



# Directives pour les receveurs des donnees salariales

## 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\_fr.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 17:22

## 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: Échéance

## 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> Ligne complémentaire <b>Short:</b> Ligne supplémentaire pour l'adresse postale
Street	<b>Translation:</b> Rue <b>Short:</b> Rue et n° maison
Postbox	<b>Translation:</b> Case postale <b>Short:</b> Case postale
Locality	<b>Translation:</b> Localité <b>Short:</b> Localité (région, province, etc.)
ZIP-Code	<b>Translation:</b> Code postal <b>Short:</b> Code postal
City	<b>Translation:</b> Ville <b>Short:</b> Ville

Country	<b>Translation:</b> Pays <b>Short:</b> Pays
---------	--

## 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> Ligne complémentaire <b>Short:</b> Ligne supplémentaire pour l'adresse postale
Street	<b>Translation:</b> Rue <b>Short:</b> Rue et n° maison
Locality	<b>Translation:</b> Localité <b>Short:</b> Localité (région, province, etc.)
ZIP-Code	<b>Translation:</b> Code postal <b>Short:</b> Code postal
City	<b>Translation:</b> Ville <b>Short:</b> Ville
Country	<b>Translation:</b> Pays <b>Short:</b> Pays

## 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> vieux identifiant
NewCredentials	<b>Translation:</b> nouveau identifiant

## 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> Informations concernant l'application <b>Short:</b> Description des données essentielles d'identification de système <b>Technical:</b> Pour l'assurance de la qualité, les informations essentielles des participants sont sauvegardées.
SystemDateTime	<b>Translation:</b> Temps du système actuelle

## 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> Informations concernant l'application <b>Short:</b> Description des données essentielles d'identification de système <b>Technical:</b> Pour l'assurance de la qualité, les informations essentielles des participants sont sauvegardées.
UmlautString	<b>Translation:</b> réseau de tréma <b>Short:</b> Arrangement des trémas pour tester les majuscules et les minuscules.
FirstOperand	<b>Translation:</b> 1. taille d'essai spécifiée <b>Short:</b> 1. l'opérande est ajoutée avec 2. l'opérande pour vérifier qu'aucune erreur de type de données ne se produit.
SecondOperand	<b>Translation:</b> 2. taille d'essai spécifiée <b>Short:</b> La 2ème opérande est ajoutée avec la 1ère opérande pour vérifier qu'aucune erreur de type de données ne se produit.
SystemDateTime	<b>Translation:</b> Temps du système actuelle
MonitoringID	<b>Translation:</b> Identificateur de surveillance <b>Short:</b> Identificateur de surveillance <b>Technical:</b> L'identificateur de surveillance est surtout utilisé dans l'application de référence pour le classement des données.

## 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> État civil <b>Short:</b> État civil de la personne
ValidAsOf	<b>Translation:</b> L'état civil est valable à partir de <b>Short:</b> L'état civil de la personne est valable à partir de

## 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> Nom <b>Short:</b> Nom de l'entreprise
Owner	<b>Translation:</b> Propriétaire de l'entreprise <b>Short:</b> Propriétaire de l'entreprise
Address	<b>Translation:</b> Adresse <b>Short:</b> Adresse de l'entreprise
UID-BFS	<b>Translation:</b> Numéro d'identification IDE-OFS <b>Short:</b> Numéro d'identification d'entreprise / IDE-OFS
Delegate	<b>Translation:</b> Remplaçant <b>Short:</b> Remplaçant de l'entreprise

## 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> Nom ou nom RC/UID-OSF <b>Short:</b> Nom ou nom RC de l'entreprise/UID-OSF
ComplementaryLine	<b>Translation:</b> Ligne complémentaire <b>Short:</b> Lignes complémentaires pour noms des divisions, désignations des filiales, etc.

## 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> Prénom <b>Short:</b> Prénom du propriétaire de l'entreprise
Lastname	<b>Translation:</b> Nom <b>Short:</b> Nom du propriétaire de l'entreprise

## 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**

<b>Name</b>	<b>CompanyWorkingTimeRefType</b>
-------------	----------------------------------

#### 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**

<b>Name</b>	<b>ContactInstitutionType</b>
-------------	-------------------------------

#### 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	<b>Translation:</b> Nom
EmailAddress	<b>Translation:</b> Adresse e-mail
PhoneNumber	<b>Translation:</b> Numéro de téléphone
MobilePhoneNumber	<b>Translation:</b> Numéro de téléphone mobile

### 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: Clé
Password	Translation: Mot de passe

## 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: Nom Short: Nom du mandataire
Owner	Translation: Propriétaire de l'entreprise Short: Propriétaire de l'entreprise
Address	Translation: Adresse Short: Adresse de l'entreprise
UID-BFS	Translation: Numéro d'identification IDE-OFS Short: Numéro d'identification d'entreprise / IDE-OFS

## 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: Message de dialogue

Short: Un format d'échange de données simple et flexible

## 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 (fr):</b> Erreur État

#### 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> Erreur: Code de l'état
Error	<b>Translation:</b> Erreur
Warning	<b>Translation:</b> Avertissement
Info	<b>Translation:</b> Informations

### 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 (fr):</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: Nom
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: Remarque

## 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: Niveau de qualité

DescriptionCode

Translation: Code de la description

Description

Translation: Description

## 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> Problème <b>Short:</b> Signaler les problèmes lors de la réponse
optional	<b>Translation:</b> réponse facultative <b>Short:</b> Marquage pour une réponse facultative

## 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> Identification
sectionIDRef	<b>Translation:</b> Référence Section Identificateur

## 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	<b>Translation:</b> Numéro d'assurance sociale <b>Short:</b> Numéro d'assurance sociale
EmployeeNumber	<b>Translation:</b> Numéro de personnel <b>Short:</b> L'identification des personnes est en règle générale numérique, et le

	fabricant du logiciel ou l'entreprise peut le choisir librement (par ex. 1254).
Lastname	<b>Translation:</b> Nom <b>Short:</b> Nom de famille
Firstname	<b>Translation:</b> Prénom <b>Short:</b> Prénom(s) de la personne
Sex	<b>Translation:</b> Sexe
DateOfBirth	<b>Translation:</b> Date de naissance <b>Short:</b> Date de naissance de la personne
Nationality	<b>Translation:</b> Nationalité <b>Short:</b> Les codes permettant une attribution cor-recte de la nationalité sont déposés au-près de l'ONU (ISO 3166) <b>Technical:</b> Élargissement 11=inconnu 22=apatride : Les codes ISO n'ont jamais été vérifiés sur le Prod-Distri auparavant.: Dans le contrôle de qualité, niveau plausibilité, le document est testé contre les codes ISO 3166 et un avertissement est généré au cas d'un code inconnu.
CivilStatus	<b>Translation:</b> État civil <b>Short:</b> État civil de la personne

## 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: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	<b>Translation:</b> Informations concernant l'application <b>Short:</b> Description des données essentielles d'identification de système <b>Technical:</b> Pour l'assurance de la qualité, les informations essentielles des participants sont sauvegardées.
-----------	--

## 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	<b>Translation:</b> Informations concernant l'application <b>Short:</b> Description des données essentielles d'identification de système <b>Technical:</b> Pour l'assurance de la qualité, les informations essentielles des participants sont sauvegardées.
Timestamp	<b>Translation:</b> Horodatage <b>Short:</b> L'horodatage (en anglais timestamping) est un mécanisme qui consiste à associer une date et une heure à un événement, une information ou une donnée informatique. Il a généralement pour but d'enregistrer l'instant auquel une opération a été effectuée.
NextCheck	<b>Translation:</b> prochain contrôle
RegisteredMaintenance	<b>Translation:</b> Fenêtre de maintenance enregistrée

## 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> Informations concernant l'application <b>Short:</b> Description des données essentielles d'identification de système <b>Technical:</b> Pour l'assurance de la qualité, les informations essentielles des participants sont sauvegardées.
SystemDateTime	<b>Translation:</b> Temps du système actuelle

## 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> Informations concernant l'application <b>Short:</b> Description des données essentielles d'identification de système <b>Technical:</b> Pour l'assurance de la qualité, les informations essentielles des participants sont sauvegardées.
SystemDateTime	<b>Translation:</b> Temps du système actuelle
MonitoringID	<b>Translation:</b> Identificateur de surveillance <b>Short:</b> Identificateur de surveillance <b>Technical:</b> L'identificateur de surveillance est surtout utilisé dans l'application de référence pour le classement des données.

## 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: Message pour l'émetteur

## 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: Rubrique
Description	Translation: Description
sectionID	Translation: Identificateur Section

## 2.23 Complex Type: Social-InsurancIdentificationType

Name	Social-InsurancIdentificationType
------	-----------------------------------

### Schema Component Representation

```
<xs:complexType name="Social-InsurancIdentificationType">
  <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: Création
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: depuis
until	Translation: jusqu'à

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

<b>Name</b>	<b>UID-BFS-UnknownType</b>
-------------	----------------------------

#### 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**

<b>Name</b>	<b>UserAgentType</b>
-------------	----------------------

#### 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: Producteur de l'application
Name	Translation: Nom du produit
Version	Translation: Version du produit
StandardVersion	Translation: Swissdec Version standard
Certificate	Translation: Certificat

## 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> Durée de travail hebdomadaire individuelle (heures)
WeeklyLessons	<b>Translation:</b> Durée de travail hebdomadaire individuelle (leçons)
companyWeeklyHoursAndLessonsIDRef	<b>Translation:</b> référence aux horaires et leçons hebdomadaires prévus par l'entreprise pour le lieu de travail en question

## 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> Heures par semaine <b>Short:</b> Nombre d'heures par semaine
WeeklyLessons	<b>Translation:</b> Leçons hebdomadaires applicables <b>Short:</b> Leçons hebdomadaires applicables au lieu

## 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> Adresse géographique <b>Short:</b> Adresse géographique des lieux de travail
CompanyWorkingTime	<b>Translation:</b> Heures de travail <b>Short:</b> Heures de travail
workplaceID	<b>Translation:</b> Identificateur du lieu de travail

## 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> Durée de travail <b>Short:</b> Durée de travail fixée individuellement
EntryDate	<b>Translation:</b> Date de l'entrée <b>Short:</b> Date d'entrée dans l'entreprise
WithdrawalDate	<b>Translation:</b> Date de sortie <b>Short:</b> Date de sortie de l'entreprise

**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> Remarques
--------------	-------------------------------

**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> Informations concernant l'application <b>Short:</b> Description des données essentielles d'identification de système <b>Technical:</b> Pour l'assurance de la qualité, les informations essentielles des participants sont sauvegardées.
CompanyName	<b>Translation:</b> Nom de l'entreprise <b>Short:</b> Description des données clés de l'entreprise

TransmissionDate	<b>Translation:</b> Date et Temps de transmission
RequestID	<b>Translation:</b> Identificateur de la demande
LanguageCode	<b>Translation:</b> Code de langage
MonitoringID	<b>Translation:</b> Identificateur de surveillance <b>Short:</b> Identificateur de surveillance <b>Technical:</b> L'identificateur de surveillance est surtout utilisé dans l'application de référence pour le classement des données.

## 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}
<b>Documentation</b>	<b>Translation (fr):</b> Type de catégorie d'assurance <b>Translation (fr):</b> Tipo Codice categoria assicurativa

### 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 (fr):</b> Type Numéro REE

### 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>  {'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'  }'</p> <p><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'}</p>
<b>Documentation</b>	<p><b>Translation (fr):</b>  Type de canton</p> <p><b>Short description (fr):</b>  Type de canton Suisse</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>	<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>
<b>Documentation</b>	<p><b>Translation (fr):</b></p> <p>Type de canton Suisse (EX inclu)</p> <p><b>Short description (fr):</b></p> <p>Type de canton Suisse, EX inclu (EXterne ou EXPatriates) qui n'habitent pas en Suisse</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**

<b>Name</b>	<b>CategoryPredefinedType</b>
<b>Content</b>	Built-in XSD Type: NMTOKEN value = {'HEN' 'HEY' 'MEN' 'MEY' 'NON' 'NOY' 'SFN'}
<b>Documentation</b>	<b>Translation (fr):</b> Catégories prédéfinies <b>Short description (fr):</b> Catégories prédéfinies <b>Domain description (fr):</b> Valeurs possibles: HEN, HEY: Honoraires de membres du CA aux PIS domiciliés à l'étranger; MEN, MEY: Prestations issues de participations de collaborateur en faveur de PIS domiciliés à l'étranger; NON, NOY: Non soumis à l'impôt à la source; SFN: Accord spécial avec la France;

## 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 value = {'unknown' 'single' 'married' 'widowed' 'divorced' 'separated' 'registeredPartner-ship' 'partnershipDissolvedByLaw' 'partnershipDissolvedByDeath' 'partnershipDissolvedByDeclarationOfLost'}
<b>Documentation</b>	<b>Translation (fr):</b>

Type d'état civil

**Short description (fr):**

Type d'état civil

**Domain description (fr):**

Valeurs possibles: unknown = inconnu; single = célibataire; married = marié(e); widowed = veuve/veuf; divorced = divorcé(e); separated = séparé(e); registeredPartnership = partenariat enregistré; partnershipDissolvedByLaw = partenariat dissous judiciairement; partnershipDissolvedByDeath = partenariat dissous par décès; partnershipDissolvedByDeclarationOfLost = partenariat dissous ensuite de déclaration d'absence;

## 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>	<p><b>Translation (fr):</b> Code de la description</p> <p><b>Technical description (fr):</b> Severity: Information Le code 9999 est toujours réservé pour un texte libre. Le reste des codes sont spécifiés dynamique dans AcknowledgementNotification.</p>

## 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> = [^@]+@[^\.]+\.\.+
<b>Documentation</b>	<b>Translation (fr):</b> Adresse mél

## 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 (fr):</b> Mode du company de travail <b>Short description (fr):</b> Indiquez le mode du company de travail <b>Domain description (fr):</b> Valeurs possible: indefiniteSalaryMth = contrat à durée indéterminée avec salaire mensuel (indefinite duration with

salary per month) indefiniteSalaryMthAWT = contrat à durée indéterminée avec salaire mensuel et temps du travail annuel (indefinite duration with salary per month and annual working time model) indefiniteSalaryHrs = contrat à durée indéterminée avec salaire horaire (indefinite duration with salary per month) indefiniteSalaryNoTimeConstraint = contrat à durée indéterminée avec provision, forfait, salaire à la tâche (indefinite duration with commission, lump sum, piece rate) fixedSalaryMth = contrat à durée déterminée avec salaire mensuel (fixed/temporary duration salary per month) fixedSalaryHrs = contrat à durée déterminée avec salaire horaire (fixed/temporary duration salary per hour) fixedSalaryNoTimeConstraint = contrat à durée déterminée avec provision, forfait, salaire à la tâche (fixed/temporary duration with commission, lump sum, piece rate) apprentice = contrat d'apprentissage internshipContract = contrat de stage

## 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 (fr):</b> Code de l'erreur

#### 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 (fr):</b> Heures ou leçons

#### 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 (fr):</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 (fr):</b> Code de langage

## 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**

<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 (fr):</b> Numéro de la commune

## Schema Component Representation

```
<xs:simpleType name="MunicipalityIDType">
  <xs:restriction base="xs:int">
    <xs:minInclusive value="1"/>
  </xs:restriction>
</xs:simpleType>
```

```

<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 (fr):</b> Nationalité</p> <p><b>Short description (fr):</b> Les codes permettant une attribution cor-recte de la nationalité sont déposés au-près de l'ONU (ISO 3166)</p> <p><b>Technical description (fr):</b> Élargissement 11=inconnu 22=apatride : Les codes ISO n'ont jamais été vérifiés sur le Prod-Distri auparavant.: Dans le contrôle de qualité, niveau plausibilité, le document est testé contre les codes ISO 3166 et un avertissement est généré au cas d'un code inconnu.</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 (fr):</b> Pourcent

## 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 (fr):</b> Position professionnelle <b>Short description (fr):</b> Position professionnelle de la personne salariée <b>Domain description (fr):</b> Valeurs possibles: highestCadre: Cadre supérieur; middleCadre: Cadre moyen; lowerCadre: Cadre inférieur; lowestCadre: Responsable de l'exécution des travaux; noCadre: Sans fonction de cadre;

## 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**

<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 (fr):</b> Niveaux de qualité</p> <p><b>Short description (fr):</b> Niveaux de qualité de la transmission</p> <p><b>Technical description (fr):</b> Il existe les niveaux suivants: Validity: premier niveau; Plausibility: deuxième niveau; Acceptance: troisième niveau; Comment: quatrième niveau;</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>	<p>Built-in XSD Type: NMTOKEN</p> <p><i>value</i> =</p> <p>{'shortTerm-L' 'annual-B' 'settled-C' 'crossBorder-G' 'asylumSeeker-N' 'needForProtection-S' 'NotificationProcedureForShorttermWork90Days' 'NotificationProcedureForShorttermWork120Days' 'ProvisionallyAdmittedForeigners-F' 'ResidentForeignNationalWithGainfulEmployment-Ci' 'othersNotSwiss'}</p>
<b>Documentation</b>	<p><b>Translation (fr):</b></p> <p>Catégories de permis de séjour</p> <p><b>Short description (fr):</b></p> <p>Catégories de permis de séjour pour étrangers</p> <p><b>Domain description (fr):</b></p> <p>Valeurs possibles: shortTerm-L = Permis de courte durée (cat. L); annual-B = Permis annuel (cat. B); settled-C = Permis d'établissement (cat. C) ; crossBorder-G = Frontaliers (cat. G) ; asylumSeeker-N = requérants d'asile (N); needForProtection-S = Personnes à protéger (S); NotificationProcedureForShorttermWork90Days = procédure d'annonce pour les activités lucratives de courte durée (90 jours); NotificationProcedureForShorttermWork120Days = procédure d'annonce pour les activités lucratives de courte durée (120 jours); ProvisionallyAdmittedForeigners (F) = étrangers admis provisoirement (F); ResidentForeignNationalWithGainfulEmployment (Ci) = permis de séjour avec activité lucrative (Ci); othersNotSwiss = Autres (sans les Suisses) ;</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 <i>value</i> = {'accepted' 'acceptedWithWarning' 'rejected'}
<b>Documentation</b>	<b>Translation (fr):</b> Code de la réponse

### 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 (fr):</b> Code de la réponse

### 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 (fr):</b> Absolu type de montant de salaire <b>Short description (fr):</b> Type de montant de salaire non signé

### 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 (fr):</b> Type de montant de salaire

### 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 (fr):</b> Identificateur Section type

## 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 (fr):</b> Type de sexe <b>Short description (fr):</b> F féminin; M masculin

## 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 (fr):</b> SimpleBoolean <b>Short description (fr):</b> Booléen

## 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**

<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 (fr):</b> Formulaire standard d'identification

## 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 (fr):</b> Type N° d'AS à treize chiffres

## 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 (fr):</b> Barème IS

## 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 (fr):</b> Type Numéro d'identification IDE-OFS

## 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 (fr):</b> oui, non ou inconnu

### 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 (fr):</b> Code postal <b>Short description (fr):</b> Code postal

### 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

