Europass XML Schema V3.1.0

Documentation and Change-log from v3.0.3 version
Table of Contents

Europass Supporting Semantic Interoperability .............................................................................. 1
Europass XML Vocabulary .................................................................................................................. 1
Structure of this Document ................................................................................................................ 1
1  Change-log ................................................................................................................................... 2
   1.1  Change-log from version v3.0.3 ......................................................................................... 2
2  Namespaces (Main, included and imported) .................................................................................. 6
   2.1  Europass Namespace .............................................................................................................. 6
   2.2  Imported Namespaces ............................................................................................................. 8
3  Versioning of Europass XML Schema Definitions ........................................................................ 9
4  Europass XML Document elements; their structure and data types ............................................. 10
   4.1  Skills Passport ....................................................................................................................... 10
   4.2  SkillsPassport/locale ............................................................................................................. 11
   4.3  SkillsPassport/DocumentInfo ............................................................................................... 13
       4.3.1  DocumentInfo/XSDVersion ......................................................................................... 15
   4.4  SkillsPassport/PrintingPreferences ..................................................................................... 15
   4.5  SkillsPassport/LearnerInfo .................................................................................................. 18
       4.5.1  SkillsPassport/LearnerInfo/Identification .................................................................... 20
       4.5.2  SkillsPassport/LearnerInfo/Headline .......................................................................... 34
       4.5.3  SkillsPassport/LearnerInfo/WorkExperienceList and WorkExperience ....................... 36
       4.5.4  SkillsPassport/LearnerInfo/EducationList and Education ........................................... 40
       4.5.5  SkillsPassport/LearnerInfo/Skills ............................................................................... 46
       4.5.6  SkillsPassport/LearnerInfo/AchievementList and Achievement ................................. 59
       4.5.7  SkillsPassport/LearnerInfo/Documentation and ReferenceTo ...................................... 61
   4.6  SkillsPassport/AttachmentList ............................................................................................. 63
   4.7  SkillsPassport/CoverLetter .................................................................................................. 65
       4.7.1  SkillsPassport/CoverLetter/Addressee ......................................................................... 66
       4.7.2  SkillsPassport/CoverLetter/Letter ............................................................................... 68
       4.7.3  SkillsPassport/CoverLetter/Documentation .................................................................. 72
5  Generic Data Types ....................................................................................................................... 74
   5.1  LabelType ............................................................................................................................. 74
   5.2  OccupationalFieldType ......................................................................................................... 74
   5.3  AddressInfoType .................................................................................................................. 75
<table>
<thead>
<tr>
<th></th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.4</td>
<td>CountryType</td>
</tr>
<tr>
<td>5.5</td>
<td>ContactAddressType</td>
</tr>
<tr>
<td>5.6</td>
<td>OrganisationType</td>
</tr>
<tr>
<td>5.7</td>
<td>AbstractExperienceListType</td>
</tr>
<tr>
<td>5.8</td>
<td>ExperienceListType</td>
</tr>
<tr>
<td>5.9</td>
<td>ExperienceType</td>
</tr>
<tr>
<td>5.10</td>
<td>PeriodType</td>
</tr>
<tr>
<td>5.11</td>
<td>DateType</td>
</tr>
<tr>
<td>5.12</td>
<td>IntraDocumentDocumentationType</td>
</tr>
<tr>
<td>5.13</td>
<td>InterDocumentDocumentationType</td>
</tr>
<tr>
<td>5.14</td>
<td>ExtraDocumentDocumentationType</td>
</tr>
<tr>
<td>5.15</td>
<td>FileDataType</td>
</tr>
<tr>
<td>5.16</td>
<td>SkillType</td>
</tr>
<tr>
<td>5.17</td>
<td>CEFLanguageLevelType</td>
</tr>
</tbody>
</table>
Europass XML Schema V3.1.0

Documentation and Change-log from v3.0.3

Europass Supporting Semantic Interoperability

Europass promotes semantic interoperability by defining a specific vocabulary according to which the information contained in Europass Documents is expressed. This vocabulary is defined according to an XML schema, which describes the constraints on the structure and on the contents of Europass Documents.

Individuals, who use the Europass Online Editors to produce their Europass Curriculum Vitae or Europass Language Passport, have the option to receive the document in Europass XML format or PDF format with the XML attached. The most important benefit of this option is that it allows individuals to reuse their personal data and save time and effort when accessing the Europass online editors at a later point, or other systems (job portals, employment/admission services, etc.) that “understand” the Europass vocabulary.

Europass XML Vocabulary

The Europass XML Vocabulary is considered to be relatively simple, clear and self-explaining, while remaining sound and extensible, and also as close as possible to other related vocabularies, as those defined by HR-XML.

This document provides a description of the latest version of the Europass XML Schema (V3.0.3) with the aim to support interested parties, who wish to manage, edit or store Europass documents within their information systems.

Structure of this Document

The document is structured as follows: First, an extensive reference to the changes and revisions from the previous version (v3.0.3) is attempted (part 1). Then, we begin the documentation of the latest version by listing the various imported and included XML Schemas, and explain that purpose each schema serves (Part 2). The main body of the document provides an in-depth description of each possible element of a Europass XML document. The order according to which the elements are described matches the hierarchy of the XML document, and starts with the root element of the XML. Each element of the Europass XML is described in detail, by emphasizing the allowed data type and providing an indicative example (Part 3). Finally, the XML data types that are either abstract, or generic or used to define multiple elements are described in detailed in a separate section, and referenced wherever necessary (Part 4).
1 Change-log

1.1 Change-log from version v3.0.3

In January-February 2014 Europass enriches Europass Documents by making available the Europass Cover Letter. The purpose of this document is to help individuals prepare a well-structured cover letter, which is advised to accompany the rest of the Europass Documents.

Similarly to the rest of the Europass Documents, the information contained in a Cover Letter is stored in the Europass XML instance accompanying the Document. For this reason the existing Europass XML Schema Definition is extended to accommodate the needs of defining a model for the Cover Letter.

The updated version is v3.1.0. Although the schema is fully backward compatible with the previous version, a minor version update is chosen, to illustrate the important changes in the schema. Those changes include 1) the enrichment of the schema with additional data types, specifically to model the Cover Letter set of information, 2) a breakdown of the main EuropassLearnerInformation schema to various separate sub-schemata in an effort to increase flexibility and reusability.

The main Europass XML Schema Definition is available under:

http://europass.cedefop.europa.eu/xml/v3.1.0/EuropassSchema.xsd

Backward compatibility

Version 3.1.0 is fully backward compatible with version 3.0.3. So, existing solutions producing Europass XML schemata will only need to update the reference to the new XSD.

The same goes for the Europass RESTful API. The change of the XML does not introduce any breaking changes to existing functionality.

Breakdown of XML Schema Definitions

Below we briefly describe the various XSD files and the nature of the elements, or data types each one defines.

EuropassSchema.xsd

This schema definition defines the root element of the Europass XML structure, SkillsPassport, and also the structure of its child elements. See 4.1 for details.

EuropassPrintingPreferences.xsd

This XML schema definition defines the structure of the SkillsPassport/PrintingPreferences element.

This schema does not introduce any structural changes from the previous version. It continues to define the elements and data types necessary in order to let an XML instance include information
about the layout and printing preferences of the generated Europass Document in ODT, DOC or PDF.

Some additions have taken place in this version, which are described in section 4.4.

Moreover, the included schema of Europass Address Formats that defines options for the “format” attribute of the element Field, is updated to version 1.2.0.

The updated XSD is http://europass.cedefop.europa.eu/xml/included/EuropassAddressFormats_V1.2.0.xsd

EuropassLearnerInformation.xsd

This XML schema definition defines the structure of the SkillsPassport/LearnerInfo element.

It contains the most changes, as its contents have been repositioned to numerous other schema definitions in an effort to increase flexibility and reusability and also improve readability and understanding.

Apart from this structural change there is almost no change in the existing data types (except for an addition in Person Name, see 4.5.1.1)

EuropassCoverLetter.xsd

This new XML schema definition defines the structure of the new SkillsPassport/CoverLetter element.

The definition includes and reuses other schema definitions to define the desired structure for the involved elements.

Details on the CoverLetter structure are given in section 4.7

ContactInformation.xsd:

AddressInfoType, the data type that defines the details of a contact/postal address is enriched to include a second address line.

DocumentInformation.xsd:

This new XML Schema definition defines the structure of the existing SkillsPassport/DocumentInfo element.

It also includes the simple data type for the VersionNumberType. See 4.3 for further details.

Refactored XML Schema Definitions

- CommonTypes.xsd: includes the definitions of the often reusable data types “LabelType”, “DateType”, “PeriodType”, “LocaleType”, “DocumentTypeType” and “CEFLanguageLevelType”.
PersonName.xsd: includes the definitions of the data types “PersonNameType”, “PersonTitleLabelType” and the simple data type “PersonTitleCodesEnumeration”. See 4.5.1.1 for details.

Language.xsd: includes the definitions of the abstract data type “LanguageType” and the extending data types “MotherLanguageType” and “ForeignLanguageType”.

Organisation.xsd: includes the definitions of the data types “OrganisationType”, “OrganisationalContactInfoType”, “OrganisationalWebsiteType”, “OrganisationalWebsiteUseType”, “EmployerType” and “BusinessSectorType”.

Certificate.xsd: includes the definitions of the data types “CertificateType” and “CertificateListType”.

Identification.xsd: includes the definitions of the data type “IdentificationType”. See 4.5.1 for details.

ContactInformation.xsd: includes the definitions of the data types “ContactInfoType”, “ContactMethodType”/“ContactMethodListType”, “TelephoneType”/“TelephoneListType”, “WebsiteType”/“WebsiteListType”, “InstantMessagingType”/“InstantMessagingListType”, “InstantMessagingUsesEnumeration”, “WebsiteUsesEnumeration”, “TelephoneUsesEnumeration”, “ContactAddressType”/“AddressInfoType”, “ContactEmailType”/“ContactEmailContactType”. See 4.5.1.2 for details.

Demographics.xsd: includes the definitions of the data types “DemographicsType”, “GenderLabelType” and “NationalityType”. See 4.5.1.4 for details.

Headline.xsd: includes the definitions of the data types “HeadlineType”, “HeadlineTypeLabelType”, “HeadlineTypesEnumeration”, “PositionType” and “PositionTypeLabelType”. See 4.5.2 for details.

OccupationalField.xsd: includes the definitions of the data types “OccupationalFieldType”. See 4.5.2 and 4.5.3.3 for details.

Experience.xsd: includes the definitions of the data types “ExperienceType”, “ExperienceListType” and the abstract data type “AbstractExperienceListType”.

EducationalExperience.xsd: includes the definitions of the data types “EducationalExperienceType”/“EducationalExperienceListType”, “EducationalLevelType” and “EducationalFieldType”. See 4.5.4 for details.

EmploymentExperience.xsd: includes the definitions of the data types “WorkExperienceType” and “WorkExperienceListType”. See 4.5.3 for details.

Skill.xsd: includes the definitions of the abstract data type “SkillType” and the base data type “GenericSkillType”.

DrivingSkill.xsd: includes the definitions of the data type “DrivingSkillType”. See 4.5.5.8 for details.

LinguisticSkill.xsd: includes the definitions of the base data type “LinguisticSkillType” and the extending “MotherTongueSkillType” and “ForeignLanguageSkillType” data types. Aslo it includes the data types “ProficiencyLevelType”, “LinguisticCertificateType”/“LinguisticCertificateListType”, “LinguisticExperienceType”/“LinguisticExperienceListType”, “LinguisticExperienceAreaType” and the simple type “LinguisticExperienceAreasEnumeration”. See 4.5.5.1 for details.
• **Achievement.xsd**: includes the definitions of the data types “AchievementType” and “AchievementListType”. See 4.5.6 for details.

• **Documentation.xsd**: includes the definitions of the data types “EuropassDocumentReferenceType”, “InternalReferenceType” and “ExternalReferenceType”, as well as “GenericDocumentationType”, “IntraDocumentDocumentationType”, “InterDocumentDocumentationType” and “ExtraDocumentDocumentationType”.

• **DigitalContent.xsd**: includes the definitions of the data types “AttachmentType”/ “AttachmentListType”, “FileDataType”, “MetadataListType”, “PhotoDataType”, the new data type “SignatureDataType” and the simple types “MimeTypeEnumeration” and “ImageMimeTypeEnumeration”. See 4.5.6 for details.

• **Letter.xsd**: includes the definitions of the data types “AchievementType” and “AchievementListType”. See 4.5.6 for details.

**Document Types**

The **DocumentTypeType** data type restricts the simple string by defining an enumeration. It is enriched to include the “ECL” text.

**Person Names**

The **PersonNameType** has been enriched to include an optional element “Title” that adheres to the newly defined type of **PersonTitleLabelType**.

**Signature**

The **IdentificationType** has been enriched to include an optional element “Signature” that adheres to the newly defined type of **SignatureDataType**.
2 Namespaces (Main, included and imported)

2.1 Europass Namespace


Europass also maintains under the same namespace other formal vocabularies:

*Included Schemata under the Europass namespace*

1. Europass Learner Information:
   - Expresses information about a learner’s personal data, experiences and skills.
   - Advertised at http://europass.cedefop.europa.eu/xml/v3.1.0/ EuropassLearnerInformation.xsd

2. Europass Printing Preferences:
   - Expresses information on how to display the learner’s information in the generated document; more specifically defines the order of the sections, the way to format dates and addresses and finally defines what information to hide in the finally generated document. This is mainly used by the Europass online editors in order to allow the flexibility of retaining all information in the XML, while displaying the desired subset of it in the generated document.
   - Advertised at http://europass.cedefop.europa.eu/xml/v3.1.0/ EuropassPrintingPreferences.xsd

3. Europass Document Information
   - Expresses metadata information about the specific XML instance and Europass document.
   - Advertised at http://europass.cedefop.europa.eu/xml/v3.1.0/ DocumentInformation.xsd

4. Europass Cover Letter
   - Expresses the information about a learner’s cover letter used to accompany other Europass documents.
   - Advertised at http://europass.cedefop.europa.eu/xml/v3.1.0/ CoverLetter.xsd

5. Other schemata:
   - All of the schemata are advertised under http://europass.cedefop.europa.eu/xml/v3.1.0/
     - CommonTypes.xsd
     - PersonName.xsd
     - Language.xsd
     - Organisation.xsd
     - Certificate.xsd
     - Identification.xsd
     - ContactInformation.xsd
     - Demographics.xsd
6. **ISO 3166-1 list of countries adjusted by Europass**
   - Defines the ISO list of countries and provides translations to the Europass languages. Notable differences are the country codes of United Kingdom (UK instead of GB) and Greece (EL instead of GR).
   - Advertised at
     http://europass.cedefop.europa.eu/xml/included/EUROPASS_ISOCountries_V1.5.0.xsd

7. **List of nationalities**
   - Defines the list of nationalities, as a one-to-one correspondence with the list of countries. It also provides translations to the Europass languages.
   - Advertised at
     http://europass.cedefop.europa.eu/xml/included/EuropassNationalities_V1.5.0.xsd

8. **ISO 639-1 list of languages adjusted by Europass**
   - Defines the ISO list of languages and provides translations to the Europass languages. Also it organizes the language codes to those that may exist as mother tongues (excludes languages that are considered “dead” as Ancient Greek), and those that may exist as foreign language knowledge.
   - Advertised at
     http://europass.cedefop.europa.eu/xml/included/EUROPASS_ISOLanguages_V1.6.0.xsd

9. **ISCO 88 COM list of occupations adjusted by Europass**
   - The International Standard Classification of Occupations and the International Labour Organisation proposed the ISCO 88, and ISCO 88 COM is the European Union variant of ISCO 88. The list is further filtered by Europass by adding a fifth level.
   - The vocabulary provides translations in most of the Europass languages of the 5th level group codes, differentiated by gender wherever applicable
   - Advertised at
     http://europass.cedefop.europa.eu/xml/included/EUROPASS_ISCO_88_COM_V1.3.0.xsd
2.2 Imported Namespaces

In order to reuse information, Europass imports some external XML Schemas which define formal vocabularies. These vocabularies are lists of terms, developed by an independent authority and adopted by Europass. Europass creates and manages the corresponding XML Schemas. Since these vocabularies are not property of Europass, they are not contained in the same namespace.

1. **NACE List of Business Sectors**
   - Nomenclature statistique des activités économiques dans la Communauté européenne: List of NACE codes for business sectors, as listed in the "Competition" related pages of the European Commission’s website (http://ec.europa.eu/comm/competition/mergers/cases/index/n International Standard)
   - Advertised at: http://europass.cedefop.europa.eu/xml/imported/NACE_COM_V1.0.0.xsd

2. **List of Educational Fields**
   - Advertised at: http://europass.cedefop.europa.eu/xml/imported/ISCED97_V1.0.0.xsd

3. **List of European Driving Licence Codes**
   - Namespace: driving: http://europass.cedefop.europa.eu/EUDriving
   - Advertised at: http://europass.cedefop.europa.eu/xml/imported/EUDrivinglicence_V1.1.0.xsd

4. **European Qualifications Framework (EQF)**
   - The European Qualifications Framework (EQF) acts as a translation device to make national qualifications more readable across Europe, promoting workers’ and learners' mobility between countries and facilitating their lifelong learning. The core of the EQF concerns eight reference levels describing what a learner knows, understands and is able to do – 'learning outcomes'. Levels range from basic (Level 1) to advanced (Level 8). The EQF applies to all types of education, training and qualifications, from school education to academic, professional and vocational. (http://ec.europa.eu/education/lifelong-learning-policy/eqf_en.htm)
   - Namespace: eqf: http://europass.cedefop.europa.eu/EQF/08
   - Advertised at: http://europass.cedefop.europa.eu/xml/imported/EQF_08_V1.0.0.xsd
3 Versioning of Europass XML Schema Definitions

Europass XML Schema Definition is versioned according to the pattern Major.Minor.Patch.

Starting from version 3.1.0 the version of the schema is visible in the Http URL under which the schema is available.


The version of the Europass XML Schema changes according to the following:

- The **major** version will change when there are ground-breaking changes (such as the change from 2.0 to 3.0)
- The **minor** version will change when a new document is introduced that brings along substantial changes to the schema (such as the CoverLetter, v3.0.x to v3.1.y)
- The **patch** version will change when small non breaking changes are introduced, e.g. adding a new sub-element in a type definition, changing the version of an imported or included schema, etc.

**XSDVersion Element**

The Europass Schema defines that all XML instances need to include a specific element that denotes that compatibility of the specific document with a specific family of schema definitions.

This compatibility is expressed against a Major.Minor version. Not including the patch version here means that the specific XML instance should continue to be valid against the updated schema.

Europass Services use this specific element to understand what level of backward compatibility is supported for the specific document, and what kind of transformations need to be applied to the document so that it is brought to be valid against the latest schema.

The position of this element in an XML instance is: /SkillsPassport/DocumentInfo/XSDVersion and more details are given in 4.3.1.
4 Europass XML Document elements; their structure and data types

4.1 Skills Passport

SkillsPassport is the root element of any Europass XML document. This element actually defines an individual’s portfolio that includes information coming from the Curriculum Vitae, Language Passport or other Europass documents, as well other non-Europass documents.

It is the actual template describing and organizing the entire set of personal information, learning achievements and training periods, work experiences, skills and competences of the learner (LearnerInfo). Moreover, it defines the way that the learner’s information appears in a “printable” Europass document—considering the printing preferences about the order and the format—(PrintingPreferences). This element also includes all the digital documents attached to this XML document to server as supporting material to this portfolio (AttachmentList). Finally, the SkillsPassport also contains some metadata about the XML document itself organized under a suitable sub-element (DocumentInfo).

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Child Element</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Content Type</td>
<td></td>
</tr>
<tr>
<td>Sequence(S):</td>
<td>The child elements in the XML document MUST appear in the order they are declared in the XSD schema.</td>
<td></td>
</tr>
<tr>
<td>Choice(C):</td>
<td>Only one of the child elements described in the XSD schema can appear in the XML document.</td>
<td></td>
</tr>
<tr>
<td>All(A):</td>
<td>The child elements described in the XSD schema can appear in the XML document in any order.</td>
<td></td>
</tr>
</tbody>
</table>
SkillsPassport is the root element of any Europass XML document. This element actually defines a person’s portfolio that includes information coming from the Curriculum Vitae, Language Passport or other Europass documents, as well other non-Europass documents.

**Instance:**

```xml
<SkillsPassport locale="">
   <DocumentInfo>{1,1}</DocumentInfo>
   <PrintingPreferences>{0,1}</PrintingPreferences>
   <LearnerInfo>{1,1}</LearnerInfo>
   <AttachmentList>{0,1}</AttachmentList>
</SkillsPassport>
```

**Indicative example:**

```xml
<SkillsPassport xmlns="http://europass.cedefop.europa.eu/Europass"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://europass.cedefop.europa.eu/xml/V3.1.0/EuropassSchema.xsd" locale="en">
   <DocumentInfo/>
   <PrintingPreferences/>
   <LearnerInfo/>
   <AttachmentList/>
</SkillsPassport>
```

### 4.2 SkillsPassport @locale

The attribute “locale” refers to the language of translation of the included information.

The content type of the locale attribute is **LocaleType**. This type defines a restriction on any string by defining a specific pattern. This pattern is comprised of two lowercase and two uppercase letters, separated by underscore. E.g. de_DE or sv_SE.

**LocaleType** is a simpleType with acceptable value of **xsd:string** that follows the followed pattern restriction: `[a-z][2](_[A-Z][2])`
<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport @locale</td>
<td>europass:LocaleType - S(1/1) [restriction]:xsd:string [pattern]:[a-z][2][-_][A-Z][2])*</td>
<td>The translation language of the included information.</td>
</tr>
</tbody>
</table>
4.3 SkillsPassport/ DocumentInfo

The element DocumentInfo contains metadata information about the specific document. It adheres to the data type DocumentInfoType. This data type defines a sequence of five mandatory sub-elements (DocumentType, CreationDate, LastUpdateDate, XSDVersion, Generator) and one optional (Comment).

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport/DocumentInfo</td>
<td>DocumentType: europass:DocumentTypeType- S(1/1) CreationDate: xsd:dateTime- S(1/1) LastUpdateDate: xsd:dateTime -</td>
<td>Contains metadata information about the specific document, like the date the XML was issued/produced, the xsd version with...</td>
</tr>
<tr>
<td>Instance:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>&lt;DocumentInfo&gt;</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>&lt;DocumentType&gt;{1,1}&lt;/DocumentType&gt;</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>&lt;CreationDate&gt;{1,1}&lt;/CreationDate&gt;</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>&lt;LastUpdateDate&gt;{1,1}&lt;/LastUpdateDate&gt;</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>&lt;XSDVersion&gt;{1,1}&lt;/XSDVersion&gt;</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>&lt;Generator&gt;{1,1}&lt;/Generator&gt;</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>&lt;Comment&gt;{0,1}&lt;/Comment&gt;</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>&lt;/DocumentInfo&gt;</code></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Indicative example:</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;DocumentInfo&gt;</code></td>
</tr>
<tr>
<td><code>&lt;DocumentType&gt;ECV&lt;/DocumentType&gt;</code></td>
</tr>
<tr>
<td><code>&lt;CreationDate&gt;2012-03-01T00:00:00.000+03:00&lt;/CreationDate&gt;</code></td>
</tr>
<tr>
<td><code>&lt;LastUpdateDate&gt;2012-03-01T00:00:00.000Z&lt;/LastUpdateDate&gt;</code></td>
</tr>
<tr>
<td><code>&lt;XSDVersion&gt;V3.1&lt;/XSDVersion&gt;</code></td>
</tr>
<tr>
<td><code>&lt;Generator&gt;XML Editor&lt;/Generator&gt;</code></td>
</tr>
<tr>
<td><code>&lt;Comment&gt;Example CV XML according to XSD v3.1.0&lt;/Comment&gt;</code></td>
</tr>
<tr>
<td><code>&lt;/DocumentInfo&gt;</code></td>
</tr>
</tbody>
</table>
4.3.1 DocumentInfo/XSDVersion

This element adheres to the CurrentVersionNumberType data type. This data type restricts the VersionNumberType and defines that the element must have a specific value that corresponds to the Major.Minor version of this XML Schema Definition.

The VersionNumberType in turn is a data type that restricts texts by defining the acceptable pattern for describing the version. According to the pattern the text should start with «V» and continue with the «Major.Minor» numbers.

4.4 SkillsPassport/PrintingPreferences

The PrintingPreferences element includes the preferences of the individual related to how to display the information included in the Europass XML when printed into readable format (e.g. PDF, ODT, DOC). This element gives the flexibility of deciding which information to show/hide in a printed document and how to format the dates, addresses, etc. Its presence is useful in the cases where the specific XML is fed to the Europass online editors or web services in order to receive a “printable” Europass document.

This element adheres to the PrintingPreferencesType data type, which is defined in details in the included XML schema “EuropassPrintingPreferences”. This data type defines a sequence of Document sub-elements, each following the data type DocumentPreferencesType.

The DocumentPreferencesType data type defines a required attribute, named type, which shows for which Europass document these printing preferences are applicable. The set of printing preferences are thus defined on a per-document basis. However we could envision an expanded XML which will include multiple Document elements, each defining its own printing preferences and thus describing a different “printable” document.

The DocumentPreferencesType also defines that it accepts a sequence of multiple Field elements.

A Field element is described by at least one attribute, the name. It may also have extra attributes, such as show, order and format.

The name attribute is actually a path to the section in the Europass Document to which this specific printing preference refers to.

The show attribute defines whether the specific section of the Europass Document is to be shown in the “printable” document or not. This section may be a specific field (e.g. date of birth), a specific section (Specific work experience item) or an entire list (e.g. List of Foreign languages).

The order attribute is described as a simple list, which defines the order with which the sections that correspond to the simple list items are to be displayed in the “printable” document.
The **format** attribute defines a pattern according to which the dates or addresses are to be formatted.

The **position** attribute is described as a simple list, which defines the way a section is positioned in the “printable” document.
Correspond to the simple list items are to be displayed in the “printable” document.

- **format:** defines a pattern according to which the dates or addresses are to be formatted.
- **position:** defines a the accepted values for the position of a section.

**Instance:**

```xml
<PrintingPreferences>
  <Document type="">{1,unbounded}</Document>
  <Field format="" name="" order="" show="" position="">{0,unbounded}</Field>
</PrintingPreferences>
```

**Indicative sample:**

```xml
<PrintingPreferences>
  <Document type="ECV">
    <Field
      name="LearnerInfo"
      show="true"
   />
  </Document>
</PrintingPreferences>
```
4.5 SkillsPassport/ LearnerInfo

LearnerInfo is a core element of Europass schema, as it includes all information about personal data, learning achievements, work experiences, skills, competences, diplomas and other miscellaneous information.

The LearnerInfo element adheres to the data type LearnerInfoType, which is presented in details below. This data type defines a specific sequence of child elements.

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Child Element</td>
<td>Content Type</td>
<td>Sequence(S)</td>
</tr>
</tbody>
</table>

- **Identification**:
- **Headline**:
- **WorkExperienceList**:
- **EducationList**:
- **Skills**
- **AchievementList**
- **Documentation**
<table>
<thead>
<tr>
<th><strong>Attributes(@)</strong></th>
<th>Identification: europass:IndentificationType - S(0/1)</th>
<th>A core element of Europass schema, as it includes all information about personal data, learning achievements, work experiences, skills, competences, diplomas and other miscellaneous information.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>/SkillsPassport/LearnerInfo</strong></td>
<td>Headline: [complexType] - S(0/1)</td>
<td>WorkExperienceList: [complexType] - S(0/1)</td>
</tr>
<tr>
<td><strong>/SkillsPassport/LearnerInfo/Identification</strong></td>
<td>PersonaName: europass:PersonNameType - S(0/1)</td>
<td>Contains the personal information of the individual, which includes at least the name, and optionally contact information, demographics, photo and signature.</td>
</tr>
<tr>
<td><strong>/SkillsPassport/LearnerInfo/Headline</strong></td>
<td>Type: europass:HeadlineTypeLabelType - S(0/1)</td>
<td>Contains a headline label for the current document. It is optional and may accommodate various cases.</td>
</tr>
<tr>
<td><strong>/SkillsPassport/LearnerInfo/WorkExperienceList</strong></td>
<td>WorkExperience: europass:WorkExperienceType - S(0/*)</td>
<td>Contains an list of work experiences. The order of those experiences is defined by the related printing preferences.</td>
</tr>
<tr>
<td><strong>/SkillsPassport/LearnerInfo/EducationList</strong></td>
<td>Education: europass:EducationalExperienceType-S(0/*)</td>
<td>Contains an list of learning achievements or a training periods (formal or not). The order of those experiences is defined by the related printing preferences.</td>
</tr>
<tr>
<td><strong>/SkillsPassport/LearnerInfo/Skills</strong></td>
<td>Linguistic: europass:LinguisticSkillType-S(0/1)</td>
<td>Contains a sequence of skills and competences that the learner has acquired during any formal or informal experience.</td>
</tr>
<tr>
<td><strong>/SkillsPassport/LearnerInfo/AchievementList</strong></td>
<td>Achievement: europass:AchievementList - S(0/*)</td>
<td>Contains a list of additional information about the various achievements of an individual, such as participation to conferences, workshops, memberships to organisations, list of publications, etc.</td>
</tr>
</tbody>
</table>
Instance

```xml
<LearnerInfo>
  <Identification>{0,1}</Identification>
  <Headline>{0,1}</Headline>
  <WorkExperienceList>{0,1}</WorkExperienceList>
  <EducationList>{0,1}</EducationList>
  <Skills>{0,1}</Skills>
  <AchievementList>{0,1}</AchievementList>
  <Documentation>{0,1}</Documentation>
</LearnerInfo>
```

### 4.5.1 SkillsPassport/LearnerInfo/Identification

The Identification element comprises a subset of elements that organize all the personal information of the individual. Its data type is IdentificationType. This type defines that the elements need to at least include the person’s name, and optionally contact information, demographics and photo.

![Identification Type Diagram]

Instance:

```xml
<Identification>
  <PersonName>{1,1}</PersonName>
  <ContactInfo>{0,1}</ContactInfo>
  <Demographics>{0,1}</Demographics>
  <Photo>{0,1}</Photo>
</Identification>
```
### 4.5.1.1 Identification/PersonName

The **PersonName** element contains the full name of an individual that identifies the owner of this document. The element consists of two sub-elements of data type any string.

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>/SkillsPassport/LearnerInfo/Identification/PersonName</td>
<td>Title: PersonTitleLabelType - S(0/1) FirstName: xsd:string - S(0/1) Surname: xsd:string - S(0/1)</td>
<td>Contains the person title, name and the surname of an individual.</td>
</tr>
</tbody>
</table>

**Instance:**

```xml
<PersonName>
  <Title>(0,1)</Title>
  <FirstName>(0,1)</FirstName>
  <Surname>(0,1)</Surname>
</PersonName>
```

**Indicative Example:**

```xml
<PersonName>
  <Title>
    <Code>dr</Code>
    <Label>Dr.</Label>
  </Title>
</PersonName>
```
4.5.1.2 Identification/PersonName/ Title

The Title element adheres to type PersonTitleLabelType. This type extend the base type LabelType by defining that the Code may have any of the values defined according to the simple data type PersonTitlesCodeEnumeration. This enumeration defines the following values: “mr”, “ms”, “mrs”, “miss”, “dr”.

4.5.1.3 Identification/ContactInfo

The ContactInfo element groups the available means with which an individual may be contacted. Its type is ContactInfoType and defines an optional sequence of sub-elements, each representing a difference type of contact info.

- Address
- Email
- TelephoneList
- WebsiteList
- InstantMessagingList
- ContactMethodList

## Elements and Attributes

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>/SkillsPassport/LearnerInfo /Identification/ContactInfo</td>
<td>Address: ContactAddressType - S(0/1) Email: ContactEmailType - S(0/1) TelephoneList: [complexType]- S(0/1) WebsiteList: [complexType]- S(0/1) InstanceMessagingList:[complexType]- S(0/1) ContactMethodList:[complexType]- S(0/1)</td>
<td>Contains all the available methods of contacting individual.</td>
</tr>
</tbody>
</table>

**Instance:**

<ContactInfo>
4.5.1.3.1 Identification/ContactInfo/Address
The Address element represents the residence address of an individual. Its data type is the ContactAddressType (described in 5.5)

4.5.1.3.2 Identification/ContactInfo/Email
The Email element defines the email address of an individual. Its data type is ContactEmailType, which restricts the ContactMethodType (described in 4.5.1.3.6.1) by defining that the Contact sub-element needs to be a string that follows the specified pattern

\[^@]+@[^@]+..+

Elements and Attributes

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>/SkillsPassport/LearnerInfo/Identification/ContactInfo/Email</td>
<td>Contact: xsd:string -S(0/1) [pattern]:[^@]+@[^@]+..+ Use: xsd:string - S(0/0)</td>
<td>Defines the email address of an individual.</td>
</tr>
</tbody>
</table>

Indicative Example:

```xml
<Email>
  <Contact>chuck@chucknorrisfacts.com</Contact>
</Email>
```
### 4.5.1.3.3 Identification/ContactInfo/ TelephoneList

The **TelephoneList** element defines a list of **Telephone** elements. Each Telephone element corresponds to the valid telephone number of owned by an individual.

A Telephone element adheres to the **TelephoneType**, which restricts the **ContactMethodType** (described in 4.5.1.3.6.1) by specifying that the **Use** element needs to follow the data type **TelephoneNumberUseLabelType**. The TelephoneUseLabelType data-type restricts the **europass:LabelType** (described in 5.1) by defining a specific list of accepted values for the **Use/Code** element (home, work, mobile).

### Elements and Attributes

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>/SkillsPassport/LearnerInfo/Identification/ContactInfo/TelephoneList</td>
<td>Telephone: europass:TelephoneType - S(0/unbounded)</td>
<td>Lists the telephone numbers owned by the individual.</td>
</tr>
<tr>
<td>/SkillsPassport/LearnerInfo/Identification/ContactInfo/TelephoneList/Telephone</td>
<td>Contact: xsd:string - S(0/1) Use : europass:TelephoneUseLabelType - S(0/1)</td>
<td>A telephone number owned by the individual.</td>
</tr>
</tbody>
</table>

### Indicative Example:
4.5.1.3.4 Identification/ContactInfo/WebsiteList

The WebsiteList element defines a list of Website elements. Each Website element corresponds to a URI of a website that either belongs to-, refers to- or presents the work of- and individual.

A Website element adheres to the WebsiteType, which restricts the ContactMethodType (described in 4.5.1.3.6.1) by specifying that the Label element needs to be any valid URI, and the Use element needs to follow the data type WebsiteUseLabelType. The WebsiteUseLabelType data-type restricts the europass:LabelType (described in 5.1) by defining a specific list of accepted values for the Use/Code element (personal, business, blog, portfolio)

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Child Element</td>
<td>Content Type</td>
<td>Sequence(S)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(minOccurs/maxOccurs)</td>
</tr>
</tbody>
</table>
### Indicative Example:

```xml
<WebsiteList>
  <Website>
    <Contact>http://chucknorrisfacts.com</Contact>
    <Use>
      <Code>blog</Code>
      <Label>Blog</Label>
    </Use>
  </Website>
</WebsiteList>
```

### 4.5.1.3.5 Identification/ContactInfo/ InstantMessagingList

The **InstantMessagingList** element defines a list of **InstantMessaging** elements. Each InstantMessaging element corresponds to a valid instant messaging account owned by an individual.

A **InstantMessaging** element adheres to the **InstantMessagingType**, which restricts the **ContactMethodType** (described in 4.5.1.3.6.1) by specifying that the **Use** element needs to follow the data type **InstantMessagingUseLabelType**. The **InstantMessagingUseLabelType** data-type restricts the **europass:LabelType** (described in 5.1) by defining a specific list of accepted values for the **Use/Code** element (gtalk, skype, icq, aim, msn, yahoo).
<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>/SkillsPassport/LearnerInfo/Identification/ContactInfo/InstantMessagingList</td>
<td>InstantMessaging: europass:InstantMessagingType - S(0/unbounded)</td>
<td>Lists the instant messaging accounts owned by the individual</td>
</tr>
<tr>
<td>/SkillsPassport/LearnerInfo/Identification/ContactInfo/InstantMessagingList/InstantMessaging</td>
<td>Contact: xsd:anyType - S(0/1) Use: europass:InstantMessagingUseLabelType - S(0/1)</td>
<td>An instant messaging account</td>
</tr>
</tbody>
</table>

### Indicative Example:

```
<InstantMessagingList>
  <InstantMessaging>
    <Contact>chuck.norris</Contact>
    <Use>
      <Code>gtalk</Code>
      <Label>Google Talk</Label>
    </Use>
  </InstantMessaging>
</InstantMessagingList>
```

**4.5.1.3.6 Identification/ContactInfo/ ContactMethodList**

This element lists any other `ContactMethod` elements not already defined. `ContactMethodType` defines the use of the contact method. This might be different depending on the contact method chosen. This is an extension point to accommodate the definition of other contact method, so long as they follow the constraints of the specific contact type.

**4.5.1.3.6.1 Identification/ContactInfo/ ContactMethodList/ ContactMethod**

A `ContactMethod` element adheres to the `ContactMethodType` data type, which defines a sequence of two sub-elements: `Contact` and `Use`, with `Use` being optional. The `Contact` element includes the actual contact information (e.g. the telephone number), while the `Use` element informs on when/how to use this contact method (e.g. business hours or not).

The `Contact` element is of `xsd:anyType`, so that each `ContactMethod` can further specify it, while the `Use` element is of `europass:LabelType`, defined in 5.1.
### Elements and Attributes

<table>
<thead>
<tr>
<th>Child Element</th>
<th>Content Type</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ContactMethod:</td>
<td>Sequence(S)</td>
<td>Lists any other available contact methods of the individual.</td>
</tr>
<tr>
<td>Identification/ContactInfo/ContactMethodList</td>
<td>Choice(C)</td>
<td></td>
</tr>
<tr>
<td>/SkillsPassport/LearnerInfo/Identification/ContactInfo/ContactMethodList/ContactMethod</td>
<td>All(A)</td>
<td>Contains information about any other available contact method of the individual.</td>
</tr>
</tbody>
</table>

### Compositions

<table>
<thead>
<tr>
<th>Attributes(@)</th>
</tr>
</thead>
</table>

### 4.5.1.4 Identification/Demographics

The **Demographics** element contains demographics-related information about the individual. It adheres to the data type **DemographicsType**, which defines a sequence of optional sub-elements for the date of birth, the gender and a list of applicable nationalities.
<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Child Element</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Content Type</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sequence(S)</td>
<td>Choice(C)</td>
</tr>
<tr>
<td></td>
<td>(minOccurs/maxOccurs)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Attributes(@)</td>
<td></td>
</tr>
</tbody>
</table>

**Definition**

Contains demographics-related information about the individual. It includes the birthdate, gender and list of nationalities.

**Instance:**

```xml
<Demographics>
  <Birthdate day="" month="" year="">{0,1}</Birthdate>
  <Gender>{0,1}</Gender>
  <NationalityList>{0,1}</NationalityList>
</Demographics>
```

**4.5.1.4.1 Identification/Demographics/Birthdate**

Birthdate adheres to the data type **DataType**. This type defines that an element needs to have at least one attribute for the year, and may optionally have another two attributes for the month and year.

**Definition**

Defines the date of birth of the individual.

**Indicative Example:**

```xml
<Birthdate year="1940" month="03" day="10"/>
```

**4.5.1.4.2 Identification/Demographics/Gender**

The Gender element defines the gender of the individual. It adheres to the data type **GenderLabelType**. This data type restricts the **europass:LabelType** (described in 5.1) by specifying that the **Code** sub-element may only be one of M or F, which correspond to male and
female. The Label sub-element is the label of the Gender, translated to the language of the document.

Indicative Example:

```xml
<Gender>
  <Code>M</Code>
  <Label>Male</Label>
</Gender>
```

### 4.5.1.4.3 1.2.1.3.3 Identification/Demographics/NationalityList

The **NationalityList** element lists **Nationality** elements. Each Nationality element adheres to the data type of **NationalityType**. This type restricts the **Code** sub-element needs to one of the values defined by the `europass:nationalityCountryCode` in the included schema of “EuropassNationalities”. The **Label** sub-element is the translation of the specific nationality in the language of the document. E.g. code: DE, label: German.

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Child Element</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Content Type</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sequence(S)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Choice(C)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>All(A)</td>
<td>(minOccurs/maxOccurs)</td>
</tr>
</tbody>
</table>
### Attributes(@)

<table>
<thead>
<tr>
<th>Path</th>
<th>Attributes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/SkillsPassport/LearnerInfo/Identification/Demographics/NationalityList</td>
<td>Nationality: NationalityType - S(0/unbounded)</td>
<td>Defines a list of all nationalities of an individual.</td>
</tr>
<tr>
<td>/SkillsPassport/LearnerInfo/Identification/Demographics/NationalityList/Nationality</td>
<td>Code: nationalityCountryCode- S(0/1)</td>
<td>Defines a specific nationality. The type of the Code element is defined in the included schema &quot;EuropassNationalities&quot;.</td>
</tr>
</tbody>
</table>

#### Instance:

```xml
<Nationality>
  <Code>{0,1}</Code>
  <Label>{0,1}</Label>
</Nationality>
```

#### Indicative Example:

```xml
<NationalityList>
  <Nationality>
    <Code>US</Code>
    <Label>American</Label>
  </Nationality>
  <Nationality>
    <Label>Citizen of the world</Label>
  </Nationality>
</NationalityList>
```

### 4.5.1.5 Identification /Photo

The Photo element includes the base-64 encoded bytes of a JPEG or PNG image file that represents the personal photo of an individual. The element adheres to the PhotoDataType, which restricts the FileDataType (described in 5.13) by excluding the Name sub-element and specifying the MimeType sub-element must follow the ImageMimeTypeEnumeration that defines the following values: image/jpeg, image/pjpeg, image/png and image/x-png.
<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
</table>
| /SkillsPassport/LearnerInfo/Identification/Photo | MimeType: europass:MimeTypeEnumeration [S(1/1)]  
Data: xsd:base64Binary [S(1/1)]  
MetadataList: europass:MetadataListType [S(0/1)] | Contains the base-64 encoded bytes of a JPEG or PNG image file that represents the personal photo of an individual |
| /SkillsPassport/LearnerInfo/Identification/Photo/MimeType | Restriction base  
europass:MimeTypeEnumeration [Enumeration]:  
value="image/jpeg"  
value="image/pjpeg"  
value="image/png"  
value="image/x-png"  
value="application/pdf" | Defines the MimeType of the Photo element. |

**Instance:**

```xml
<Photo>
  <MimeType>{1,1}</MimeType>
  <Data>{1,1}</Data>
  <MetadataList>{0,1}</MetadataList>
</Photo>
```

**Indicative Example:**
4.5.1.6 Identification /Signature

The Signature element includes the base-64 encoded bytes of a JPEG or PNG image file that represents the personal signature of an individual. The element adheres to the SignatureDataType, which is identical to the PhotoDataType (described in 4.5.1.5).

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>/SkillsPassport/LearnerInfo/Identification/Signature</td>
<td>MimeType: europass:MimeTypeEnumeration -S(1/1) Data: xsd:base64Binary - S(1/1) MetadataList: europass:MetadataListType - S(0/1)</td>
<td>Contains the base-64 encoded bytes of a JPEG or PNG image file that represents the personal signature of an individual</td>
</tr>
<tr>
<td>/SkillsPassport/LearnerInfo/Identification/Signature/MimeType</td>
<td>Restriction base oneuropass:MimeTypeEnumeration [Enumeration]: value=&quot;image/jpeg&quot; value=&quot;image/pjpeg&quot; value=&quot;image/png&quot; value=&quot;image/x-png&quot; value=&quot;application/pdf&quot;</td>
<td>Defines the MimeType of the Signature element.</td>
</tr>
</tbody>
</table>

Instance:

```xml
<Signature>
  <MimeType>[1,1]</MimeType>
  <Data>[1,1]</Data>
  <MetadataList>[0,1]</MetadataList>
</Signature>
```

Indicative Example:

```xml
<Signature>
  <MimeType>image/jpeg</MimeType>
  <Data>!-- Photo based-64 encoded bytes go here --></Data>
  <MetadataList>
    <Metadata key="signature-dimensions" value="250x150"/>
  </MetadataList>
</Signature>
```
4.5.2 SkillsPassport/LearnerInfo/Headline

The Headline element includes the motivation of the individual for authoring this specific document. It defines a suitable text that may be used to either reveal the purpose of the document (e.g. applying for specific job position or training) or to describe the job position held or pursued.

The element adheres to a complex type that defines a sequence of two sub-elements:

- The Type sub-element follows the data type of HeadlineTypeLabelType. This type in turn restricts the LabelType by defining that the Code sub-element may only be one of the following values: preferred_job, job_applied_for, studies_applied_for and position. The Label sub-element is actually the translation of the type in the language of the document.
- The Description sub-element follows the data type of OccupationalFieldType (described in 5.2).

Instance:

```xml
<Headline>
  <Type>{1,1}</Type>
  <Description>{1,1}</Description>
</Headline>
```

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport /LearnerInfo Headline</td>
<td>Type:europass:HeadlineTypeLabelType - S(0/1) Description:europass:OccupationalFieldType - S(0/1)</td>
<td>Contains a headline label for the current document. It is optional and may accommodate various cases.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Child Element</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content Type</td>
</tr>
<tr>
<td>Sequence(S)</td>
</tr>
</tbody>
</table>
**HeadlineTypeLabelType**

<table>
<thead>
<tr>
<th>Code: europass:HeadlineTypeLabelType - S(0/1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>[enumeration]: &quot;preferred_job&quot;</td>
</tr>
<tr>
<td>[enumeration]: &quot;job_applied_for&quot;</td>
</tr>
<tr>
<td>[enumeration]: &quot;studies_applied_for&quot;</td>
</tr>
<tr>
<td>[enumeration]: &quot;position&quot;</td>
</tr>
<tr>
<td>[enumeration]: &quot;personal_statement&quot;</td>
</tr>
</tbody>
</table>

**Label: xsd:string - S(0/1)**

Defines the accepted types of headline. The **Code** may only be one of the defined enumeration values. The **Label** is actually the translated text of the type to the language of the document.
4.5.3 SkillsPassport/LearnerInfo/WorkExperienceList and WorkExperience

This section of the XML document lists the work positions that an individual held. For each work experience one can find further information on the period, the position title, the employer and the activities that concern it.

The section is described by a WorkExperienceList element, which accepts a list of WorkExperience elements. Each WorkExperience element adheres to the data type WorkExperienceType.

The WorkExperienceType data type extends the ExperienceType (described in 5.9) data type by further defining three sub-elements. That said, a WorkExperience element must have a Period element, denoting the period during which the work experience took place, and a Position element, denoting the position title held. Optionally it may also include the elements Description and Documentation (inherited from the ExperienceType) and also the elements Activities and Employer (added by the WorkExperienceType).

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport/LearnerInfo/WorkExperienceList</td>
<td>WorkExperience: europass:WorkExperienceType - S(0/unbounded)</td>
<td>Lists the work experiences of an individual.</td>
</tr>
</tbody>
</table>
SkillsPassport / LearnerInfo / WorkExperienceList / WorkExperience

<table>
<thead>
<tr>
<th>Extension base europass:ExperienceType</th>
<th>Period: europass:PeriodType - S(0/1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description: xsd:string - S(0/1)</td>
<td>Documentation:</td>
</tr>
<tr>
<td></td>
<td>europass:IntraDocumentDocumentationType - S(0/1)</td>
</tr>
<tr>
<td>Position: OccupationalFieldtype - S(0/1)</td>
<td>Activities: xsd:string - S(0/1)</td>
</tr>
<tr>
<td>Employer: europass:EmployerType - S(0/1)</td>
<td></td>
</tr>
</tbody>
</table>

Extends the generic ExperienceType to include further information about a work experience, like the Position, Activities and Employer.

Instance:

```xml
<WorkExperienceList>
  <WorkExperience>{0, unbounded}</WorkExperience>
</WorkExperienceList>

and

<WorkExperience>
  <Period>{0, 1}</Period>
  <Description>{0, 1}</Description>
  <Documentation>{0, 1}</Documentation>
  <Position>{0, 1}</Position>
  <Activities>{0, 1}</Activities>
  <Employer>{0, 1}</Employer>
</WorkExperience>

Indicative Example:

```xml
<WorkExperienceList>
  <WorkExperience>
    <Period>
      <From year="2001"/>
      <Current>true</Current>
    </Period>
    <Documentation>
      <ReferenceTo idref="ATT_1"/>
    </Documentation>
    <Position><Label>Martial Arts Instructor</Label></Position>
    <Activities>Main activities and responsibilities include preparing class plans.</Activities>
    <Employer>
      <Name>School of Ninjutsu</Name>
      <ContactInfo>
        <Address>
          <Contact>
            <Country>
              <Code>JP</Code>
              <Label>Japan</Label>
            </Country>
          </Contact>
        </Address>
        <Website>
          <Contact>http://mybusiness.com</Contact>
        </Website>
      </ContactInfo>
    </Employer>
  </WorkExperience>
</WorkExperienceList>
```
4.5.3.1 WorkExperience/Period

The Period element defines the period during which the specific work experience took place. It follows the PeriodType (described in 5.10).

Instance:

```xml
<Period>
  <From day="" month="" year="">{0,1}</From>
  <To day="" month="" year="">{0,1}</To>
  <Current>{0,1}</Current>
</Period>
```

Indicative Example:

```xml
<Period>
  <From year="2001" month="10"/>
  <Current>true</Current>
</Period>
```

4.5.3.2 WorkExperience/Documentation

The Documentation element provides a list of references to digital documents that are attached to the specific Europass Document and included in the XML Document, and which are related to this specific work experience.

An individual can attach one or more relevant documents that verify or describe the specific work experience included in the Europass Document.

The Documentation element follows the IntraDocumentDocumentationType data type (described in Error! Reference source not found.).

4.5.3.3 WorkExperience/Position

The Position element described the position held by the individual during her work experience. It follows the OccupationalFieldType data type (described in 5.2).
Instance:

```xml
<Position>
  <Code>{0,1}</Code>
  <Label>{0,1}</Label>
</Position>
```

Indicative Example:

```xml
<Position>
  <Label>Martial Arts Instructor</Label>
</Position>
```

4.5.3.4 WorkExperience/Employer

The Employer element describes the organisation with which the individual cooperated during her work experience.

The Employer element adheres to the EmployerType data type, which extends the OrganisationType data type (described in 5.6), by specifying the additional optional element Sector, for describing the business sector to which the employer belongs to.

The Sector element follows the data type europass:BusinessSectorType which is defined in the imported namespace “nace:http://europass.cedefop.europa.eu/NACE”
<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport/</td>
<td>Extension europass:OrganisationType</td>
<td>Describes the organisation with which the individual cooperated during her work experience.</td>
</tr>
<tr>
<td>LearnerInfo/</td>
<td>Name: xsd:string - S(0/1)</td>
<td></td>
</tr>
<tr>
<td>WorkExperienceList/</td>
<td>ContactInfo:europass:OrganisationalContactInfoType - S(0/1)</td>
<td></td>
</tr>
<tr>
<td>WorkExperience/</td>
<td>Sector:europass:BusinessSectorType - S(0/1)</td>
<td></td>
</tr>
<tr>
<td>Employer</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Indicative Example:**

```xml
<Employer>
  <Name>School of Ninjutsu</Name>
  <ContactInfo>
    <Address>
      <Country>
        <Code>JP</Code>
        <Label>Japan</Label>
      </Country>
      </Address>
      <Website>
        <Contact>http://mybusiness.com</Contact>
        <Use><Code>business</Code></Use>
        </Website>
    </ContactInfo>
    <Sector>
      <Code>Q</Code>
      <Label>Human health and social work</Label>
    </Sector>
  </ContactInfo>
</Employer>
```

### 4.5.4 SkillsPassport/LearnerInfo/EducationList and Education

This section of the XML document lists the education or training experiences that an individual attended. For each education experience one can find further information on the period, the awarded title, the organisation providing the education or training, the activities that were carried out during this experience, as well as information on the educational field and educational level to which this experience is classified.

The section is described by an **EducationList** element, which accepts a list of **Education** elements. Each Education element adheres to the data type **EducationalExperienceType**.

The **EducationalExperienceType** data type extends the **ExperienceType** (described in 5.9) data type by further defining three sub-elements. That said, a
**EducationalExperienceType** element must have a **Period** element, denoting the period during which the education experience took place, and a **Title** element, denoting the title awarded. Optionally it may also include the elements **Description** and **Documentation** (inherited from the **ExperienceType**) and also the elements **Activities, Organisation, Level** and **Field** (added by the **EducationalExperienceType**).

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport / LearnerInfo/ EducationList</td>
<td><strong>Content Type</strong>&lt;br&gt;**Sequence(S)</td>
<td><strong>Definition</strong></td>
</tr>
<tr>
<td><strong>Extension base europass:ExperienceType</strong></td>
<td><strong>Description: xsd:string - S(0/1)</strong>&lt;br&gt;<strong>Documentation: europass:IntraDocumentDocumentationType - S(0/1)</strong>&lt;br&gt;<strong>Title: OccupationalFieldDataType - S(0/1)</strong></td>
<td><strong>Extension base europass:ExperienceType</strong> to include further information about an education experience, like the Title, Activities, Organisation, Level and Field.</td>
</tr>
<tr>
<td>SkillsPassport / LearnerInfo/ EducationList/ Education</td>
<td><strong>Content Type</strong>&lt;br&gt;**Sequence(S)</td>
<td><strong>Definition</strong></td>
</tr>
<tr>
<td><strong>Education: europass:EducationExperienceType - S(1/unbounded)</strong></td>
<td>Lists all education or training experiences of an individual.</td>
<td></td>
</tr>
</tbody>
</table>
Activities: xsd:string - S(0/1)
Organisation: europass:OrganisationType - S(0/1)
Level: europass:EducationalLevelType - S(0/1)
Field: europass:EducationalFieldType - S(0/1)

Instance:

```
<EducationList>
  <Education>(0, unbounded)</Education>
</EducationList>

and

```
<Education>
  <Period>(0, 1)</Period>
  <Description>(0, 1)</Description>
  <Documentation>(0, 1)</Documentation>
  <Title>(0, 1)</Title>
  <Activities>(0, 1)</Activities>
  <Organisation>(0, 1)</Organisation>
  <Level>(0, 1)</Level>
  <Field>(0, 1)</Field>
</Education>

Indicative example:

```
<EducationList>
  <Education>
    <Period>
      <From year="1995" month="10" day="05"/>
      <To year="2000" month="09" day="30"/>
    </Period>
    <Documentation>
      <ReferenceTo idref="ATT_1"/>
      <ReferenceTo idref="ATT_2"/>
    </Documentation>
    <Title>Martial Arts</Title>
    <Activities>Principal Subjects covered</Activities>
    <Organisation>
      <Name>The University of Chicago</Name>
      <ContactInfo>
        <Address>
          <Contact>
            <Country>
              <Code>US</Code>
              <Label>United States</Label>
            </Country>
          </Contact>
        </Address>
      </ContactInfo>
    </Organisation>
  </Education>
</EducationList>
```
4.5.4.1 Education/Period

The Period element defines the period during which the specific education or training experience took place. It follows the PeriodType (described in 5.10).

Instance:

```xml
<Period>
  <From day="" month="" year="">{1,1}</From>
  <To day="" month="" year="">{0,1}</To>
  <Current>{0,1}</Current>
</Period>
```

Indicative Example:

```xml
<Period>
  <From year="2001" year="10"/>
  <To year="2004" year="12"/>
</Period>
```

4.5.4.2 Education/Documentation

The Documentation element provides a list of references to digital documents that are attached to the specific Europass Document and included in the XML Document, and which are related to this specific education or training experience.

An individual can attach one or more relevant documents that verify or describe the specific education experience included in the Europass Document.

The Documentation element follows the IntraDocumentDocumentationType data type (described in Error! Reference source not found.).

4.5.4.3 Education/Organisation

The Organisation element provides information about the organisation that provided the specific education or training experience. The element adheres to the data type OrganisationType (described in 5.6).
4.5.4.4 Education/Level

The **Level** element provides information on how this specific education or training experience could be classified according to a specific classification scheme.

The Level element adheres to the **EducationalLevelType** data type. This type restricts the **LabelType** by specifying that the Code element must be of type **eqf:level**, as this is defined in the imported namespace “eqf:http://europass.cedefop.europa.eu/EQF/08”

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
</table>
| SkillsPassport/LearnerInfo/EducationList/Education/Level | **Code**: eqf:level - S(0/1)  
**Label**: xsd:string - S(0/1) | Describes the level to which this education is classified.  
The Code element when present must take one of the enumeration values defined by the **eqf:level** type from the imported namespace of  
eqf:http://europass.cedefop.europa.eu/EQF/08 |

**Indicative example:**

```
<Level>
  <Code>6</Code>
  <Label>EQF 6</Label>
</Level>
```
4.5.4.5 Education/Field

The **Field** element provides information on how could the field this specific education or training experience relates to according to a specific taxonomy.

The Field element adheres to the **EducationalField** type data type. This type restricts the **LabelType** by specifying that the Code element must be of type **isced:field-level2**, as this is defined in the imported namespace “isced:http://europass.cedefop.europa.eu/ISCED/97”

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
</table>
| SkillsPassport/LearnerInfo/EducationList/Education/Field | Code: isced:field-level2 - S(0/1) 
Label: xsd:string - S(0/1) | Describes the field this specific education or training experience relates to according to a specific taxonomy. The Code element when present must take one of the enumeration values defined by the isced:field-level2 type from the imported namespace of isced:http://europass.cedefop.europa.eu/ISCED/97 |

**Indicative example:**

```xml
<Field>
  <Code>21</Code>
  <Label>Arts</Label>
</Field>
```
4.5.5 SkillsPassport/LearnerInfo/Skills

The Skills element contains a sequence of skills and competences that the individual has acquired during any formal or informal experience and has decided to include them in the Europass document. The skills are organised into seven categories:

1. Linguistic skills: including details about the mother tongues and the foreign languages that the individual speaks;
2. Communication skills
3. Organisational skills
4. Job-related skills: job related or other technical skills, valuable for the job.
5. Computer skills: skills that reveal knowledge of working with information systems.
6. Driving skills: driving skills that are verified by the driving licences owned.
7. Other: any other skill that is relevant and provided added value to the individual.

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport / LearnerInfo / Skills</td>
<td>Linguistic: europass:LinguisticSkillType- S(0/1)</td>
</tr>
<tr>
<td></td>
<td>Communication: europass:GenericSkillType- S(0/1)</td>
</tr>
<tr>
<td></td>
<td>Organisational: europass:GenericSkillType- S(0/1)</td>
</tr>
<tr>
<td></td>
<td>JobRelated: europass:GenericSkillType- S(0/1)</td>
</tr>
<tr>
<td></td>
<td>Computer: europass:GenericSkillType- S(0/1)</td>
</tr>
<tr>
<td></td>
<td>Driving: europass:DrivingSkillType- S(0/1)</td>
</tr>
<tr>
<td></td>
<td>Other: europass:GenericSkillType- S(0/1)</td>
</tr>
</tbody>
</table>

Defines the skills and competences that the individual has acquired during any formal or informal experience.

Instance:
<Skills>
  <Linguistic>{0,1}</Linguistic>
  <Communication>{0,1}</Communication>
  <Organisational>{0,1}</Organisational>
  <JobRelated>{0,1}</JobRelated>
  <Computer>{0,1}</Computer>
  <Driving>{0,1}</Driving>
  <Other>{0,1}</Other>
</Skills>

4.5.5.1 Skills/Linguistic

The **Linguistic** element refers to the ability of the individual to communicate in various languages. It adheres to the **LinguisticSkillType**, which defines a sequence of two sub-elements: **MotherTongueList** and **ForeignLanguageList**, described in detail below.

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport / LearnerInfo / Skills/Linguistic</td>
<td>MotherTongueList: [complexType]S(0/1) ForeignLanguageList: [complexType]S(0/1)</td>
<td>Refers to the ability of the individual to communicate in various languages/</td>
</tr>
</tbody>
</table>

**Instance:**

```
<Linguistic>
  <MotherTongueList>{0,1}</MotherTongueList>
  <ForeignLanguageList>{0,1}</ForeignLanguageList>
</Linguistic>
```

4.5.5.2 Skills/Linguistic/MotherTongueList and MotherTongue

The **MotherTongueList** element, lists the languages in which the individual communicates as a native speaker. The data type of the element defines a sequence of **MotherTongue** elements each adhering to the data type **MotherTongueSkillType**.
The **MotherTongueSkillType** restricts the **SkillType** (described in 5.16) by defining that it includes only two sub-elements: Description and Documentation. Moreover, Description adheres to the data type MotherLanguageType.

The **MotherLanguageType** restricts the **LabelType** (described in 5.1) by defining that the Code element must have one of the enumeration values specified by the **europass:motherCode** data type defined in the included schema “EuropassISOLanguages”.

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport / LearnerInfo/ Skills/Linguistic/ MotherTongueList</td>
<td><strong>Compositions</strong>&lt;br&gt;Child Element <strong>Content Type</strong>&lt;br&gt;Sequence(S)</td>
<td>Lists the languages in which the individual communicates as a native speaker (mother tongue)</td>
</tr>
<tr>
<td></td>
<td><strong>Description</strong>&lt;br&gt;europass:MotherTongueSkillType-S(0/unbounded)</td>
<td></td>
</tr>
<tr>
<td>SkillsPassport / LearnerInfo/ Skills/Linguistic/ MotherTongueList/ MotherTongue</td>
<td><strong>Restriction on europass:SkillType</strong>&lt;br&gt;Description: europass:MotherLanguageType-S(0/1)&lt;br&gt;Documentation: europass:IntraDocumentDocumentationType-S(0/1)</td>
<td>A languages in which the individual communicates as a native speaker (mother tongue)</td>
</tr>
<tr>
<td>Data Type</td>
<td>Compositions</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------</td>
<td>------------</td>
</tr>
<tr>
<td></td>
<td>Child Element</td>
<td>Defines that this element includes information about mother language. In case the Code element is present, then this should take a value from the enumeration defined by the europass:motherCode in the included schema &quot;EuropassISOLanguages&quot;.</td>
</tr>
<tr>
<td></td>
<td>Content Type</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sequence(S)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Choice(C)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>All(A)</td>
</tr>
<tr>
<td></td>
<td>(minOccurs/maxOccurs)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Attributes(@)</td>
<td></td>
</tr>
<tr>
<td><strong>MotherLanguageType</strong></td>
<td>Code: europass:motherCode - S(0/1)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[restriction]:xsd:string</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[enumeration]</td>
<td>el</td>
</tr>
<tr>
<td></td>
<td></td>
<td>de</td>
</tr>
<tr>
<td></td>
<td>Label:xsd:string - S(0/1)</td>
<td></td>
</tr>
</tbody>
</table>

**Instance:**

```xml
<MotherTongueList>
  <MotherTongue>{0, unbounded}</MotherTongue>
</MotherTongueList>
```

**Indicative example:**

```xml
<MotherTongueList>
  <MotherTongue>
    <Description>{0, 1}</Description>
    <Documentation>{0, 1}</Documentation>
  </MotherTongue>
</MotherTongueList>
```

### 4.5.5.3 Skills/Linguistic/ForeignLanguageList and ForeignLanguage

The **ForeignLanguageList** element, lists the languages in which the individual communicates at variant levels of competence. The data type of the element defines a sequence of **ForeignLanguage** elements each adhering to the data type **ForeignLanguageSkillType**.
The **ForeignLanguageSkillType** restricts the **SkillType** (described in 5.16) by specifying the data type of the elements that describe the level of knowledge of the language (**ProficiencyLevel**) and related linguistic diplomas (**VerifiedBy**) and/or experiences (**AcquiredDuring**). It also specifies that the **Description** element must adhere to the data type **ForeignLanguageType**.

The **ForeignLanguageType** restricts the **LabelType** (described in 5.1) by defining that the **Code** element must have one of the enumeration values specified by the **europass:foreignCode** data type defined in the included schema “EuropassISOLanguages”.

### Elements and Attributes

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport / ForeignLanguage:</td>
<td><strong>Child Element</strong>&lt;br&gt;<strong>Content Type</strong>&lt;br&gt;<strong>Sequence(S)</strong></td>
<td>Lists the languages in</td>
</tr>
</tbody>
</table>
A language in which the individual communicates at variant levels of competence.

Instance:

```xml
<ForeignLanguageList>
  <ForeignLanguage>(0, unbounded)</ForeignLanguage>
</ForeignLanguageList>

Indicative example:

```xml
<ForeignLanguage>
  <Description>[0,1]</Description>
  <ProficiencyLevel>[0,1]</ProficiencyLevel>
  <AcquiredDuring>[0,1]</AcquiredDuring>
  <VerifiedBy>[0,1]</VerifiedBy>
  <Documentation>[0,1]</Documentation>
</ForeignLanguage>

<ForeignLanguage>
  <Description>
    <Code>ja</Code>
    <Label>Japanese</Label>
  </Description>
  <ProficiencyLevel>
    <Listening>C1</Listening>
    <Reading>C1</Reading>
    <SpokenInteraction>C1</SpokenInteraction>
    <SpokenProduction>C1</SpokenProduction>
    <Writing>C1</Writing>
  </ProficiencyLevel>
  <VerifiedBy>
    <Certificate>
      <Title>Cambridge Proficiency</Title>
    </Certificate>
    <Certificate>
      <Title>Michigan Proficiency</Title>
    </Certificate>
  </VerifiedBy>
</ForeignLanguage>```
4.5.5.3.1 *Skills/Linguistic/ForeignLanguageList/ForeignLanguage/ProficiencyLevel*

The **ProficiencyLevel** element demonstrates the level of competence for the specific linguistic skill. The evaluation is performed on a self-assessment basis and is based on the classification defined by the Common European Framework of Reference for Languages (CEFR).

The element follows a complex type that defines a sequence of sub-elements each adhering to the data type **CEFLanguageLevelType**, described in 5.17.

### Compositions

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SkillsPassport</strong> / <strong>LearnerInfo</strong> / <strong>Skills</strong> / <strong>Linguistic</strong> / <strong>ForeignLanguageList</strong> / <strong>ForeignLanguage</strong> / <strong>ProficiencyLevel</strong></td>
<td>Listening: europass:CEFLanguageLevelType - S(0/1) Reading: europass:CEFLanguageLevelType - S(0/1) SpokenInteraction: europass:CEFLanguageLevelType - S(0/1) SpokenProduction: europass:CEFLanguageLevelType - S(0/1) Writing: europass:CEFLanguageLevelType</td>
<td>A detailed break-down of an individual’s self assessment about her level of knowledge of a foreign language. The CEF language level is comprised of one letter (either A, B or C) and one digit (either 1 or 2). E.g. A1, B2, C1. The levels...</td>
</tr>
</tbody>
</table>
4.5.5.3.2 Skills/Linguistic/ForeignLanguageList/ForeignLanguage/AquiredDuring

The AcquiredDuring element lists the experiences that helped an individual to acquire the specific linguistic skill. The data type of this element is the LinguisticExperienceListType. This type restricts the abstract type AbstractExperienceListType (defined in 5.7) by defining that the included Experience element will be of type LinguisticExperienceType instead of anyType.

4.5.5.3.2.1 ForeingLanguage/AcquiredDuring/Experience

The Experience element adheres to the LinguisticExperienceType data type. This type models an experience during which the individual has acquired or improved her knowledge of a foreign language. The data type defines a list of sub-elements, as detailed in the table below:

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport /LearnerInfo/ Skills/Linguistic/ ForeignLanguageList/</td>
<td>Period: europass:PeriodType- S(0/1) Description: xsd:string- S(0/1) Documentation:</td>
<td>A detailed break-down of an experience in an individual’s life that led to the enrichment</td>
</tr>
<tr>
<td>ForeignLanguage/AcquiredDuring/Experience</td>
<td>europass:IntraDocumentDocumentationType - S(0/1)</td>
<td>of the knowledge of a foreign language.</td>
</tr>
<tr>
<td></td>
<td>europass:LinguisticExperienceAreaType - S(0/1)</td>
<td>Period: the start and end (optionally) dates of the experience.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Description: a text description of what was the experience about.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Documentation: a list of material that are attached to the current information set.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Area: a categorisation of this linguistic experience.</td>
</tr>
</tbody>
</table>

### 4.5.5.3.2.2 ForeignLanguage/AcquiredDuring/Experience/Area

The **Area** element of a Linguistic Experience adheres to the **LinguisticExperienceAreaType** data type. This type defines the category under which the specific linguistic experience may be categorized. The type restricts the generic **LabelType** by specifying that the **Code** element, if it exists, it must accept values that are part of a specific enumeration:

- studying_training_language
- work_language
- living_traveling_language
- mediating_groups_language

### 4.5.5.3.3 Skills/Linguistic/ForeignLanguageList/ForeignLanguage/VerifiedBy

The **VerifiedBy** element lists the certificates that verify the knowledge of the specific language. The data type of this element is the **CertificateListType** that consists of one or more **Certificate** elements, each adhering to the **LinguisticCertificateType**.

The **LinguisticCertificateType** restricts the **CertificateType** (described in 5.16) by restricting the allowed type for the Level element in order to be **CIELanguageLevelType** (described in 5.17).
<table>
<thead>
<tr>
<th>SkillsPassport /LearnerInfo/ Skills/Linguistic/ ForeignLanguageList/ ForeignLanguage/ VerifiedBy</th>
<th>(minOccurs/maxOccurs) Attributes(@)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certificate: europass:LinguisticCertificateType-S(0/unbounded)</td>
<td>A list of certificates that verify a linguistic skill</td>
</tr>
</tbody>
</table>

| SkillsPassport /LearnerInfo/ Skills/Linguistic/ ForeignLanguageList/ ForeignLanguage/ VerifiedBy/Certificate | Title: xsd:string-S(0/1) AwardingBody:xsd:string-S(0/1) Date: europass:DateType-S(0/1) Level: europass:CEFLanguageLevelType - S(0/1) | A certificate that verifies linguistic skill |

**Instance:**

```xml
<VerifiedBy>
  <Certificate>{0,unbounded}</Certificate>
</VerifiedBy>
```

**Indicative example:**

```xml
<VerifiedBy>
  <Certificate>
    <Title>CambridgeProficiency</Title>
    <Level>C2</Level>
  </Certificate>
</VerifiedBy>
```

### 4.5.5.4 Skills/Communication

The **Communication** element includes information which proves that the individual has specific communication skills, such as good ability to adapt to multicultural environments or team spirit.

The data type of this element is **GenericSkillType**. This data type restricts the generic data type **SkillType** (described in 5.16), by defining a generic skill, which is modeled only by an unstructured (free text) **Description** element and a **Documentation** element, which lists the references to the related attached digital documents.
Elements and Attributes

<table>
<thead>
<tr>
<th>SkillType (restriction base)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

Compositions

Child Element

- Description
- ProficiencyLevel
- AcquiredDuring
- VerifiedBy
- Documentation

Content Type

- Sequence(S) | Choice(C) | All(A)
- (minOccurs/maxOccurs)
- Attributes(@)

Attributes:

- Description: xsd:string - S(0/1)
- Documentation: DocumentationType - S(0/1)

Definition:

Defines the communication skills of an individual.

Instance:

```xml
<Communication>
  <Description>{0,1}</Description>
  <Documentation>{0,1}</Documentation>
</Communication>
```

Indicative example:

```xml
<Communication>
  <Description>Team spirit; Good ability to adapt to multicultural environments, gained through my work experience abroad.</Description>
</Communication>
```
4.5.5.5 Skills/Organisational

The Organisational element includes information which proves that the individual has specific organisational skills, such as leadership or project management skills.

The data type of this element is GenericSkillType, described in 4.5.5.4.

Instance:

```
<Organisational>
  <Description>{0,1}</Description>
  <Documentation>{0,1}</Documentation>
</Organisational>
```

Indicative example:

```
<Organisational>
  <Description>Leadership (currently responsible for a team of 10 people); Sense of organisation.</Description>
</Organisational>
```

4.5.5.6 Skills/JobRelated

The JobRelated element includes information which proves that the individual has specific technical skills, necessary for a specific job, such as good command of quality assurance methods.

The data type of this element is GenericSkillType, described in 4.5.5.4.

Instance:

```
<JobRelated>
  <Description>{0,1}</Description>
  <Documentation>{0,1}</Documentation>
</JobRelated>
```

Indicative example:

```
<JobRelated>
  <Description>Good command of quality control processes</Description>
</JobRelated>
```

4.5.5.7 Skills/Computer

The Computer element includes information which proves that the individual has specific skills related to the use of information systems and software packages.

The data type of this element is GenericSkillType, described in 4.5.5.4.
Instance:

```xml
<Computer>
  <Description>{0,1}</Description>
  <Documentation>{0,1}</Documentation>
</Computer>
```

Indicative example:

```xml
<Computer>
  <Description>Good command of Microsoft Office™ tools (Word™, Excel™ and PowerPoint™); Basic knowledge of graphic design applications (Adobe Illustrator™, PhotoShop™).
</Description>
</Computer>
```

4.5.5.8 Skills/Driving

This element defines a driving skill e.g. driving licence of type B. It has a complex type `DrivingSkillType`, based on a `SkillType` restriction which is modeled with one `Description` element that verifies the driving license of an individual and may also accept `Documentation` information such as the attachment of a driving license copy.

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport/</td>
<td>Child Element</td>
<td>Defines the driving skills of an individual.</td>
</tr>
<tr>
<td>LearnerInfo/</td>
<td>Content Type</td>
<td></td>
</tr>
<tr>
<td>Skills/Driving</td>
<td>Sequence(S)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Choice(C)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>All(A)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(minOccurs/maxOccurs)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Attributes(@)</td>
<td></td>
</tr>
</tbody>
</table>
Driving/Description

Instance:

```xml
<Driving>
  <Description>{0,1}
    <Licence>{0, unbounded}</Licence>
  </Description>
  <Documentation>{0,1}</Documentation>
</Driving>
```

Indicative example:

```xml
<Driving>
  <Description>
    <Licence>A</Licence>
    <Licence>B</Licence>
    <Licence>BE</Licence>
  </Description>
</Driving>
```

4.5.5.9 Skills/Other

The Other element includes any other skills that the individual has and which adds value to her profile. For example, artistic skills may be included here.

The data type of this element is `GenericSkillType`, described in 4.5.5.4.

Instance:

```xml
<Other>
  <Description>{0,1}</Description>
  <Documentation>{0,1}</Documentation>
</Other>
```

Indicative example:

```xml
<Other>
  <Description>Carpentry skills acquired through voluntary activities.</Description>
</Other>
```

4.5.6 SkillsPassport/LearnerInfo/AchievementList and Achievement

This section of the XML document lists any additional information about the various achievements of an individual, such as participation to conferences, workshops, memberships to organisations, list of publications, etc.
The section is described by the **AchievementList** element, which accepts a list of **Achievement** elements. Each **Achievement** element adheres to the data type **AchievementType**.

The **AchievementType** defines a sequence of three sub-elements: **Title**, **Description** and **Documentation**.

The **Title** element describes to what kind of achievement this section refers to. It adheres to the **AchievementTypeLabelType** data type. This data type restrict the **LabelType** (described in 5.1) by specifying that the Code element must have one of the specified values: "honors_awards", "publications", "projects", "citations", "memberships", "conferences", "seminars", workshops" or "references".

The **Description** element provides information on what this specific achievement entails. It is a simple text (**xsd:string**).

The **Documentation** element provides a list of references to attached digital documents that are included in the XML Document and are related to this achievement. The element follows the data type **DocumentationType** (described in Error! Reference source not found.).

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport / LearnerInfo/ AchievementList</td>
<td>Achievement: <strong>AchievementListType</strong> - S(0/1)</td>
<td>Defines a list of achievements.</td>
</tr>
</tbody>
</table>
| SkillsPassport / LearnerInfo/ AchievementList/ Achievement | Title: **europass:AchievementTypeLabelType** - S(0/1)  
Description: **xsd:string** - S(0/1)  
Documentation: **europass:IntraDocumentDocumentationType** - S(0/1) | Defines an achievement |
<table>
<thead>
<tr>
<th>Data Type</th>
<th>Compositions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Composition</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Child Element</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Content Type</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Sequence(S)</strong></td>
</tr>
<tr>
<td></td>
<td>**</td>
</tr>
<tr>
<td></td>
<td><strong>(minOccurs/maxOccurs)</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Attributes(@)</strong></td>
</tr>
<tr>
<td><strong>AchievementTypeLabelType</strong></td>
<td><strong>Restriction on Label Type</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Code</strong> must be one of</td>
</tr>
<tr>
<td></td>
<td><strong>[enumeration]</strong>: honors_awards</td>
</tr>
<tr>
<td></td>
<td><strong>[enumeration]</strong>: publications</td>
</tr>
<tr>
<td></td>
<td><strong>[enumeration]</strong>: citations</td>
</tr>
<tr>
<td></td>
<td><strong>[enumeration]</strong>: projects</td>
</tr>
<tr>
<td></td>
<td><strong>[enumeration]</strong>: memberships</td>
</tr>
<tr>
<td></td>
<td><strong>[enumeration]</strong>: seminars</td>
</tr>
<tr>
<td></td>
<td><strong>[enumeration]</strong>: conferences</td>
</tr>
<tr>
<td></td>
<td><strong>[enumeration]</strong>: workshops</td>
</tr>
<tr>
<td></td>
<td><strong>[enumeration]</strong>: references</td>
</tr>
</tbody>
</table>

**Definition**

Defines the content and structure of an element that is described as an achievement.

**Indicative example:**

```xml
<AchievementList>
    <Achievement>
        <Title>
            <Code>projects</Code>
            <Label>Projects</Label>
        </Title>
        <Description>Project 1; Project 2</Description>
    </Achievement>
    <Achievement>
        <Title>
            <Code>publications</Code>
            <Label>Publications</Label>
        </Title>
        <Description>Publication 1; Publication 2</Description>
    </Achievement>
    <Achievement>
        <Title>
            <Label>Theatre Participations</Label>
        </Title>
        <Description>Theatre 1; Theatre 2</Description>
    </Achievement>
</AchievementList>
```

**4.5.7 SkillsPassport/LearnerInfo/Documentation and ReferenceTo**

An individual can attach one or more relevant documents that verify or support the information included in the specific Europass Document.
Thus, Documentation element comprises a list of ReferenceTo elements each referencing an Attachment element that corresponds to an uploaded digital document, the bytes of which are already included in the XML document.

The Documentation element adheres to the data type DocumentationType (described in Error! Reference source not found.).
4.6 SkillsPassport/AttachmentList

The **AttachmentList** element defines a list of **Attachment** elements. Each Attachment element corresponds to any digital document (PDF, JPEG or PNG format) that an individual has attached to her Europass document to support/evidence of her personal data, learning achievements, work experiences, skills, competences, diplomas, etc.

The **Attachment** element adheres to the data type **AttachmentType**. This data type extends the **FileDataType** (defined in 5.12) by defining a **Description** sub-element and a required **id** attribute. The actual bytes of the digital attachment are included in the **Data** sub-element as raw base-64 encoded bytes.

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkillsPassport/AttachmentList</td>
<td>Attachment: <strong>europass:AttachmentType</strong> - S(0/unbounded)</td>
<td>A list of digital documents attached to the specific Europass document as evidence of the mentioned experiences, skills and competences.</td>
</tr>
<tr>
<td>SkillsPassport/AttachmentList/Attachment</td>
<td><strong>europass:AttachmentType</strong>&lt;br&gt;<strong>id</strong>: xsd:ID</td>
<td>A digital document attached to the specific Europass document.</td>
</tr>
</tbody>
</table>
**Description:** xsd:string - S(0/unbounded)
Name: xsd:string - S(0/1)
MimeType: europass:MimeTypeEnumeration - S(1/1)
Data: xsd:base64Binary - S(1/1)
MetadataList: europass:MetadataListType - S(0/1)

Must have a unique identifier attribute and a sequence of sub-elements that describe the attachment in detail.

**Instance:**

```xml
<AttachmentList>
   <Attachment id=""
      <Name>{0,1}</Name>
      <MimeType>{1,1}</MimeType>
      <Data>{1,1}</Data>
      <MetadataList>{0,1}</MetadataList>
   </Attachment>
</AttachmentList>
```

**Indicative example:**

```xml
<AttachmentList>
   <Attachment id="ATT_1"
      <Name>licence.pdf</Name>
      <MimeType>application/pdf</MimeType>
      <Data>!---base-64 encoded bytes go here --></Data>
      <MetadataList>
         <Metadata key="number-of-pages" value="5"/>
      </MetadataList>
      <Description>Copy of the </Description>
   </Attachment>
   <Attachment id="ATT_2"
      <Name>My scanned Diploma </Name>
      <MimeType>image/jpeg</MimeType>
      <Data>!---base-64 encoded bytes go here --></Data>
   </Attachment>
</AttachmentList>
```
4.7 SkillsPassport/CoverLetter

This section of the XML describes the information about a Cover Letter that is good to accompany the rest Europass documents.

A Cover Letter is usually a letter addressed to a person that represents the organisation in which the individuals pursues an work, training, education, volunteering, or other kind of placement. It is includes contact information details about the author, as well as the Addressee. The main area of the letter states the main argumentation of the author defending his/her eligibility for the placement. The letter usually concludes with a closing salutation where the author “signs”.

The various sections are defined in details by the Europass XML Schema, and more specifically EuropassCoverLetter.xsd.

The entire structure for the cover letter is grouped under the core element CoverLetter. It adheres to type CoverLetterType, which is presented in details below.

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
</table>
| /SkillsPassport/CoverLetter | Addressee: europass:AddresseeType - S(0/1)  
Letter: europass:LetterType - S(0/1)  
Documentation: europass:GenericDocumentationType - S(0/1) | A core element of Europass schema, as it includes all information about the cover letter. |
| /SkillsPassport/CoverLetter/Addressee | PersonName: europass:PersonNameType - S(1/1)  
Position: europass:OccupationalField - S(0/1)  
Organisation: europass:OrganisationType - S(0/1) | Contains the personal information of the individual, to which the cover letter is addressed. Such as the title and name, the position and the organisation which she/he represents. |
### 4.7.1 SkillsPassport/CoverLetter/Addressee

The entire structure for the addressee of the cover letter is grouped under the element `Addressee`. It adheres to type `AddresseeType`, which is presented in details below.
<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>/SkillsPassport/CoverLetter/Addressee</td>
<td>PersonName: europass:PersonNameType - S(1/1) Position: europass:OccupationalFieldType - S(0/1) Organisation:europass:OrgansationType-S(0/1)</td>
<td>Contains the personal information of the individual, to which the cover letter is addressed. Such as the title and name, the position and the organisation which she/he represents.</td>
</tr>
</tbody>
</table>

**Instance**

```xml
/Addressee
  <PersonName>{0,1}</PersonName>
  <Position>{0,1}</Position>
  <Organisation>{0,1}</Organisation>
</Addressee>
```

**Example**

```xml
/Addressee
  <PersonName>
  <Title>
    <Code>dr</Code>
    <Label>Dr.</Label>
  </Title>
  <FirstName>John</FirstName>
  <Surname>Stuart</Surname>
  <Position>
    <Code>12332</Code>
    <Label>Human resource manager</Label>
  </Position>
  <Organisation>
    <Name>Clipper emergency center</Name>
    <ContactInfo>
      <Address>
        <AddressLine>Wall street 42</AddressLine>
        <PostalCode>SW1P 3AT</PostalCode>
        <Municipality>London</Municipality>
        <Country>
          <Code>UK</Code>
          <Label>United Kingdom</Label>
        </Country>
      </Address>
      <Contact>
      </Contact>
    </ContactInfo>
  </Organisation>
</Addressee>
```
4.7.2 SkillsPassport/CoverLetter/Letter

The entire structure for the letter per se is grouped under the element Letter. It adheres to type LetterType, which is presented in details below.

![Diagram of LetterType structure]

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>/SkillsPassport/ CoverLetter / Letter</td>
<td>Localisation:europass:LetterLocalisationType -S(0/1) OpeningSalutation:europass:OpeningSalutationType -S(0/1) SubjectLine:xsd:string -S(0/1) MainBody:europass:BodyType -S(0/1) ClosingSalutation:europass:ClosingSalutationType -S(1/1)</td>
<td>Contains the main content of the cover letter, the argumentation of the individual about her/his eligibility for the pursued placement.</td>
</tr>
<tr>
<td>/SkillsPassport/ CoverLetter / Letter / Localisation</td>
<td></td>
<td>Contains information about the date and place of the writing of the letter.</td>
</tr>
<tr>
<td>/SkillsPassport/ CoverLetter / Letter / OpeningSalutation</td>
<td></td>
<td>Contains the opening salutation preferred for this letter.</td>
</tr>
<tr>
<td>/SkillsPassport/ CoverLetter / Letter / MainBody</td>
<td></td>
<td>Contains the main body of this letter.</td>
</tr>
<tr>
<td>/SkillsPassport/ CoverLetter / Letter / ClosingSalutation</td>
<td></td>
<td>Contains the preferred closing salutation for the letter.</td>
</tr>
</tbody>
</table>
**Instance**

```xml
<Letter>
  <Localisation>{0,1}</Localisation>
  <SubjectLine>{0,1}</SubjectLine>
  <OpeningSalutation>{0,1}</OpeningSalutation>
  <MainBody>{0,1}</MainBody>
  <ClosingSalutation>{0,1}</ClosingSalutation>
</Letter>
```

**Example**

```xml
<Letter>
  <Localisation>
    <Date year="2013" month="-10" day="---15"/>
    <Municipality>Birmingham</Municipality>
  </Place>
</Localisation>
<SubjectLine>Ref. IT support officer/2013/01/AD</SubjectLine>
<OpeningSalutation>
  <Label>Dear Mr.</Label>
  <Surname>Stuart</Surname>
</OpeningSalutation>
<Body>
  &lt;p&gt;I would like to express my interest ...&lt;/p&gt;
  &lt;p&gt;I am confident that the experience acquired in my present job...&lt;/p&gt;
</Body>
<Closing>
  &lt;p&gt;I am available for interview ...&lt;/p&gt;
</Closing>
</Letter>
```

**4.7.2.1 SkillsPassport/CoverLetter/Letter/Localisation**

The entire structure for the date and place information of the letter is grouped under the element `Localisation`. It adheres to type `LetterLocalisationType`, which is presented in details below.
4.7.2.2 SkillsPassport/CoverLetter/Letter/OpeningSalutation

The entire structure for the opening salutation of the letter is grouped under the element **OpeningSalutation**. It adheres to type **OpeningSalutationType**, which is presented in details below.

More specifically, the Code needs to adhere to the pattern:

opening-salut-[1-9][1][0-9]{1}?(-impersonal)?
4.7.2.3 SkillsPassport/CoverLetter/Letter/Body

The entire structure for the main body of the letter is grouped under the element **MainBody**. It adheres to type **BodyType**, which is presented in details below.

![Diagram of BodyType]

4.7.2.4 SkillsPassport/CoverLetter/Letter/ClosingSalutation

The entire structure for the closing salutation of the letter is grouped under the element **ClosingSalutation**. It adheres to type **ClosingSalutationType**, which is presented in details below.

More specifically, the Code needs to adhere to the pattern:

closing-salut-[1-9][1][0-9]*[1])?
4.7.3 SkillsPassport/CoverLetter/Documentation

The entire structure for the documentation of the cover letter is grouped under the element Documentation. It adheres to type GenericDocumentationType, which is presented in details below.

<table>
<thead>
<tr>
<th>Elements and Attributes</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>/SkillsPassport/</td>
<td></td>
<td>Contains diverse references to other resources that accompany the cover letter. Those resources may be other Europass Documents, list of references to material attached to this XML instance, or even list of references to external resources, outside this specific XML. The element may contain an optional child element, Heading, which will indicate the preferred heading/label of this section.</td>
</tr>
<tr>
<td>CoverLetter/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Documentation</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Heading:europass:InterDocumentationHeadingLabelType - S(0/1)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>InterDocument:europass:InterDocumentDocumentationType - S(0/1)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IntraDocument:europass:IntraDocumentDocumentationType - S(0/1)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ExtraDocument:europass:ExtraDocumentDocumentationType - S(0/1)</td>
<td></td>
</tr>
<tr>
<td>/SkillsPassport/</td>
<td>Code: xsd:string -S(0/1)</td>
<td>Defines the preferred heading/label of this section. The Code needs to adhere to the pattern: ^heading_[1-9]$</td>
</tr>
<tr>
<td>CoverLetter/</td>
<td>Label : xsd:string - S(0/1)</td>
<td></td>
</tr>
<tr>
<td>Documentation/</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Heading</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Code</td>
<td>xsd:string</td>
<td>S(0/1)</td>
</tr>
<tr>
<td>Label</td>
<td>xsd:string</td>
<td>S(0/1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
/SkillsPassport/CoverLetter/Documentation/InterDocument | See 5.13 | Defines a list of references to other Europass Documents.

/SkillsPassport/CoverLetter/Documentation/IntraDocument | See 5.12 | Defines a list of references to resources included within this specific XML instance.

/SkillsPassport/CoverLetter/Documentation/ExtraDocument | See 5.14 | Defines a list of references to resources outside the scope of this XML instance.

**Instance**

```xml
<Documentation>
  <InterDocument>{0,1}</InterDocument>
  <IntraDocument>{0,1}</IntraDocument>
  <ExtraDocument>{0,1}</ExtraDocument>
</Documentation>
```
5 Generic Data Types

5.1 LabelType

The **LabelType** data type defines a sequence of two sub-elements **Code** and **Label**. This data type is used to describe any information that may be referenced back to a taxonomy based on the Code element. The Label element includes the translation of this text to the language of the document.

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
</table>
| **LabelType** | Code:xsd:string - S(0/1)  
Label:xsd:string - S(0/1) | Describes any information that may be referenced back to a taxonomy. |

5.2 OccupationalFieldType

The **OccupationalFieldType** data type restricts the **LabelType** (described in 5.1) by defining that the **Code** sub-element must correspond to the occupational field code type is defined in the included schema "EuropassiSCO_88_COM". The **Label** sub-element is actually the translation of the type in the language of the document.
### 5.3 AddressInfoType

The **AddressInfoType** data type defines a sequence of four sub-elements: **AddressLine**, **PostalCode**, **Municipality** and **Country**, out of which only **Country** is mandatory. This data type is used to describe an address by providing the address-line, usually referring to the residence’s street and street number, the municipality, the postal code and the country.

The minimum required information is the **Country** element, which adheres to the **CountryType** data-type (described in 5.4).

#### Instance:

```xml
<Contact>
  <AddressLine>[0,1]</AddressLine>
  <AddressLine2>[0,1]</AddressLine2>
</Contact>
```
<PostalCode>{0,1}</PostalCode>
<Municipality>{0,1}</Municipality>
<Country>{0,1}</Country>
</Contact>

**Indicative Example:**

```xml
<Contact>
  <AddressLine>12 Strawberry Hille</AddressLine>
  <AddressLine2>Bld B, Apt 12</AddressLine2>
  <PostalCode>12345</PostalCode>
  <Municipality>London</Municipality>
  <Country>
    <Code>UK</Code>
    <Label>United Kingdom</Label>
  </Country>
</Contact>
```

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
</table>
| **AddressInfoType** | AddressLine: xsd:string -S(0/1)  
AddressLine2: xsd:string -S(0/1)  
PostalCode : xsd:string - S(0/1)  
Municipality: xsd:string - S(0/1)  
Country: europass:CountryType- S(0/1) | Describes the details of a postal address. |

**5.4 CountryType**

The **CountryType** data type restricts the **LabelType** (defined in 5.1) by defining that the **Code** element must have one of the values defined by the `europass:countryCode` defined in the included schema "EuropassISOCountries". E.g. code: FR, label: France.
<table>
<thead>
<tr>
<th>Data Type</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CountryType</strong></td>
<td>Code: <code>xsd:string</code> - S(0/1) Label : <code>xsd:string</code> - S(0/1)</td>
<td>Defines the structure of an element that describes a country.</td>
</tr>
</tbody>
</table>

**Instance:**

```
<Country>
  <Code>{0,1}</Code>
  <Label>{0,1}</Label>
</Country>
```

**Indicative Example:**

```
<Country>
  <Code>UK</Code>
  <Label>United Kingdom</Label>
</Country>
```

### 5.5 ContactAddressType

The **ContactAddressType**, restricts the **ContactMethodType** (described in 4.5.1.3.6.1) by defining that the **Contact** element adheres to the data type **AddressInfoType** (described in 5.2), and that the **Use** element is not necessary.

**Address**, as all the rest contact method elements that are based on the **ContactMethodType** data type, consists of two elements **Contact** and **Use**.
Contact element is of type AddressInfoType that defines an address which is described by the address-line, usually referring to the residence’s street and street number, the municipality, postal code and country. The minimum required information is the Country element.

The Use element defines the use of the contact method, e.g. home or work address.

### 5.6 OrganisationType

The OrganisationType data type defines a sequence of two sub-elements: Name and ContactInfo. It is used when an element needs to describe an Organisation. An organisation is described at least by a Name. Optionally its description may include a ContactInfo element.

The optional ContactInfo element adheres to the type OrganisationalContactInfoType data type. This type defines a sequence of two sub-elements: Address and Website, both being optional.

The Address element adheres to the data type ContactAddressType (described in 5.5), while the Website adheres to the data type OrganisationalWebsiteType.

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Child Element</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Content Type</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sequence(S)</td>
<td>Choice(C)</td>
</tr>
</tbody>
</table>
### Attributes (@)

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OrganisationType</strong></td>
<td>Describes an organisation.</td>
</tr>
</tbody>
</table>
| **OrganisationalContactInfoType** | Address: xsd:string -S(0/1)  
Website : europass:OrganisationalWebsiteType-S(0/1)                                                                 |
| **OrganisationalWebsiteType** | Restriction on europass:ContactMethodType  
Use: [complexType]-S(0/1)  
Restriction on europass:LabelType  
Code: fixed to “business”.                                                                 |

### 5.7 AbstractExperienceListType

The `europass:AbstractExperienceListType` is an abstract type that defines list of Experience elements. The elements may be of any type.

### 5.8 ExperienceListType

The `europass:ExperienceListType` defines a list of Experience elements. The elements must be of type `europass:ExperienceType`.

### 5.9 ExperienceType

The `europass:ExperienceType` data type defines a sequence of three sub-elements: Period, Description and Documentation, out of which the Period is mandatory. This data type is used to describe any experience that is relevant for the curriculum vitae of an individual.
| ExperienceType Attributes(@) |
|-----------------------------|--------------------------------------------------|
| Period:europass:PeriodType - S(0/1) | Period: Defines the period (starting date and ending date or a Boolean indicator that the period extends until today) |
| Description:xsd:string - S(0/1) | Description: A text describing the experience |
| Documentation:europasss:DocumentationType - S(0/1) | Documentation: A List of references to attached digital documents that provide evidence or complement this experience (e.g. recommendation letter) |

### 5.10 PeriodType

The **europass:PeriodType** data type defines a sequence three sub elements: From, To and Current. Its purpose is to describe a period, with starting date and either an end date or a Boolean indicator of whether this period extends until today. The end date and the current indicator are optional elements.

#### Data Type Compositions

<table>
<thead>
<tr>
<th>Child Element</th>
<th>Content Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence(S)</td>
<td>Choice(C)</td>
</tr>
<tr>
<td>(minOccurs/maxOccurs)</td>
<td>Attributes(@)</td>
</tr>
</tbody>
</table>

#### Definition

<table>
<thead>
<tr>
<th>PeriodType From:europass:DateType - S(1/1)</th>
<th>From: the date that the period starts</th>
</tr>
</thead>
<tbody>
<tr>
<td>To:europass:DateType- S(0/1)</td>
<td>To: the date that the period ends</td>
</tr>
<tr>
<td>Current:xsd:boolean - S(0/1)</td>
<td>Current: a Boolean indicator that</td>
</tr>
<tr>
<td></td>
<td>denotes whether this period extends</td>
</tr>
<tr>
<td></td>
<td>until today.</td>
</tr>
</tbody>
</table>
5.11 DateType

The **DateType** data type defines an element that may have up to three attributes: `day`, `month` and `year`, out of which `year` is required. Its purpose is to describe a date, which consists of the day, the month and the year.

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>DateType</td>
<td>day : xsd:gDay - optional</td>
<td>Defines date information, which consists of day, month and year. The date must include at least one year element.</td>
</tr>
<tr>
<td></td>
<td>month: xsd:gMonth - optional</td>
<td></td>
</tr>
<tr>
<td></td>
<td>year: xsd:gYear - required</td>
<td></td>
</tr>
</tbody>
</table>

5.12 IntraDocumentDocumentationType

The **IntraDocumentDocumentationType** data type defines a list of **ReferenceTo** elements, each being a reference to an attached digital document.

The **ReferenceTo** element adheres to the **InternalReferenceType** data type which defines that the element needs to have an attribute, named `idref`, which actually is a reference to the id of an element within the same XML document. The references element is an **Attachment** element, found somewhere in the list of SkillsPassport/AttachmentList (described in 4.6).
### Data Type

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>IntraDocumentDocumentationType</td>
<td>ReferenceTo:InternalReferenceType (0, unbounded)</td>
<td>Lists ReferenceTo elements that reference digital documents attached to this specific Europass Document and included in the XML document.</td>
</tr>
<tr>
<td>InternalReferenceType</td>
<td>\textit{idref} \textit{xsd:IDREF} \textit{- required}</td>
<td>Defines that the \textit{idref} attribute needs to reference the id of an existing element in the XML Document.</td>
</tr>
</tbody>
</table>

#### Instance:

```xml
<Documentation>
  <ReferenceTo idref="">{0,unbounded}</ReferenceTo>
</Documentation>
```

#### Indicative example:

```xml
<Documentation>
  <ReferenceTo idref="ATT_1"/>
  <ReferenceTo idref="ATT_2"/>
</Documentation>
```

### 5.13 InterDocumentDocumentationType

The \textit{InterDocumentDocumentationType} data type defines a list of \textit{ReferencedDocument} elements, each being a reference to a Europass Document acronym name, as defined by the DocumentTypeType data type.
The **ReferencedDocument** element adheres to the **EuropassDocumentReferenceType** data type which defines that the element needs to have an attribute, named `ref`, which actually is a reference to the acronym of a Europass Document.

### Data Type

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>InternalReferenceType</strong></td>
<td>Ref europass:DocumentTypeType - required</td>
<td>Defines that the <code>ref</code> attribute needs to reference the acronym of a Europass Document.</td>
</tr>
</tbody>
</table>

**Instance:**

```
<Documentation>
  <ReferencedDocument ref="">{0,unbounded}</ReferencedDocument>
</Documentation>
```

**Indicative example:**

```
<InterDocument>
  <ReferencedDocument ref="ECV"/>
  <ReferencedDocument ref="ESP"/>
  <ReferencedDocument ref="ELP"/>
</InterDocument>
```

### 5.14 ExtraDocumentDocumentationType

The **ExtraDocumentDocumentationType** data type defines a list of **ReferencedResource** elements, each being a reference to an external resource, that is a resource outside this XML, e.g. a public URI.
The **ReferencedResource** element adheres to the **ExternalReferenceType** data type which defines that the element needs to have a child element, named **Description**, which will describe the nature of the references external resource. It may also have an optional attribute, named **href**, which will provide the URI of the referenced resource.

**Data Type**

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExtraDocumentDocumentationType</td>
<td>Sequence(S)</td>
<td>Lists ReferencedResource elements that reference external (outside this XML) resources.</td>
</tr>
<tr>
<td>ExternalReferenceType</td>
<td>Description: xsd:string - S(0,unbounded)</td>
<td>Defines that the element must have a child Description element, and a href attribute.</td>
</tr>
</tbody>
</table>

**Instance:**

```xml
<ExtraDocument>
  <ReferencedResource href="">
    <Description>{0,unbounded}</Description>
  </ReferencedResource>
</ExtraDocument>
```

**Indicative example:**

```xml
<ExtraDocument>
  <ReferencedResource>
    <Description>List of Citations</Description>
  </ReferencedResource>
  <ReferencedResource href="http://myvideocv.com/jim.burnett">
    <Description>Video CV</Description>
  </ReferencedResource>
</ExtraDocument>
```
### 5.15 FileDataType

The **FileDataType** data type defines a sequence of elements and is used to describe an attached digital file.

![Diagram of FileDataType]

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
</table>
| **FileDataType**| Name: xsd:string - S(0/1)  
MimeType: europass:MimeTypeEnumeration - S(1/1)  
Data: xsd:base64Binary - S(1/1)  
MetadataList: europass:MetadataListType - S(0/1) | Describes a digital document.  
**Name**: defines the original file-name.  
**MimeType**: defines the file type of the document and must follow the MimeTypeEnumeration data type.  
**Data**: includes the raw base-64 encoded bytes of the document.  
**MetadataList**: lists Metadata elements, each offering some extra information about this document. |
| **MimeTypeEnumeration** | Restriction on xsd:string - S(1/1)  
[enumeration] : image/jpeg  
[enumeration] : image/pjpeg  
[enumeration] : image/png  
[enumeration] : image/x-png  
[enumeration] : application/pdf | Restricts the string by defining an enumeration of allowed file mime types. |
| **MetadataListType** | Metadata: europass:MetadataType - S(0/unbounded) | Lists Metadata elements, each offering some extra information about this document. |
| **MetadataType** | key: xsd:string  
value: xsd:string | Defines specific metadata information identified by the key attribute and with text specified by the value attribute. |
5.16 SkillType

The **SkillType** data type defines any skill or competence that an individual demonstrates. A **SkillType** defines a sequence of sub-elements: **Description, ProficiencyLevel, AcquiredDuring, VerifiedBy** and **Documentation**.

The **Description** element provides a text-based description of this skill.

The **ProficiencyLevel** denotes the level of expertise of the individual when it comes to the specific skill.

The **AcquiredDuring** element includes a list of Experiences in which the individual participated in, and which had as learning outcome the acquisition of this skill or competence.

The **VerifiedBy** element includes a list of Certificates each of which verifies that the individual has the specific skill or competence.

Finally, the **Documentation** element lists references to attached digital documents that are included in the XML Document and that provide evidence for the possession of this skill by the learner.
ExperienceListType

Experience: europass: AbstractExperienceListType-S(1/unbounded)
Defines a list of Experience elements, each adhering to xsd:anyType

CertificateListType

Certificate: europass:CertificateType-S(1/unbounded)
Defines a list of Certificate elements, each adhering to the CertificateType

CertificateType

Title: xsd:string-S(1/1)
AwardingBody: xsd:string-S(0/1)
Date: europass:DateType-S(0/1)
Level: xsd:anyType-S(0/1)
Defines information about a certificate related to the awarded title, the date of awarding, the body that awarded it and finally the level against a specific classification to which this certificate corresponds.

5.17CEFLanguageLevelType

The CEFLanguageLevelType data type defines a specific pattern for the CEF language level. According to this pattern, an element that adheres to this type must be a string that is comprised of one letter (either A, B or C) and one digit (either 1 or 2). E.g. A1, B2, C1. The levels are defined by the Common European Framework of Reference for Languages (CEFR).

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Compositions</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>CEFLanguageLevelType</td>
<td>Restriction xsd:string</td>
<td>Defines a string that is comprised of one letter (either A, B or C) and one digit (either 1 or 2). E.g. A1, B2, C1</td>
</tr>
</tbody>
</table>