



Published on **XML.com** <http://www.xml.com/pub/a/2000/11/29/schemas/structuresref.html>  
 See [this](#) if you're having trouble printing code examples

## W3C XML Schema Structures Reference

By Eric van der Vlist

### xs:all

Compositor describing an unordered group of elements.

```
<xs:all
  id                = xs:ID
  maxOccurs         = "1" ; "1"
  minOccurs         = ( "0" | "1" ) : "1"
  {any attributes with non-schema namespace}
  >
  Content: (xs:annotation?, xs:element *)
```

</xs:all>

May be included in: [xs:complexType](#) (reference or local definition), [xs:complexType](#) (global definition), [xs:extension](#) (complex content), [xs:restriction](#) (complex content)

### xs:all (within a group)

Compositor describing an unordered group of elements. The number of occurrences cannot be defined when xs:all is used within a group.

```
<xs:all
  id                = xs:ID
  {any attributes with non-schema namespace}
  >
  Content: (xs:annotation?, xs:element *)
```

</xs:all>

May be included in: [xs:group](#) (definition)

### xs:annotation

Informative data for human or electronic agents.

```
<xs:annotation
  id                = xs:ID
  {any attributes with non-schema namespace}
  >
  Content: (xs:appinfo | xs:documentation)*
```

</xs:annotation>

May be included in: [xs:all](#), [xs:all](#) (within a group), [xs:any](#), [xs:anyAttribute](#), [xs:attribute](#) (reference or local definition), [xs:attribute](#) (global definition), [xs:attributeGroup](#) (reference or local definition), [xs:attributeGroup](#) (global definition), [xs:choice](#), [xs:choice](#) (within a group), [xs:complexContent](#), [xs:complexType](#) (reference or local definition), [xs:complexType](#) (global definition), [xs:element](#) (within xs:all), [xs:element](#) (reference or local definition), [xs:element](#) (global definition), [xs:enumeration](#), [xs:extension](#) (complex content), [xs:extension](#) (simple content), [xs:field](#), [xs:fractionDigits](#), [xs:group](#) (reference), [xs:group](#) (definition), [xs:import](#), [xs:include](#), [xs:key](#), [xs:keyref](#), [xs:length](#), [xs:list](#), [xs:maxExclusive](#), [xs:maxInclusive](#), [xs:maxLength](#), [xs:minExclusive](#), [xs:minInclusive](#), [xs:minLength](#), [xs:notation](#), [xs:pattern](#), [xs:redefine](#), [xs:restriction](#) (complex content), [xs:restriction](#) (simple type), [xs:restriction](#) (simple content), [xs:schema](#), [xs:selector](#), [xs:sequence](#) (within a group), [xs:sequence](#), [xs:simpleContent](#), [xs:simpleType](#) (reference or local definition), [xs:simpleType](#) (global definition), [xs:totalDigits](#), [xs:union](#), [xs:unique](#), [xs:whiteSpace](#)

### xs:any

Wildcard to replace any element.

```
<xs:any
  id                = xs:ID
  maxOccurs         = ( xs:nonNegativeInteger | "unbounded" ) : "1"
  minOccurs         = xs:nonNegativeInteger : "1"
  namespace         = ( ( "##any" | "##other" ) | list of ( xs:anyURI |
    ( "##targetNamespace" | "##local" ) ) ) : "##any"
  processContents   = ( "skip" | "lax" | "strict" ) : "strict"
  {any attributes with non-schema namespace}
  >
  Content: (xs:annotation?)
```

</xs:any>

May be included in: [xs:choice](#), [xs:choice](#) (within a group), [xs:sequence](#) (within a group), [xs:sequence](#)

### xs:anyAttribute

Wildcard to replace any attribute.

```
<xs:anyAttribute
  id                = xs:ID
  namespace         = ( ( "##any" | "##other" ) | list of ( xs:anyURI |
    ( "##targetNamespace" | "##local" ) ) ) : "##any"
  processContents   = ( "skip" | "lax" | "strict" ) : "strict"
  {any attributes with non-schema namespace}
  >
  Content: (xs:annotation?)
```

</xs:anyAttribute>

May be included in: [xs:attributeGroup](#) (global definition), [xs:complexType](#) (reference or local definition), [xs:complexType](#) (global definition), [xs:extension](#) (complex

content), xs:extension (simple content), xs:restriction (complex content), xs:restriction (simple content)

**xs:appinfo**

Information for applications.

```
<xs:appinfo
    source = xs:anyURI
>
Content: ({any})*
```

&lt;/xs:appinfo&gt;

May be included in: xs:annotation

**xs:attribute (reference or local definition)**

Reference to a global attribute declaration or local definition (local definitions cannot be referenced).

```
<xs:attribute
    default                = xs:string
    fixed                  = xs:string
    form                   = ( "qualified" | "unqualified" )
    id                     = xs:ID
    name                   = xs:NCName
    ref                    = xs:QName
    type                   = xs:QName
    use                    = ( "prohibited" | "optional" | "required" ) :
                           "optional"

    {any attributes with non-schema namespace}
>
Content: ((xs:annotation?), ( xs:simpleType ?))
```

&lt;/xs:attribute&gt;

May be included in: [xs:attributeGroup](#) (global definition), [xs:complexType](#) (reference or local definition), [xs:complexType](#) (global definition), [xs:extension](#) (complex content), [xs:extension](#) (simple content), [xs:restriction](#) (complex content), [xs:restriction](#) (simple content)

**xs:attribute (global definition)**

Global attribute declaration which can be referenced within the same schema or by other schemas.

<code>&lt;xs:attribute</code>		
	<code>default</code>	<code>= xs:string</code>
	<code>fixed</code>	<code>= xs:string</code>
	<code>id</code>	<code>= xs:ID</code>
	<code>name</code>	<code>= xs:NCName</code>
	<code>type</code>	<code>= xs:QName</code>
	<code>{any attributes with non-schema namespace}</code>	

>  
Content: (xs:annotation?, *xs:simpleType* ?)

&lt;/xs:attribute&gt;

May be included in: xs:schema

**xs:attributeGroup** (reference or local definition)

Reference to a global attributes group declaration or local definition (local definitions cannot be referenced).

```
<xs:attributeGroup
    id                = xs:ID
    ref               = xs:QName
    {any attributes with non-schema namespace}
>
Content: (xs:annotation?)
```

&lt;/xs:attributeGroup&gt;

May be included in: [xs:attributeGroup](#) (global definition), [xs:complexType](#) (reference or local definition), [xs:complexType](#) (global definition), [xs:extension](#) (complex content), [xs:extension](#) (simple content), [xs:restriction](#) (complex content), [xs:restriction](#) (simple content)

**xs:attributeGroup** (global definition)

Global attributes group declaration which can be referenced within the same schema or by other schemas.

```
<xs:attributeGroup
  id                    = xs:ID
  name                  = xs:NCName
  {any attributes with non-schema namespace}
>
Content: (xs:annotation?, ((xs:attribute | xs:attributeGroup)*,
xs:anyAttribute?))
```

&lt;/xs:attributeGroup&gt;

May be included in: xs:redefine, xs:schema

**xs:choice**

Compositor to define group of mutually exclusive elements or compositors.

```
<xs:choice
    id                    = xs:ID
    maxOccurs             = ( xs:nonNegativeInteger | "unbounded" ) : "1"
    minOccurs             = xs:nonNegativeInteger : "1"
    {any attributes with non-schema namespace}
>
Content: (xs:annotation?, ( xs:element | xs:group | xs:choice | xs:sequence |
xs:any)*)
```

</xs:choice>  
May be included in: [xs:choice](#), [xs:choice \(within a group\)](#), [xs:complexType \(reference or local definition\)](#), [xs:complexType \(global definition\)](#), [xs:extension \(complex content\)](#), [xs:restriction \(complex content\)](#), [xs:sequence \(within a group\)](#), [xs:sequence](#)

**xs:choice (within a group)**  
Compositor to define group of mutually exclusive elements or compositors. The number of occurrences cannot be defined when xs:choice is used within a group.  
<xs:choice  
    id = xs:ID  
    {any attributes with non-schema namespace}  
    >  
    Content: ([xs:annotation?](#), ( [xs:element](#) | [xs:group](#) | [xs:choice](#) | [xs:sequence](#) | [xs:any](#) )\*)  
</xs:choice>  
May be included in: [xs:group \(definition\)](#)

**xs:complexContent**  
Derivation of a simple type to complex content.  
<xs:complexContent  
    id = xs:ID  
    mixed = xs:boolean  
    {any attributes with non-schema namespace}  
    >  
    Content: (([xs:annotation?](#)), ( [xs:restriction](#) | [xs:extension](#) ))  
</xs:complexContent>  
May be included in: [xs:complexType \(reference or local definition\)](#), [xs:complexType \(global definition\)](#)

**xs:complexType (reference or local definition)**  
Reference to a global complex type declaration or local definition (local definitions cannot be referenced).  
<xs:complexType  
    id = xs:ID  
    mixed = xs:boolean : "false"  
    {any attributes with non-schema namespace}  
    >  
    Content: ([xs:annotation?](#), ([xs:simpleContent](#) | [xs:complexContent](#) | (( [xs:group](#) | [xs:all](#) | [xs:choice](#) | [xs:sequence](#) )?, (( [xs:attribute](#) | [xs:attributeGroup](#) )\*, [xs:anyAttribute?](#)))))  
</xs:complexType>  
May be included in: [xs:element \(within xs:all\)](#), [xs:element \(reference or local definition\)](#), [xs:element](#)

(global definition)

---

**xs:complexType (global definition)**  
Global declaration of a complex type which can be referenced within the same schema of by other schemas.  
<xs:complexType  
    abstract = xs:boolean : "false"  
    block = ( "#all" | list of ( "extension" | "restriction" ) )  
    final = ( "#all" | list of ( "extension" | "restriction" ) )  
    id = xs:ID  
    mixed = xs:boolean : "false"  
    name = xs:NCName  
    {any attributes with non-schema namespace}  
    >  
    Content: ([xs:annotation?](#), ([xs:simpleContent](#) | [xs:complexContent](#) | (( [xs:group](#) | [xs:all](#) | [xs:choice](#) | [xs:sequence](#) )?, (( [xs:attribute](#) | [xs:attributeGroup](#) )\*, [xs:anyAttribute?](#)))))  
</xs:complexType>  
May be included in: [xs:redefine](#), [xs:schema](#)

**xs:documentation**  
Human targeted documentation.  
<xs:documentation  
    source = xs:anyURI  
    xml:lang = xml:lang  
    >  
    Content: ({any})\*  
</xs:documentation>  
May be included in: [xs:annotation](#)

**xs:element (within xs:all)**  
Reference to a global element declaration or local definition (local definitions cannot be referenced). The number of occurrences can only be 0 or 1 when xs:element is used within xs:all..  
<xs:element  
    block = ( "#all" | list of ( "extension" | "restriction" | "substitution" ) )  
    default = xs:string  
    fixed = xs:string  
    form = ( "qualified" | "unqualified" )  
    id = xs:ID  
    maxOccurs = ( "0" | "1" ) : "1"  
    minOccurs = ( "0" | "1" ) : "1"

XML.com: W3C XML Schema Structures Reference	XML.com: W3C XML Schema Structures Reference
Page 7 of 19	Page 8 of 19
<div> <div> <div>name</div> <div>= xs:NCName</div> </div> <div> <div>nillable</div> <div>= xs:boolean : "false"</div> </div> <div> <div>ref</div> <div>= xs:QName</div> </div> <div> <div>type</div> <div>= xs:QName</div> </div> <div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> </div> <div> <div>Content: (xs:annotation?, ( <u>xs:simpleType</u>   <u>xs:complexType</u> )?, (xs:unique   xs:key   xs:keyref)*)</div> </div> <div>&lt;/xs:element&gt;</div> <div>May be included in: <u>xs:all</u>, <u>xs:all (within a group)</u></div> </div>	<div> <div> <div>fixed</div> <div>= xs:string</div> </div> <div> <div>id</div> <div>= xs:ID</div> </div> <div> <div>name</div> <div>= xs:NCName</div> </div> <div> <div>nillable</div> <div>= xs:boolean : "false"</div> </div> <div> <div>substitutionGroup</div> <div>= xs:QName</div> </div> <div> <div>type</div> <div>= xs:QName</div> </div> <div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> </div> <div> <div>Content: (xs:annotation?, ( <u>xs:simpleType</u>   <u>xs:complexType</u> )?, (xs:unique   xs:key   xs:keyref)*)</div> </div> <div>&lt;/xs:element&gt;</div> <div>May be included in: <u>xs:schema</u></div> </div>
<div> <div> <div><b>xs:element (reference or local definition)</b></div> <div>Reference to a global element declaration or local definition (local definitions cannot be referenced).</div> </div> <div> <div>&lt;xs:element</div> <div> <div> <div>block</div> <div>= ( "#all"   list of ( "extension"   "restriction"   "substitution" ) )</div> </div> <div> <div>default</div> <div>= xs:string</div> </div> <div> <div>fixed</div> <div>= xs:string</div> </div> <div> <div>form</div> <div>= ( "qualified"   "unqualified" )</div> </div> <div> <div>id</div> <div>= xs:ID</div> </div> <div> <div>maxOccurs</div> <div>= ( xs:nonNegativeInteger   "unbounded" ) : "1"</div> </div> <div> <div>minOccurs</div> <div>= xs:nonNegativeInteger : "1"</div> </div> <div> <div>name</div> <div>= xs:NCName</div> </div> <div> <div>nillable</div> <div>= xs:boolean : "false"</div> </div> <div> <div>ref</div> <div>= xs:QName</div> </div> <div> <div>type</div> <div>= xs:QName</div> </div> <div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> </div> <div> <div>Content: (xs:annotation?, ( <u>xs:simpleType</u>   <u>xs:complexType</u> )?, (xs:unique   xs:key   xs:keyref)*)</div> </div> <div>&lt;/xs:element&gt;</div> <div>May be included in: <u>xs:choice</u>, <u>xs:choice (within a group)</u>, <u>xs:sequence (within a group)</u>, <u>xs:sequence</u></div> </div> </div> </div>	<div> <div> <div><b>xs:enumeration</b></div> <div>Facet to restrict a datatype to a finite set of values.</div> </div> <div> <div>&lt;xs:enumeration</div> <div> <div> <div>id</div> <div>= xs:ID</div> </div> <div> <div>value</div> <div>= anySimpleType</div> </div> <div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> </div> <div> <div>Content: (xs:annotation?)</div> </div> <div>&lt;/xs:enumeration&gt;</div> <div>May be included in: <u>xs:restriction (simple type)</u>, <u>xs:restriction (simple content)</u></div> </div> </div> </div>
<div> <div> <div><b>xs:element (global definition)</b></div> <div>Global element declaration which can be referenced within the same schema or by other schemas.</div> </div> <div> <div>&lt;xs:element</div> <div> <div> <div>abstract</div> <div>= xs:boolean : "false"</div> </div> <div> <div>block</div> <div>= ( "#all"   list of ( "extension"   "restriction"   "substitution" ) )</div> </div> <div> <div>default</div> <div>= xs:string</div> </div> <div> <div>final</div> <div>= ( "#all"   list of ( "extension"   "restriction" ) )</div> </div> </div> </div> </div>	<div> <div> <div><b>xs:extension (complex content)</b></div> <div>Extension of a complex content model.</div> </div> <div> <div>&lt;xs:extension</div> <div> <div> <div>base</div> <div>= xs:QName</div> </div> <div> <div>id</div> <div>= xs:ID</div> </div> <div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> </div> <div> <div>Content: ((xs:annotation?), (( <u>xs:group</u>   <u>xs:all</u>   <u>xs:choice</u>   <u>xs:sequence</u> )?, (( <u>xs:attribute</u>   <u>xs:attributeGroup</u> )*, xs:anyAttribute?)))</div> </div> <div>&lt;/xs:extension&gt;</div> <div>May be included in: <u>xs:complexContent</u></div> </div> </div> </div>
<div> <div> <div><b>xs:extension (simple content)</b></div> <div>Extension of a simple content model.</div> </div> <div> <div>&lt;xs:extension</div> <div> <div> <div>base</div> <div>= xs:QName</div> </div> <div> <div>id</div> <div>= xs:ID</div> </div> <div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> </div> </div> </div> </div>	
http://www.xml.com/lpt/a/2000/11/29/schemas/structuresref.html	http://www.xml.com/lpt/a/2000/11/29/schemas/structuresref.html
4/13/2003	4/13/2003

XML.com: W3C XML Schema Structures Reference

Page 9 of 19

>

Content: (xs:annotation?, (( xs:attribute | xs:attributeGroup )\*, xs:anyAttribute?))

</xs:extension>

May be included in: xs:simpleContent

xs:field

Definition of the field to be used for a uniqueness constraint.

<xs:field

id = xs:ID

xpath = xs:token

{any attributes with non-schema namespace}

>

Content: (xs:annotation?)

</xs:field>

May be included in: xs:key, xs:keyref, xs:unique

xs:fractionDigits

Facet to define the number of fractional digits of a numerical datatype.

<xs:fractionDigits

fixed = xs:boolean : "false"

id = xs:ID

value = xs:nonNegativeInteger

{any attributes with non-schema namespace}

>

Content: (xs:annotation?)

</xs:fractionDigits>

May be included in: xs:restriction (simple type), xs:restriction (simple content)

xs:group (reference)

Reference to a global elements group declaration or local definition (local definitions cannot be referenced).

<xs:group

id = xs:ID

maxOccurs = ( xs:nonNegativeInteger | "unbounded" ) : "1"

minOccurs = xs:nonNegativeInteger : "1"

ref = xs:QName

{any attributes with non-schema namespace}

>

Content: (xs:annotation?)

</xs:group>

May be included in: xs:choice, xs:choice (within a group), xs:complexType (reference or local

XML.com: W3C XML Schema Structures Reference

Page 10 of 19

definition), xs:complexType (global definition), xs:extension (complex content), xs:restriction (complex content), xs:sequence (within a group), xs:sequence

xs:group (definition)

Global elements group declaration which can be referenced within the same schema of by other schemas.

<xs:group

maxOccurs = anySimpleType

minOccurs = anySimpleType

name = xs:NCName

ref = anySimpleType

>

Content: (xs:annotation?, ( xs:all | xs:choice | xs:sequence ))

</xs:group>

May be included in: xs:redefine, xs:schema

xs:import

Import of a W3C XML Schema for another namespace.

<xs:import

id = xs:ID

namespace = xs:anyURI

schemaLocation = xs:anyURI

{any attributes with non-schema namespace}

>

Content: (xs:annotation?)

</xs:import>

May be included in: xs:schema

xs:include

Inclusion of a W3C XML Schema for the same target namespace.

<xs:include

id = xs:ID

schemaLocation = xs:anyURI

{any attributes with non-schema namespace}

>

Content: (xs:annotation?)

</xs:include>

May be included in: xs:schema

xs:key

Definition of a key.

http://www.xml.com/lpt/a/2000/11/29/schemas/structuresref.html

4/13/2003

http://www.xml.com/lpt/a/2000/11/29/schemas/structuresref.html

4/13/2003

XML.com: W3C XML Schema Structures Reference Page 11 of 19	XML.com: W3C XML Schema Structures Reference Page 12 of 19
<div> <div> <div>&lt;xs:key</div> <div> <div>id</div> <div>= xs:ID</div> </div> <div> <div>name</div> <div>= xs:NCName</div> </div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> <div>Content: ((<a href="#">xs:annotation?</a>), (<a href="#">xs:selector</a>, <a href="#">xs:field+</a>))</div> </div> <div>&lt;/xs:key&gt;</div> <div>May be included in: <a href="#">xs:element (within xs:all)</a>, <a href="#">xs:element (reference or local definition)</a>, <a href="#">xs:element (global definition)</a></div> </div> <hr/> <div> <div> <div><b>xs:keyref</b></div> <div>Definition of a key reference.</div> </div> <div> <div>&lt;xs:keyref</div> <div> <div>id</div> <div>= xs:ID</div> </div> <div> <div>name</div> <div>= xs:NCName</div> </div> <div> <div>refer</div> <div>= xs:QName</div> </div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> <div>Content: ((<a href="#">xs:annotation?</a>), (<a href="#">xs:selector</a>, <a href="#">xs:field+</a>))</div> </div> <div>&lt;/xs:keyref&gt;</div> <div>May be included in: <a href="#">xs:element (within xs:all)</a>, <a href="#">xs:element (reference or local definition)</a>, <a href="#">xs:element (global definition)</a></div> </div> <hr/> <div> <div> <div><b>xs:length</b></div> <div>Facet to define the length of a value.</div> </div> <div> <div>&lt;xs:length</div> <div> <div>fixed</div> <div>= xs:boolean : "false"</div> </div> <div> <div>id</div> <div>= xs:ID</div> </div> <div> <div>value</div> <div>= xs:nonNegativeInteger</div> </div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> <div>Content: (<a href="#">xs:annotation?</a>)</div> </div> <div>&lt;/xs:length&gt;</div> <div>May be included in: <a href="#">xs:restriction (simple type)</a>, <a href="#">xs:restriction (simple content)</a></div> </div> <hr/> <div> <div> <div><b>xs:list</b></div> <div>Derivation by list.</div> </div> <div> <div>&lt;xs:list</div> <div> <div>id</div> <div>= xs:ID</div> </div> <div> <div>itemType</div> <div>= xs:QName</div> </div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> </div> </div>	<div> <div>Content: ((<a href="#">xs:annotation?</a>), (<a href="#">xs:simpleType ?</a>))</div> <div>&lt;/xs:list&gt;</div> <div>May be included in: <a href="#">xs:simpleType (reference or local definition)</a>, <a href="#">xs:simpleType (global definition)</a></div> </div> <hr/> <div> <div> <div><b>xs:maxExclusive</b></div> <div>Facet to define a maximum (exclusive) value.</div> </div> <div> <div>&lt;xs:maxExclusive</div> <div> <div>fixed</div> <div>= xs:boolean : "false"</div> </div> <div> <div>id</div> <div>= xs:ID</div> </div> <div> <div>value</div> <div>= anySimpleType</div> </div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> <div>Content: (<a href="#">xs:annotation?</a>)</div> </div> <div>&lt;/xs:maxExclusive&gt;</div> <div>May be included in: <a href="#">xs:restriction (simple type)</a>, <a href="#">xs:restriction (simple content)</a></div> </div> <hr/> <div> <div> <div><b>xs:maxInclusive</b></div> <div>Facet to define a maximum (inclusive) value.</div> </div> <div> <div>&lt;xs:maxInclusive</div> <div> <div>fixed</div> <div>= xs:boolean : "false"</div> </div> <div> <div>id</div> <div>= xs:ID</div> </div> <div> <div>value</div> <div>= anySimpleType</div> </div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> <div>Content: (<a href="#">xs:annotation?</a>)</div> </div> <div>&lt;/xs:maxInclusive&gt;</div> <div>May be included in: <a href="#">xs:restriction (simple type)</a>, <a href="#">xs:restriction (simple content)</a></div> </div> <hr/> <div> <div> <div><b>xs:maxLength</b></div> <div>Facet to define a maximum length.</div> </div> <div> <div>&lt;xs:maxLength</div> <div> <div>fixed</div> <div>= xs:boolean : "false"</div> </div> <div> <div>id</div> <div>= xs:ID</div> </div> <div> <div>value</div> <div>= xs:nonNegativeInteger</div> </div> <div>{any attributes with non-schema namespace}</div> <div>&gt;</div> <div>Content: (<a href="#">xs:annotation?</a>)</div> </div> <div>&lt;/xs:maxLength&gt;</div> <div>May be included in: <a href="#">xs:restriction (simple type)</a>, <a href="#">xs:restriction (simple content)</a></div> </div> <hr/> <div> <div> <div><b>xs:minExclusive</b></div> </div> </div>
<a href="http://www.xml.com/lpt/a/2000/11/29/schemas/structuresref.html">http://www.xml.com/lpt/a/2000/11/29/schemas/structuresref.html</a> 4/13/2003	<a href="http://www.xml.com/lpt/a/2000/11/29/schemas/structuresref.html">http://www.xml.com/lpt/a/2000/11/29/schemas/structuresref.html</a> 4/13/2003

XML.com: W3C XML Schema Structures Reference Page 13 of 19	XML.com: W3C XML Schema Structures Reference Page 14 of 19
<p>Facet to define a minimum (exclusive) value.</p> <pre> &lt;xs:minExclusive     fixed                = xs:boolean : "false"     id                   = xs:ID     value                = anySimpleType     {any attributes with non-schema namespace}     &gt;     Content: (<a href="#">xs:annotation?</a>) &lt;/xs:minExclusive&gt; May be included in:  <a href="#">xs:restriction (simple type)</a>, <a href="#">xs:restriction (simple content)</a> </pre> <hr/> <p><b>xs:minInclusive</b> Facet to define a minimum (inclusive) value.</p> <pre> &lt;xs:minInclusive     fixed                = xs:boolean : "false"     id                   = xs:ID     value                = anySimpleType     {any attributes with non-schema namespace}     &gt;     Content: (<a href="#">xs:annotation?</a>) &lt;/xs:minInclusive&gt; May be included in:  <a href="#">xs:restriction (simple type)</a>, <a href="#">xs:restriction (simple content)</a> </pre> <hr/> <p><b>xs:minLength</b> Facet to define a minimum length.</p> <pre> &lt;xs:minLength     fixed                = xs:boolean : "false"     id                   = xs:ID     value                = xs:nonNegativeInteger     {any attributes with non-schema namespace}     &gt;     Content: (<a href="#">xs:annotation?</a>) &lt;/xs:minLength&gt; May be included in:  <a href="#">xs:restriction (simple type)</a>, <a href="#">xs:restriction (simple content)</a> </pre> <hr/> <p><b>xs:notation</b> Declaration of a notation.</p> <pre> &lt;xs:notation     id                   = xs:ID     name                 = xs:NCName     public               = xs:boolean </pre>	<pre>     system               = xs:anyURI     {any attributes with non-schema namespace}     &gt;     Content: (<a href="#">xs:annotation?</a>) &lt;/xs:notation&gt; May be included in:  <a href="#">xs:schema</a> </pre> <hr/> <p><b>xs:pattern</b> Facet to define a regular expression pattern constraint.</p> <pre> &lt;xs:pattern     id                   = xs:ID     value                = anySimpleType     {any attributes with non-schema namespace}     &gt;     Content: (<a href="#">xs:annotation?</a>) &lt;/xs:pattern&gt; May be included in:  <a href="#">xs:restriction (simple type)</a>, <a href="#">xs:restriction (simple content)</a> </pre> <hr/> <p><b>xs:redefine</b> Import of a W3C XML Schema for the same namespace with possible override.</p> <pre> &lt;xs:redefine     id                   = xs:ID     schemaLocation       = xs:anyURI     {any attributes with non-schema namespace}     &gt;     Content: (<a href="#">xs:annotation</a>   <a href="#">xs:simpleType</a>   <a href="#">xs:complexType</a>   <a href="#">xs:group</a>   <a href="#">xs:attributeGroup</a>)* &lt;/xs:redefine&gt; May be included in:  <a href="#">xs:schema</a> </pre> <hr/> <p><b>xs:restriction (complex content)</b> Derivation of a complex content model by restriction.</p> <pre> &lt;xs:restriction     base                 = xs:QName     id                   = xs:ID     {any attributes with non-schema namespace}     &gt;     Content: (<a href="#">xs:annotation?</a>, ( <a href="#">xs:group</a>   <a href="#">xs:all</a>   <a href="#">xs:choice</a>   <a href="#">xs:sequence</a> )?,     (( <a href="#">xs:attribute</a>   <a href="#">xs:attributeGroup</a> )*, <a href="#">xs:anyAttribute?</a>)) &lt;/xs:restriction&gt; May be included in:  <a href="#">xs:complexContent</a> </pre> <hr/>
<a href="http://www.xml.com/lpt/a/2000/11/29/schemas/structuresref.html">http://www.xml.com/lpt/a/2000/11/29/schemas/structuresref.html</a> 4/13/2003	<a href="http://www.xml.com/lpt/a/2000/11/29/schemas/structuresref.html">http://www.xml.com/lpt/a/2000/11/29/schemas/structuresref.html</a> 4/13/2003

**xs:restriction (simple type)**

Derivation of a simple datatype by restriction.

&lt;xs:restriction

base = xs:QName  
 id = xs:ID  
 {any attributes with non-schema namespace}  
 >

Content: (([xs:annotation?](#)), ( [xs:simpleType ?](#), ([xs:minExclusive](#) | [xs:minInclusive](#) | [xs:maxExclusive](#) | [xs:maxInclusive](#) | [xs:totalDigits](#) | [xs:fractionDigits](#) | [xs:length](#) | [xs:minLength](#) | [xs:maxLength](#) | [xs:enumeration](#) | [xs:whiteSpace](#) | [xs:pattern](#))\*))

&lt;/xs:restriction&gt;

May be included in: [xs:simpleType](#) (reference or local definition), [xs:simpleType](#) (global definition)**xs:restriction (simple content)**

Derivation of a simple content model by restriction.

&lt;xs:restriction

base = xs:QName  
 id = xs:ID  
 {any attributes with non-schema namespace}  
 >

Content: ([xs:annotation?](#), ( [xs:simpleType ?](#), ([xs:minExclusive](#) | [xs:minInclusive](#) | [xs:maxExclusive](#) | [xs:maxInclusive](#) | [xs:totalDigits](#) | [xs:fractionDigits](#) | [xs:length](#) | [xs:minLength](#) | [xs:maxLength](#) | [xs:enumeration](#) | [xs:whiteSpace](#) | [xs:pattern](#))\*?), (( [xs:attribute](#) | [xs:attributeGroup](#) )\*, [xs:anyAttribute?](#)))

&lt;/xs:restriction&gt;

May be included in: [xs:simpleContent](#)**xs:schema**

Document element of a W3C XML Schema.

&lt;xs:schema

attributeFormDefault = ( "qualified" | "unqualified" ) : "unqualified"  
 blockDefault = ( "#all" | list of ( "extension" | "restriction" | "substitution" ) ) : ""  
 elementFormDefault = ( "qualified" | "unqualified" ) : "unqualified"  
 finalDefault = ( "#all" | list of ( "extension" | "restriction" ) ) : ""  
 id = xs:ID  
 targetNamespace = xs:anyURI  
 version = xs:token  
 xml:lang = xml:lang  
 {any attributes with non-schema namespace}  
 >  
 Content: (([xs:include](#) | [xs:import](#) | [xs:redefine](#) | [xs:annotation](#))\*,

(([xs:simpleType](#) | [xs:complexType](#) | [xs:group](#) | [xs:attributeGroup](#)) | [xs:element](#) | [xs:attribute](#) | [xs:notation](#)), [xs:annotation](#))\*))

&lt;/xs:schema&gt;

May be included in:

**xs:selector**

Definition of the the path selecting an element for a uniqueness constraint.

&lt;xs:selector

id = xs:ID  
 xpath = xs:token  
 {any attributes with non-schema namespace}  
 >  
 Content: ([xs:annotation?](#))

&lt;/xs:selector&gt;

May be included in: [xs:key](#), [xs:keyref](#), [xs:unique](#)**xs:sequence (within a group)**

Compositor to define an ordered group of elements. The number of occurrences cannot be defined when xs:all is used within a group.

&lt;xs:sequence

id = xs:ID  
 {any attributes with non-schema namespace}  
 >  
 Content: ([xs:annotation?](#), ( [xs:element](#) | [xs:group](#) | [xs:choice](#) | [xs:sequence](#) | [xs:any](#))\*))

&lt;/xs:sequence&gt;

May be included in: [xs:group](#) (definition)**xs:sequence**

Compositor to define an ordered group of elements.

&lt;xs:sequence

id = xs:ID  
 maxOccurs = ( [xs:nonNegativeInteger](#) | "unbounded" ) : "1"  
 minOccurs = [xs:nonNegativeInteger](#) : "1"  
 {any attributes with non-schema namespace}  
 >  
 Content: ([xs:annotation?](#), ( [xs:element](#) | [xs:group](#) | [xs:choice](#) | [xs:sequence](#) | [xs:any](#))\*))

&lt;/xs:sequence&gt;

May be included in: [xs:choice](#), [xs:choice](#) (within a group), [xs:complexType](#) (reference or local definition), [xs:complexType](#) (global definition), [xs:extension](#) (complex content), [xs:restriction](#) (complex content), [xs:sequence](#) (within a group), [xs:sequence](#)



**xs:simpleContent**

Simple content model declaration.

```
<xs:simpleContent
  id                    = xs:ID
  {any attributes with non-schema namespace}
>
  Content: ((xs:annotation?), ( xs:restriction | xs:extension ))
</xs:simpleContent>
May be included in: xs:complexType (reference or local definition), xs:complexType (global definition)
```

**xs:simpleType (reference or local definition)**

Reference to a global simple type declaration or local definition (local definitions cannot be referenced).

```
<xs:simpleType
  id                    = xs:ID
  {any attributes with non-schema namespace}
>
  Content: (xs:annotation?, (xs:restriction | xs:list | xs:union))
</xs:simpleType>
May be included in: xs:attribute (reference or local definition), xs:attribute (global definition),
xs:element (within xs:all), xs:element (reference or local definition), xs:element
(global definition), xs:list, xs:restriction (simple type), xs:restriction (simple content), xs:union
```

**xs:simpleType (global definition)**

Global simple type declaration which can be referenced within the same schema of by other schemas.

```
<xs:simpleType
  final                = ( "#all" | ( "list" | "union" | "restriction" ) )
  id                   = xs:ID
  name                 = xs:NCName
  {any attributes with non-schema namespace}
>
  Content: (xs:annotation?, (xs:restriction | xs:list | xs:union))
</xs:simpleType>
May be included in: xs:redefine, xs:schema
```

**xs:totalDigits**

Facet to define the total number of digits of a numeric datatype.

```
<xs:totalDigits
  fixed                = xs:boolean : "false"
```

```
  id                    = xs:ID
  value                = xs:positiveInteger
  {any attributes with non-schema namespace}
>
  Content: (xs:annotation?)
</xs:totalDigits>
May be included in: xs:restriction (simple type), xs:restriction (simple content)
```

**xs:union**

Derivation of simple datatypes by union.

```
<xs:union
  id                    = xs:ID
  memberTypes          = list of xs:QName
  {any attributes with non-schema namespace}
>
  Content: ((xs:annotation?), ( xs:simpleType *))
</xs:union>
May be included in: xs:simpleType (reference or local definition), xs:simpleType (global definition)
```

**xs:unique**

Definition of a uniqueness constraint.

```
<xs:unique
  id                    = xs:ID
  name                 = xs:NCName
  {any attributes with non-schema namespace}
>
  Content: ((xs:annotation?), (xs:selector, xs:field+))
</xs:unique>
May be included in: xs:element (within xs:all), xs:element (reference or local definition), xs:element
(global definition)
```

**xs:whiteSpace**

Facet to define the white space behavior.

```
<xs:whiteSpace
  fixed                = xs:boolean : "false"
  id                   = xs:ID
  value                = ( "preserve" | "replace" | "collapse" )
  {any attributes with non-schema namespace}
>
  Content: (xs:annotation?)
</xs:whiteSpace>
```

May be included in: [xs:restriction \(simple type\)](#), [xs:restriction \(simple content\)](#)

---

**XML.com** Copyright © 1998-2003 O'Reilly & Associates, Inc.