



Richtlinien fuer Lohndatenreceiver

Lohnstandard-CH (ELM) / SalaryDeclaration

Eine Gemeinschaftsarbeit der / En cooperation avec / In collaborazione con eAHV, SUVA,
SSK/Stv-BE, BFS und SVV.

Technical Schemadocumentation (TechDoc)

Filename: CommonTechDoc_de.pdf

Status: final

Schema: Common.xsd

Namespace: <http://www.swissdec.ch/schema/sd/20200220/Common>

Schemaversion: 20200220 / 0.0 (major / minor)

Generated with an adaption of xs3p (<http://titanium.dstc.edu.au/xml/xs3p/>)

Ausgabe / Edition 30.04.2020 17:55

Table of Contents

1. SchemaDocumentProperties.....	4
3. GlobalDefinitions.....	4
2. 33 Complex Type : AccessInformationType	4
2. 4 Complex Type : AddressExtensionType	5
2. 29 Complex Type : AddressType	5
2. 3 Complex Type : AddressWorkplaceType	6
2. 54 Complex Type : AnswerAmountType	6
2. 50 Complex Type : AnswerBooleanType	7
2. 52 Complex Type : AnswerDateTimeType	7
2. 51 Complex Type : AnswerDateType	7
2. 49 Complex Type : AnswerDoubleType	8
2. 48 Complex Type : AnswerIntegerType	8
2. 47 Complex Type : AnswerStringType	8
2. 53 Complex Type : AnswerYesNoUnknownType	9
2. 34 Complex Type : BalanceType	9
2. 32 Complex Type : ChangeCredentialsType	9
2. 98 Complex Type : CheckInteroperabilityResponseType	10
2. 97 Complex Type : CheckInteroperabilityType	10
2. 24 Complex Type : CivilStatusAndDateType	11
2. 1 Complex Type : CompanyDescriptionBaseType	11
2. 2 Complex Type : CompanyDescriptionType	12
2. 15 Complex Type : CompanyNameType	12
2. 16 Complex Type : CompanyOwnerType	13
2. 6 Complex Type : CompanyWorkingTimeIDType	13
2. 11 Complex Type : CompanyWorkingTimeRefType	14
2. 18 Complex Type : ContactInstitutionType	14
2. 31 Complex Type : CredentialsType	14
2. 19 Complex Type : DelegateType	15
2. 41 Complex Type : DialogMessagesType	15
2. 42 Complex Type : DialogMessageType	16
2. 25 Complex Type : EmptyType	16
2. 94 Complex Type : FaultStateType	17
2. 12 Complex Type : HoursRefType	17
2. 35 Complex Type : IncidentCaseIDType	17
2. 36 Complex Type : IncidentCaseIDWithMapType	18
2. 13 Complex Type : LessonsRefType	18
2. 104 Complex Type : MessageType	19
2. 28 Complex Type : NameAndAddressType	19
2. 87 Complex Type : NotificationsType	19
2. 88 Complex Type : NotificationType	20
2. 46 Complex Type : ParagraphAnswerType	20
2. 44 Complex Type : ParagraphType	21
2. 45 Complex Type : ParagraphValueType	21
2. 21 Complex Type : ParticularsBase2Type	22
2. 20 Complex Type : ParticularsBaseType	22
2. 22 Complex Type : ParticularsType	23
2. 101 Complex Type : PingConsumerResponseType	23
2. 99 Complex Type : PingConsumerType	24
2. 96 Complex Type : PingResponseType	24
2. 95 Complex Type : PingType	25
2. 102 Complex Type : PlannedMaintenanceType	25
2. 43 Complex Type : PreviousType	26
2. 103 Complex Type : ProducerMessagesType	26
2. 100 Complex Type : RegisteredMaintenanceType	26

2.55	Complex Type : SectionType	27
2.23	Complex Type : Social-InsuranceIdentificationType	27
2.40	Complex Type : StoryBaseType	28
2.37	Complex Type : TaxAtSourceCategoryType	28
2.27	Complex Type : TimePeriodType	28
2.17	Complex Type : UID-BFS-UnknownType	29
2.60	Complex Type : UserAgentType	29
2.10	Complex Type : WeeklyHoursAndLessonsIDType	30
2.14	Complex Type : WeeklyHoursAndLessonsRefType	30
2.9	Complex Type : WeeklyHoursAndLessonsType	30
2.7	Complex Type : WeeklyHoursIDType	31
2.8	Complex Type : WeeklyLessonsIDType	31
2.5	Complex Type : WorkplaceType	31
2.30	Complex Type : WorkType	32
2.86	Model Group : NotificationsGroup	33
2.59	Model Group : RequestContextBaseGroup	33
2.62	Simple Type : AssuranceCategoryCodeType	34
2.63	Simple Type : BUR-REE-NumberType	34
2.65	Simple Type : CantonAddressType	35
2.64	Simple Type : CantonAndEXType	36
2.38	Simple Type : CategoryPredefinedType	37
2.66	Simple Type : CivilStatusType	37
2.89	Simple Type : DescriptionCodeType	38
2.67	Simple Type : EmailAddressType	39
2.83	Simple Type : EmploymentContractType	39
2.26	Simple Type : EmptySimpleType	40
2.93	Simple Type : FaultCodeType	41
2.82	Simple Type : HoursOrLessonsType	41
2.68	Simple Type : IDType	41
2.69	Simple Type : InstanceRefIDType	42
2.70	Simple Type : LanguageCodeType	42
2.61	Simple Type : MonitoringIDType	43
2.71	Simple Type : MunicipalityIDType	43
2.72	Simple Type : NationalityType	44
2.73	Simple Type : NotEmptyStringType	44
2.74	Simple Type : PercentType	45
2.84	Simple Type : PositionType	45
2.90	Simple Type : QualityLevelType	46
2.75	Simple Type : ResidenceCategoryType	46
2.92	Simple Type : ResponseCodeAAwRType	47
2.91	Simple Type : ResponseCodeType	48
2.77	Simple Type : SalaryAmountAbsoluteType	48
2.76	Simple Type : SalaryAmountType	49
2.58	Simple Type : SectionIDType	49
2.78	Simple Type : SexType	50
2.79	Simple Type : SimpleBooleanType	50
2.56	Simple Type : StandardFormIDType	51
2.80	Simple Type : SV-AS-NumberType	51
2.39	Simple Type : TaxAtSourceCodeType	52
2.81	Simple Type : UID-BFSType	52
2.57	Simple Type : YesNoUnknownType	53
2.85	Simple Type : ZIP-CodeType	53

1. Schema Document Properties

Target Namespace	http://www.swissdec.ch/schema/common/20200220/Common
Version	0.0
Element and Attribute Namespaces	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.

Declared Namespaces

Prefix	Namespace
xml	http://www.w3.org/XML/1998/namespace
c	http://www.swissdec.ch/schema/common/20200220/Common
xs	http://www.w3.org/2001/XMLSchema

Schema Component Representation

```
<xs:schema targetNamespace="http://www.swissdec.ch/schema/common/20200220/Common"
  elementFormDefault="qualified" attributeFormDefault="unqualified" version="0.0">
  ...
</xs:schema>
```

3. Global Definitions

2.33 Complex Type: **AccessInformationType**

Name	AccessInformationType
------	-----------------------

Schema Component Representation

```
<xs:complexType name="AccessInformationType">
  <xs:sequence>
    <xs:element name="Url" type="xs:string"/>
    <xs:element name="ExpiryDate" type="xs:dateTime"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

Url	Translation: URL
ExpiryDate	Translation: Ablaufdatum

2.4 Complex Type: AddressExtensionType

Name	AddressExtensionType
------	----------------------

Schema Component Representation

```
<xs:complexType name="AddressExtensionType">
  <xs:complexContent>
    <xs:extension base="c:AddressWorkplaceType">
      <xs:sequence>
        <xs:element name="Canton" type="c:CantonAddressType" minOccurs="0"/>
        <xs:element name="MunicipalityID" type="c:MunicipalityIDType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Documentation: Schema Component Representation

2.29 Complex Type: AddressType

Name	AddressType
------	-------------

Schema Component Representation

```
<xs:complexType name="AddressType">
  <xs:sequence>
    <xs:element name="ComplementaryLine" type="xs:string" minOccurs="0"/>
    <xs:element name="Street" type="xs:string" minOccurs="0"/>
    <xs:element name="Postbox" type="xs:string" minOccurs="0"/>
    <xs:element name="Locality" type="xs:string" minOccurs="0"/>
    <xs:element name="ZIP-Code" type="c:ZIP-CodeType"/>
    <xs:element name="City" type="xs:string"/>
    <xs:element name="Country" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

ComplementaryLine	Translation: Zusatzzeile Short: Zusatzzeile für die Postadresse
Street	Translation: Strasse Short: Strasse und Hausnummer
Postbox	Translation: Postfach Short: Postfach
Locality	Translation: Lokalität Short: Lokalität wie Region, Provinz
ZIP-Code	Translation: Postleitzahl Short: Postleitzahl
City	Translation: Ort Short: Ort

Country	Translation: Land Short: Land
---------	--

2.3 Complex Type: **AddressWorkplaceType**

Name	AddressWorkplaceType
-------------	-----------------------------

Schema Component Representation

```
<xs:complexType name="AddressWorkplaceType">
  <xs:sequence>
    <xs:element name="ComplementaryLine" type="xs:string" minOccurs="0"/>
    <xs:element name="Street" type="xs:string" minOccurs="0"/>
    <xs:element name="Locality" type="xs:string" minOccurs="0"/>
    <xs:element name="ZIP-Code" type="c:ZIP-CodeType"/>
    <xs:element name="City" type="xs:string"/>
    <xs:element name="Country" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

ComplementaryLine	Translation: Zusatzzeile Short: Zusatzzeile für die Postadresse
Street	Translation: Strasse Short: Strasse und Hausnummer
Locality	Translation: Lokalität Short: Lokalität wie Region, Provinz
ZIP-Code	Translation: Postleitzahl Short: Postleitzahl
City	Translation: Ort Short: Ort
Country	Translation: Land Short: Land

2.54 Complex Type: **AnswerAmountType**

Name	AnswerAmountType
-------------	-------------------------

Schema Component Representation

```
<xs:complexType name="AnswerAmountType">
  <xs:sequence>
    <xs:element name="Default" type="c:SalaryAmountType" minOccurs="0"/>
    <xs:element name="Value" type="c:SalaryAmountType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

2.50 Complex Type: **AnswerBooleanType**

Name	AnswerBooleanType
------	-------------------

Schema Component Representation

```
<xs:complexType name="AnswerBooleanType">
  <xs:sequence>
    <xs:element name="Default" type="xs:boolean" minOccurs="0"/>
    <xs:element name="Value" type="xs:boolean" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

2.52 Complex Type: **AnswerDateTimeType**

Name	AnswerDateTimeType
------	--------------------

Schema Component Representation

```
<xs:complexType name="AnswerDateTimeType">
  <xs:sequence>
    <xs:element name="Default" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="Value" type="xs:dateTime" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

2.51 Complex Type: **AnswerDateType**

Name	AnswerDateType
------	----------------

Schema Component Representation

```
<xs:complexType name="AnswerDateType">
  <xs:sequence>
    <xs:element name="Default" type="xs:date" minOccurs="0"/>
    <xs:element name="Value" type="xs:date" minOccurs="0"/>
  </xs:sequence>
```

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

Documentation: Schema Component Representation

2.49 Complex Type: AnswerDoubleType

Name	AnswerDoubleType
------	------------------

Schema Component Representation

```
<xs:complexType name="AnswerDoubleType">
  <xs:sequence>
    <xs:element name="Default" type="xs:double" minOccurs="0"/>
    <xs:element name="Value" type="xs:double" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

2.48 Complex Type: AnswerIntegerType

Name	AnswerIntegerType
------	-------------------

Schema Component Representation

```
<xs:complexType name="AnswerIntegerType">
  <xs:sequence>
    <xs:element name="Default" type="xs:integer" minOccurs="0"/>
    <xs:element name="Value" type="xs:integer" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

2.47 Complex Type: AnswerStringType

Name	AnswerStringType
------	------------------

Schema Component Representation

```
<xs:complexType name="AnswerStringType">
  <xs:sequence>
    <xs:element name="Default" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```



```
<xs:element name="Value" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

2.53 Complex Type: AnswerYesNoUnknownType

Name	AnswerYesNoUnknownType
------	------------------------

Schema Component Representation

```
<xs:complexType name="AnswerYesNoUnknownType">
  <xs:sequence>
    <xs:element name="Default" type="c:YesNoUnknownType" minOccurs="0"/>
    <xs:element name="Value" type="c:YesNoUnknownType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

2.34 Complex Type: BalanceType

Name	BalanceType
------	-------------

Schema Component Representation

```
<xs:complexType name="BalanceType">
  <xs:choice>
    <xs:element name="FavourCompany" type="c:SalaryAmountAbsoluteType"/>
    <xs:element name="FavourInstitution" type="c:SalaryAmountAbsoluteType"/>
  </xs:choice>
</xs:complexType>
```

Documentation: Schema Component Representation

2.32 Complex Type: ChangeCredentialsType

Name	ChangeCredentialsType
------	-----------------------

Schema Component Representation

```
<xs:complexType name="ChangeCredentialsType">
```

```

<xs:sequence>
  <xs:element name="OldCredentials" type=" c:CredentialsType "/>
  <xs:element name="NewCredentials" type=" c:CredentialsType "/>
</xs:sequence>
</xs:complexType>

```

Documentation: Schema Component Representation

OldCredentials	Translation: alte Zugangsdaten
NewCredentials	Translation: neue Zugangsdaten

2.98 Complex Type: CheckInteroperabilityResponseType

Name	CheckInteroperabilityResponseType
------	-----------------------------------

Schema Component Representation

```

<xs:complexType name="CheckInteroperabilityResponseType">
  <xs:sequence>
    <xs:element name="UserAgent" type=" c:UserAgentType "/>
    <xs:element name="UmlautStringIsCorrect" type=" c:SimpleBooleanType "/>
    <xs:element name="FirstOperandIsCorrect" type=" c:SimpleBooleanType "/>
    <xs:element name="UmlautString" type=" xs:string "/>
    <xs:element name="AdditionResult" type=" c:SalaryAmountType "/>
    <xs:element name="SubtractionResult" type=" c:SalaryAmountType "/>
    <xs:element name="SystemDateTime" type=" xs:dateTime "/>
  </xs:sequence>
</xs:complexType>

```

Documentation: Schema Component Representation

UserAgent	Translation: Applikationsinformation Short: Beschreibung der wesentlichen System-Identifikations-Daten Technical: Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert
SystemDateTime	Translation: Aktuelle Systemzeit

2.97 Complex Type: CheckInteroperabilityType

Name	CheckInteroperabilityType
------	---------------------------

Schema Component Representation

```

<xs:complexType name="CheckInteroperabilityType">
  <xs:sequence>
    <xs:element name="UserAgent" type=" c:UserAgentType "/>
    <xs:element name="UmlautString" type=" xs:string "/>
    <xs:element name="FirstOperand" type=" c:SalaryAmountType "/>
    <xs:element name="SecondOperand" type=" c:SalaryAmountType "/>
    <xs:element name="SystemDateTime" type=" xs:dateTime "/>
  </xs:sequence>
</xs:complexType>

```

```
<xs:element name="MonitoringID" type=" c:MonitoringIDType " minOccurs="0"/>
</xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

UserAgent	Translation: Applikationsinformation Short: Beschreibung der wesentlichen System-Identifikations-Daten Technical: Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert
UmlautString	Translation: Anordnung von Umlauten Short: Anordnung von Umlauten zum Testen der gross klein Schreibung.
FirstOperand	Translation: 1. vorgegebene Testgrösse Short: 1. Operand wird mit 2. Operand zusammengerechnet zum testen, dass keine Datentyp Fehler passieren.
SecondOperand	Translation: 2. vorgegebene Testgrösse Short: 2. Operand wird mit 1. Operand zusammengerechnet zum testen, dass keine Datentyp Fehler passieren.
SystemDateTime	Translation: Aktuelle Systemzeit
MonitoringID	Translation: ÜberwachungsID Short: ÜberwachungsID Technical: Die ÜberwachungsID wird vorallem in der Referenzapplikation zur Aufteilung der Daten verwendet

2.24 Complex Type: **CivilStatusAndDateType**

Name	CivilStatusAndDateType
------	------------------------

Schema Component Representation

```
<xs:complexType name="CivilStatusAndDateType">
  <xs:sequence>
    <xs:element name="Status" type=" c:CivilStatusType "/>
    <xs:element name="ValidAsOf" type=" xs:date " minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

Status	Translation: Zivilstand Short: Zivilstand der Person
ValidAsOf	Translation: Zivilstand ist gültig ab Short: Zivilstand der Person ist gültig ab

2.1 Complex Type: **CompanyDescriptionBaseType**

Name	CompanyDescriptionBaseType
------	----------------------------

Schema Component Representation

```
<xs:complexType name="CompanyDescriptionBaseType">
  <xs:sequence>
    <xs:element name="Name" type="c:CompanyNameType"/>
    <xs:element name="Owner" type="c:CompanyOwnerType" minOccurs="0"/>
    <xs:element name="Address" type="c:AddressType"/>
    <xs:element name="UID-BFS" type="c:UID-BFS-UnknownType"/>
    <xs:element name="Delegate" type="c:DelegateType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

Name	Translation: Name Short: Name des Unternehmens
Owner	Translation: Firmeninhaber Short: Inhaber des Unternehmens
Address	Translation: Adresse Short: Adresse des Unternehmens
UID-BFS	Translation: Unternehmens UID-BFS Short: Unternehmens-Identifikationsnummer / UID-BFS
Delegate	Translation: Stellvertreter Short: Stellvertreter des Unternehmens

2.2 Complex Type: **CompanyDescriptionType**

Name	CompanyDescriptionType
-------------	-------------------------------

Schema Component Representation

```
<xs:complexType name="CompanyDescriptionType">
  <xs:complexContent>
    <xs:extension base="c:CompanyDescriptionBaseType">
      <xs:sequence>
        <xs:element name="Workplace" type="c:WorkplaceType" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Documentation: Schema Component Representation

2.15 Complex Type: **CompanyNameType**

Name	CompanyNameType
-------------	------------------------

Schema Component Representation

```
<xs:complexType name="CompanyNameType">
  <xs:sequence>
    <xs:element name="HR-RC-Name" type="xs:string"/>
    <xs:element name="ComplementaryLine" type="xs:string" minOccurs="0" maxOccurs="2"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

HR-RC-Name	Translation: Name oder UID-BFS/HR-Name Short: Name oder UID-BFS/HR-Name des Unternehmens
ComplementaryLine	Translation: Zusatzzeile Short: Zusatzzeilen für Abteilungsamen, Filialbezeichnungen usw.

2.16 Complex Type: **CompanyOwnerType**

Name	CompanyOwnerType
-------------	-------------------------

Schema Component Representation

```
<xs:complexType name="CompanyOwnerType">
  <xs:sequence>
    <xs:element name="Firstname" type="xs:string"/>
    <xs:element name="Lastname" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

Firstname	Translation: Vorname Short: Vorname des Firmeninhabers
Lastname	Translation: Nachname Short: Nachname des Firmeninhabers

2.6 Complex Type: **CompanyWorkingTimeIDType**

Name	CompanyWorkingTimeIDType
-------------	---------------------------------

Schema Component Representation

```
<xs:complexType name="CompanyWorkingTimeIDType">
  <xs:choice>
    <xs:element name="WeeklyHours" type="c:WeeklyHoursIDType"/>
    <xs:element name="WeeklyLessons" type="c:WeeklyLessonsIDType"/>
    <xs:element name="WeeklyHoursAndLessons" type="c:WeeklyHoursAndLessonsIDType"/>
  </xs:choice>
```

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

Documentation: Schema Component Representation

2.11 Complex Type: **CompanyWorkingTimeRefType**

Name	CompanyWorkingTimeRefType
------	---------------------------

Schema Component Representation

```
<xs:complexType name="CompanyWorkingTimeRefType">
  <xs:choice>
    <xs:element name="WeeklyHours" type="c:HoursRefType"/>
    <xs:element name="WeeklyLessons" type="c:LessonsRefType"/>
    <xs:element name="WeeklyHoursAndLessons" type="c:WeeklyHoursAndLessonsRefType"/>
  </xs:choice>
</xs:complexType>
```

Documentation: Schema Component Representation

2.18 Complex Type: **ContactInstitutionType**

Name	ContactInstitutionType
------	------------------------

Schema Component Representation

```
<xs:complexType name="ContactInstitutionType">
  <xs:sequence>
    <xs:element name="Name" type="c:NotEmptyStringType"/>
    <xs:element name="EmailAddress" type="c:EmailAddressType" minOccurs="0"/>
    <xs:element name="PhoneNumber" type="xs:string" minOccurs="0"/>
    <xs:element name="MobilePhoneNumber" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

Name	Translation: Name
EmailAddress	Translation: Email Adresse
PhoneNumber	Translation: Telefonnummer
MobilePhoneNumber	Translation: Handy Telefonnummer

2.31 Complex Type: **CredentialsType**

Name	CredentialsType
------	-----------------

Schema Component Representation

```
<xs:complexType name="CredentialsType">
  <xs:sequence>
    <xs:element name="Key" type="xs:string"/>
    <xs:element name="Password" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

Key	Translation: Schlüssel
Password	Translation: Passwort

2.19 Complex Type: **DelegateType**

Name	DelegateType
------	--------------

Schema Component Representation

```
<xs:complexType name="DelegateType">
  <xs:sequence>
    <xs:element name="Name" type="c:CompanyNameType"/>
    <xs:element name="Owner" type="c:CompanyOwnerType" minOccurs="0"/>
    <xs:element name="Address" type="c:AddressType"/>
    <xs:element name="UID-BFS" type="c:UID-BFSType"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

Name	Translation: Name Short: Name des Stellvertreters
Owner	Translation: Firmeninhaber Short: Firmeninhaber
Address	Translation: Adresse Short: Adresse des Unternehmens
UID-BFS	Translation: Unternehmens UID-BFS Short: Unternehmens-Identifikationsnummer / UID-BFS

2.41 Complex Type: **DialogMessagesType**

Name	DialogMessagesType
------	--------------------

Schema Component Representation

```
<xs:complexType name="DialogMessagesType">
```

```

<xs:sequence>
  <xs:element name="DialogMessage" type=" c:DialogMessageType " maxOccurs="unbounded">
    <xs:key name="SectionID-Key3">
      <xs:selector xpath="c:Section"/>
      <xs:field xpath="@sectionID"/>
    </xs:key>
    <xs:keyref name="" refer="c:SectionID-Key3">
      <xs:selector xpath="c:Paragraph"/>
      <xs:field xpath="@sectionIDRef"/>
    </xs:keyref>
  </xs:element>
</xs:sequence>
</xs:complexType>

```

Documentation: Schema Component Representation

DialogMessage

Translation: Dialog Message

Short: Einfaches, flexibles Datenaustauschformat

2.42 Complex Type: DialogMessageType

Name	DialogMessageType
------	-------------------

Schema Component Representation

```

<xs:complexType name="DialogMessageType">
  <xs:complexContent>
    <xs:extension base=" c:StoryBaseType ">
      <xs:sequence>
        <xs:element name="StandardDialogID" type=" c:StandardFormIDType "/>
        <xs:element name="Previous" type=" c:PreviousType " minOccurs="0"/>
        <xs:element name="Title" type=" c:IDType " minOccurs="0"/>
        <xs:element name="Description" type=" xs:token " minOccurs="0"/>
        <xs:element name="Paragraph" type=" c:ParagraphType " maxOccurs="unbounded"/>
        <xs:element name="Section" type=" c:SectionType " minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

Documentation: Schema Component Representation

2.25 Complex Type: EmptyType

Name	EmptyType
------	-----------

Schema Component Representation

```

<xs:complexType name="EmptyType"/>

```


Documentation: Schema Component Representation

2.94 Complex Type: **FaultStateType**

Name	FaultStateType
Documentation	Translation (de): Fehler Status

Schema Component Representation

```
<xs:complexType name="FaultStateType">
  <xs:sequence>
    <xs:element name="Code" type=" c:FaultCodeType "/>
    <xs:element name="Error" type=" c:NotificationsType " minOccurs="0"/>
    <xs:element name="Warning" type=" c:NotificationsType " minOccurs="0"/>
    <xs:element name="Info" type=" c:NotificationsType " minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

Code	Translation: Fehler-Status-Code
Error	Translation: Fehler
Warning	Translation: Warnungen
Info	Translation: Informationen

2.12 Complex Type: **HoursRefType**

Name	HoursRefType
-------------	---------------------

Schema Component Representation

```
<xs:complexType name="HoursRefType">
  <xs:simpleContent>
    <xs:extension base=" c:HoursOrLessonsType ">
      <xs:attribute name="companyWeeklyHoursIDRef" type=" c:InstanceRefIDType " use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Documentation: Schema Component Representation

2.35 Complex Type: **IncidentCaseIDType**

Name	IncidentCaseIDType
Documentation	Translation (de): IncidentCaseID

Schema Component Representation

```
<xs:complexType name="IncidentCaseIDType">
  <xs:simpleContent>
    <xs:extension base="c:IDType"/>
  </xs:simpleContent>
</xs:complexType>
```

Documentation: Schema Component Representation

2.36 Complex Type: IncidentCaseIDWithMapType

Name	IncidentCaseIDWithMapType
-------------	----------------------------------

Schema Component Representation

```
<xs:complexType name="IncidentCaseIDWithMapType">
  <xs:simpleContent>
    <xs:extension base="c:IncidentCaseIDType">
      <xs:attribute name="incidentCaseIDMap" type="c:InstanceRefIDType" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Documentation: Schema Component Representation

2.13 Complex Type: LessonsRefType

Name	LessonsRefType
-------------	-----------------------

Schema Component Representation

```
<xs:complexType name="LessonsRefType">
  <xs:simpleContent>
    <xs:extension base="c:HoursOrLessonsType">
      <xs:attribute name="companyWeeklyLessonsIDRef" type="c:InstanceRefIDType" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Documentation: Schema Component Representation

2.104 Complex Type: **MessageType**

Name	MessageType
------	-------------

Schema Component Representation

```
<xs:complexType name="MessageType">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="language" type="c:LanguageCodeType" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Documentation: Schema Component Representation

2.28 Complex Type: **NameAndAddressType**

Name	NameAndAddressType
------	--------------------

Schema Component Representation

```
<xs:complexType name="NameAndAddressType">
  <xs:sequence>
    <xs:element name="Name" type="xs:string"/>
    <xs:element name="Address" type="c:AddressType" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

Name	Translation: Name
Address	Translation: Adresse

2.87 Complex Type: **NotificationsType**

Name	NotificationsType
------	-------------------

Schema Component Representation

```
<xs:complexType name="NotificationsType">
  <xs:sequence>
    <xs:element name="Notification" type="c:NotificationType" maxOccurs="unbounded"/>
  </xs:sequence>
```

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

Documentation: Schema Component Representation

Notification

Translation: Hinweis

2.88 Complex Type: NotificationType

Name	NotificationType
------	------------------

Schema Component Representation

```
<xs:complexType name="NotificationType">
  <xs:sequence>
    <xs:element name="QualityLevel" type="c:QualityLevelType"/>
    <xs:element name="DescriptionCode" type="c:DescriptionCodeType"/>
    <xs:element name="Description" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

QualityLevel

Translation: Qualitäts-Stufe

DescriptionCode

Translation: Code der Beschreibung

Description

Translation: Beschreibungstext

2.46 Complex Type: ParagraphAnswerType

Name	ParagraphAnswerType
------	---------------------

Schema Component Representation

```
<xs:complexType name="ParagraphAnswerType">
  <xs:sequence>
    <xs:choice>
      <xs:element name="String" type="c:AnswerStringType"/>
      <xs:element name="Integer" type="c:AnswerIntegerType"/>
      <xs:element name="Double" type="c:AnswerDoubleType"/>
      <xs:element name="Boolean" type="c:AnswerBooleanType"/>
      <xs:element name="Date" type="c:AnswerDateType"/>
      <xs:element name="DateTime" type="c:AnswerDateTimeType"/>
      <xs:element name="YesNoUnknown" type="c:AnswerYesNoUnknownType"/>
      <xs:element name="Amount" type="c:AnswerAmountType"/>
    </xs:choice>
    <xs:element name="Problem" type="c:NotEmptyStringType" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="optional" type="c:EmptySimpleType" use="optional"/>
</xs:complexType>
```

Documentation: Schema Component Representation

Problem	Translation: Probleme Short: Probleme beim Beantworten mitteilen
optional	Translation: optionale Antwort Short: Kennzeichnung für eine optionale Antwort

2.44 Complex Type: ParagraphType

Name	ParagraphType
------	---------------

Schema Component Representation

```
<xs:complexType name="ParagraphType">
  <xs:sequence>
    <xs:element name="ID" type="xs:short"/>
    <xs:element name="Label" type="xs:token"/>
    <xs:choice>
      <xs:element name="Value" type="c:ParagraphValueType" minOccurs="0"/>
      <xs:element name="Answer" type="c:ParagraphAnswerType" minOccurs="0"/>
    </xs:choice>
  </xs:sequence>
  <xs:attribute name="sectionIDRef" type="c:SectionIDType"/>
</xs:complexType>
```

Documentation: Schema Component Representation

ID	Translation: Identifikation
sectionIDRef	Translation: Abschnitt Identifikation Referenz

2.45 Complex Type: ParagraphValueType

Name	ParagraphValueType
------	--------------------

Schema Component Representation

```
<xs:complexType name="ParagraphValueType">
  <xs:choice>
    <xs:element name="String" type="xs:string"/>
    <xs:element name="Integer" type="xs:integer"/>
    <xs:element name="Double" type="xs:double"/>
    <xs:element name="Boolean" type="xs:boolean"/>
    <xs:element name="Date" type="xs:date"/>
    <xs:element name="DateTime" type="xs:dateTime"/>
    <xs:element name="YesNoUnknown" type="c:YesNoUnknownType"/>
    <xs:element name="Amount" type="c:SalaryAmountType"/>
  </xs:choice>
</xs:complexType>
```

Documentation: Schema Component Representation

2.21 Complex Type: **ParticularsBase2Type**

Name	ParticularsBase2Type
------	----------------------

Schema Component Representation

```
<xs:complexType name="ParticularsBase2Type">
  <xs:complexContent>
    <xs:extension base="c:ParticularsBaseType">
      <xs:sequence>
        <xs:element name="Address" type="c:AddressType"/>
        <xs:element name="EmailAddress" type="c:EmailAddressType" minOccurs="0"/>
        <xs:element name="PhoneNumber" type="xs:string" minOccurs="0"/>
        <xs:element name="MobilePhoneNumber" type="xs:string" minOccurs="0"/>
        <xs:element name="ResidenceCanton" type="c:CantonAndEXType"/>
        <xs:element name="MunicipalityID" type="c:MunicipalityIDType" minOccurs="0"/>
        <xs:element name="ResidenceCategory" type="c:ResidenceCategoryType" minOccurs="0"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Documentation: Schema Component Representation

2.20 Complex Type: **ParticularsBaseType**

Name	ParticularsBaseType
------	---------------------

Schema Component Representation

```
<xs:complexType name="ParticularsBaseType">
  <xs:sequence>
    <xs:element name="Social-Insuranceldentification" type="c:Social-InsuranceldentificationType"/>
    <xs:element name="EmployeeNumber" type="xs:string" minOccurs="0"/>
    <xs:element name="Lastname" type="xs:string"/>
    <xs:element name="Firstname" type="xs:string"/>
    <xs:element name="Sex" type="c:SexType"/>
    <xs:element name="DateOfBirth" type="xs:date"/>
    <xs:element name="Nationality" type="c:NationalityType"/>
    <xs:element name="CivilStatus" type="c:CivilStatusAndDateType"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

Social-Insuranceldentification	Translation: Sozialversicherungsnummer Short: Sozialversicherungsnummer
EmployeeNumber	Translation: Personalnummer Short: Die Personalnummer ist in der Regel numerisch und vom Software-

	Hersteller bzw. dem Unternehmen frei wählbar (z.B. 1254)
Lastname	Translation: Nachname Short: Nachname
Firstname	Translation: Vorname Short: Vorname
Sex	Translation: Geschlecht
DateOfBirth	Translation: Geburtsdatum Short: Geburtsdatum der Person
Nationality	Translation: Staatsangehörigkeit Short: Die Codes für eine korrekte Zuteilung der Staatsangehörigkeit sind bei der UNO hinterlegt (ISO 3166) Technical: Erweiterung 11= unbekannt 22= staatenlos : Die ISO-Codes wurden auf dem Prod-Distri noch nie geprüft: In der Qualitätsstufe Plausibilität kann evtl. gegen die ISO 3166 Codes geprüft und eine Warnung angezeigt werden. Achtung: Probleme mit bestehenden Datenbeständen (Änderungen der Ländernamen und Verwechslungen)
CivilStatus	Translation: Zivilstandsangaben Short: Zivilstand der Person und gültig ab Datum

2.22 Complex Type: **ParticularsType**

Name	ParticularsType
------	-----------------

Schema Component Representation

```
<xs:complexType name="ParticularsType">
  <xs:complexContent>
    <xs:extension base="c:ParticularsBase2Type">
      <xs:sequence>
        <xs:element name="LanguageCode" type="c:LanguageCodeType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Documentation: Schema Component Representation

2.101 Complex Type: **PingConsumerResponseType**

Name	PingConsumerResponseType
------	--------------------------

Schema Component Representation

```
<xs:complexType name="PingConsumerResponseType">
  <xs:sequence>
    <xs:element name="UserAgent" type="c:UserAgentType"/>
    <xs:element name="Timestamp" type="xs:dateTime"/>
  </xs:sequence>
</xs:complexType>
```

```

<xs:choice minOccurs="0">
  <xs:element name="PlannedMaintenance" type=" c:PlannedMaintenanceType "/>
  <xs:element name="NoPlannedMaintenance" type=" c:EmptyType "/>
</xs:choice>
</xs:sequence>
</xs:complexType>

```

Documentation: Schema Component Representation

UserAgent

Translation: Applikationsinformation

Short: Beschreibung der wesentlichen System-Identifikations-Daten

Technical: Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert

2.99 Complex Type: PingConsumerType

Name

PingConsumerType

Schema Component Representation

```

<xs:complexType name="PingConsumerType">
  <xs:sequence>
    <xs:element name="UserAgent" type=" c:UserAgentType "/>
    <xs:element name="Timestamp" type=" xs:dateTime "/>
    <xs:element name="NextCheck" type=" xs:dateTime "/>
    <xs:element name="RegisteredMaintenance" type=" c:RegisteredMaintenanceType " minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

Documentation: Schema Component Representation

UserAgent

Translation: Applikationsinformation

Short: Beschreibung der wesentlichen System-Identifikations-Daten

Technical: Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert

Timestamp

Translation: Zeitstempel

Short: Ein Zeitstempel (englisch timestamp) wird benutzt, um einem Ereignis einen eindeutigen Zeitpunkt zuzuordnen.

NextCheck

Translation: nächste Überprüfung

RegisteredMaintenance

Translation: Registriertes Wartungsfenster

2.96 Complex Type: PingResponseType

Name

PingResponseType

Schema Component Representation

```

<xs:complexType name="PingResponseType">

```



```

<xs:sequence>
  <xs:element name="UserAgent" type=" c:UserAgentType "/>
  <xs:element name="SystemDateTime" type=" xs:dateTime "/>
</xs:sequence>
</xs:complexType>

```

Documentation: Schema Component Representation

UserAgent	Translation: Applikationsinformation Short: Beschreibung der wesentlichen System-Identifikations-Daten Technical: Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert
SystemDateTime	Translation: Aktuelle Systemzeit

2.95 Complex Type: PingType

Name	PingType
------	----------

Schema Component Representation

```

<xs:complexType name="PingType">
  <xs:sequence>
    <xs:element name="UserAgent" type=" c:UserAgentType "/>
    <xs:element name="SystemDateTime" type=" xs:dateTime "/>
    <xs:element name="MonitoringID" type=" c:MonitoringIDType " minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

```

Documentation: Schema Component Representation

UserAgent	Translation: Applikationsinformation Short: Beschreibung der wesentlichen System-Identifikations-Daten Technical: Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert
SystemDateTime	Translation: Aktuelle Systemzeit
MonitoringID	Translation: ÜberwachungsID Short: ÜberwachungsID Technical: Die ÜberwachungsID wird vorallem in der Referenzapplikation zur Aufteilung der Daten verwendet

2.102 Complex Type: PlannedMaintenanceType

Name	PlannedMaintenanceType
------	------------------------

Schema Component Representation

```

<xs:complexType name="PlannedMaintenanceType">
  <xs:sequence>

```

```

<xs:element name="Start" type="xs:dateTime"/>
<xs:element name="End" type="xs:dateTime"/>
<xs:element name="ProducerMessages" type="c:ProducerMessagesType">
  <xs:unique name="language">
    <xs:selector xpath="c:Message"/>
    <xs:field xpath="@language"/>
  </xs:unique>
</xs:element>
</xs:sequence>
</xs:complexType>

```

Documentation: Schema Component Representation

ProducerMessages

Translation: Nachricht für den Transmitter

2.43 Complex Type: PreviousType

Name	PreviousType
------	--------------

Schema Component Representation

```

<xs:complexType name="PreviousType">
  <xs:choice>
    <xs:element name="RequestStoryID" type="c:IDType"/>
    <xs:element name="ResponseStoryID" type="c:IDType"/>
  </xs:choice>
</xs:complexType>

```

Documentation: Schema Component Representation

2.103 Complex Type: ProducerMessagesType

Name	ProducerMessagesType
------	----------------------

Schema Component Representation

```

<xs:complexType name="ProducerMessagesType">
  <xs:sequence>
    <xs:element name="Message" type="c:MessageType" minOccurs="3" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

```

Documentation: Schema Component Representation

2.100 Complex Type: RegisteredMaintenanceType

Name	RegisteredMaintenanceType
-------------	----------------------------------

Schema Component Representation

```
<xs:complexType name="RegisteredMaintenanceType">
  <xs:sequence>
    <xs:element name="Start" type="xs:dateTime"/>
    <xs:element name="End" type="xs:dateTime"/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

2.55 Complex Type: **SectionType**

Name	SectionType
-------------	--------------------

Schema Component Representation

```
<xs:complexType name="SectionType">
  <xs:sequence>
    <xs:element name="Heading" type="c:IDType" minOccurs="0"/>
    <xs:element name="Description" type="xs:token" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="sectionID" type="c:SectionIDType" use="required"/>
</xs:complexType>
```

Documentation: Schema Component Representation

Heading	Translation: Überschrift
Description	Translation: Beschreibung
sectionID	Translation: Abschnitt Identifikation

2.23 Complex Type: **Social-InsuranceldentificationType**

Name	Social-InsuranceldentificationType
-------------	---

Schema Component Representation

```
<xs:complexType name="Social-InsuranceldentificationType">
  <xs:choice>
    <xs:element name="SV-AS-Number" type="c:SV-AS-NumberType"/>
    <xs:element name="unknown" type="c:EmptyType"/>
  </xs:choice>
</xs:complexType>
```

Documentation: Schema Component Representation

2.40 Complex Type: **StoryBaseType**

Name	StoryBaseType
------	---------------

Schema Component Representation

```
<xs:complexType name="StoryBaseType">
  <xs:sequence>
    <xs:element name="Creation" type=" xs:dateTime "/>
    <xs:element name="StoryID" type=" c:IDType "/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

Creation	Translation: Erstellung
StoryID	Translation: StoryID

2.37 Complex Type: **TaxAtSourceCategoryType**

Name	TaxAtSourceCategoryType
------	-------------------------

Schema Component Representation

```
<xs:complexType name="TaxAtSourceCategoryType">
  <xs:choice>
    <xs:element name="TaxAtSourceCode" type=" c:TaxAtSourceCodeType "/>
    <xs:element name="CategoryPredefined" type=" c:CategoryPredefinedType "/>
    <xs:element name="CategoryOpen" type=" c:IDType "/>
  </xs:choice>
</xs:complexType>
```

Documentation: Schema Component Representation

2.27 Complex Type: **TimePeriodType**

Name	TimePeriodType
------	----------------

Schema Component Representation

```
<xs:complexType name="TimePeriodType">
  <xs:sequence>
    <xs:element name="from" type=" xs:date "/>
    <xs:element name="until" type=" xs:date "/>
  </xs:sequence>
</xs:complexType>
```

```
</xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

from	Translation: von
until	Translation: bis

2.17 Complex Type: **UID-BFS-UnknownType**

Name	UID-BFS-UnknownType
-------------	----------------------------

Schema Component Representation

```
<xs:complexType name="UID-BFS-UnknownType">
  <xs:choice>
    <xs:element name="UID" type=" c:UID-BFSType "/>
    <xs:element name="Unknown" type=" c:EmptyType "/>
  </xs:choice>
</xs:complexType>
```

Documentation: Schema Component Representation

2.60 Complex Type: **UserAgentType**

Name	UserAgentType
-------------	----------------------

Schema Component Representation

```
<xs:complexType name="UserAgentType">
  <xs:sequence>
    <xs:element name="Producer" type=" c:NotEmptyStringType "/>
    <xs:element name="Name" type=" c:NotEmptyStringType "/>
    <xs:element name="Version" type=" c:NotEmptyStringType "/>
    <xs:element name="StandardVersion" type=" xs:decimal "/>
    <xs:element name="Certificate" type=" c:NotEmptyStringType "/>
  </xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

Producer	Translation: Applikations-Hersteller
Name	Translation: Produkt-Name
Version	Translation: Produkt-Version
StandardVersion	Translation: Swissdec Standard Version
Certificate	Translation: Zertifikat

2.10 Complex Type: **WeeklyHoursAndLessonsIDType**

Name	WeeklyHoursAndLessonsIDType
------	-----------------------------

Schema Component Representation

```
<xs:complexType name="WeeklyHoursAndLessonsIDType">
  <xs:complexContent>
    <xs:extension base="c:WeeklyHoursAndLessonsType">
      <xs:attribute name="companyWeeklyHoursAndLessonsID" type="c:InstanceRefIDType" use="required"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Documentation: Schema Component Representation

2.14 Complex Type: **WeeklyHoursAndLessonsRefType**

Name	WeeklyHoursAndLessonsRefType
------	------------------------------

Schema Component Representation

```
<xs:complexType name="WeeklyHoursAndLessonsRefType">
  <xs:sequence>
    <xs:element name="WeeklyHours" type="c:HoursOrLessonsType"/>
    <xs:element name="WeeklyLessons" type="c:HoursOrLessonsType"/>
  </xs:sequence>
  <xs:attribute name="companyWeeklyHoursAndLessonsIDRef" type="c:InstanceRefIDType" use="required"/>
</xs:complexType>
```

Documentation: Schema Component Representation

WeeklyHours	Translation: Individuelle wöchentliche Arbeitszeit (Stunden)
WeeklyLessons	Translation: Individuelle wöchentliche Arbeitszeit (Lektionen)
companyWeeklyHoursAndLessonsIDRef	Translation: Referenz zu den Wochenstunden und Wochenlektionen des Arbeitsorts im Unternehmen

2.9 Complex Type: **WeeklyHoursAndLessonsType**

Name	WeeklyHoursAndLessonsType
------	---------------------------

Schema Component Representation

```
<xs:complexType name="WeeklyHoursAndLessonsType">
  <xs:sequence>
    <xs:element name="WeeklyHours" type="c:HoursOrLessonsType"/>
    <xs:element name="WeeklyLessons" type="c:HoursOrLessonsType"/>
  </xs:sequence>
</xs:complexType>
```

```
</xs:sequence>
</xs:complexType>
```

Documentation: Schema Component Representation

WeeklyHours	Translation: Stunden pro Woche Short: Anzahl Stunden pro Woche
WeeklyLessons	Translation: Lektionen pro Woche Short: Anzahl Lektionen pro Woche

2.7 Complex Type: **WeeklyHoursIDType**

Name	WeeklyHoursIDType
------	-------------------

Schema Component Representation

```
<xs:complexType name="WeeklyHoursIDType">
  <xs:simpleContent>
    <xs:extension base="c:HoursOrLessonsType">
      <xs:attribute name="companyWeeklyHoursID" type="c:InstanceRefIDType" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Documentation: Schema Component Representation

2.8 Complex Type: **WeeklyLessonsIDType**

Name	WeeklyLessonsIDType
------	---------------------

Schema Component Representation

```
<xs:complexType name="WeeklyLessonsIDType">
  <xs:simpleContent>
    <xs:extension base="c:HoursOrLessonsType">
      <xs:attribute name="companyWeeklyLessonsID" type="c:InstanceRefIDType" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Documentation: Schema Component Representation

2.5 Complex Type: **WorkplaceType**

Name	WorkplaceType
------	---------------

Schema Component Representation

```
<xs:complexType name="WorkplaceType">
  <xs:sequence>
    <xs:choice minOccurs="0">
      <xs:element name="BUR-REE-Number" type="c:BUR-REE-NumberType"/>
      <xs:element name="InHouseID" type="c:IDType"/>
    </xs:choice>
    <xs:element name="AddressExtended" type="c:AddressExtensionType"/>
    <xs:element name="CompanyWorkingTime" type="c:CompanyWorkingTimeIDType" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="workplaceID" type="c:InstanceRefIDType" use="required"/>
</xs:complexType>
```

Documentation: Schema Component Representation

AddressExtended	Translation: Geografische Adresse Short: Geografische Adresse der Arbeitsorte
CompanyWorkingTime	Translation: Arbeitszeit Short: Arbeitszeit
workplaceID	Translation: Arbeitsort Identifikation

2.30 Complex Type: **WorkType**

Name	WorkType
-------------	-----------------

Schema Component Representation

```
<xs:complexType name="WorkType">
  <xs:sequence>
    <xs:element name="WorkingTime">
      <xs:complexType>
        <xs:choice>
          <xs:element name="Steady">
            <xs:complexType>
              <xs:sequence>
                <xs:choice>
                  <xs:element name="WeeklyHours" type="c:HoursOrLessonsType"/>
                  <xs:element name="WeeklyLessons" type="c:HoursOrLessonsType"/>
                  <xs:element name="WeeklyHoursAndLessons" type="c:WeeklyHoursAndLessonsType"/>
                </xs:choice>
                <xs:element name="ActivityRate" type="c:PercentType"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
          <xs:element name="Unsteady" type="c:EmptyType"/>
        </xs:choice>
      </xs:complexType>
    </xs:element>
    <xs:element name="EntryDate" type="xs:date"/>
    <xs:element name="WithdrawalDate" type="xs:date" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```


Documentation: Schema Component Representation

WorkingTime	Translation: Arbeitszeit Short: Individuell vereinbarte Arbeitszeit
EntryDate	Translation: Eintrittsdatum Short: Eintrittsdatum ins Unternehmen
WithdrawalDate	Translation: Austrittsdatum Short: Austrittsdatum aus dem Unternehmen

2.86 Model Group: NotificationsGroup

Name

Schema Component Representation

```
<xs:group name="NotificationsGroup">
  <xs:sequence>
    <xs:element name="Notification" type="c:NotificationType" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:group>
```

Documentation: Schema Component Representation

Notification	Translation: Hinweise
--------------	------------------------------

2.59 Model Group: RequestContextBaseGroup

Name

Schema Component Representation

```
<xs:group name="RequestContextBaseGroup">
  <xs:sequence>
    <xs:element name="UserAgent" type="c:UserAgentType"/>
    <xs:element name="CompanyName" type="c:CompanyNameType"/>
    <xs:element name="TransmissionDate" type="xs:dateTime"/>
    <xs:element name="RequestID" type="c:IDType"/>
    <xs:element name="LanguageCode" type="c:LanguageCodeType"/>
    <xs:element name="MonitoringID" type="c:MonitoringIDType" minOccurs="0"/>
  </xs:sequence>
</xs:group>
```

Documentation: Schema Component Representation

UserAgent	Translation: Applikationsinformation Short: Beschreibung der wesentlichen System-Identifikations-Daten Technical: Zur Qualitätsicherung des Übermittlungsprozesses werden die wesentlichen Informationen der beteiligten gesichert
CompanyName	Translation: Unternehmensdaten Short: Beschreibung der wesentlichen Unternehmensdaten

TransmissionDate	Translation: Übertragungszeitpunkt
RequestID	Translation: AuftragsID
LanguageCode	Translation: Sprachcode
MonitoringID	Translation: ÜberwachungsID Short: ÜberwachungsID Technical: Die ÜberwachungsID wird vorallem in der Referenzapplikation zur Aufteilung der Daten verwendet

2.62 Simple Type: **AssuranceCategoryCodeType**

Name	AssuranceCategoryCodeType
Content	Built-in XSD Type: NMTOKEN <i>pattern</i> = [A-Z0-9]{2}
Documentation	Translation (de): Versicherungskategorie-Code Typ

Schema Component Representation

```
<xs:simpleType name="AssuranceCategoryCodeType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:pattern value="[A-Z0-9]{2}" />
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.63 Simple Type: **BUR-REE-NumberType**

Name	BUR-REE-NumberType
Content	Built-in XSD Type: NMTOKEN <i>pattern</i> = [A-Z][0-9]{8}
Documentation	Translation (de): BUR-Nummer Typ

Schema Component Representation

```
<xs:simpleType name="BUR-REE-NumberType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:pattern value="[A-Z][0-9]{8}" />
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.65 Simple Type: **CantonAddressType**

Name	CantonAddressType
Content	<p>Built-in XSD Type: NMTOKEN</p> <p><i>value =</i> {'AG' 'AI' 'AR' 'BE' 'BL' 'BS' 'FR' 'GE' 'GL' 'GR' 'JU' 'LU' 'NE' 'NW' 'OW' 'SG' 'SH' 'SO' 'SZ' 'TG' 'TI' 'UR' 'VD' 'VS' 'ZG' 'ZH' 'EX'}</p> <p><i>value =</i> {'AG' 'AI' 'AR' 'BE' 'BL' 'BS' 'FR' 'GE' 'GL' 'GR' 'JU' 'LU' 'NE' 'NW' 'OW' 'SG' 'SH' 'SO' 'SZ' 'TG' 'TI' 'UR' 'VD' 'VS' 'ZG' 'ZH'}</p>
Documentation	<p>Translation (de): KantonsTyp</p> <p>Short description (de): KantonsTyp der Schweiz</p>

Schema Component Representation

```

<xs:simpleType name="CantonAddressType">
  <xs:restriction base="c:CantonAndEXType">
    <xs:enumeration value="AG"/>
    <xs:enumeration value="AI"/>
    <xs:enumeration value="AR"/>
    <xs:enumeration value="BE"/>
    <xs:enumeration value="BL"/>
    <xs:enumeration value="BS"/>
    <xs:enumeration value="FR"/>
    <xs:enumeration value="GE"/>
    <xs:enumeration value="GL"/>
    <xs:enumeration value="GR"/>
    <xs:enumeration value="JU"/>
    <xs:enumeration value="LU"/>
    <xs:enumeration value="NE"/>
    <xs:enumeration value="NW"/>
    <xs:enumeration value="OW"/>
    <xs:enumeration value="SG"/>
    <xs:enumeration value="SH"/>
    <xs:enumeration value="SO"/>
    <xs:enumeration value="SZ"/>
    <xs:enumeration value="TG"/>
    <xs:enumeration value="TI"/>
    <xs:enumeration value="UR"/>
    <xs:enumeration value="VD"/>
    <xs:enumeration value="VS"/>
    <xs:enumeration value="ZG"/>
    <xs:enumeration value="ZH"/>
  </xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.64 Simple Type: **CantonAndEXType**

Name	CantonAndEXType
Content	<p>Built-in XSD Type: NMTOKEN</p> <p><i>value =</i></p> <p>{'AG' 'AI' 'AR' 'BE' 'BL' 'BS' 'FR' 'GE' 'GL' 'GR' 'JU' 'LU' 'NE' 'NW' 'OW' 'SG' 'SH' 'SO' 'SZ' 'TG' 'TI' 'UR' 'VD' 'VS' 'ZG' 'ZH' 'EX'}</p>
Documentation	<p>Translation (de): KantonsTyp (inkl. EX)</p> <p>Short description (de): KantonsTyp der Schweiz (inkl. EX)</p> <p>Domain description (de): Der KantonsTyp der Schweiz inkl. EX (EXtern oder EXpatriates), die nicht in der Schweiz wohnen</p>

Schema Component Representation

```

<xs:simpleType name="CantonAndEXType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="AG"/>
    <xs:enumeration value="AI"/>
    <xs:enumeration value="AR"/>
    <xs:enumeration value="BE"/>
    <xs:enumeration value="BL"/>
    <xs:enumeration value="BS"/>
    <xs:enumeration value="FR"/>
    <xs:enumeration value="GE"/>
    <xs:enumeration value="GL"/>
    <xs:enumeration value="GR"/>
    <xs:enumeration value="JU"/>
    <xs:enumeration value="LU"/>
    <xs:enumeration value="NE"/>
    <xs:enumeration value="NW"/>
    <xs:enumeration value="OW"/>
    <xs:enumeration value="SG"/>
    <xs:enumeration value="SH"/>
    <xs:enumeration value="SO"/>
    <xs:enumeration value="SZ"/>
    <xs:enumeration value="TG"/>
    <xs:enumeration value="TI"/>
    <xs:enumeration value="UR"/>
    <xs:enumeration value="VD"/>
    <xs:enumeration value="VS"/>
    <xs:enumeration value="ZG"/>
    <xs:enumeration value="ZH"/>
    <xs:enumeration value="EX"/>
  </xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.38 Simple Type: **CategoryPredefinedType**

Name	CategoryPredefinedType
Content	Built-in XSD Type: NMTOKEN value = {'HEN' 'HEY' 'MEN' 'MEY' 'NON' 'NOY' 'SFN'}
Documentation	Translation (de): Vordefinierte Kategorien Short description (de): Vordefinierte Kategorien Domain description (de): Mögliche Werte: HEN, HEY: VR-Honorar an qSP mit Wohnsitz im Ausland; MEN, MEY: Leistungen aus exportierten Mitarbeiterbeteiligungen an qSP mit Wohnsitz im Ausland; NON, NOY: Nicht quellensteuerpflichtig; SFN: Sondervereinbarung mit Frankreich;

Schema Component Representation

```

<xs:simpleType name="CategoryPredefinedType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="HEN"/>
    <xs:enumeration value="HEY"/>
    <xs:enumeration value="MEN"/>
    <xs:enumeration value="MEY"/>
    <xs:enumeration value="NON"/>
    <xs:enumeration value="NOY"/>
    <xs:enumeration value="SFN"/>
  </xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.66 Simple Type: **CivilStatusType**

Name	CivilStatusType
Content	Built-in XSD Type: NMTOKEN value = {'unknown' 'single' 'married' 'widowed' 'divorced' 'separated' 'registeredPartner-ship' 'partnershipDissolvedByLaw' 'partnershipDissolvedByDeath' 'partnershipDissolvedByDeclarationOfLost'}
Documentation	Translation (de):

ZivilstandsTyp**Short description (de):**

ZivilstandsTyp

Domain description (de):

Mögliche Werte: unknown = unbekannt; single = ledig; married = verheiratet; widowed = verwitwet; divorced = geschieden; separated = getrennt; registeredPartnership = eingetragene Partnerschaft partnershipDissolvedByLaw = gerichtlich aufgelöste Partnerschaft partnershipDissolvedByDeath = durch Tod aufgelöste Partnerschaft partnershipDissolvedByDeclarationOfLost = durch Verschollenerklärung aufgelöste Partnerschaft

Schema Component Representation

```
<xs:simpleType name="CivilStatusType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="single"/>
    <xs:enumeration value="married"/>
    <xs:enumeration value="widowed"/>
    <xs:enumeration value="divorced"/>
    <xs:enumeration value="separated"/>
    <xs:enumeration value="registeredPartnership"/>
    <xs:enumeration value="partnershipDissolvedByLaw"/>
    <xs:enumeration value="partnershipDissolvedByDeath"/>
    <xs:enumeration value="partnershipDissolvedByDeclarationOfLost"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation**2.89 Simple Type: DescriptionCodeType**

Name	DescriptionCodeType
Content	Built-in XSD Type: nonNegativeInteger <i>total no. of digits = 7</i>
Documentation	Translation (de): Beschreibungscode Technical description (de): Severity: Information Code 9999 ist immer für Freitext reserviert. Die restlichen Code sind dynamisch in AcknowledgementNotification spezifiziert.

Schema Component Representation

```
<xs:simpleType name="DescriptionCodeType">
```

```

<xs:restriction base="xs:nonNegativeInteger">
  <xs:totalDigits value="7"/>
</xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.67 Simple Type: **EmailAddressType**

Name	EmailAddressType
Content	Built-in XSD Type: string <i>pattern</i> = [^@]+@[^\.]+\..+
Documentation	Translation (de): Email Adresse

Schema Component Representation

```

<xs:simpleType name="EmailAddressType">
  <xs:restriction base="xs:string">
    <xs:pattern value="[^@]+@[^\.]+\..+"/>
  </xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.83 Simple Type: **EmploymentContractType**

Name	EmploymentContractType
Content	Built-in XSD Type: NMTOKEN <i>value</i> = {'indefiniteSalaryMth' 'indefiniteSalaryMthAWT' 'indefiniteSalaryHrs' 'indefiniteSalaryNoTimeConstraint' 'fixedSalaryMth' 'fixedSalaryHrs' 'fixedSalaryNoTimeConstraint' 'apprentice' 'internshipContract'}
Documentation	Translation (de): Arbeitsvertragsart Short description (de): Arbeitsvertragsart angeben Domain description (de): Mögliche Werte: indefiniteSalaryMth = unbefristeter Vertrag mit Monatslohn (indefinite duration with salary per month) in-

definiteSalaryMthAWT = unbefristeter Vertrag mit Monatslohn und Jahresarbeitszeit (indefinite duration with salary per month and annual working time model) indefiniteSalaryHrs = unbefristeter Vertrag mit Stundenlohn (indefinite duration with salary per month) indefiniteSalaryNoTimeConstraint = unbefristeter Vertrag mit Provision, Pauschale, Akkordlohn (indefinite duration with commission, lump sum, piece rate) fixedSalaryMth = befristeter Vertrag mit Monatslohn (fixed/temporary duration salary per month) fixedSalaryHrs = befristeter Vertrag mit Stundenlohn (fixed/temporary duration salary per hour) fixedSalaryNoTimeConstraint = befristeter Vertrag mit Provision, Pauschale, Akkordlohn (fixed/temporary duration with commission, lump sum, piece rate) apprentice = Lehrvertrag internshipContract = Praktikumsvertrag

Schema Component Representation

```
<xs:simpleType name="EmploymentContractType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="indefiniteSalaryMth"/>
    <xs:enumeration value="indefiniteSalaryMthAWT"/>
    <xs:enumeration value="indefiniteSalaryHrs"/>
    <xs:enumeration value="indefiniteSalaryNoTimeConstraint"/>
    <xs:enumeration value="fixedSalaryMth"/>
    <xs:enumeration value="fixedSalaryHrs"/>
    <xs:enumeration value="fixedSalaryNoTimeConstraint"/>
    <xs:enumeration value="apprentice"/>
    <xs:enumeration value="internshipContract"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.26 Simple Type: **EmptySimpleType**

Name	EmptySimpleType
Content	Built-in XSD Type: string value = {}

Schema Component Representation

```
<xs:simpleType name="EmptySimpleType">
  <xs:restriction base="xs:string">
    <xs:enumeration value=""/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.93 Simple Type: **FaultCodeType**

Name	FaultCodeType
Content	Built-in XSD Type: NMTOKEN <i>value</i> = {'NOT_accepted' 'NOT_plausible' 'NOT_valid'}
Documentation	Translation (de): Fehler-Code

Schema Component Representation

```
<xs:simpleType name="FaultCodeType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="NOT_accepted"/>
    <xs:enumeration value="NOT_plausible"/>
    <xs:enumeration value="NOT_valid"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.82 Simple Type: **HoursOrLessonsType**

Name	HoursOrLessonsType
Content	Built-in XSD Type: decimal <i>pattern</i> = [0-9]+\.[0-9]{2}
Documentation	Translation (de): Stunden oder Lektionen

Schema Component Representation

```
<xs:simpleType name="HoursOrLessonsType">
  <xs:restriction base="xs:decimal">
    <xs:pattern value="[0-9]+\.[0-9]{2}" />
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.68 Simple Type: **IDType**

Name	IDType
Content	Built-in XSD Type: string

	<i>length</i> >= 1
--	--------------------

Schema Component Representation

```
<xs:simpleType name="IDType">
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="255"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.69 Simple Type: InstanceRefIDType

Name	InstanceRefIDType
Content	Built-in XSD Type: string <i>pattern</i> = #.*
Documentation	Translation (de): Instance Reference ID

Schema Component Representation

```
<xs:simpleType name="InstanceRefIDType">
  <xs:restriction base="xs:string">
    <xs:pattern value="#.*"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.70 Simple Type: LanguageCodeType

Name	LanguageCodeType
Content	Built-in XSD Type: NMTOKEN <i>value</i> = {'de' 'fr' 'it' 'en'}
Documentation	Translation (de): Sprachcode

Schema Component Representation

```
<xs:simpleType name="LanguageCodeType">
```

```

<xs:restriction base="xs:NMTOKEN">
  <xs:enumeration value="de"/>
  <xs:enumeration value="fr"/>
  <xs:enumeration value="it"/>
  <xs:enumeration value="en"/>
</xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.61 Simple Type: **MonitoringIDType**

Name	MonitoringIDType
Content	Built-in XSD Type: string <i>length</i> >= 1

Schema Component Representation

```

<xs:simpleType name="MonitoringIDType">
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="32"/>
  </xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.71 Simple Type: **MunicipalityIDType**

Name	MunicipalityIDType
Content	Built-in XSD Type: int 1 <= <i>value</i> <= 9999 <i>total no. of digits</i> = 4
Documentation	Translation (de): Gemeindenummer

Schema Component Representation

```

<xs:simpleType name="MunicipalityIDType">
  <xs:restriction base="xs:int">
    <xs:minInclusive value="1"/>
    <xs:maxInclusive value="9999"/>
    <xs:totalDigits value="4"/>
  </xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.72 Simple Type: **NationalityType**

Name	NationalityType
Content	Built-in XSD Type: string <i>pattern</i> = [A-Z][A-Z]11 22
Documentation	<p>Translation (de): Staatsangehörigkeit</p> <p>Short description (de): Die Codes für eine korrekte Zuteilung der Staatsangehörigkeit sind bei der UNO hinterlegt (ISO 3166)</p> <p>Technical description (de): Erweiterung 11= unbekannt 22= staatenlos : Die ISO-Codes wurden auf dem Prod-Distri noch nie geprüft: In der Qualitätsstufe Plausibilität kann evtl. gegen die ISO 3166 Codes geprüft und eine Warnung angezeigt werden. Achtung: Probleme mit bestehenden Datenbeständen (Änderungen der Ländernamen und Verwechslungen)</p>

Schema Component Representation

```
<xs:simpleType name="NationalityType">
  <xs:restriction base="xs:string">
    <xs:pattern value="[A-Z][A-Z]11|22"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.73 Simple Type: **NotEmptyStringType**

Name	NotEmptyStringType
Content	Built-in XSD Type: string <i>length</i> >= 1

Schema Component Representation

```
<xs:simpleType name="NotEmptyStringType">
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
  </xs:restriction>
</xs:simpleType>
```

```
</xs:simpleType>
```

Documentation: Schema Component Representation

2.74 Simple Type: **PercentType**

Name	PercentType
Content	Built-in XSD Type: decimal <i>pattern</i> = [0-9]+\.[0-9]{2}
Documentation	Translation (de): Prozent

Schema Component Representation

```
<xs:simpleType name="PercentType">
  <xs:restriction base="xs:decimal">
    <xs:pattern value="[0-9]+\.[0-9]{2}" />
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.84 Simple Type: **PositionType**

Name	PositionType
Content	Built-in XSD Type: NMTOKEN <i>value</i> = {'highestCadre' 'middleCadre' 'lowerCadre' 'lowestCadre' 'noCadre'}
Documentation	Translation (de): Berufliche Stellung Short description (de): Berufliche Stellung des Arbeitnehmers Domain description (de): Mögliche Werte: highestCadre: Oberes Kader; middleCadre: Mittleres Kader; lowerCadre: Unteres Kader; lowestCadre: Unterstes Kader; noCadre: ohne Kaderfunktion;

Schema Component Representation

```
<xs:simpleType name="PositionType">
```

```

<xs:restriction base="xs:NMTOKEN">
  <xs:enumeration value="highestCadre"/>
  <xs:enumeration value="middleCadre"/>
  <xs:enumeration value="lowerCadre"/>
  <xs:enumeration value="lowestCadre"/>
  <xs:enumeration value="noCadre"/>
</xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.90 Simple Type: **QualityLevelType**

Name	QualityLevelType
Content	Built-in XSD Type: NMTOKEN <i>value</i> = {'Validity' 'Plausibility' 'Acceptance' 'Comment'}
Documentation	Translation (de): Qualitäts-Stufen Short description (de): Qualitäts-Stufen der Übermittlung Technical description (de): Folgende Stufen gelten Validity: erste Stufe; Plausibility: zweite Stufe; Acceptance: dritte Stufe; Comment: vierte Stufe;

Schema Component Representation

```

<xs:simpleType name="QualityLevelType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="Validity"/>
    <xs:enumeration value="Plausibility"/>
    <xs:enumeration value="Acceptance"/>
    <xs:enumeration value="Comment"/>
  </xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.75 Simple Type: **ResidenceCategoryType**

Name	ResidenceCategoryType
Content	Built-in XSD Type: NMTOKEN <i>value</i> = {'shortTerm-L' 'annual-B' 'settled-C' 'crossBorder-G' 'asylumSeeker-'

	N' 'needForProtection-S' 'NotificationProcedureForShorttermWork90Days' 'NotificationProcedureForShorttermWork120Days' 'ProvisionallyAdmittedForeigners-F' 'ResidentForeignNationalWithGainfulEmployment-Ci' 'othersNotSwiss'}
Documentation	<p>Translation (de): Aufenthaltskategorien</p> <p>Short description (de): Aufenthaltskategorien für Ausländer</p> <p>Domain description (de): Mögliche Werte: shortTerm-L = Kurzaufenthalter (L); annual-B = Jahresaufenthalter (B); settled-C = Niedergelassene (C); crossBorder-G = Grenzgänger (G); asylumSeeker-N = Asylsuchender (N); needForProtection-S = Schutzbedürftige (S); NotificationProcedureForShorttermWork90Days = Meldeverfahren für kurzfristige Erwerbstätigkeit (bis 90 Tage, Schengenregelung); NotificationProcedureForShorttermWork120Days = Meldeverfahren für kurzfristige Erwerbstätigkeit bis 120 Tage; ProvisionallyAdmittedForeigners (F) = Vorläufig aufgenommene Ausländer; ResidentForeignNationalWithGainfulEmployment (Ci) = Aufenthaltsbewilligung mit Erwerbstätigkeit; othersNotSwiss = Andere (nicht Schweizer) ;</p>

Schema Component Representation

```
<xs:simpleType name="ResidenceCategoryType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="shortTerm-L"/>
    <xs:enumeration value="annual-B"/>
    <xs:enumeration value="settled-C"/>
    <xs:enumeration value="crossBorder-G"/>
    <xs:enumeration value="asylumSeeker-N"/>
    <xs:enumeration value="needForProtection-S"/>
    <xs:enumeration value="NotificationProcedureForShorttermWork90Days"/>
    <xs:enumeration value="NotificationProcedureForShorttermWork120Days"/>
    <xs:enumeration value="ProvisionallyAdmittedForeigners-F"/>
    <xs:enumeration value="ResidentForeignNationalWithGainfulEmployment-Ci"/>
    <xs:enumeration value="othersNotSwiss"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.92 Simple Type: ResponseCodeAAwRType

Name	ResponseCodeAAwRType
Content	Built-in XSD Type: NMTOKEN value = {'accepted' 'acceptedWithWarning' 'rejected'}
Documentation	

	Translation (de): Antwort-Code
--	--

Schema Component Representation

```
<xs:simpleType name="ResponseCodeAAwRType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="accepted"/>
    <xs:enumeration value="acceptedWithWarning"/>
    <xs:enumeration value="rejected"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.91 Simple Type: ResponseCodeType

Name	ResponseCodeType
Content	Built-in XSD Type: NMTOKEN <i>value</i> = {'accepted' 'acceptedWithWarning'}
Documentation	Translation (de): Antwort-Code

Schema Component Representation

```
<xs:simpleType name="ResponseCodeType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="accepted"/>
    <xs:enumeration value="acceptedWithWarning"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.77 Simple Type: SalaryAmountAbsoluteType

Name	SalaryAmountAbsoluteType
Content	Built-in XSD Type: decimal <i>pattern</i> = [\-]?[0-9]+\.[0-9]{2} <i>pattern</i> = [0-9]+\.[0-9]{2}
Documentation	Translation (de): Absolut-LohnbetragsTyp

Short description (de):

Absolut-LohnbetragsTyp ohne Vorzeichen

Schema Component Representation

```
<xs:simpleType name="SalaryAmountAbsoluteType">
  <xs:restriction base="c:SalaryAmountType">
    <xs:pattern value="[0-9]+\.[0-9]{2}"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation**2.76 Simple Type: SalaryAmountType**

Name	SalaryAmountType
Content	Built-in XSD Type: decimal <i>pattern</i> = [\-]?[0-9]+\.[0-9]{2}
Documentation	Translation (de): LohnbetragsTyp

Schema Component Representation

```
<xs:simpleType name="SalaryAmountType">
  <xs:restriction base="xs:decimal">
    <xs:pattern value="[\-]?[0-9]+\.[0-9]{2}"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation**2.58 Simple Type: SectionIDType**

Name	SectionIDType
Content	Built-in XSD Type: string <i>pattern</i> = #.*
Documentation	Translation (de): Abschnitt Identifikation Typ

Schema Component Representation

```
<xs:simpleType name="SectionIDType">
```

```
<xs:restriction base="xs:string">
  <xs:pattern value="#.*"/>
</xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.78 Simple Type: **SexType**

Name	SexType
Content	Built-in XSD Type: NMTOKEN <i>value</i> = {'F' 'M'}
Documentation	Translation (de): Geschlechts Typ Short description (de): F weiblich; M männlich

Schema Component Representation

```
<xs:simpleType name="SexType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="F"/>
    <xs:enumeration value="M"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.79 Simple Type: **SimpleBooleanType**

Name	SimpleBooleanType
Content	Built-in XSD Type: boolean <i>pattern</i> = true
Documentation	Translation (de): SimpleBoolean Short description (de): Boolean

Schema Component Representation

```
<xs:simpleType name="SimpleBooleanType">
  <xs:restriction base="xs:boolean">
```

```

<xs:pattern value="true"/>
<xs:pattern value="false"/>
</xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.56 Simple Type: **StandardFormIDType**

Name	StandardFormIDType
Content	Built-in XSD Type: string <i>length</i> >= 1 <i>pattern</i> = notStandard [0-9]{4}\.[0-9]{4}\.[0-9]{4}-[0-9]{3}
Documentation	Translation (de): Standard Formular ID

Schema Component Representation

```

<xs:simpleType name="StandardFormIDType">
  <xs:restriction base="c:IDType">
    <xs:pattern value="notStandard|[0-9]{4}\.[0-9]{4}\.[0-9]{4}-[0-9]{3}"/>
  </xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.80 Simple Type: **SV-AS-NumberType**

Name	SV-AS-NumberType
Content	Built-in XSD Type: NMTOKEN <i>pattern</i> = [0-9]{3}\.[0-9]{4}\.[0-9]{4}\.[0-9]{2} <i>length</i> <= 16
Documentation	Translation (de): 13-stellige SV-Nr Typ

Schema Component Representation

```

<xs:simpleType name="SV-AS-NumberType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:pattern value="[0-9]{3}\.[0-9]{4}\.[0-9]{4}\.[0-9]{2}"/>
    <xs:maxLength value="16"/>
  </xs:restriction>
</xs:simpleType>

```

Documentation: Schema Component Representation

2.39 Simple Type: **TaxAtSourceCodeType**

Name	TaxAtSourceCodeType
Content	Built-in XSD Type: NMTOKEN <i>pattern</i> = [A-Z][0-9][Y,N]
Documentation	Translation (de): QST-Code

Schema Component Representation

```
<xs:simpleType name="TaxAtSourceCodeType">
  <xs:restriction base=" xs:NMTOKEN ">
    <xs:pattern value="[A-Z][0-9][Y,N]"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.81 Simple Type: **UID-BFSType**

Name	UID-BFSType
Content	Built-in XSD Type: NMTOKEN <i>pattern</i> = CHE-[0-9]{3}\.[0-9]{3}\.[0-9]{3}
Documentation	Translation (de): Unternehmens UID-BFS Typ

Schema Component Representation

```
<xs:simpleType name="UID-BFSType">
  <xs:restriction base=" xs:NMTOKEN ">
    <xs:pattern value="CHE-[0-9]{3}\.[0-9]{3}\.[0-9]{3}"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.57 Simple Type: **YesNoUnknownType**

Name	YesNoUnknownType
Content	Built-in XSD Type: NMTOKEN <i>value</i> = {'yes' 'no' 'unknown'}
Documentation	Translation (de): ja, nein oder unbekannt

Schema Component Representation

```
<xs:simpleType name="YesNoUnknownType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="yes"/>
    <xs:enumeration value="no"/>
    <xs:enumeration value="unknown"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation

2.85 Simple Type: **ZIP-CodeType**

Name	ZIP-CodeType
Content	Built-in XSD Type: string <i>length</i> >= 1
Documentation	Translation (de): PLZ Typ Short description (de): Postleitzahl

Schema Component Representation

```
<xs:simpleType name="ZIP-CodeType">
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="15"/>
  </xs:restriction>
</xs:simpleType>
```

Documentation: Schema Component Representation