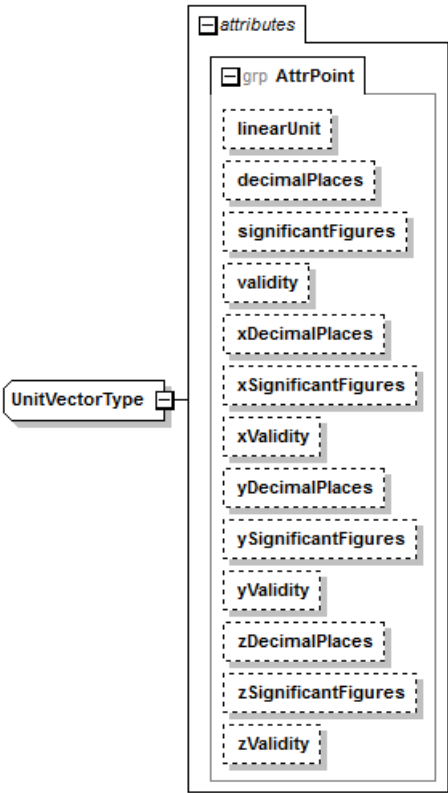
### element TransformRotationType/YDirection

diagram							
type	<a href="#">UnitVectorSimpleType</a>						
properties	content    simple						
facets	<table><tr><td>Kind</td><td>Value</td><td>Annotation</td></tr><tr><td>length</td><td>3</td><td></td></tr></table>	Kind	Value	Annotation	length	3	
Kind	Value	Annotation					
length	3						
annotation	<p>documentation</p> <p>The YDirection element is a unit vector that describes the y rotation. When the origin is (0,0,0), the components of the Y direction are the coordinates in the 'after' coordinate system of the point (0,1,0) in the 'before' system.</p>						

element **TransformRotationType/ZDirection**

diagram			
type	<a href="#">UnitVectorSimpleType</a>		
properties	content	simple	
facets	Kind	Value	Annotation
	length	3	
annotation	documentation The ZDirection element is a unit vector that describes the z rotation. When the Origin is (0,0,0), the components of the z Direction are the coordinates in the 'after' coordinate system of the point (0,0,1) in the 'before' system.		

complexType **UnitVectorType**

diagram						
type	extension of <a href="#">UnitVectorSimpleType</a>					
properties	base	UnitVectorSimpleType				
used by	elements	<a href="#">SweepType/DirBeg</a> <a href="#">PlaneXType/Direction</a> <a href="#">AxisType/Direction</a> <a href="#">LatitudeLongitudeSweepType/DirMeridianPrime</a> <a href="#">OrientedLatitudeLongitudeSweepType/DirNorthPole</a> <a href="#">PlaneType/Normal</a> <a href="#">ActualUnitVectorType</a>				
	complexType					
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	<b>xs:token</b>				
	<a href="#">decimalPlaces</a>	<b>xs:nonNegativeInteger</b>				

	<a href="#">significantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">validity</a> <b>ValidityEnumType</b> <a href="#">xDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">xSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">xValidity</a> <b>ValidityEnumType</b> <a href="#">yDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">ySignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">yValidity</a> <b>ValidityEnumType</b> <a href="#">zDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">zSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">zValidity</a> <b>ValidityEnumType</b>
annotation	documentation The UnitVectorType is a unit vector in which the accuracy of the components may be specified. The linearUnit attribute included in AttrPoint must not be used since unit vectors do not have length units.

### complexType UserDataXMLType

diagram	
used by	element <a href="#">AttributeUserType/UserDataXML</a>

### complexType VectorType

diagram	
type	extension of <a href="#">VectorSimpleType</a>

properties	base VectorSimpleType					
facets	Kind	Value	Annotation			
	length	3				
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">linearUnit</a>	xs:token				
	<a href="#">decimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">significantFigures</a>	xs:nonNegativeInteger				
	<a href="#">validity</a>	ValidityEnumType				
	<a href="#">xDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">xSignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">xValidity</a>	ValidityEnumType				
	<a href="#">yDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">ySignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">yValidity</a>	ValidityEnumType				
	<a href="#">zDecimalPlaces</a>	xs:nonNegativeInteger				
	<a href="#">zSignificantFigures</a>	xs:nonNegativeInteger				
	<a href="#">zValidity</a>	ValidityEnumType				
annotation	documentation The VectorType defines a vector in which the length units and accuracy of the components may be specified.					

simpleType **D2Type**

type	restriction of <a href="#">ListDoubleType</a>		
properties	base	ListDoubleType	
used by	complexType	<a href="#">AngleRangeType</a>	
facets	Kind	Value	Annotation
	length	2	
annotation	documentation The D2Type is an array of two double values.		

simpleType **D3Type**

type	restriction of <a href="#">ListDoubleType</a>		
properties	base ListDoubleType		
used by	attribute	<a href="#">AttributeD3Type/@value</a>	
facets	Kind	Value	Annotation
	length	3	
annotation	documentation The D3Type is an array of three double values.		

simpleType **D4Type**

type	restriction of <a href="#">ListDoubleType</a>		
properties	base	ListDoubleType	
facets	Kind	Value	Annotation
	length	4	
annotation	documentation The D4Type is an array of four double values.		



**simpleType DoublePositiveType**

type	restriction of <b>xs:double</b>
properties	base <b>xs:double</b>
facets	Kind Value Annotation minExclusive 0.0
annotation	documentation The DoublePositiveType is a positive double type. The value must be greater than zero.

**simpleType I2Type**

type	restriction of <a href="#">ListIntType</a>
properties	base ListIntType
used by	attribute <a href="#">AttributeI2Type/@value</a>
facets	Kind Value Annotation length 2
annotation	documentation The I2Type is an array of two integer values.

**simpleType ListBoolType**

type	list of <b>xs:boolean</b>
properties	base <b>xs:boolean</b>
annotation	documentation The ListBoolType is an array of Boolean numbers.

**simpleType ListDoubleType**

type	list of <b>xs:double</b>
properties	base <b>xs:double</b>
used by	complexTypes <a href="#">ArrayDoubleType</a> <a href="#">ArrayPoint2dType</a> <a href="#">ArrayPointType</a> <a href="#">ArrayUnitVectorType</a> simpleTypes <a href="#">D2Type</a> <a href="#">D3Type</a> <a href="#">D4Type</a> <a href="#">ParameterRangeType</a> <a href="#">Point2dSimpleType</a> <a href="#">PointSimpleType</a> <a href="#">UnitVector2dSimpleType</a> <a href="#">UnitVectorSimpleType</a> <a href="#">VectorSimpleType</a>
annotation	documentation The ListDoubleType is an array of double values.

**simpleType ListIntType**

type	list of <b>xs:integer</b>
properties	base <b>xs:integer</b>
used by	complexTypes <a href="#">ArrayI2Type</a> <a href="#">ArrayI3Type</a> <a href="#">ArrayIntType</a> simpleType <a href="#">I2Type</a>
annotation	documentation The ListIntType is a list of integer numbers.

**simpleType ListNaturalType**

type	list of <a href="#">NaturalType</a>
properties	base NaturalType

used by	complexType <a href="#">ArrayNaturalType</a>
annotation	documentation The ListNaturalType is an array of Natural numbers.

**simpleType ListUnsignedByteType**

type	list of <b>xs:unsignedByte</b>
properties	base <b>xs:unsignedByte</b>
used by	complexType <a href="#">ArrayUnsignedByteType</a>
annotation	documentation The ListUnsignedByteType is an array of unsigned byte values.

**simpleType NaturalType**

type	restriction of <b>xs:unsignedInt</b>
properties	base <b>xs:unsignedInt</b>
used by	simpleType <a href="#">ListNaturalType</a> attributes <a href="#">ArrayReferenceType/@N</a> <a href="#">ArrayReferenceFullType/@N</a> <a href="#">AttributesType/@N</a>
facets	Kind Value Annotation minInclusive 1
annotation	documentation The NaturalType is the natural number type (integer > 0).

**simpleType ParameterRangeType**

type	restriction of <a href="#">ListDoubleType</a>
properties	base ListDoubleType
facets	Kind Value Annotation length 2
annotation	documentation The ParameterRangeBoundsType is an array of two double values used to represent parameter bounds. The first entry in the list is the first bound of the parameter, and the second entry is the second bound. The second entry must be larger or smaller than the first.

**simpleType Point2dSimpleType**

type	restriction of <a href="#">ListDoubleType</a>
properties	base ListDoubleType
used by	elements <a href="#">LineSegment2dGroup/EndPoint</a> <a href="#">LineSegment2dGroup/StartPoint</a>
facets	Kind Value Annotation length 2
annotation	documentation The Point2dSimpleType is an array of two double values used to represent a 2D point. The first entry in the list is the X value and the second entry is the Y value.

**simpleType PointSimpleType**

type	restriction of <a href="#">ListDoubleType</a>
properties	base ListDoubleType

used by	elements complexType	<a href="#">LineSegmentGroup/EndPoint</a> <a href="#">CoordinateSystemCoreType/Origin</a> <a href="#">LineSegmentGroup/StartPoint</a> <a href="#">PointType</a>
facets	Kind Value Annotation length 3	
annotation	documentation	The PointSimpleType is an array of three double values used to represent the Cartesian three-dimensional coordinates. The first entry in the list is the X value, the second entry is the Y value, and the third entry is the Z value.

**simpleType QIFIdType**

type	restriction of <b>xs:unsignedInt</b>	
properties	base xs:unsignedInt	
used by	complexType attribute	<a href="#">QIFReferenceType</a> <a href="#">QIFReferenceFullType/@asmPath</a>
annotation	documentation	The QIFIdType defines an unsigned integer with no leading zeros. All 'id' attributes in all QIF schema files are of QIFIdType.

**simpleType QPIdReferenceType**

type	restriction of <b>xs:token</b>	
properties	base xs:token	
used by	elements	<a href="#">QPIdFullReferenceType/DocumentQPId</a> <a href="#">QPIdFullReferenceType/ItemQPId</a>
annotation	documentation	The QPIdReferenceType defines the text representation of the universally unique identifier described in the standard ISO/IEC 9834-8. As a number, it has 128 bits. As a text string it is represented by 32 hexadecimal digits displayed in five groups separated by hyphens in the form 8-4-4-4-12, for a total of 36 characters. example: fd43400a-29bf-4ec6-b96c-e2f846eb6ff6

**simpleType QPIdType**

type	restriction of <b>xs:token</b>	
properties	base xs:token	
used by	attribute	<a href="#">AttributeQPIdType/@value</a>
annotation	documentation	The QPIdType (QIF Persistent Identifier Type) is the text representation of the universally unique identifier described in the standard ISO/IEC 9834-8. As a number, it has 128 bits. As a text string it is represented by 32 hexadecimal digits displayed in five groups separated by hyphens in the form 8-4-4-4-12, for a total of 36 characters. example: fd43400a-29bf-4ec6-b96c-e2f846eb6ff6

**simpleType UnitVector2dSimpleType**

type	restriction of <a href="#">ListDoubleType</a>	
properties	base ListDoubleType	
facets	Kind Value Annotation length 2	
annotation	documentation	The UnitVectorSimpleType is an array of three double values. used to represent a 2D unit vector. The first entry in the list is the X component (I) and the second entry is the Y component (J). A UnitVectorSimpleType should be of unit length to the limits of accuracy of the executing system, and in any event must not be less than 0.99999999 or greater than 1.00000001.

simpleType **UnitVectorSimpleType**

type	restriction of <a href="#">ListDoubleType</a>
properties	base ListDoubleType
used by	elements <a href="#">TransformRotationType/XDirection</a> <a href="#">TransformRotationType/YDirection</a> complexType <a href="#">TransformRotationType/ZDirection</a> <a href="#">UnitVectorType</a>
facets	Kind Value Annotation length 3
annotation	documentation The UnitVectorSimpleType is an array of three double values. used to represent a 3D unit vector. The first entry in the list is the X component (I), the second entry is the Y component (J), and the third entry is the Z component (K). A UnitVectorSimpleType should be of unit length to the limits of accuracy of the executing system, and in any event must not be less than 0.99999999 or greater than 1.00000001.

simpleType **ValidityEnumType**

type	restriction of <b>xs:NMTOKEN</b>
properties	base xs:NMTOKEN
used by	attributes <a href="#">AttrPoint/@validity</a> <a href="#">AttrPoint/@xValidity</a> <a href="#">AttrPoint/@yValidity</a> <a href="#">AttrPoint/@zValidity</a>
facets	Kind Value Annotation enumeration REPORTED enumeration DUMMY enumeration MOOT enumeration DERIVED enumeration SET
annotation	documentation The ValidityEnumType enumerates values that describe the validity of a coordinate or set of coordinates. A coordinate can be present and correct in a report (REPORTED). A coordinate may be missing from a report but will have a value determined by one of several means: it may be an arbitrary value (DUMMY), it may be an arbitrary value that is meaningless such as the third coordinate in two-dimensional measurement (MOOT), it may be derived (calculated) from other values (DERIVED), or it may be set to a value (SET) typically the nominal value.

simpleType **VectorSimpleType**

type	restriction of <a href="#">ListDoubleType</a>
properties	base ListDoubleType
used by	complexType <a href="#">VectorType</a>
facets	Kind Value Annotation length 3
annotation	documentation The VectorSimpleType is an array of three double values used to represent a 3D vector. The first entry in the list is the X component (I), the second entry is the Y component (J), and the third entry is the Z component (K).

attributeGroup **AttrActualPoint**

Attribute Group: <a href="#">ActualPointType</a> <a href="#">ActualUnitVectorType</a>						
used by	complexType					
attributes	Name	Type	Use	Default	Fixed	Annotation
	<a href="#">combinedUncertainty</a>	xs:decimal				
	<a href="#">meanError</a>	xs:decimal				
	<a href="#">xCombinedUncertainty</a>	xs:decimal				
	<a href="#">xMeanError</a>	xs:decimal				

	<a href="#">yCombinedUncertainty</a> <b>xs:decimal</b> <a href="#">yMeanError</a> <b>xs:decimal</b> <a href="#">zCombinedUncertainty</a> <b>xs:decimal</b> <a href="#">zMeanError</a> <b>xs:decimal</b>
annotation	documentation The AttrActualPoint attributeGroup provides attributes for the combined uncertainty and mean error of actual 3D points and vectors. Uncertainty and/or error may be specified that applies to all three values (X, Y and Z) and/or uncertainty and/or error may be specified individually for X, Y, and Z. If both types of accuracy are given, the individual values should be used, not the group value.

**attribute AttrActualPoint/@combinedUncertainty**

type	<b>xs:decimal</b>
------	-------------------

**attribute AttrActualPoint/@meanError**

type	<b>xs:decimal</b>
------	-------------------

**attribute AttrActualPoint/@xCombinedUncertainty**

type	<b>xs:decimal</b>
------	-------------------

**attribute AttrActualPoint/@xMeanError**

type	<b>xs:decimal</b>
------	-------------------

**attribute AttrActualPoint/@yCombinedUncertainty**

type	<b>xs:decimal</b>
------	-------------------

**attribute AttrActualPoint/@yMeanError**

type	<b>xs:decimal</b>
------	-------------------

**attribute AttrActualPoint/@zCombinedUncertainty**

type	<b>xs:decimal</b>
------	-------------------

**attribute AttrActualPoint/@zMeanError**

type	<b>xs:decimal</b>
------	-------------------

**attributeGroup AttrPoint**

used by	complexTypes <a href="#">ArrayPointType</a> <a href="#">ArrayUnitVectorType</a> <a href="#">LineSegmentType</a> <a href="#">PointType</a> <a href="#">TransformMatrixType</a> <a href="#">UnitVectorType</a> <a href="#">VectorType</a>					
attributes	Name <a href="#">linearUnit</a>	Type <b>xs:token</b>	Use	Default	Fixed	Annotation

	<a href="#">decimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">significantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">validity</a> <b><a href="#">ValidityEnumType</a></b> <a href="#">xDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">xSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">xValidity</a> <b><a href="#">ValidityEnumType</a></b> <a href="#">yDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">ySignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">yValidity</a> <b><a href="#">ValidityEnumType</a></b> <a href="#">zDecimalPlaces</a> <b>xs:nonNegativeInteger</b> <a href="#">zSignificantFigures</a> <b>xs:nonNegativeInteger</b> <a href="#">zValidity</a> <b><a href="#">ValidityEnumType</a></b>
annotation	documentation The AttrPoint attributeGroup provides attributes for length unit and measures of the accuracy of the coordinates or components of 3D points and vectors. Accuracy may be specified that applies to all three values (X, Y and Z) and/or accuracy may be specified individually for X, Y, and Z. If both types of accuracy are given, the individual values should be used, not the group value.

attribute **AttrPoint/@linearUnit**

type	<b>xs:token</b>
------	-----------------

attribute **AttrPoint/@decimalPlaces**

type	<b>xs:nonNegativeInteger</b>
------	------------------------------

attribute **AttrPoint/@significantFigures**

type	<b>xs:nonNegativeInteger</b>
------	------------------------------

attribute **AttrPoint/@validity**

type	<b><a href="#">ValidityEnumType</a></b>		
facets	Kind	Value	Annotation
	enumeration	REPORTED	
	enumeration	DUMMY	
	enumeration	MOOT	
	enumeration	DERIVED	
	enumeration	SET	

attribute **AttrPoint/@xDecimalPlaces**

type	<b>xs:nonNegativeInteger</b>
------	------------------------------

attribute **AttrPoint/@xSignificantFigures**

type	<b>xs:nonNegativeInteger</b>
------	------------------------------

attribute **AttrPoint/@xValidity**

type	<a href="#">ValidityEnumType</a>		
facets	Kind	Value	Annotation
	enumeration	REPORTED	
	enumeration	DUMMY	
	enumeration	MOOT	
	enumeration	DERIVED	
	enumeration	SET	

attribute **AttrPoint/@yDecimalPlaces**

type	<b>xs:nonNegativeInteger</b>
------	------------------------------

attribute **AttrPoint/@ySignificantFigures**

type	<b>xs:nonNegativeInteger</b>
------	------------------------------

attribute **AttrPoint/@yValidity**

type	<a href="#">ValidityEnumType</a>		
facets	Kind	Value	Annotation
	enumeration	REPORTED	
	enumeration	DUMMY	
	enumeration	MOOT	
	enumeration	DERIVED	
	enumeration	SET	

attribute **AttrPoint/@zDecimalPlaces**

type	<b>xs:nonNegativeInteger</b>
------	------------------------------

attribute **AttrPoint/@zSignificantFigures**

type	<b>xs:nonNegativeInteger</b>
------	------------------------------

attribute **AttrPoint/@zValidity**

type	<a href="#">ValidityEnumType</a>		
facets	Kind	Value	Annotation
	enumeration	REPORTED	
	enumeration	DUMMY	
	enumeration	MOOT	
	enumeration	DERIVED	
	enumeration	SET	

