

Element

Maps to UML Attribute or UML AssociationEnd with stereotype XSDelement.

- annotation – to UML Attribute or UML AssociationEnd documentation.
- default - to initial UML Attribute or UML AssociationEnd value.
- maxOccurs - to multiplicity upper range. Value unbounded maps to asterisk in UML.
- minOccurs – to multiplicity lower range.
- name – to UML Attribute or UML AssociationEnd name.
- type or content (simpleType | complexType) – to UML Attribute or UML AssociationEnd type.

Other properties maps to corresponding tagged values.

XML representation summary as element Element Information Item

```
<element
  abstract = boolean : false
  block = (#all | List of (extension | restriction | substitution))
  default = string
  final = (#all | List of (extension | restriction))
  fixed = string
  form = (qualified | unqualified) id = ID
  maxOccurs = (nonNegativeInteger | unbounded) : 1
  minOccurs = nonNegativeInteger : 1
  name = NCName
  nillable = boolean : false
  ref = oName
  substitutionGroup = Oname type = OName
  {any attributes with non-schema namespace...}>
  Content: (annotation?, ((simpleType | complexType)?, (unique | key | keyr
</element>
```

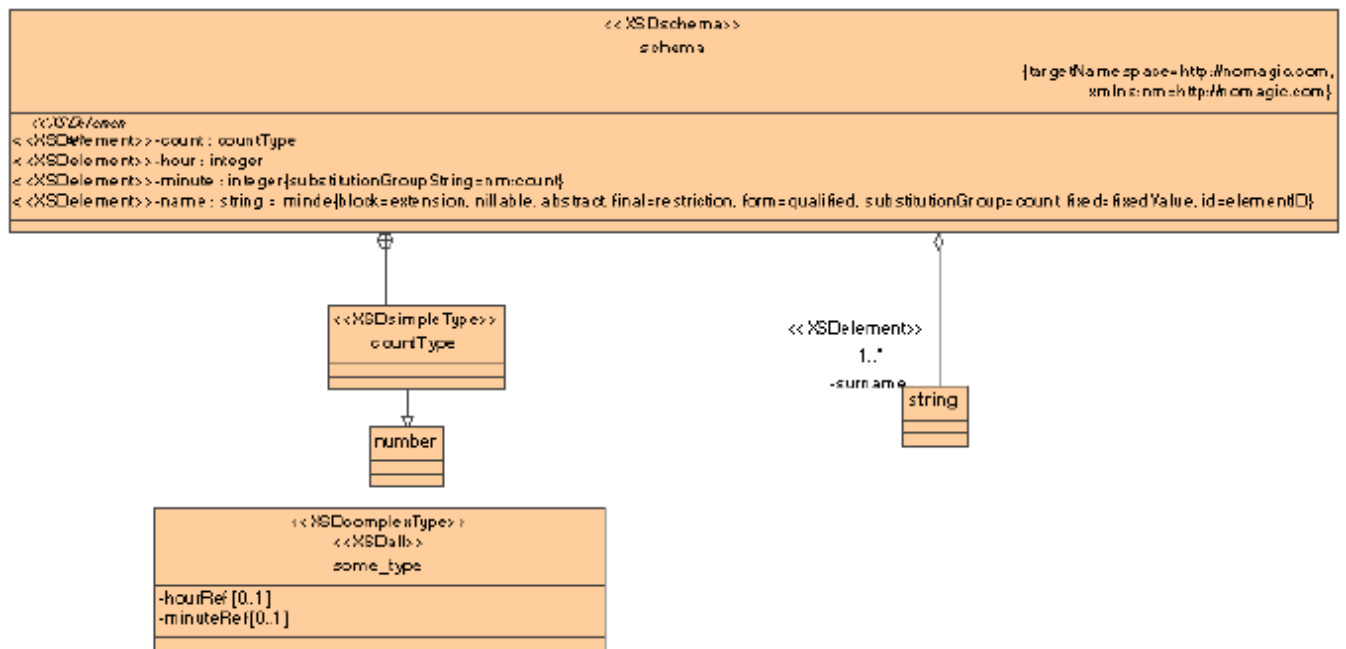
ref value is generated from ref or refString TaggedValue. One of ref or name must be present, but not both.

If ref is present, then all of <complexType>, <simpleType>, <key>, <keyref>, <unique>, nillable, default, fixed, form, block and type must be absent, i.e. only minOccurs, maxOccurs, and id are allowed in addition to ref, along with <annotation>.

XML representation summary as element Element Information Item (Continued)

```
<xs:element name = "PurchaseOrder" type = "PurchaseOrderType"/>

<xs:element name = "gift">
  <xs:complexType>
    <xs:sequence>
      <xs:element name = "birthday" type = "xs:date"/>
      <xs:element ref = "PurchaseOrder"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```



Example of element UML Model.

XML representation summary as element Element Information Item (Continued)

```
<xs:schema xmlns:nm = "http://nomagic.com" xmlns:xs =
"http://www.w3.org/2001/XMLSchema" targetNamespace = "http://nomagic.com" >

  <xs:element name = "name" type = "xs:string" default = "minde" id = "elementID"
abstract = "true" block = "extension" final = "restriction" fixed = "fixedValue" form =
"qualified" nillable = "true" substitutionGroup = "nm:count" >

    <xs:annotation >

        <xs:documentation >element name documentation</xs:documentation>
    </xs:annotation>

</xs:element>

<xs:element name = "count" >

    <xs:annotation >

        <xs:documentation >element count documentation</xs:documentation>
    </xs:annotation>

    <xs:simpleType>

        <xs:restriction base = "xs:number" />

    </xs:simpleType>
</xs:element>

<xs:element name = "hour" type = "xs:integer" />

<xs:element name = "minute" type = "xs:integer" substitutionGroup "nm:count" />

<xs:element name = "surname" type = "xs:string" minOccurs "1" maxOccurs = "unbounded" />

<xs:complexType name = "some_type" >

    <xs:all >

        <xs:element ref = "nm:hour" minOccurs = "0" maxOccurs = "1" >

            <xs:annotation >

                <xs:documentation >hour ref documentation</xs:documentation>
            </xs:annotation>

        </xs:element>

        <xs:element ref = "nm:minute" minOccurs = "0" maxOccurs = "1" />

    </xs:all>
</xs:complexType>
```