

XML Schema Documentation

Table of Contents

- [Schema Document Properties](#)
- [Global Declarations](#)
 - [Element: VATConsignment](#)
- [Global Definitions](#)
 - [Complex Type: Data_Type](#)
 - [Complex Type: VATDeclaration_Type](#)
 - [Complex Type: VATPeriod_Type](#)
 - [Simple Type: GridNumberCode](#)

[top](#)

Schema Document Properties

Target Namespace	http://www.minfin.fgov.be/VATConsignment
Version	1.0
Language	en
Element and Attribute Namespaces	<ul style="list-style-type: none">• Global element and attribute declarations belong to this schema's target namespace.• By default, local element declarations belong to this schema's target namespace.• By default, local attribute declarations have no namespace.
Schema Composition	<ul style="list-style-type: none">• This schema imports schema(s) from the following namespace(s):<ul style="list-style-type: none">◦ http://www.minfin.fgov.be/InputCommon (at IntervatInputCommon_v0_7.xsd)

Declared Namespaces

Prefix	Namespace
Default namespace	http://www.minfin.fgov.be/VATConsignment
xml	http://www.w3.org/XML/1998/namespace
xs	http://www.w3.org/2001/XMLSchema
common	http://www.minfin.fgov.be/InputCommon

Schema Component Representation

```
<xs:schema xml:lang="en" targetNamespace="http://www.minfin.fgov.be/VATConsignment"
  elementFormDefault="qualified" attributeFormDefault="unqualified" version="1.0">
  <xs:import namespace="http://www.minfin.fgov.be/InputCommon"
    schemaLocation="IntervatInputCommon_v0_7.xsd" />
  ...
</xs:schema>
```

[top](#)

Global Declarations

Element: VATConsignment

Name	VATConsignment
Type	Locally-defined complex type
Nilable	no
Abstract	no
Documentation	Envoi regroupé de déclarations TVA périodiques, déposé par un mandataire Gegroepeerde verzending van periodische btw aangiften, voorgelegd door een gevolmachtigde

XML Instance Representation

```
<VATConsignment
  VATDeclarationsNbr="xs:positiveInteger [1] ?">
  <Representative> common:Representative Type </Representative> [0..1] ?
  <RepresentativeReference> common:RepresentativeReference Type </RepresentativeReference>
  [0..1] ?
  <VATDeclaration> VATDeclaration Type </VATDeclaration> [1..*] ?
</VATConsignment>
```

Schema Component Representation

```
<xs:element name="VATConsignment">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Representative" type="common:Representative Type" minOccurs="0"/>
      <xs:element name="RepresentativeReference" type="common:RepresentativeReference Type"
        minOccurs="0"/>
      <xs:element name="VATDeclaration" type="VATDeclaration Type" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="VATDeclarationsNbr" type="xs:positiveInteger" use="required"/>
  </xs:complexType>
</xs:element>
```

[top](#)

Global Definitions

Complex Type: **Data_Type**

Super-types:	None
Sub-types:	None

Name	Data_Type
<u>Abstract</u>	no

XML Instance Representation

```
<...>
  <Amount
    GridNumber="GridNumberCode [1] ?"> [1..*] ?
    common:PositiveAmount Type
  </Amount>
</...>
```

Schema Component Representation

```
<xs:complexType name="Data_Type">
  <xs:sequence>
    <xs:element name="Amount" maxOccurs="unbounded">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="common:PositiveAmount Type"
            <xs:attribute name="GridNumber" type="GridNumberCode" use="required"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

[top](#)

Complex Type: **VATDeclaration_Type**

Super-types:	None
Sub-types:	None

Name	VATDeclaration_Type
<u>Abstract</u>	no

Documentation

Déclaration TVA périodique, pour un déclarant et une période

Periodieke aangifte, voor een aangever en een periode

XML Instance Representation

```
<...
  SequenceNumber="xs:positiveInteger [1] ?"
  DeclarantReference="common:DeclarantReference_Type [0..1] ?">
    <ReplacedVATDeclaration> common:IntervatDeclarationReference_Type </ReplacedVATDeclaration>
    [0..1] ?
    <Declarant> common:Declarant_Type </Declarant> [1] ?
    <Period> VATPeriod_Type </Period> [1] ?
    <Data> Data_Type </Data> [1] ?
    <ClientListingNihil> common:YesNo_Type </ClientListingNihil> [1] ?
    <Ask
      Restitution="common:YesNo_Type [1] ?"
      Payment="common:YesNo_Type [1] ?"/> [1] ?
    <FileAttachment> common:FileAttachment_Type </FileAttachment> [0..*] ?
    <Comment> common:Comment_Type </Comment> [0..1] ?
  </...>
```

Schema Component Representation

```
<xs:complexType name="VATDeclaration_Type">
  <xs:sequence>
    <xs:element name="ReplacedVATDeclaration" type="common:IntervatDeclarationReference_Type"
      minOccurs="0"/>
    <xs:element name="Declarant" type="common:Declarant_Type" />
    <xs:element name="Period" type="VATPeriod_Type" />
    <xs:element name="Data" type="Data_Type" />
    <xs:element name="ClientListingNihil" type="common:YesNo_Type" />
    <xs:element name="Ask">
      <xs:complexType>
        <xs:attribute name="Restitution" type="common:YesNo_Type" use="required"/>
        <xs:attribute name="Payment" type="common:YesNo_Type" use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="FileAttachment" type="common:FileAttachment_Type" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="Comment" type="common:Comment_Type" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="SequenceNumber" type="xs:positiveInteger" use="required"/>
  <xs:attribute name="DeclarantReference" type="common:DeclarantReference_Type" />
</xs:complexType>
```

[top](#)

Complex Type: VATPeriod_Type

Super-types:	None
Sub-types:	None

Name

VATPeriod_Type

Abstract

no

XML Instance Representation

```
<...>
  Start Choice [1]
    <Month> common:Month_Type </Month> [1] ?
    <Quarter> common:Quarter_Type </Quarter> [1] ?
  End Choice
  <Year> common:Year_Type </Year> [1] ?
</...>
```

Schema Component Representation

```
<xs:complexType name="VATPeriod_Type">
  <xs:sequence>
    <xs:choice>
      <xs:element name="Month" type="common:Month_Type" />
      <xs:element name="Quarter" type="common:Quarter_Type" />
    </xs:choice>
  </xs:sequence>
</xs:complexType>
```

```
<xs:element name="Year" type="common:Year Type"/>
</xs:sequence>
</xs:complexType>
```

[top](#)

Simple Type: **GridNumberCode**

Super-types: [xs:integer](#) < **GridNumberCode** (by restriction)

Sub-types: None

Name GridNumberCode

Content

- Base XSD Type: integer
- *value* comes from list:
{'0'|'1'|'2'|'3'|'44'|'45'|'46'|'47'|'48'|'49'|'54'|'55'|'56'|'57'|'59'|'61'|'62'|'63'|'64'|'71'|'72'|'81'|'82'|'83'|'84'|'85'|'86'|'87'|'88'|'91'}

Schema Component Representation

```
<xs:simpleType name="GridNumberCode">
  <xs:restriction base="xs:integer">
    <xs:enumeration value="0"/>
    <xs:enumeration value="1"/>
    <xs:enumeration value="2"/>
    <xs:enumeration value="3"/>
    <xs:enumeration value="44"/>
    <xs:enumeration value="45"/>
    <xs:enumeration value="46"/>
    <xs:enumeration value="47"/>
    <xs:enumeration value="48"/>
    <xs:enumeration value="49"/>
    <xs:enumeration value="54"/>
    <xs:enumeration value="55"/>
    <xs:enumeration value="56"/>
    <xs:enumeration value="57"/>
    <xs:enumeration value="59"/>
    <xs:enumeration value="61"/>
    <xs:enumeration value="62"/>
    <xs:enumeration value="63"/>
    <xs:enumeration value="64"/>
    <xs:enumeration value="71"/>
    <xs:enumeration value="72"/>
    <xs:enumeration value="81"/>
    <xs:enumeration value="82"/>
    <xs:enumeration value="83"/>
    <xs:enumeration value="84"/>
    <xs:enumeration value="85"/>
    <xs:enumeration value="86"/>
    <xs:enumeration value="87"/>
    <xs:enumeration value="88"/>
    <xs:enumeration value="91"/>
  </xs:restriction>
</xs:simpleType>
```

[top](#)