Schema **lensType.xsd**

schema location: **D:\projects\XML-Interfaces\xsd\generic\lens\lensType.xsd**

Complex types
lensType

schema location: **D:\projects\XML-Interfaces\xsd\generic/lens/refractionType.xsd**

Complex types
refractionType

schema location: **D:\projects\XML-Interfaces\xsd\generic/lens/optionsType.xsd**

Complex types
optionsType

schema location: **D:\projects\XML-Interfaces\xsd\generic/lens/preCalcType.xsd**

Complex types
preCalcType

schema location: **D:\projects\XML-Interfaces\xsd\generic/lens/cylinderType.xsd**

Complex types
cylinderType

schema location: **D:\projects\XML-Interfaces\xsd\generic/lens/prismType.xsd**

Complex types
prismType

complexType **lensType**

diagram	

lens-code
type xs:string

lens-id
type xs:string

Temporaer vorhanden zu
Kompatibilitaetszwecken -
wird in Kuerze entfernt!

edi-code
type xs:integer

Eigentlich redundant -
eventuell ueber Entfernung
nachdenken?!

product-line
type xs:integer

diameter
type
Bestelldurchmesser

description
type

lensType

refraction
type refractionType

decentration
type
0..2

modify-thickness-flag
type xs:boolean
Dickenaenderung zulaessig

optima-flag
type xs:boolean

nur noch vorübergehend aus
Kompatibilitätsgründen
vorhanden und wird
demnächst entfernt - bitte
optima-flag unter
frame-data, bzw.
frame-source verwenden!

options
type optionsType

children	**lens-code lens-id edi-code product-line diameter description refraction decentration modify-thickness-flag optima-flag options**

source	`<xs:complexType name="lensType">`
	`<xs:sequence>`
	`<xs:choice>`
	`<xs:element name="lens-code">`
	`<xs:simpleType>`
	`<xs:restriction base="xs:string">`
	`<xs:minLength value="1"/>`
	`<xs:maxLength value="6"/>`
	`</xs:restriction>`
	`</xs:simpleType>`
	`</xs:element>`
	`<xs:element name="lens-id">`

```xml
      <xs:annotation>
        <xs:documentation>Temporaer vorhanden zu Kompatibilitaetszwecken - wird in Kuerze
entfernt!</xs:documentation>
      </xs:annotation>
      <xs:simpleType>
       <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
        <xs:maxLength value="6"/>
       </xs:restriction>
      </xs:simpleType>
     </xs:element>
     <xs:element name="edi-code">
      <xs:annotation>
        <xs:documentation>Eigentlich redundant - eventuell ueber Entfernung nachdenken?!</xs:documentation>
      </xs:annotation>
      <xs:simpleType>
       <xs:restriction base="xs:integer">
        <xs:minInclusive value="-9999"/>
        <xs:maxInclusive value="9999"/>
       </xs:restriction>
      </xs:simpleType>
     </xs:element>
    </xs:choice>
    <xs:element name="product-line">
     <xs:simpleType>
      <xs:restriction base="xs:integer">
       <xs:minInclusive value="0"/>
      </xs:restriction>
     </xs:simpleType>
    </xs:element>
    <xs:element name="diameter">
     <xs:annotation>
       <xs:documentation>Bestelldurchmesser</xs:documentation>
     </xs:annotation>
     <xs:complexType>
      <xs:sequence>
       <xs:element name="physical">
        <xs:annotation>
          <xs:documentation>Physikalischer Durchmesser</xs:documentation>
        </xs:annotation>
        <xs:simpleType>
         <xs:restriction base="xs:integer">
          <xs:minInclusive value="1"/>
          <xs:maxInclusive value="99"/>
         </xs:restriction>
        </xs:simpleType>
       </xs:element>
       <xs:element name="optical" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Optisch wirksamer Durchmesser - nur vorhanden, falls unterschiedlich vom physikalischen
Durchmesser (also bei vordezentrierten Glaesern)</xs:documentation>
        </xs:annotation>
        <xs:simpleType>
         <xs:restriction base="xs:integer">
          <xs:minInclusive value="1"/>
          <xs:maxInclusive value="99"/>
         </xs:restriction>
        </xs:simpleType>
       </xs:element>
      </xs:sequence>
     </xs:complexType>
    </xs:element>
    <xs:element name="description" minOccurs="0">
     <xs:complexType>
      <xs:sequence>
       <xs:element name="name" type="xs:string" minOccurs="0"/>
       <xs:element name="ce-text" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
       <xs:element name="note" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
       <xs:element name="lens-bag-name" type="xs:string" minOccurs="0"/>
      </xs:sequence>
     </xs:complexType>
    </xs:element>
    <xs:element name="refraction" type="refractionType"/>
    <xs:element name="decentration" minOccurs="0" maxOccurs="2">
     <xs:complexType>
      <xs:sequence>
       <xs:element name="length">
```

```xml
            <xs:simpleType>
             <xs:restriction base="xs:float">
              <xs:minInclusive value="0.1"/>
              <xs:maxInclusive value="40.0"/>
             </xs:restriction>
            </xs:simpleType>
           </xs:element>
           <xs:element name="direction">
            <xs:simpleType>
             <xs:restriction base="xs:float">
              <xs:minInclusive value="0.0"/>
              <xs:maxInclusive value="360.0"/>
             </xs:restriction>
            </xs:simpleType>
           </xs:element>
          </xs:sequence>
          <xs:attribute name="origin" use="optional" default="internal">
           <xs:simpleType>
            <xs:restriction base="xs:string">
             <xs:enumeration value="internal"/>
             <xs:enumeration value="customer"/>
            </xs:restriction>
           </xs:simpleType>
          </xs:attribute>
         </xs:complexType>
        </xs:element>
        <xs:element name="modify-thickness-flag" type="xs:boolean">
         <xs:annotation>
          <xs:documentation>Dickenaenderung zulaessig</xs:documentation>
         </xs:annotation>
        </xs:element>
        <xs:element name="optima-flag" type="xs:boolean">
         <xs:annotation>
          <xs:documentation>nur noch vorübergehend aus Kompatibilitätsgründen vorhanden und wird demnächst entfernt -
bitte optima-flag unter frame-data, bzw. frame-source verwenden!</xs:documentation>
         </xs:annotation>
        </xs:element>
        <xs:element name="options" type="optionsType" minOccurs="0"/>
       </xs:sequence>
      </xs:complexType>
```

element **lensType/lens-code**

diagram	
type	restriction of **xs:string**
facets	minLength 1 maxLength 6

source	`<xs:element name="lens-code">`
	`  <xs:simpleType>`
	`   <xs:restriction base="xs:string">`
	`    <xs:minLength value="1"/>`
	`    <xs:maxLength value="6"/>`
	`   </xs:restriction>`
	`  </xs:simpleType>`
	`</xs:element>`

element **lensType/lens-id**

diagram	 Temporaer vorhanden zu Kompatibilitaetszwecken - wird in Kuerze entfernt!
type	restriction of **xs:string**
facets	minLength 1 maxLength 6

annotation	documentation	Temporaer vorhanden zu Kompatibilitaetszwecken - wird in Kuerze entfernt!

source		`<xs:element name="lens-id">`
		 `<xs:annotation>`
		  `<xs:documentation>`Temporaer vorhanden zu Kompatibilitaetszwecken - wird in Kuerze entfernt!`</xs:documentation>`
		 `</xs:annotation>`
		 `<xs:simpleType>`
		  `<xs:restriction base="xs:string">`
		   `<xs:minLength value="1"/>`
		   `<xs:maxLength value="6"/>`
		  `</xs:restriction>`
		 `</xs:simpleType>`
		`</xs:element>`

element **lensType/edi-code**

diagram	 edi-code type xs:integer Eigentlich redundant - eventuell ueber Entfernung nachdenken?!
type	restriction of **xs:integer**
facets	minInclusive -9999 maxInclusive 9999
annotation	documentation Eigentlich redundant - eventuell ueber Entfernung nachdenken?!

source	`<xs:element name="edi-code">`
	 `<xs:annotation>`
	  `<xs:documentation>`Eigentlich redundant - eventuell ueber Entfernung nachdenken?!`</xs:documentation>`
	 `</xs:annotation>`
	 `<xs:simpleType>`
	  `<xs:restriction base="xs:integer">`
	   `<xs:minInclusive value="-9999"/>`
	   `<xs:maxInclusive value="9999"/>`
	  `</xs:restriction>`
	 `</xs:simpleType>`
	`</xs:element>`

element **lensType/product-line**

diagram	 product-line type xs:integer
type	restriction of **xs:integer**
facets	minInclusive 0

source	`<xs:element name="product-line">`
	 `<xs:simpleType>`
	  `<xs:restriction base="xs:integer">`
	   `<xs:minInclusive value="0"/>`
	  `</xs:restriction>`
	 `</xs:simpleType>`
	`</xs:element>`

element **lensType/diameter**

diagram	

children	**physical** **optical**

annotation	documentation Bestelldurchmesser

| source | ```xml
<xs:element name="diameter">
 <xs:annotation>
  <xs:documentation>Bestelldurchmesser</xs:documentation>
 </xs:annotation>
 <xs:complexType>
  <xs:sequence>
   <xs:element name="physical">
    <xs:annotation>
     <xs:documentation>Physikalischer Durchmesser</xs:documentation>
    </xs:annotation>
    <xs:simpleType>
     <xs:restriction base="xs:integer">
      <xs:minInclusive value="1"/>
      <xs:maxInclusive value="99"/>
     </xs:restriction>
    </xs:simpleType>
   </xs:element>
   <xs:element name="optical" minOccurs="0">
    <xs:annotation>
     <xs:documentation>Optisch wirksamer Durchmesser - nur vorhanden, falls unterschiedlich vom physikalischen Durchmesser (also bei vordezentrierten Glaesern)</xs:documentation>
    </xs:annotation>
    <xs:simpleType>
     <xs:restriction base="xs:integer">
      <xs:minInclusive value="1"/>
      <xs:maxInclusive value="99"/>
     </xs:restriction>
    </xs:simpleType>
   </xs:element>
  </xs:sequence>
 </xs:complexType>
</xs:element>
``` |

element **lensType/diameter/physical**

diagram	

type	restriction of **xs:integer**

facets	minInclusive 1 maxInclusive 99

annotation	documentation Physikalischer Durchmesser

| source | ```xml
<xs:element name="physical">
 <xs:annotation>
  <xs:documentation>Physikalischer Durchmesser</xs:documentation>
 </xs:annotation>
``` |

```
      <xs:simpleType>
       <xs:restriction base="xs:integer">
        <xs:minInclusive value="1"/>
        <xs:maxInclusive value="99"/>
       </xs:restriction>
      </xs:simpleType>
     </xs:element>
```

element **lensType/diameter/optical**

diagram	
type	restriction of **xs:integer**
facets	minInclusive 1 maxInclusive 99
annotation	documentation Optisch wirksamer Durchmesser - nur vorhanden, falls unterschiedlich vom physikalischen Durchmesser (also bei vordezentrierten Glaesern)

source	`<xs:element name="optical" minOccurs="0">`
	 `<xs:annotation>`
	  `<xs:documentation>`Optisch wirksamer Durchmesser - nur vorhanden, falls unterschiedlich vom physikalischen Durchmesser (also bei vordezentrierten Glaesern)`</xs:documentation>`
	 `</xs:annotation>`
	 `<xs:simpleType>`
	  `<xs:restriction base="xs:integer">`
	   `<xs:minInclusive value="1"/>`
	   `<xs:maxInclusive value="99"/>`
	  `</xs:restriction>`
	 `</xs:simpleType>`
	`</xs:element>`

element **lensType/description**

diagram	
children	**name ce-text note lens-bag-name**

source	`<xs:element name="description" minOccurs="0">`
	 `<xs:complexType>`
	  `<xs:sequence>`
	   `<xs:element name="name" type="xs:string" minOccurs="0"/>`
	   `<xs:element name="ce-text" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>`
	   `<xs:element name="note" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>`
	   `<xs:element name="lens-bag-name" type="xs:string" minOccurs="0"/>`
	  `</xs:sequence>`
	 `</xs:complexType>`
	`</xs:element>`

element **lensType/description/name**

diagram	name type xs:string
type	**xs:string**
source	<xs:element name="name" type="xs:string" minOccurs="0"/>

element **lensType/description/ce-text**

diagram	ce-text type xs:string
type	**xs:string**
source	<xs:element name="ce-text" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>

element **lensType/description/note**

diagram	note type xs:string
type	**xs:string**
source	<xs:element name="note" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>

element **lensType/description/lens-bag-name**

diagram	lens-bag-name type xs:string
type	**xs:string**
source	<xs:element name="lens-bag-name" type="xs:string" minOccurs="0"/>

element **lensType/refraction**

diagram	
type	**refractionType**
children	**sphere cylinder addition prism inset upset interpupillary-distance near-object-distance**
source	`<xs:element name="refraction" type="refractionType"/>`

element **lensType/decentration**

diagram	
children	**length direction**

attributes	Name	Type	Use	Default	Fixed	Annotation
	origin	xs:string	optional	internal		

source	`<xs:element name="decentration" minOccurs="0" maxOccurs="2">`
	`  <xs:complexType>`
	`   <xs:sequence>`
	`    <xs:element name="length">`
	`     <xs:simpleType>`
	`      <xs:restriction base="xs:float">`
	`       <xs:minInclusive value="0.1"/>`
	`       <xs:maxInclusive value="40.0"/>`
	`      </xs:restriction>`
	`     </xs:simpleType>`
	`    </xs:element>`

```
          <xs:element name="direction">
           <xs:simpleType>
            <xs:restriction base="xs:float">
             <xs:minInclusive value="0.0"/>
             <xs:maxInclusive value="360.0"/>
            </xs:restriction>
           </xs:simpleType>
          </xs:element>
         </xs:sequence>
         <xs:attribute name="origin" use="optional" default="internal">
          <xs:simpleType>
           <xs:restriction base="xs:string">
            <xs:enumeration value="internal"/>
            <xs:enumeration value="customer"/>
           </xs:restriction>
          </xs:simpleType>
         </xs:attribute>
        </xs:complexType>
       </xs:element>
```

element **lensType/decentration/length**

diagram	**length** type xs:float
type	restriction of **xs:float**
facets	minInclusive 0.1 maxInclusive 40.0

source	`<xs:element name="length">`
	`  <xs:simpleType>`
	`   <xs:restriction base="xs:float">`
	`    <xs:minInclusive value="0.1"/>`
	`    <xs:maxInclusive value="40.0"/>`
	`   </xs:restriction>`
	`  </xs:simpleType>`
	`</xs:element>`

element **lensType/decentration/direction**

diagram	**direction** type xs:float
type	restriction of **xs:float**
facets	minInclusive 0.0 maxInclusive 360.0

source	`<xs:element name="direction">`
	`  <xs:simpleType>`
	`   <xs:restriction base="xs:float">`
	`    <xs:minInclusive value="0.0"/>`
	`    <xs:maxInclusive value="360.0"/>`
	`   </xs:restriction>`
	`  </xs:simpleType>`
	`</xs:element>`

element **lensType/modify-thickness-flag**

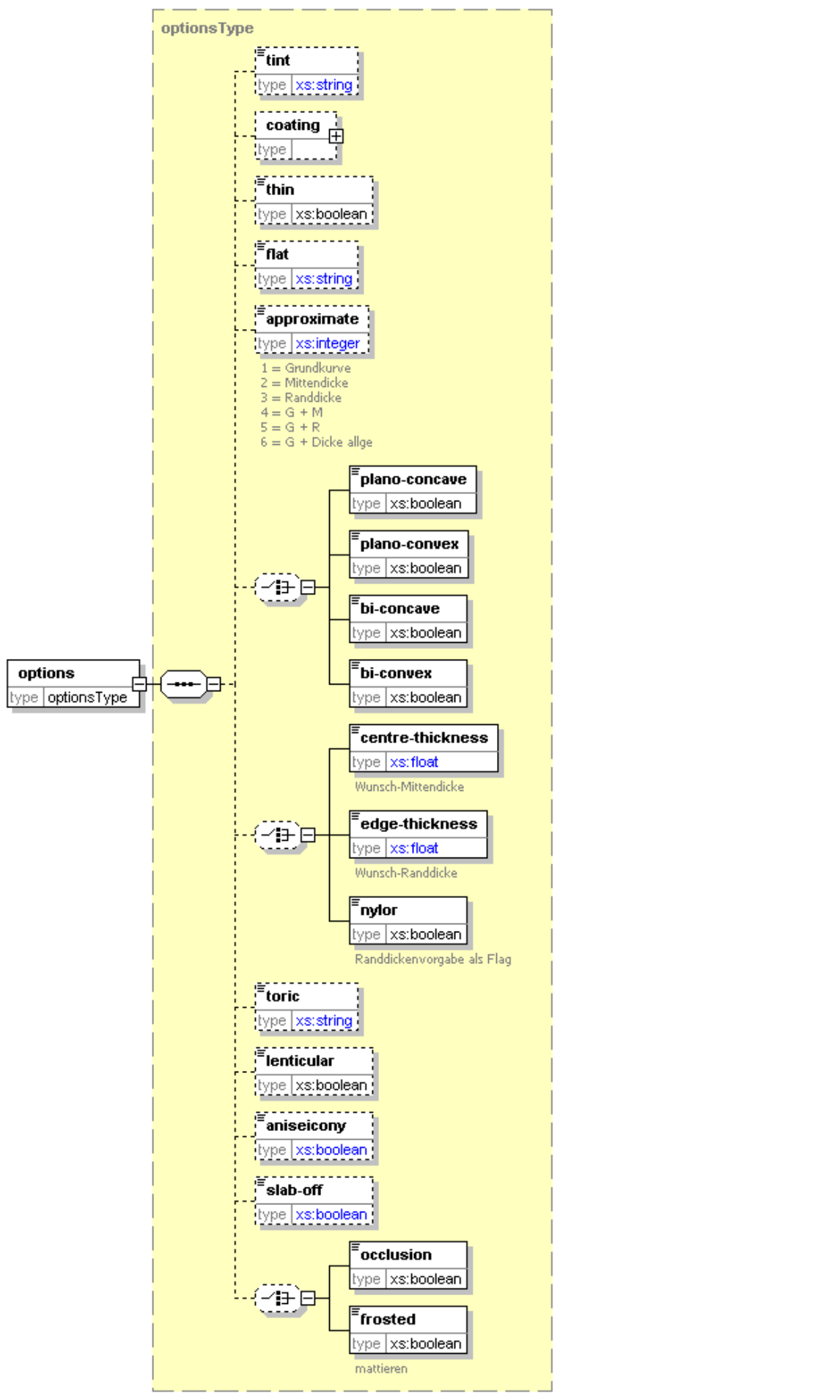|	
diagram	**modify-thickness-flag** type xs:boolean Dickenaenderung zulaessig
type	**xs:boolean**
annotation	documentation Dickenaenderung zulaessig
source	`<xs:element name="modify-thickness-flag" type="xs:boolean">`

	<xs:annotation> <xs:documentation>Dickenaenderung zulaessig</xs:documentation> </xs:annotation> </xs:element>

element **lensType/optima-flag**

diagram	optima-flag type xs:boolean nur noch vorübergehend aus Kompatibilitätsgründen vorhanden und wird demnächst entfernt - bitte optima-flag unter frame-data, bzw. frame-source verwenden!
type	**xs:boolean**
annotation	documentation nur noch vorübergehend aus Kompatibilitätsgründen vorhanden und wird demnächst entfernt - bitte optima-flag unter frame-data, bzw. frame-source verwenden!
source	<xs:element name="optima-flag" type="xs:boolean"> <xs:annotation> <xs:documentation>nur noch vorübergehend aus Kompatibilitätsgründen vorhanden und wird demnächst entfernt - bitte optima-flag unter frame-data, bzw. frame-source verwenden!</xs:documentation> </xs:annotation> </xs:element>

element **lensType/options**

diagram	

optionsType

options
type optionsType

tint
type xs:string

coating
type

thin
type xs:boolean

flat
type xs:string

approximate
type xs:integer

1 = Grundkurve
2 = Mittendicke
3 = Randdicke
4 = G + M
5 = G + R
6 = G + Dicke allge

plano-concave
type xs:boolean

plano-convex
type xs:boolean

bi-concave
type xs:boolean

bi-convex
type xs:boolean

centre-thickness
type xs:float
Wunsch-Mittendicke

edge-thickness
type xs:float
Wunsch-Randdicke

nylor
type xs:boolean
Randdickenvorgabe als Flag

toric
type xs:string

lenticular
type xs:boolean

aniseicony
type xs:boolean

slab-off
type xs:boolean

occlusion
type xs:boolean

frosted
type xs:boolean
mattieren

type	**optionsType**
children	**tint** **coating** **thin** **flat** **approximate** **plano-concave** **plano-convex** **bi-concave** **bi-convex** **centre-thickness** **edge-thickness** **nylor** **toric** **lenticular** **aniseicony** **slab-off** **occlusion** **frosted**
source	<xs:element name="options" type="optionsType" minOccurs="0"/>

complexType **refractionType**

diagram	
children	**sphere** **cylinder** **addition** **prism** **inset** **upset** **interpupillary-distance** **near-object-distance**
used by	element **lensType/refraction**

source	<xs:complexType name="refractionType">
	 <xs:sequence>
	  <xs:element name="sphere">
	   <xs:simpleType>
	    <xs:restriction base="xs:float">
	     <xs:minInclusive value="-50"/>
	     <xs:maxInclusive value="50"/>
	    </xs:restriction>
	   </xs:simpleType>
	  </xs:element>
	  <xs:element name="cylinder" type="cylinderType" minOccurs="0"/>
	  <xs:element name="addition" minOccurs="0">
	   <xs:simpleType>
	    <xs:restriction base="xs:float">
	     <xs:minInclusive value="0.25"/>
	    </xs:restriction>
	   </xs:simpleType>
	  </xs:element>
	  <xs:element name="prism" minOccurs="0" maxOccurs="2">
	   <xs:complexType>
	    <xs:complexContent>
	     <xs:extension base="prismType">
	      <xs:attribute name="pupillary-distance-correction" use="optional">
	       <xs:simpleType>
	        <xs:restriction base="xs:int">
	         <xs:enumeration value="0"/>

```xml
                    <xs:enumeration value="1"/>
                    <xs:enumeration value="2"/>
                  </xs:restriction>
                </xs:simpleType>
              </xs:attribute>
            </xs:extension>
          </xs:complexContent>
        </xs:complexType>
      </xs:element>
      <xs:element name="inset" minOccurs="0">
        <xs:complexType>
          <xs:choice>
            <xs:element name="null">
              <xs:simpleType>
                <xs:restriction base="xs:string">
                  <xs:enumeration value="null"/>
                </xs:restriction>
              </xs:simpleType>
            </xs:element>
            <xs:element name="value" type="xs:float"/>
            <xs:sequence>
              <xs:element name="z" type="xs:float"/>
              <xs:element name="q" type="xs:float">
                <xs:annotation>
                  <xs:documentation>Nah-PD</xs:documentation>
                </xs:annotation>
              </xs:element>
            </xs:sequence>
          </xs:choice>
        </xs:complexType>
      </xs:element>
      <xs:element name="upset" minOccurs="0">
        <xs:complexType>
          <xs:choice>
            <xs:element name="null" type="xs:string"/>
            <xs:element name="value" type="xs:float"/>
            <xs:sequence>
              <xs:element name="y" type="xs:float"/>
              <xs:element name="h" type="xs:float"/>
            </xs:sequence>
          </xs:choice>
        </xs:complexType>
      </xs:element>
      <xs:element name="interpupillary-distance" type="xs:float" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Monukulare PD</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="near-object-distance" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Objektabstand Nähe für individuelle Gleitsichtgläser</xs:documentation>
        </xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:integer">
            <xs:maxInclusive value="4444"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
```

element **refractionType/sphere**

diagram	
type	restriction of **xs:float**
facets	minInclusive -50 maxInclusive 50

source	`<xs:element name="sphere">`
	`  <xs:simpleType>`
	`    <xs:restriction base="xs:float">`
	`      <xs:minInclusive value="-50"/>`

```
        <xs:maxInclusive value="50"/>
       </xs:restriction>
      </xs:simpleType>
     </xs:element>
```

element **refractionType/cylinder**

diagram	
type	**cylinderType**
children	**power** **axis**
source	`<xs:element name="cylinder" type="cylinderType" minOccurs="0"/>`

element **refractionType/addition**

diagram	
type	restriction of **xs:float**
facets	minInclusive 0.25

source	`<xs:element name="addition" minOccurs="0">`
	`  <xs:simpleType>`
	`   <xs:restriction base="xs:float">`
	`    <xs:minInclusive value="0.25"/>`
	`   </xs:restriction>`
	`  </xs:simpleType>`
	`</xs:element>`

element **refractionType/prism**

diagram						
type	extension of **prismType**					
children	**power** **base**					
attributes	Name	Type	Use	Default	Fixed	Annotation
	pupillary-distance-correction	xs:int	optional			

source	`<xs:element name="prism" minOccurs="0" maxOccurs="2">`
	`  <xs:complexType>`
	`   <xs:complexContent>`
	`    <xs:extension base="prismType">`
	`     <xs:attribute name="pupillary-distance-correction" use="optional">`
	`      <xs:simpleType>`
	`       <xs:restriction base="xs:int">`
	`        <xs:enumeration value="0"/>`					

```
            <xs:enumeration value="1"/>
            <xs:enumeration value="2"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:extension>
   </xs:complexContent>
  </xs:complexType>
</xs:element>
```
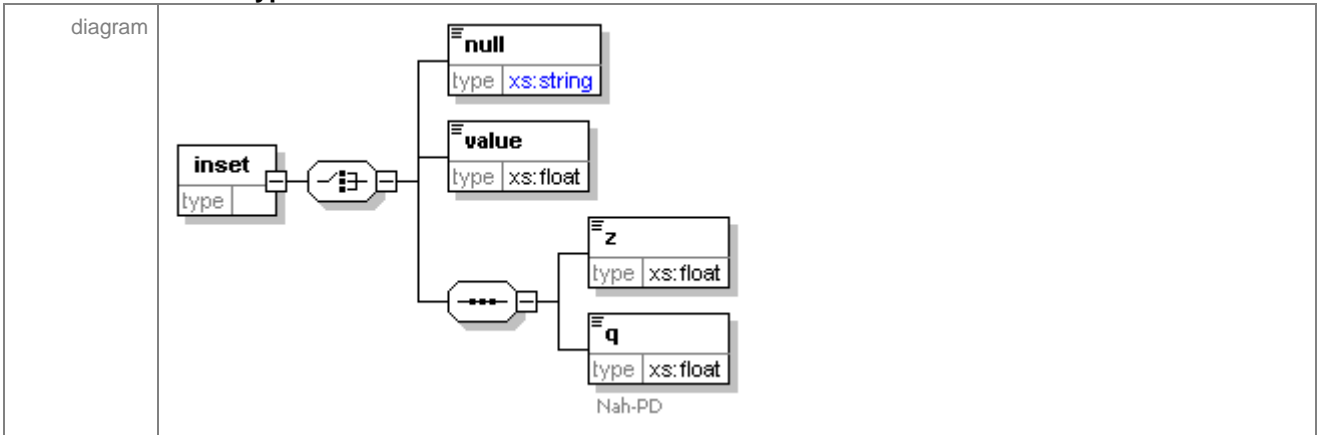
element **refractionType/inset**

diagram	 Nah-PD
children	**null value z q**

source	```<xs:element name="inset" minOccurs="0">```
	``` <xs:complexType>```
	``` <xs:choice>```
	``` <xs:element name="null">```
	``` <xs:simpleType>```
	``` <xs:restriction base="xs:string">```
	``` <xs:enumeration value="null"/>```
	``` </xs:restriction>```
	``` </xs:simpleType>```
	``` </xs:element>```
	``` <xs:element name="value" type="xs:float"/>```
	``` <xs:sequence>```
	``` <xs:element name="z" type="xs:float"/>```
	``` <xs:element name="q" type="xs:float">```
	``` <xs:annotation>```
	``` <xs:documentation>Nah-PD</xs:documentation>```
	``` </xs:annotation>```
	``` </xs:element>```
	``` </xs:sequence>```
	``` </xs:choice>```
	``` </xs:complexType>```
	```</xs:element>```
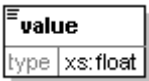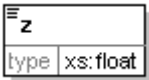
element **refractionType/inset/null**

diagram	
type	restriction of **xs:string**
facets	enumeration null

source	```<xs:element name="null">```
	``` <xs:simpleType>```
	``` <xs:restriction base="xs:string">```
	``` <xs:enumeration value="null"/>```
	``` </xs:restriction>```
	``` </xs:simpleType>```
	```</xs:element>```
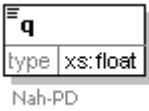
element **refractionType/inset/value**

diagram	**value** type xs:float
type	**xs:float**
source	<xs:element name="value" type="xs:float"/>
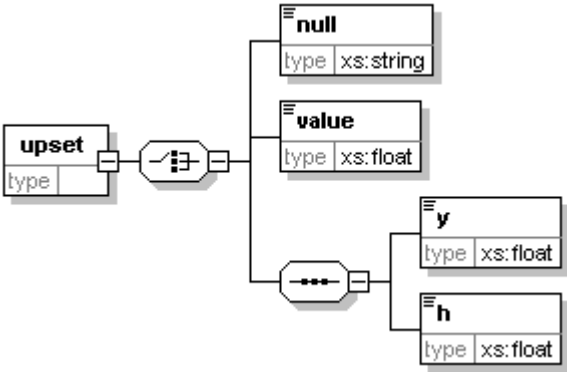
element **refractionType/inset/z**

diagram	**z** type xs:float
type	**xs:float**
source	<xs:element name="z" type="xs:float"/>

element **refractionType/inset/q**

diagram	**q** type xs:float Nah-PD
type	**xs:float**
annotation	documentation Nah-PD
source	<xs:element name="q" type="xs:float"> <xs:annotation> <xs:documentation>Nah-PD</xs:documentation> </xs:annotation> </xs:element>

element **refractionType/upset**

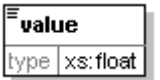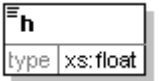|	
diagram	
children	**null value y h**
source	<xs:element name="upset" minOccurs="0"> <xs:complexType> <xs:choice> <xs:element name="null" type="xs:string"/> <xs:element name="value" type="xs:float"/> <xs:sequence> <xs:element name="y" type="xs:float"/> <xs:element name="h" type="xs:float"/> </xs:sequence> </xs:choice> </xs:complexType> </xs:element>

```
                              </xs:element>
```

element **refractionType/upset/null**

diagram	null type xs:string
type	**xs:string**
source	`<xs:element name="null" type="xs:string"/>`

element **refractionType/upset/value**

diagram	value type xs:float
type	**xs:float**
source	`<xs:element name="value" type="xs:float"/>`
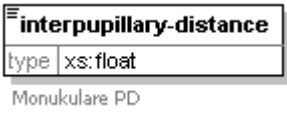
element **refractionType/upset/y**

diagram	y type xs:float
type	**xs:float**
source	`<xs:element name="y" type="xs:float"/>`

element **refractionType/upset/h**

diagram	h type xs:float
type	**xs:float**
source	`<xs:element name="h" type="xs:float"/>`

element **refractionType/interpupillary-distance**

diagram	interpupillary-distance type xs:float Monukulare PD
type	**xs:float**
annotation	documentation Monukulare PD

source	`<xs:element name="interpupillary-distance" type="xs:float" minOccurs="0">`
	`  <xs:annotation>`
	`    <xs:documentation>Monukulare PD</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

element **refractionType/near-object-distance**

diagram	near-object-distance type xs:integer Objektabstand Nähe für individuelle Gleitsichtgläser
type	restriction of **xs:integer**
facets	maxInclusive 4444
annotation	documentation Objektabstand Nähe für individuelle Gleitsichtgläser

source	`<xs:element name="near-object-distance" minOccurs="0">`
	 `<xs:annotation>`
	  `<xs:documentation>Objektabstand Nähe für individuelle Gleitsichtgläser</xs:documentation>`
	 `</xs:annotation>`
	 `<xs:simpleType>`
	  `<xs:restriction base="xs:integer">`
	   `<xs:maxInclusive value="4444"/>`
	  `</xs:restriction>`
	 `</xs:simpleType>`
	`</xs:element>`

complexType **optionsType**

diagram

tint
type | xs:string

coating
type | [+]

thin
type | xs:boolean

flat
type | xs:string

approximate
type | xs:integer

1 = Grundkurve
2 = Mittendicke
3 = Randdicke
4 = G + M
5 = G + R
6 = G + Dicke allge

optionsType

plano-concave
type | xs:boolean

plano-convex
type | xs:boolean

bi-concave
type | xs:boolean

bi-convex
type | xs:boolean

centre-thickness
type | xs:float

Wunsch-Mittendicke

edge-thickness
type | xs:float

Wunsch-Randdicke

nylor
type | xs:boolean

Randdickenvorgabe als Flag

toric
type | xs:string

lenticular
type | xs:boolean

aniseicony
type | xs:boolean

slab-off
type | xs:boolean

occlusion
type | xs:boolean

frosted
type | xs:boolean

mattieren

children	tint coating thin flat approximate plano-concave plano-convex bi-concave bi-convex centre-thickness edge-thickness nylor toric lenticular aniseicony slab-off occlusion frosted
used by	element **lensType/options**

source

```xml
<xs:complexType name="optionsType">
  <xs:sequence>
    <xs:element name="tint" minOccurs="0">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="xs:string">
            <xs:attribute name="note" type="xs:string" use="optional"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
    <xs:element name="coating" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="antireflection" minOccurs="0">
            <xs:complexType>
              <xs:simpleContent>
                <xs:extension base="xs:string">
                  <xs:attribute name="side" use="required">
                    <xs:simpleType>
                      <xs:restriction base="xs:string">
                        <xs:enumeration value="both"/>
                        <xs:enumeration value="front"/>
                        <xs:enumeration value="back"/>
                      </xs:restriction>
                    </xs:simpleType>
                  </xs:attribute>
                </xs:extension>
              </xs:simpleContent>
            </xs:complexType>
          </xs:element>
          <xs:element name="hard" type="xs:string" minOccurs="0"/>
          <xs:choice minOccurs="0">
            <xs:element name="tint">
              <xs:annotation>
                <xs:documentation>Umbra</xs:documentation>
              </xs:annotation>
              <xs:complexType>
                <xs:simpleContent>
                  <xs:extension base="xs:string">
                    <xs:attribute name="side" use="optional">
                      <xs:simpleType>
                        <xs:restriction base="xs:string">
                          <xs:enumeration value="both"/>
                          <xs:enumeration value="front"/>
                          <xs:enumeration value="back"/>
                        </xs:restriction>
                      </xs:simpleType>
                    </xs:attribute>
                  </xs:extension>
                </xs:simpleContent>
              </xs:complexType>
            </xs:element>
            <xs:element name="uv-protection" type="xs:string"/>
          </xs:choice>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="thin" type="xs:boolean" minOccurs="0"/>
    <xs:element name="flat" minOccurs="0">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="flat"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="approximate" minOccurs="0">
      <xs:annotation>
        <xs:documentation>1 = Grundkurve
2 = Mittendicke
3 = Randdicke
4 = G + M
```

```
5 = G + R
6 = G + Dicke allge</xs:documentation>
     </xs:annotation>
     <xs:complexType>
      <xs:simpleContent>
       <xs:extension base="xs:integer"/>
      </xs:simpleContent>
     </xs:complexType>
    </xs:element>
    <xs:choice minOccurs="0">
     <xs:element name="plano-concave" type="xs:boolean"/>
     <xs:element name="plano-convex" type="xs:boolean"/>
     <xs:element name="bi-concave" type="xs:boolean"/>
     <xs:element name="bi-convex" type="xs:boolean"/>
    </xs:choice>
    <xs:choice minOccurs="0">
     <xs:element name="centre-thickness">
      <xs:annotation>
       <xs:documentation>Wunsch-Mittendicke</xs:documentation>
      </xs:annotation>
      <xs:simpleType>
       <xs:restriction base="xs:float">
        <xs:minExclusive value="0.2"/>
        <xs:maxExclusive value="30.0"/>
       </xs:restriction>
      </xs:simpleType>
     </xs:element>
     <xs:element name="edge-thickness">
      <xs:annotation>
       <xs:documentation>Wunsch-Randdicke</xs:documentation>
      </xs:annotation>
      <xs:simpleType>
       <xs:restriction base="xs:float">
        <xs:minExclusive value="0.2"/>
        <xs:maxExclusive value="30.0"/>
       </xs:restriction>
      </xs:simpleType>
     </xs:element>
     <xs:element name="nylor" type="xs:boolean">
      <xs:annotation>
       <xs:documentation>Randdickenvorgabe als Flag</xs:documentation>
      </xs:annotation>
     </xs:element>
    </xs:choice>
    <xs:element name="toric" default="back" minOccurs="0">
     <xs:simpleType>
      <xs:restriction base="xs:string">
       <xs:enumeration value="front"/>
       <xs:enumeration value="back"/>
      </xs:restriction>
     </xs:simpleType>
    </xs:element>
    <xs:element name="lenticular" type="xs:boolean" minOccurs="0"/>
    <xs:element name="aniseicony" minOccurs="0">
     <xs:complexType>
      <xs:simpleContent>
       <xs:extension base="xs:boolean">
        <xs:attribute name="value" type="xs:float" use="optional"/>
       </xs:extension>
      </xs:simpleContent>
     </xs:complexType>
    </xs:element>
    <xs:element name="slab-off" minOccurs="0">
     <xs:complexType>
      <xs:simpleContent>
       <xs:extension base="xs:boolean">
        <xs:attribute name="value" use="optional">
         <xs:simpleType>
          <xs:restriction base="xs:float">
           <xs:minInclusive value="1.3"/>
          </xs:restriction>
         </xs:simpleType>
        </xs:attribute>
       </xs:extension>
      </xs:simpleContent>
     </xs:complexType>
    </xs:element>
```

```
            <xs:choice minOccurs="0">
             <xs:element name="occlusion" type="xs:boolean"/>
             <xs:element name="frosted" type="xs:boolean">
              <xs:annotation>
               <xs:documentation>mattieren</xs:documentation>
              </xs:annotation>
             </xs:element>
            </xs:choice>
           </xs:sequence>
          </xs:complexType>
```

element **optionsType/tint**

diagram	

type	extension of **xs:string**

attributes	Name	Type	Use	Default	Fixed	Annotation
	note	xs:string	optional			

source	`<xs:element name="tint" minOccurs="0">`
	`<xs:complexType>`
	`<xs:simpleContent>`
	`<xs:extension base="xs:string">`
	`<xs:attribute name="note" type="xs:string" use="optional"/>`
	`</xs:extension>`
	`</xs:simpleContent>`
	`</xs:complexType>`
	`</xs:element>`

element **optionsType/coating**

diagram	

children	**antireflection** **hard** **tint** **uv-protection**

source	`<xs:element name="coating" minOccurs="0">`
	`<xs:complexType>`
	`<xs:sequence>`
	`<xs:element name="antireflection" minOccurs="0">`
	`<xs:complexType>`
	`<xs:simpleContent>`
	`<xs:extension base="xs:string">`
	`<xs:attribute name="side" use="required">`
	`<xs:simpleType>`
	`<xs:restriction base="xs:string">`
	`<xs:enumeration value="both"/>`
	`<xs:enumeration value="front"/>`
	`<xs:enumeration value="back"/>`
	`</xs:restriction>`
	`</xs:simpleType>`
	`</xs:attribute>`
	`</xs:extension>`
	`</xs:simpleContent>`
	`</xs:complexType>`
	`</xs:element>`
	`<xs:element name="hard" type="xs:string" minOccurs="0"/>`

```
                    <xs:choice minOccurs="0">
                     <xs:element name="tint">
                       <xs:annotation>
                        <xs:documentation>Umbra</xs:documentation>
                       </xs:annotation>
                       <xs:complexType>
                        <xs:simpleContent>
                         <xs:extension base="xs:string">
                           <xs:attribute name="side" use="optional">
                            <xs:simpleType>
                             <xs:restriction base="xs:string">
                               <xs:enumeration value="both"/>
                               <xs:enumeration value="front"/>
                               <xs:enumeration value="back"/>
                             </xs:restriction>
                            </xs:simpleType>
                           </xs:attribute>
                         </xs:extension>
                        </xs:simpleContent>
                       </xs:complexType>
                     </xs:element>
                     <xs:element name="uv-protection" type="xs:string"/>
                    </xs:choice>
                  </xs:sequence>
                 </xs:complexType>
               </xs:element>
```
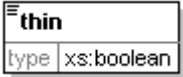
element **optionsType/coating/antireflection**

diagram	
type	extension of **xs:string**

attributes	Name	Type	Use	Default	Fixed	Annotation
	side	xs:string	required			

source	`<xs:element name="antireflection" minOccurs="0">`
	`<xs:complexType>`
	`<xs:simpleContent>`
	`<xs:extension base="xs:string">`
	`<xs:attribute name="side" use="required">`
	`<xs:simpleType>`
	`<xs:restriction base="xs:string">`
	`<xs:enumeration value="both"/>`
	`<xs:enumeration value="front"/>`
	`<xs:enumeration value="back"/>`
	`</xs:restriction>`
	`</xs:simpleType>`
	`</xs:attribute>`
	`</xs:extension>`
	`</xs:simpleContent>`
	`</xs:complexType>`
	`</xs:element>`

element **optionsType/coating/hard**

diagram	
type	**xs:string**
source	`<xs:element name="hard" type="xs:string" minOccurs="0"/>`

element **optionsType/coating/tint**

diagram	 **tint** type | xs:string Umbra
type	extension of **xs:string**
attributes	Name	Type	Use	Default	Fixed	Annotation
	side	xs:string	optional			
annotation	documentation Umbra

source	`<xs:element name="tint">`
	`<xs:annotation>`
	`<xs:documentation>Umbra</xs:documentation>`
	`</xs:annotation>`
	`<xs:complexType>`
	`<xs:simpleContent>`
	`<xs:extension base="xs:string">`
	`<xs:attribute name="side" use="optional">`
	`<xs:simpleType>`
	`<xs:restriction base="xs:string">`
	`<xs:enumeration value="both"/>`
	`<xs:enumeration value="front"/>`
	`<xs:enumeration value="back"/>`
	`</xs:restriction>`
	`</xs:simpleType>`
	`</xs:attribute>`
	`</xs:extension>`
	`</xs:simpleContent>`
	`</xs:complexType>`
	`</xs:element>`

element **optionsType/coating/uv-protection**

diagram	 **uv-protection** type | xs:string
type	**xs:string**
source	`<xs:element name="uv-protection" type="xs:string"/>`

element **optionsType/thin**

diagram	 **thin** type | xs:boolean
type	**xs:boolean**
source	`<xs:element name="thin" type="xs:boolean" minOccurs="0"/>`

element **optionsType/flat**

diagram	 **flat** type | xs:string
type	restriction of **xs:string**
facets	enumeration flat

source	`<xs:element name="flat" minOccurs="0">`
	`<xs:simpleType>`
	`<xs:restriction base="xs:string">`
	`<xs:enumeration value="flat"/>`

```
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
```

element **optionsType/approximate**

diagram	**approximate** type | xs:integer 1 = Grundkurve 2 = Mittendicke 3 = Randdicke 4 = G + M 5 = G + R 6 = G + Dicke allge
type	extension of **xs:integer**

annotation	documentation 1 = Grundkurve
	                           2 = Mittendicke
	                           3 = Randdicke
	                           4 = G + M
	                           5 = G + R
	                           6 = G + Dicke allge

source	`<xs:element name="approximate" minOccurs="0">`
	  `<xs:annotation>`
	   `<xs:documentation>`1 = Grundkurve
	2 = Mittendicke
	3 = Randdicke
	4 = G + M
	5 = G + R
	6 = G + Dicke allge`</xs:documentation>`
	  `</xs:annotation>`
	  `<xs:complexType>`
	   `<xs:simpleContent>`
	    `<xs:extension base="xs:integer"/>`
	   `</xs:simpleContent>`
	  `</xs:complexType>`
	`</xs:element>`

element **optionsType/plano-concave**

diagram	**plano-concave** type | xs:boolean
type	**xs:boolean**
source	`<xs:element name="plano-concave" type="xs:boolean"/>`

element **optionsType/plano-convex**
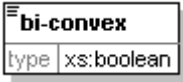
diagram	**plano-convex** type | xs:boolean
type	**xs:boolean**
source	`<xs:element name="plano-convex" type="xs:boolean"/>`

element **optionsType/bi-concave**

diagram	**bi-concave** type | xs:boolean
type	**xs:boolean**

source	<xs:element name="bi-concave" type="xs:boolean"/>

element **optionsType/bi-convex**

diagram	**bi-convex** type xs:boolean
type	**xs:boolean**
source	<xs:element name="bi-convex" type="xs:boolean"/>

element **optionsType/centre-thickness**

diagram	**centre-thickness** type xs:float Wunsch-Mittendicke	
type	restriction of **xs:float**	
facets	minExclusive	0.2
	maxExclusive	30.0
annotation	documentation	Wunsch-Mittendicke
source	<xs:element name="centre-thickness"> <xs:annotation> <xs:documentation>Wunsch-Mittendicke</xs:documentation> </xs:annotation> <xs:simpleType> <xs:restriction base="xs:float"> <xs:minExclusive value="0.2"/> <xs:maxExclusive value="30.0"/> </xs:restriction> </xs:simpleType> </xs:element>	

element **optionsType/edge-thickness**

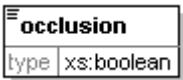|		
diagram	**edge-thickness** type xs:float Wunsch-Randdicke	
type	restriction of **xs:float**	
facets	minExclusive	0.2
	maxExclusive	30.0
annotation	documentation	Wunsch-Randdicke
source	<xs:element name="edge-thickness"> <xs:annotation> <xs:documentation>Wunsch-Randdicke</xs:documentation> </xs:annotation> <xs:simpleType> <xs:restriction base="xs:float"> <xs:minExclusive value="0.2"/> <xs:maxExclusive value="30.0"/> </xs:restriction> </xs:simpleType> </xs:element>	

element **optionsType/nylor**

diagram	**nylor** type xs:boolean Randdickenvorgabe als Flag
type	**xs:boolean**
annotation	documentation Randdickenvorgabe als Flag

source	`<xs:element name="nylor" type="xs:boolean">`
	`<xs:annotation>`
	`<xs:documentation>Randdickenvorgabe als Flag</xs:documentation>`
	`</xs:annotation>`
	`</xs:element>`

element **optionsType/toric**

diagram	**toric** type xs:string
type	restriction of **xs:string**
facets	enumeration front enumeration back

source	`<xs:element name="toric" default="back" minOccurs="0">`
	`<xs:simpleType>`
	`<xs:restriction base="xs:string">`
	`<xs:enumeration value="front"/>`
	`<xs:enumeration value="back"/>`
	`</xs:restriction>`
	`</xs:simpleType>`
	`</xs:element>`

element **optionsType/lenticular**

diagram	**lenticular** type xs:boolean
type	**xs:boolean**
source	`<xs:element name="lenticular" type="xs:boolean" minOccurs="0"/>`

element **optionsType/aniseicony**

diagram	**aniseicony** type xs:boolean					
type	extension of **xs:boolean**					
attributes	Name	Type	Use	Default	Fixed	Annotation
	value	xs:float	optional			

source	`<xs:element name="aniseicony" minOccurs="0">`
	`<xs:complexType>`
	`<xs:simpleContent>`
	`<xs:extension base="xs:boolean">`
	`<xs:attribute name="value" type="xs:float" use="optional"/>`
	`</xs:extension>`
	`</xs:simpleContent>`
	`</xs:complexType>`
	`</xs:element>`					

element **optionsType/slab-off**

diagram	
	``slab-off
type xs:boolean```
type	extension of **xs:boolean**
attributes	Name Type Use Default Fixed Annotation value xs:float optional

source	`<xs:element name="slab-off" minOccurs="0">`
	 `<xs:complexType>`
	  `<xs:simpleContent>`
	   `<xs:extension base="xs:boolean">`
	    `<xs:attribute name="value" use="optional">`
	     `<xs:simpleType>`
	      `<xs:restriction base="xs:float">`
	       `<xs:minInclusive value="1.3"/>`
	      `</xs:restriction>`
	     `</xs:simpleType>`
	    `</xs:attribute>`
	   `</xs:extension>`
	  `</xs:simpleContent>`
	 `</xs:complexType>`
	`</xs:element>`

element **optionsType/occlusion**

diagram	
	``occlusion
type xs:boolean```
type	**xs:boolean**
source	`<xs:element name="occlusion" type="xs:boolean"/>`

element **optionsType/frosted**

diagram	
	``frosted
type xs:boolean
mattieren```
type	**xs:boolean**
annotation	documentation mattieren

source	`<xs:element name="frosted" type="xs:boolean">`
	 `<xs:annotation>`
	  `<xs:documentation>mattieren</xs:documentation>`
	 `</xs:annotation>`
	`</xs:element>`

complexType **preCalcType**

diagram	
	edge-thickness-demo type: xs:boolean Randdickenverlauf Bei consult default true
	focal-type type: xs:integer 1 = Einstaerkenglas 2 = Bifokalglas 3 = Trifokalglas 4 = Gleitsichtglas
	material-category type: xs:integer 0 = Silikat 1 = Kunststoff
	refractive-index type: xs:decimal
preCalcType	**refractive-index-type** type: xs:integer 1=1.501 2=1.5251 3=1.604 4=1.706 5=1.800 6=1.600 Kunst 7=1.664 8=1.8930 9=1.586 10=1.74 11=1.533
	surface-type type: xs:integer 1=normales Glas 2=Clet Hypal, Hypal 3=frei 4=Ueberfang 5=Aphal 6=Einstaerken asphaerisch 7=Lupenglas 8=Arbeitsplatzbrille(RD+Busi 9=Individual
	phototrophic type: xs:boolean
	diameter-type type: xs:integer 0=zentriert 1=75/80 bis 55/60,60,55,50 4=80E bis 55E
	density type: xs:float

children	**edge-thickness-demo** **focal-type** **material-category** **refractive-index** **refractive-index-type** **surface-type** **phototrophic** **diameter-type** **density**

source	`<xs:complexType name="preCalcType">`
	`<xs:sequence>`
	`<xs:element name="edge-thickness-demo" type="xs:boolean" minOccurs="0">`
	`<xs:annotation>`
	`<xs:documentation>`Randdickenverlauf
	Bei consult default true`</xs:documentation>`
	`</xs:annotation>`
	`</xs:element>`

```xml
<xs:element name="focal-type" minOccurs="0">
  <xs:annotation>
    <xs:documentation>1 = Einstaerkenglas
2 = Bifokalglas
3 = Trifokalglas
4 = Gleitsichtglas</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="1"/>
      <xs:maxInclusive value="4"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="material-category" minOccurs="0">
  <xs:annotation>
    <xs:documentation>0 = Silikat
1 = Kunststoff</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="1"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="refractive-index" type="xs:decimal" minOccurs="0"/>
<xs:element name="refractive-index-type" minOccurs="0">
  <xs:annotation>
    <xs:documentation>1=1.501   2=1.5251   3=1.604   4=1.706
5=1.800   6=1.600 Kunst
7=1.664   8=1.8930
9=1.586  10=1.74
11=1.533</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="1"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="surface-type" minOccurs="0">
  <xs:annotation>
    <xs:documentation>1=normales Glas
2=Clet Hypal, Hypal
3=frei
4=Ueberfang
5=Aphal
6=Einstaerken asphaerisch
7=Lupenglas
8=Arbeitsplatzbrille(RD+Busi
9=Individual</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="1"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="phototrophic" type="xs:boolean" minOccurs="0"/>
<xs:element name="diameter-type" minOccurs="0">
  <xs:annotation>
    <xs:documentation>0=zentriert
1=75/80 bis 55/60,60,55,50
4=80E bis 55E  </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="0"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="density" type="xs:float" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
```

element **preCalcType/edge-thickness-demo**

diagram	**edge-thickness-demo** type xs:boolean Randdickenverlauf Bei consult default true
type	**xs:boolean**
annotation	documentation Randdickenverlauf Bei consult default true

source	`<xs:element name="edge-thickness-demo" type="xs:boolean" minOccurs="0">`
	 `<xs:annotation>`
	  `<xs:documentation>`Randdickenverlauf
	Bei consult default true`</xs:documentation>`
	 `</xs:annotation>`
	`</xs:element>`

element **preCalcType/focal-type**

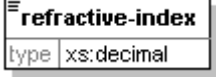|	
diagram	**focal-type** type xs:integer 1 = Einstaerkenglas 2 = Bifokalglas 3 = Trifokalglas 4 = Gleitsichtglas
type	restriction of **xs:integer**
facets	minInclusive 1 maxInclusive 4
annotation	documentation 1 = Einstaerkenglas 2 = Bifokalglas 3 = Trifokalglas 4 = Gleitsichtglas

source	`<xs:element name="focal-type" minOccurs="0">`
	 `<xs:annotation>`
	  `<xs:documentation>`1 = Einstaerkenglas
	2 = Bifokalglas
	3 = Trifokalglas
	4 = Gleitsichtglas`</xs:documentation>`
	 `</xs:annotation>`
	 `<xs:simpleType>`
	  `<xs:restriction base="xs:integer">`
	   `<xs:minInclusive value="1"/>`
	   `<xs:maxInclusive value="4"/>`
	  `</xs:restriction>`
	 `</xs:simpleType>`
	`</xs:element>`

element **preCalcType/material-category**

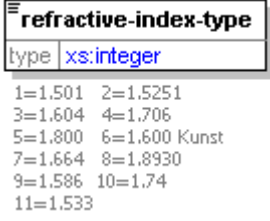|	
diagram	**material-category** type xs:integer 0 = Silikat 1 = Kunststoff
type	restriction of **xs:integer**
facets	minInclusive 0 maxInclusive 1
annotation	documentation 0 = Silikat 1 = Kunststoff

source	`<xs:element name="material-category" minOccurs="0">`
	 `<xs:annotation>`
	  `<xs:documentation>`0 = Silikat
	1 = Kunststoff`</xs:documentation>`
	 `</xs:annotation>`

```xml
    <xs:simpleType>
     <xs:restriction base="xs:integer">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="1"/>
     </xs:restriction>
    </xs:simpleType>
   </xs:element>
```

element **preCalcType/refractive-index**

diagram	**refractive-index** type | xs:decimal
type	**xs:decimal**
source	`<xs:element name="refractive-index" type="xs:decimal" minOccurs="0"/>`

element **preCalcType/refractive-index-type**

diagram	**refractive-index-type** type | xs:integer 1=1.501 2=1.5251 3=1.604 4=1.706 5=1.800 6=1.600 Kunst 7=1.664 8=1.8930 9=1.586 10=1.74 11=1.533
type	restriction of **xs:integer**
facets	minInclusive 1
annotation	documentation 1=1.501 2=1.5251 3=1.604 4=1.706 5=1.800 6=1.600 Kunst 7=1.664 8=1.8930 9=1.586 10=1.74 11=1.533

source	`<xs:element name="refractive-index-type" minOccurs="0">`
	 `<xs:annotation>`
	  `<xs:documentation>`1=1.501 2=1.5251 3=1.604 4=1.706
	5=1.800 6=1.600 Kunst
	7=1.664 8=1.8930
	9=1.586 10=1.74
	11=1.533`</xs:documentation>`
	 `</xs:annotation>`
	 `<xs:simpleType>`
	  `<xs:restriction base="xs:integer">`
	   `<xs:minInclusive value="1"/>`
	  `</xs:restriction>`
	 `</xs:simpleType>`
	`</xs:element>`

element **preCalcType/surface-type**
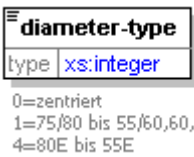
diagram	**surface-type** type | xs:integer 1=normales Glas 2=Clet Hypal, Hypal 3=frei 4=Ueberfang 5=Aphal 6=Einstaerken asphaerisch 7=Lupenglas 8=Arbeitsplatzbrille(RD+Busi 9=Individual
type	restriction of **xs:integer**

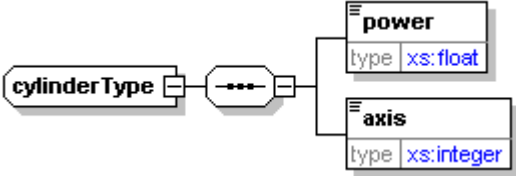|		
facets	minInclusive 1	
annotation	documentation	1=normales Glas 2=Clet Hypal, Hypal 3=frei 4=Ueberfang 5=Aphal 6=Einstaerken asphaerisch 7=Lupenglas 8=Arbeitsplatzbrille(RD+Busi 9=Individual
source	<xs:element name="surface-type" minOccurs="0"> <xs:annotation> <xs:documentation>1=normales Glas 2=Clet Hypal, Hypal 3=frei 4=Ueberfang 5=Aphal 6=Einstaerken asphaerisch 7=Lupenglas 8=Arbeitsplatzbrille(RD+Busi 9=Individual</xs:documentation> </xs:annotation> <xs:simpleType> <xs:restriction base="xs:integer"> <xs:minInclusive value="1"/> </xs:restriction> </xs:simpleType> </xs:element>	

element **preCalcType/phototrophic**

diagram	**phototrophic** type xs:boolean
type	**xs:boolean**
source	<xs:element name="phototrophic" type="xs:boolean" minOccurs="0"/>

element **preCalcType/diameter-type**

diagram	**diameter-type** type xs:integer 0=zentriert 1=75/80 bis 55/60,60,55,50 4=80E bis 55E	
type	restriction of **xs:integer**	
facets	minInclusive 0	
annotation	documentation	0=zentriert 1=75/80 bis 55/60,60,55,50 4=80E bis 55E
source	<xs:element name="diameter-type" minOccurs="0"> <xs:annotation> <xs:documentation>0=zentriert 1=75/80 bis 55/60,60,55,50 4=80E bis 55E </xs:documentation> </xs:annotation> <xs:simpleType> <xs:restriction base="xs:integer"> <xs:minInclusive value="0"/> </xs:restriction> </xs:simpleType> </xs:element>	

element **preCalcType/density**

diagram	
type	**xs:float**
source	`<xs:element name="density" type="xs:float" minOccurs="0"/>`

complexType **cylinderType**

diagram	
children	**power axis**
used by	element **refractionType/cylinder**

source	`<xs:complexType name="cylinderType">`
	`  <xs:sequence>`
	`   <xs:element name="power">`
	`    <xs:simpleType>`
	`     <xs:restriction base="xs:float">`
	`      <xs:minInclusive value="-30"/>`
	`      <xs:maxInclusive value="30"/>`
	`     </xs:restriction>`
	`    </xs:simpleType>`
	`   </xs:element>`
	`   <xs:element name="axis">`
	`    <xs:simpleType>`
	`     <xs:restriction base="xs:integer">`
	`      <xs:minInclusive value="0"/>`
	`      <xs:maxInclusive value="180"/>`
	`     </xs:restriction>`
	`    </xs:simpleType>`
	`   </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

element **cylinderType/power**

diagram		
type	restriction of **xs:float**	
facets	minInclusive	-30
	maxInclusive	30

source	`<xs:element name="power">`
	`  <xs:simpleType>`
	`   <xs:restriction base="xs:float">`
	`    <xs:minInclusive value="-30"/>`
	`    <xs:maxInclusive value="30"/>`
	`   </xs:restriction>`
	`  </xs:simpleType>`
	`</xs:element>`	

element **cylinderType/axis**

diagram	
type	restriction of **xs:integer**
facets	minInclusive 0 maxInclusive 180

source	`<xs:element name="axis">`
	 `<xs:simpleType>`
	  `<xs:restriction base="xs:integer">`
	   `<xs:minInclusive value="0"/>`
	   `<xs:maxInclusive value="180"/>`
	  `</xs:restriction>`
	 `</xs:simpleType>`
	`</xs:element>`
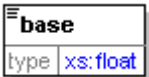
complexType **prismType**

diagram	
children	**power base**
used by	element **refractionType/prism**

source	`<xs:complexType name="prismType">`
	 `<xs:sequence>`
	  `<xs:element name="power" type="xs:float"/>`
	  `<xs:element name="base">`
	   `<xs:simpleType>`
	    `<xs:restriction base="xs:float">`
	     `<xs:minInclusive value="0.0"/>`
	     `<xs:maxInclusive value="360.0"/>`
	    `</xs:restriction>`
	   `</xs:simpleType>`
	  `</xs:element>`
	 `</xs:sequence>`
	`</xs:complexType>`

element **prismType/power**

diagram	
type	**xs:float**
source	`<xs:element name="power" type="xs:float"/>`

element **prismType/base**

diagram	
type	restriction of **xs:float**
facets	minInclusive 0.0 maxInclusive 360.0
source	`<xs:element name="base">`

```
    <xs:simpleType>
     <xs:restriction base="xs:float">
      <xs:minInclusive value="0.0"/>
      <xs:maxInclusive value="360.0"/>
     </xs:restriction>
    </xs:simpleType>
   </xs:element>
```

XML Schema documentation generated with **XML Spy** Schema Editor **www.xmlspy.com**