



# Direttive per trasmettitore di dati salariali

## Direttive standard salari 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\_it.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 17:42

## 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 .....	36
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 .....	40
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 .....	52
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: Data di scadenza

## 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	<b>Translation:</b> Riga complementare <b>Short:</b> Riga complementare per l'indirizzo posta-le
Street	<b>Translation:</b> Via <b>Short:</b> Via e numero
Postbox	<b>Translation:</b> Casella postale <b>Short:</b> Casella postale
Locality	<b>Translation:</b> Località <b>Short:</b> Località come regione, provincia
ZIP-Code	<b>Translation:</b> Numero postale d'avviamento <b>Short:</b> Numero postale d'avviamento
City	<b>Translation:</b> Luogo <b>Short:</b> Luogo

Country	<b>Translation:</b> Nazione
	<b>Short:</b> Nazione

## 2.3 Complex Type: **AddressWorkplaceType**

<b>Name</b>	<b>AddressWorkplaceType</b>
-------------	-----------------------------

### 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	<b>Translation:</b> Riga complementare <b>Short:</b> Riga complementare per l'indirizzo posta-le
Street	<b>Translation:</b> Via <b>Short:</b> Via e numero
Locality	<b>Translation:</b> Località <b>Short:</b> Località come regione, provincia
ZIP-Code	<b>Translation:</b> Numero postale d'avviamento <b>Short:</b> Numero postale d'avviamento
City	<b>Translation:</b> Luogo <b>Short:</b> Luogo
Country	<b>Translation:</b> Nazione <b>Short:</b> Nazione

## 2.54 Complex Type: **AnswerAmountType**

<b>Name</b>	<b>AnswerAmountType</b>
-------------	-------------------------

### 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	<b>Translation:</b> Vecchie credenziali
NewCredentials	<b>Translation:</b> Nuove credenziali

## 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	<b>Translation:</b> Informazioni sull'applicazione <b>Short:</b> Descrizione dei dati essenziali per l'identificazione del sistema <b>Technical:</b> Per garantire la qualità, le informazioni essenziali dei partecipanti sono salvati.
SystemDateTime	<b>Translation:</b> Tempo di sistema attuale

## 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	<b>Translation:</b> Informazioni sull'applicazione <b>Short:</b> Descrizione dei dati essenziali per l'identificazione del sistema <b>Technical:</b> Per garantire la qualità, le informazioni essenziali dei partecipanti sono salvati.
UmlautString	<b>Translation:</b> rete di diaeresi <b>Short:</b> Disposizione delle dieresi per il test delle lettere maiuscole e minuscole.
FirstOperand	<b>Translation:</b> 1. dimensione di prova specificata <b>Short:</b> 1. l'operando viene aggiunto con 2. l'operando per verificare che non ci siano errori di tipo i dati non si verificano.
SecondOperand	<b>Translation:</b> 2. dimensione di prova specificata <b>Short:</b> Il 2° operando viene aggiunto con il 1° operando per verificare che non ci siano errori di il tipo di dati non si verifica.
SystemDateTime	<b>Translation:</b> Tempo di sistema attuale
MonitoringID	<b>Translation:</b> Identificatore di sorveglianza <b>Short:</b> Identificatore di sorveglianza <b>Technical:</b> L'identificatore di sorveglianza è utilizzato principalmente nell'applicazione di riferimento per la classificazione dei dati.

## 2.24 Complex Type: **CivilStatusAndDateType**

<b>Name</b>	<b>CivilStatusAndDateType</b>
-------------	-------------------------------

## 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	<b>Translation:</b> Stato civile <b>Short:</b> Stato civile della persona
ValidAsOf	<b>Translation:</b> Lo stato civile è valido da <b>Short:</b> Lo stato civile della persona è valido dal

## 2.1 Complex Type: **CompanyDescriptionBaseType**

<b>Name</b>	<b>CompanyDescriptionBaseType</b>
-------------	-----------------------------------

## 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	<b>Translation:</b> Nome <b>Short:</b> Nome dell'azienda
Owner	<b>Translation:</b> Titolare <b>Short:</b> Titolare
Address	<b>Translation:</b> Indirizzo <b>Short:</b> Indirizzo dell'azienda
UID-BFS	<b>Translation:</b> Numero d'identificazione IDI-UST <b>Short:</b> Numero d'identificazione delle imprese (IDI-UST)
Delegate	<b>Translation:</b> Sostituire <b>Short:</b> Rappresentante della società

## 2.2 Complex Type: **CompanyDescriptionType**

<b>Name</b>	<b>CompanyDescriptionType</b>
-------------	-------------------------------

### 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**

<b>Name</b>	<b>CompanyNameType</b>
-------------	------------------------

## 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	<b>Translation:</b> nome o nome RC/UID-UST <b>Short:</b> nome o nome RC della società
ComplementaryLine	<b>Translation:</b> Riga complementare <b>Short:</b> Riga complementare per il nome della divisione, denominazioni di filiali ecc.

## 2.16 Complex Type: **CompanyOwnerType**

<b>Name</b>	<b>CompanyOwnerType</b>
-------------	-------------------------

### 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	<b>Translation:</b> Nome <b>Short:</b> Nome del titolare
Lastname	<b>Translation:</b> Cognome <b>Short:</b> Cognome del titolare

## 2.6 Complex Type: **CompanyWorkingTimeIDType**

<b>Name</b>	<b>CompanyWorkingTimeIDType</b>
-------------	---------------------------------

### 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>
```

```
</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: Nome
EmailAddress	Translation: Indirizzo e-mail
PhoneNumber	Translation: Numero di telefono
MobilePhoneNumber	Translation: Numero di cellulare

### 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: Chiave
Password	Translation: Password

## 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: Nome Short: Nome del rappresentante
Owner	Translation: Titolare Short: Titolare
Address	Translation: Indirizzo Short: Indirizzo dell'azienda
UID-BFS	Translation: Numero d'identificazione IDI-UST Short: Numero d'identificazione delle imprese (IDI-UST)

## 2.41 Complex Type: DialogMessagesType

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

### Schema Component Representation

```

<xs:complexType name="DialogMessageType">
  <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: Messaggio di dialogo

Short: Formato di scambio dati semplice e flessibile

## 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**

<b>Name</b>	<b>FaultStateType</b>
<b>Documentation</b>	<b>Translation (it):</b> Stato di errore

#### 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	<b>Translation:</b> Errore: Codice di stato
Error	<b>Translation:</b> Errore
Warning	<b>Translation:</b> Attenzione
Info	<b>Translation:</b> Informazioni

### 2.12 Complex Type: **HoursRefType**

<b>Name</b>	<b>HoursRefType</b>
-------------	---------------------

#### 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**

<b>Name</b>	<b>IncidentCaseIDType</b>
<b>Documentation</b>	<b>Translation (it):</b> 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

<b>Name</b>	<b>IncidentCaseIDWithMapType</b>
-------------	----------------------------------

### 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

<b>Name</b>	<b>LessonsRefType</b>
-------------	-----------------------

### 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: Nome
Address	Translation: Indirizzo

## 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: Note

## 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: Livello di qualità

DescriptionCode

Translation: Codice di descrizione

Description

Translation: Descrizione

## 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	<b>Translation:</b> Problema <b>Short:</b> Segnalare i problemi quando si risponde
optional	<b>Translation:</b> risposta opzionale <b>Short:</b> Marcatura per una risposta opzionale

## 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	<b>Translation:</b> Identificazione
sectionIDRef	<b>Translation:</b> Riferimento Sezione Identificazione

## 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-InsurancelIdentification" type="c:Social-InsurancelIdentificationType"/>
    <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-InsurancelIdentification	<b>Translation:</b> Numero assicurazione sociale <b>Short:</b> Numero assicurazione sociale
EmployeeNumber	<b>Translation:</b> Numero personale <b>Short:</b> Il numero personale è di regola numerico e può essere definito libera-

	mente dal produttore del programma rispettivamente dall'azienda. (per esempio 1254)
Lastname	<b>Translation:</b> Cognome <b>Short:</b> Cognome
Firstname	<b>Translation:</b> Nome <b>Short:</b> Nome
Sex	<b>Translation:</b> Sesso
DateOfBirth	<b>Translation:</b> Data di nascita <b>Short:</b> Data di nascita della persona
Nationality	<b>Translation:</b> Nazionalità <b>Short:</b> I codici per l'attribuzione corretta della cittadinanza sono disponibili nella norme ONU (ISO 3166) <b>Technical:</b> Estensione 11= sconosciuto 22= apolide : I codici ISO non sono mai stati controllati sul Prod-Distri Nella plausibilità del livello di qualità può essere possibile verificare la plausibilità rispetto ai codici ISO 3166 e visualizzare un'avvertenza.
CivilStatus	<b>Translation:</b> Stato civile <b>Short:</b> Stato civile con data di validità

## 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:** Informazioni sull'applicazione

**Short:** Descrizione dei dati essenziali per l'identificazione del sistema

**Technical:** Per garantire la qualità, le informazioni essenziali dei partecipanti sono salvati.

## 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:** Informazioni sull'applicazione

**Short:** Descrizione dei dati essenziali per l'identificazione del sistema

**Technical:** Per garantire la qualità, le informazioni essenziali dei partecipanti sono salvati.

Timestamp

**Translation:** data e ora

**Short:** Il timestamping è un meccanismo che consiste nell'associare una data e l'ora di un evento, di un'informazione o di un dato informatico. Ha di solito per registrare l'ora in cui è stata eseguita un'operazione.

NextCheck

**Translation:** prossima ispezione

RegisteredMaintenance

**Translation:** Finestra di manutenzione registrata

## 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	<b>Translation:</b> Informazioni sull'applicazione <b>Short:</b> Descrizione dei dati essenziali per l'identificazione del sistema <b>Technical:</b> Per garantire la qualità, le informazioni essenziali dei partecipanti sono salvati.
SystemDateTime	<b>Translation:</b> Tempo di sistema attuale

## 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	<b>Translation:</b> Informazioni sull'applicazione <b>Short:</b> Descrizione dei dati essenziali per l'identificazione del sistema <b>Technical:</b> Per garantire la qualità, le informazioni essenziali dei partecipanti sono salvati.
SystemDateTime	<b>Translation:</b> Tempo di sistema attuale
MonitoringID	<b>Translation:</b> Identificatore di sorveglianza <b>Short:</b> Identificatore di sorveglianza <b>Technical:</b> L'identificatore di sorveglianza è utilizzato principalmente nell'applicazione di riferimento per la classificazione dei dati.

## 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: Messaggio per il trasmettitore

## 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: Rubrica
Description	Translation: Descrizione
sectionID	Translation: Identificazione Sezione

## 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: Creazione
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: dalla
until	Translation: fino alla

### 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: Produttore della domanda
Name	Translation: Nome del prodotto
Version	Translation: Versione del prodotto
StandardVersion	Translation: Swissdec Versione standard
Certificate	Translation: Certificato

## 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	<b>Translation:</b> Durata del lavoro settimanale individuale (ore)
WeeklyLessons	<b>Translation:</b> Durata del lavoro settimanale individuale (lezioni)
companyWeeklyHoursAndLessonsIDRef	<b>Translation:</b> riferimento alle ore e alle lezioni settimanali nel luogo di lavoro dell'azienda

## 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	<b>Translation:</b> Ore settimanali <b>Short:</b> Numero di ore settimanali
WeeklyLessons	<b>Translation:</b> Lezioni settimanali conteggiate <b>Short:</b> Numero di lezioni settimanali conteggiate

## 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	<b>Translation:</b> Indirizzo geografico <b>Short:</b> Indirizzo geografico dei luoghi di lavoro
CompanyWorkingTime	<b>Translation:</b> Orario di lavoro <b>Short:</b> Orario di lavoro
workplaceID	<b>Translation:</b> Identificazione del luogo di lavoro

## 2.30 Complex Type: **WorkType**

<b>Name</b>	<b>WorkType</b>
-------------	-----------------

## 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	<b>Translation:</b> Durata di lavoro <b>Short:</b> Durata di lavoro convenuto individualmente.
EntryDate	<b>Translation:</b> Data di entrata <b>Short:</b> Data di entrata nell'impresa
WithdrawalDate	<b>Translation:</b> Data di uscita <b>Short:</b> Data di uscita dall'impresa

## 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	<b>Translation:</b> Note
--------------	--------------------------

## 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	<b>Translation:</b> Informazioni sull'applicazione <b>Short:</b> Descrizione dei dati essenziali per l'identificazione del sistema <b>Technical:</b> Per garantire la qualità, le informazioni essenziali dei partecipanti sono salvati.
CompanyName	<b>Translation:</b> Nome dell'azienda <b>Short:</b> Descrizione dei principali dati aziendali

TransmissionDate	<b>Translation:</b> Data e ora di trasmissione
RequestID	<b>Translation:</b> Identificatore di applicazione
LanguageCode	<b>Translation:</b> Codice lingua
MonitoringID	<b>Translation:</b> Identificatore di sorveglianza <b>Short:</b> Identificatore di sorveglianza <b>Technical:</b> L'identificatore di sorveglianza è utilizzato principalmente nell'applicazione di riferimento per la classificazione dei dati.

## 2.62 Simple Type: **AssuranceCategoryCodeType**

<b>Name</b>	<b>AssuranceCategoryCodeType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN <i>pattern</i> = [A-Z0-9]{2}

### 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**

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

### 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**

<b>Name</b>	<b>CantonAddressType</b>
<b>Content</b>	<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> <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'}</p>
<b>Documentation</b>	<p><b>Translation (it):</b></p> <p>Tipo di cantone</p> <p><b>Short description (it):</b></p> <p>Tipo di cantone Svizzera</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**

<b>Name</b>	<b>CantonAndEXType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN <i>value =</i> {'AG' 'AI' 'AR' 'BE' 'BL' 'BS' 'FR' 'GE' 'GL' 'GR' 'JU' 'LU' 'NE' 'N W' 'OW' 'SG' 'SH' 'SO' 'SZ' 'TG' 'TI' 'UR' 'VD' 'VS' 'ZG' 'ZH' 'EX' }
<b>Documentation</b>	<b>Translation (it):</b> Tipo di Cantone (incl. EX) <b>Short description (it):</b> Il tipo di cantone di Svizzera incl. EX (EXtern o EXpatriates) che non vivono in Svizzera

## 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**

<b>Name</b>	<b>CategoryPredefinedType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN <i>value</i> = {'HEN' 'HEY' 'MEN' 'MEY' 'NON' 'NOY' 'SFN'}
<b>Documentation</b>	<b>Translation (it):</b> Categorie predefinite <b>Short description (it):</b> Categorie predefinite <b>Domain description (it):</b> Valeurs possibili: HEN, HEY: Onorario CdA a PIF con domicilio all'estero; MEN, MEY: Prestazioni derivanti da partecipazioni dei collaboratori esportate a PIF con domicilio all'estero; NON, NOY: Non soggetto all'imposta alla fonte; SFN: Accordo speciale con la Francia;

## 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**

<b>Name</b>	<b>CivilStatusType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN <i>value</i> = {'unknown' 'single' 'married' 'widowed' 'divorced' 'separated' 'registeredPartner-ship' 'partnershipDissolvedByLaw' 'partnershipDissolvedByDeath' 'partnershipDissolvedByDeclarationOfLost'}
<b>Documentation</b>	<b>Translation (it):</b> Tipo di stato civile <b>Short description (it):</b> Type d'état civil <b>Domain description (it):</b> Valori possibili: unknown = sconosciuto; single = celibe/Nu-

bile; married = sposato(a); widowed = vedova(a); divorced = divorziato(a); separated = separato(a); registeredPartnership = unione registrata; partnershipDissolvedByLaw = unione sciolta dal tribunale; partnershipDissolvedByDeath = unione sciolta per decesso; partnershipDissolvedByDeclarationOfLost = unione sciolta per dichiarazione di scomparsa;

## 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**

<b>Name</b>	<b>DescriptionCodeType</b>
<b>Content</b>	Built-in XSD Type: nonNegativeInteger <i>total no. of digits = 7</i>
<b>Documentation</b>	<b>Translation (it):</b> Codice di descrizione <b>Technical description (it):</b> Severity: Informazioni Il codice 9999 è sempre riservato ad un testo libero. Il resto dei codici sono specificati dinamicamente in RiconoscimentoNotifica.

## 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**

<b>Name</b>	<b>EmailAddressType</b>
<b>Content</b>	Built-in XSD Type: string <i>pattern</i> = <code>^[^@]+@[^\.]+\.\.+</code>
<b>Documentation</b>	<b>Translation (it):</b> indirizzo e-mail

## 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**

<b>Name</b>	<b>EmploymentContractType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN <i>value</i> = {'indefiniteSalaryMth','indefiniteSalaryMthAWT','indefiniteSalaryHrs','indefiniteSalaryNoTimeConstraint','fixedSalaryMth','fixedSalaryHrs','fixedSalaryNoTimeConstraint','apprentice','internshipContract'}
<b>Documentation</b>	<b>Translation (it):</b> Modalità di lavoro in azienda <b>Short description (it):</b> Indicare la modalità dell'azienda che lavora <b>Domain description (it):</b> Valori possibili: indefiniteSalaryMth = contratto a tempo indeterminato con stipendio mensile (indefinite duration with salary per month) indefiniteSalaryMthAWT = contratto a tempo indeterminato con stipendio mensile e l'orario di lavoro annuale (indefinite duration with salary per month and annual working time model) indefiniteSalaryHrs = contratto a lungo termine a tempo indeterminato con salario orario (indefinite duration with salary per month) indefiniteSalaryNoTimeConstraint =

contratto a tempo indeterminato con acconto, somma forfettaria, salario a cottimo (indefinite duration with commission, lump sum, piece rate) fixedSalaryMth = contratto a tempo determinato con paga mensile (fixed/temporary duration salary per month) fixedSalaryHrs = contratto a tempo determinato con retribuzione oraria (fixed/temporary duration salary per hour) fixedSalaryNoTimeConstraint = contratto a tempo determinato con accantonamento, somma forfettaria, stipendio per posto di lavoro (fixed/temporary duration with commission, lump sum, piece rate) apprentice = accordo di apprendimento internshipContract = contratto di tirocinio

## 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**

<b>Name</b>	<b>EmptySimpleType</b>
<b>Content</b>	Built-in XSD Type: string <i>value</i> = {""}

## 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**



<b>Name</b>	<b>FaultCodeType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN <i>value</i> = {'NOT_accepted' 'NOT_plausible' 'NOT_valid'}
<b>Documentation</b>	<b>Translation (it):</b> Codice di errore

### 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**

<b>Name</b>	<b>HoursOrLessonsType</b>
<b>Content</b>	Built-in XSD Type: decimal <i>pattern</i> = [0-9]+\.[0-9]{2}
<b>Documentation</b>	<b>Translation (it):</b> Ore o lezioni

### 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**

<b>Name</b>	<b>IDType</b>
<b>Content</b>	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

<b>Name</b>	<b>InstanceRefIDType</b>
<b>Content</b>	Built-in XSD Type: string <i>pattern</i> = #.*
<b>Documentation</b>	<b>Translation (it):</b> 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

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

## Schema Component Representation

```
<xs:simpleType name="LanguageCodeType">
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="de"/>
  </xs:restriction>
</xs:simpleType>
```

```

<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**

<b>Name</b>	<b>MonitoringIDType</b>
<b>Content</b>	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**

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

#### 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**

<b>Name</b>	<b>NationalityType</b>
<b>Content</b>	Built-in XSD Type: string <i>pattern</i> = [A-Z][A-Z]11 22
<b>Documentation</b>	<p><b>Translation (it):</b> Nazionalità</p> <p><b>Short description (it):</b> I codici per l'attribuzione corretta della cittadinanza sono disponibili nella norme ONU (ISO 3166)</p> <p><b>Technical description (it):</b> Estensione 11= sconosciuto 22= apolide : I codici ISO non sono mai stati controllati sul Prod-Distri Nella plausibilità del livello di qualità può essere possibile verificare la plausibilità rispetto ai codici ISO 3166 e visualizzare un'avvertenza.</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**

<b>Name</b>	<b>NotEmptyStringType</b>
<b>Content</b>	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>
```

### Documentation: Schema Component Representation

## 2.74 Simple Type: **PercentType**

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

### 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**

<b>Name</b>	<b>PositionType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN <i>value</i> = {'highestCadre' 'middleCadre' 'lowerCadre' 'lowestCadre' 'noCadre'}
<b>Documentation</b>	<b>Translation (it):</b> Posizione professionale <b>Short description (it):</b> Posizione professionale del dipendente <b>Domain description (it):</b> Valori possibili: highestCadre: Quadro superiore; middleCadre: Quadro medio; lowerCadre: Quadro inferiore; lowestCadre: Responsabile dell'esecuzione di lavori; noCadre: Senza funzione di quadro;

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

```

<xs:enumeration value="lowestCadre"/>
<xs:enumeration value="noCadre"/>
</xs:restriction>
</xs:simpleType>

```

## Documentation: Schema Component Representation

### 2.90 Simple Type: **QualityLevelType**

<b>Name</b>	<b>QualityLevelType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN value = {'Validity' 'Plausibility' 'Acceptance' 'Comment'}
<b>Documentation</b>	<p><b>Translation (it):</b> Livelli di qualità</p> <p><b>Short description (it):</b> Livelli di qualità della trasmissione</p> <p><b>Technical description (it):</b> Ci sono i seguenti livelli: Validity: primo livello; Plausibility: secondo livello; Acceptance: terzo livello; Comment: quarto piano;</p>

## 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**

<b>Name</b>	<b>ResidenceCategoryType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN value = {'shortTerm-L' 'annual-B' 'settled-C' 'crossBorder-G' 'asylumSeeker-N' 'needForProtection-S' 'NotificationProcedureForShorttermWork90Days' 'NotificationProcedureForShorttermWork120Days' 'ProvisionallyAdmittedForeigners-F' 'ResidentForeignNatio

	nalWithGainfulEmployment-Ci' othersNotSwiss'}
<b>Documentation</b>	<p><b>Translation (it):</b> Categoria di permesso</p> <p><b>Short description (it):</b> Categoria di permesso di soggiorno</p> <p><b>Domain description (it):</b> Valore possibile: shortTerm-L = Dimorante temporaneo (L); annual-B = Dimorante annuale (B); settled-C = Domiciliato (C); crossBorder-G = Frontaliere (G); asylumSeeker-N = Richiedente d'asilo (N); needForProtection-S = Bisognoso di protezione (S); NotificationProcedureForShortterm-Work90Days = Procedura di notifica per attività lucrativa di breve durata (90 giorni); NotificationProcedureForShortterm-Work120Days = Procedura di notifica per attività lucrativa di breve durata (120 giorni); ProvisionallyAdmittedForeigners (F) = Persona ammessa provvisoria-mente (F); ResidentForeign-NationalWithGainfulEmployment (Ci) = Permesso di dimora con attività lucrativa (Ci); othersNotSwiss = Altro (non sviz-zero);</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

<b>Name</b>	<b>ResponseCodeAAwRType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN value = {'accepted' 'acceptedWithWarning' 'rejected'}
<b>Documentation</b>	<p><b>Translation (it):</b> Codice di risposta</p>

## 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

<b>Name</b>	<b>ResponseCodeType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN <i>value</i> = {'accepted' 'acceptedWithWarning'}
<b>Documentation</b>	<b>Translation (it):</b> Codice di risposta

## 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

<b>Name</b>	<b>SalaryAmountAbsoluteType</b>
<b>Content</b>	Built-in XSD Type: decimal <i>pattern</i> = [\-]?[0-9]+\.[0-9]{2} <i>pattern</i> = [0-9]+\.[0-9]{2}
<b>Documentation</b>	<b>Translation (it):</b> Salario assoluto Tipo <b>Short description (it):</b> Salario assoluto Tipo non firmato



## 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**

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

## 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**

<b>Name</b>	<b>SectionIDType</b>
<b>Content</b>	Built-in XSD Type: string <i>pattern</i> = #.*
<b>Documentation</b>	<b>Translation (it):</b> Identificazione Sezione tipo

## 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**

<b>Name</b>	<b>SexType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN <i>value</i> = {'F' 'M'}
<b>Documentation</b>	<b>Translation (it):</b> Tipo di genere <b>Short description (it):</b> F femminile; M maschile

## 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**

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

## Schema Component Representation

```
<xs:simpleType name="SimpleBooleanType">
  <xs:restriction base="xs:boolean">
    <xs:pattern value="true"/>
    <xs:pattern value="false"/>
  </xs:restriction>
</xs:simpleType>
```

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

## Documentation: Schema Component Representation

### 2.56 Simple Type: **StandardFormIDType**

<b>Name</b>	<b>StandardFormIDType</b>
<b>Content</b>	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}
<b>Documentation</b>	<b>Translation (it):</b> Formulario standard 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**

<b>Name</b>	<b>SV-AS-NumberType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN <i>pattern</i> = [0-9]{3}\.[0-9]{4}\.[0-9]{4}\.[0-9]{2} <i>length</i> <= 16
<b>Documentation</b>	<b>Translation (it):</b> Tipo il numero assicurazione sociale è di 13 cifre

## 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**

<b>Name</b>	<b>TaxAtSourceCodeType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN <i>pattern</i> = [A-Z][0-9][Y,N]
<b>Documentation</b>	<b>Translation (it):</b> Codice tariffario IF

#### 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**

<b>Name</b>	<b>UID-BFSType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN <i>pattern</i> = CHE-[0-9]{3}\.[0-9]{3}\.[0-9]{3}
<b>Documentation</b>	<b>Translation (it):</b> Numero d'identificazione IDI-UST

#### 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**

<b>Name</b>	<b>YesNoUnknownType</b>
-------------	-------------------------

<b>Content</b>	Built-in XSD Type: NMTOKEN <i>value</i> = {'yes' 'no' 'unknown'}
<b>Documentation</b>	<b>Translation (it):</b> sì, no o sconosciuto

### 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

<b>Name</b>	<b>ZIP-CodeType</b>
<b>Content</b>	Built-in XSD Type: string <i>length</i> >= 1
<b>Documentation</b>	<b>Translation (it):</b> Codice postale <b>Short description (it):</b> Codice postale

### 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