Schema documentation for Employment.xsd

december 8, 2018

Table of Contents

Namespace: "employment" 4
Schema(s) 4
Main schema Employment.xsd 4
Element(s) 4
Element employment 4
Element employmentType / control 5
Element controlType / recordId 6
Element controlType / status 6
Element controlType / maintenanceAgency 6
Element maintenanceAgencyType / agencyCode 7
Element maintenanceAgencyType / otherAgencyCode 7
Element maintenanceAgencyType / agencyName 8
Element controlType / languageDeclaration 8
Element languageDeclarationType / language 9
Element languageDeclarationType / script 9
Element controlType / conventionDeclaration 10
Element conventionDeclarationType / code 11
Element conventionDeclarationType / citation 11
Element conventionDeclarationType / description 11
Element controlType / ownValueListDeclaration 12
Element ownValueListDeclarationType / code 12
Element ownValueListDeclarationType / citation 13
Element ownValueListDeclarationType / description 13
Element controlType / maintenanceHistory 14
Element maintenanceHistoryType / maintenanceHistoryEvent 14
Element additionalElements 15
Element ownElement 16
Element ownElement / ownElementDescription 17
Element ownElement / ownElement 17
Element value 18
Element property 18
Element attribute 19
Element ownElementType / ownElement 20
Element additionalXMLData 21
Element appendix 21
Element AppendixType / eSignature 23
Element eSignatureComplexType / signature 23
Element employmentType / employmentDescription 24
Element employmentDescriptionType / employmentId 26
Element employmentDescriptionType / applicationDocuments 26
Element applicationDocumentsType / applicationDocument 27
Element applicationDocumentType / description 28
Element employmentDescriptionType / positions 28
Element positionsType / position 28
Element positionType / term 29
Element positionType / description 30
Element employmentDescriptionType / employmentDates 30
Element employmentDatesType / employmentDate 31
Element employmentDateType / startDate 31
Element employmentDateType / endDate 32
Element employmentDescriptionType / employmentTerminations 32
Element employmentTerminationsType / employmentTermination 32
Element employmentTerminationType / terminationDecision 33
Element terminationDecisionType / description 34
Element employmentTerminationType / terminationApplication 34
Element terminationApplicationType / description 35
Element employmentDescriptionType / firstWorkDay 35
Element employmentDescriptionType / contracts 36
Element contractsType / contract 36
Element contractType / description 37
Element employmentDescriptionType / agreements 37
Element agreementsType / agreement 38
Element agreementType / description 38

Element employmentDescriptionType / workplaces ... 38
Element workplacesType / workplace ... 39
Element employmentDescriptionType / jobLocations ... 40
Element jobLocationsType / jobLocation ... 40
Element employmentDescriptionType / weeklyWorkingTime ... 41
Element weeklyWorkingTimeType / description ... 41
Element employmentDescriptionType / typeOfWorkingTime ... 42
Element employmentDescriptionType / typeOfEmployment ... 42
Element employmentDescriptionType / yearlyVacationDays ... 42
Element employmentDescriptionType / maxAllowedSavedVacationDays ... 43
Element employmentDescriptionType / maxAllowedVacationDaysTakenInAdvance ... 43
Element employmentDescriptionType / sideLineOccupations ... 43
Element sideLineOccupationsType / approvedOccupation ... 44
Element approvedOccupationType / description ... 45
Element employmentDescriptionType / salaryAgreements ... 45
Element salaryAgreementsType / salaryAgreement ... 45
Element salaryAgreementType / salaryPeriod ... 47
Element salaryPeriodType / date ... 47
Element salaryPeriodType / dateRange ... 47
Element dateRangeType / startDate ... 48
Element dateRangeType / endDate ... 48
Element salaryPeriodType / dateSet ... 49
Element dateSetType / date ... 49
Element dateSetType / dateRange ... 49
Element employmentDescriptionType / fringeBenefits ... 50
Element fringeBenefitsType / fringeBenefit ... 50
Element fringeBenefitType / description ... 51
Element fringeBenefitType / value ... 51
Element fringeBenefitType / fringeBenefitPeriod ... 52
Element fringeBenefitPeriodType / date ... 52
Element fringeBenefitPeriodType / dateRange ... 53
Element fringeBenefitPeriodType / dateSet ... 53
Element employmentDescriptionType / paydays ... 53
Element paydaysType / payday ... 54
Element employmentDescriptionType / retirementRegulations ... 54
Element retirementRegulationsType / retirementRegulation ... 55
Element retirementRegulationType / description ... 55
Element retirementRegulationType / value ... 55
Element employmentDescriptionType / retirementAge ... 56
Element employmentDescriptionType / otherTerms ... 56
Element employmentDescriptionType / testimonials ... 56
Element testimonialsType / testimonial ... 57
Element testimonialType / description ... 57
Element employmentDescriptionType / references ... 58
Element referencesType / reference ... 58
Element referenceType / description ... 59
Element employmentDescriptionType / relations ... 59
Element relationsType / relationToParent ... 59
Element relationToParentType / relationEntry ... 60
Element relationsType / relationToChild ... 61
Element relationToChildType / relationEntry ... 61
Element otherTermsType / otherTerm ... 62
Complex Type(s) ... 62
Complex Type employmentType ... 62
Complex Type controlType ... 63
Complex Type maintenanceAgencyType ... 64
Complex Type otherAgencyCodeType ... 65
Complex Type languageDeclarationType ... 65
Complex Type languageType ... 66
Complex Type scriptType ... 66
Complex Type conventionDeclarationType ... 67
Complex Type citationType ... 67
Complex Type ownValueListDeclarationType ... 68
Complex Type maintenanceHistoryType ... 69
Complex Type maintenanceHistoryEventType ... 69
Complex Type additionalElementsType ... 70
Complex Type ownElementType ... 71
Complex Type ExtendingComplexType ... 72
Complex Type AppendixType ... 72
Complex Type eSignatureComplexType ... 74
Complex Type employmentDescriptionType ... 75
Complex Type applicationDocumentsType ... 79

Complex Type applicationDocumentType ... 79
Complex Type positionsType ... 80
Complex Type positionType ... 80
Complex Type termType ... 81
Complex Type employmentDatesType ... 81
Complex Type employmentDateType ... 82
Complex Type employmentTerminationsType ... 82
Complex Type employmentTerminationType ... 82
Complex Type terminationDecisionType ... 83
Complex Type terminationApplicationType ... 84
Complex Type contractsType ... 85
Complex Type contractType ... 85
Complex Type agreementsType ... 86
Complex Type agreementType ... 86
Complex Type workplacesType ... 87
Complex Type workplaceType ... 87
Complex Type jobLocationsType ... 88
Complex Type jobLocationType ... 88
Complex Type weeklyWorkingTimeType ... 89
Complex Type sideLineOccupationsType ... 89
Complex Type approvedOccupationType ... 90
Complex Type salaryAgreementsType ... 91
Complex Type salaryAgreementType ... 91
Complex Type salaryPeriodType ... 93
Complex Type dateRangeType ... 93
Complex Type dateSetType ... 94
Complex Type fringeBenefitsType ... 94
Complex Type fringeBenefitType ... 95
Complex Type fringeBenefitPeriodType ... 95
Complex Type paydaysType ... 96
Complex Type retirementRegulationsType ... 96
Complex Type retirementRegulationType ... 96
Complex Type testimonialsType ... 97
Complex Type testimonialType ... 97
Complex Type referencesType ... 98
Complex Type referenceType ... 98
Complex Type relationsType ... 99
Complex Type relationToParentType ... 100
Complex Type relationToChildType ... 100
Complex Type otherTermsType ... 101
Simple Type(s) ... 102
Simple Type statusType ... 102
Simple Type agencyCodeType ... 102
Simple Type otherAgencyCodePrefix ... 102
Simple Type eventTypeType ... 103
Simple Type agentTypeType ... 103
Simple Type typeType ... 104
Simple Type ownDate ... 104
Simple Type salaryTypeType ... 105
Simple Type currencyType ... 105
Namespace: "" ... 112
Attribute(s) ... 112
Attribute otherAgencyCodeType / @agencyCodePrefix ... 112
Attribute otherAgencyCodeType / @otherAgencyCodePrefix ... 112
Attribute languageType / @languageCode ... 112
Attribute languageType / @language ... 112
Attribute scriptType / @scriptCode ... 113
Attribute scriptType / @script ... 113
Attribute citationType / @href ... 113
Attribute maintenanceHistoryEventType / @eventType ... 113
Attribute maintenanceHistoryEventType / @eventDateTime ... 114
Attribute maintenanceHistoryEventType / @agentType ... 114
Attribute maintenanceHistoryEventType / @otherAgentType ... 114
Attribute maintenanceHistoryEventType / @agent ... 115
Attribute attribute / @name ... 115
Attribute attribute / @dataType ... 115
Attribute attribute / @format ... 115
Attribute ownElementType / @name ... 115
Attribute ownElementType / @dataType ... 116
Attribute ownElementType / @format ... 116
Attribute eSignatureComplexType / @present ... 116
Attribute eSignatureComplexType / @dateESignatureIsVerified ... 116

Attribute AppendixType / @disposable 116
Attribute AppendixType / @name 117
Attribute AppendixType / @description 117
Attribute AppendixType / @fileFormat 117
Attribute AppendixType / @originalFileFormat 117
Attribute AppendixType / @path 118
Attribute AppendixType / @eSignatureHaveExisted 118
Attribute applicationDocumentType / @href 118
Attribute termType / @code 118
Attribute termType / @source 118
Attribute positionType / @href 119
Attribute terminationDecisionType / @href 119
Attribute terminationApplicationType / @href 119
Attribute employmentTerminationType / @typeOfTermination 119
Attribute employmentTerminationType / @otherTypeOfTermination 120
Attribute contractType / @href 120
Attribute agreementType / @href 120
Attribute workplaceType / @code 120
Attribute workplaceType / @name 121
Attribute jobLocationType / @city 121
Attribute jobLocationType / @country 121
Attribute weeklyWorkingTimeType / @unit 121
Attribute weeklyWorkingTimeType / @numberOfUnits 122
Attribute approvedOccupationType / @type 122
Attribute approvedOccupationType / @notificationDate 122
Attribute approvedOccupationType / @decisionDate 122
Attribute approvedOccupationType / @href 122
Attribute salaryAgreementType / @salaryType 123
Attribute salaryAgreementType / @otherSalaryType 123
Attribute salaryAgreementType / @salaryCode 123
Attribute salaryAgreementType / @amount 124
Attribute salaryAgreementType / @currency 124
Attribute testimonialType / @href 128
Attribute referenceType / @href 128
Attribute relationToParentType / relationEntry / @parentOrganisationReferenceCode 128
Attribute relationToParentType / @typeOfRelation 128
Attribute relationToParentType / @href 129
Attribute relationToChildType / relationEntry / @childOrganisationReferenceCode 129
Attribute relationToChildType / @typeOfRelation 129
Attribute relationToChildType / @href 129

Namespace: "employment"

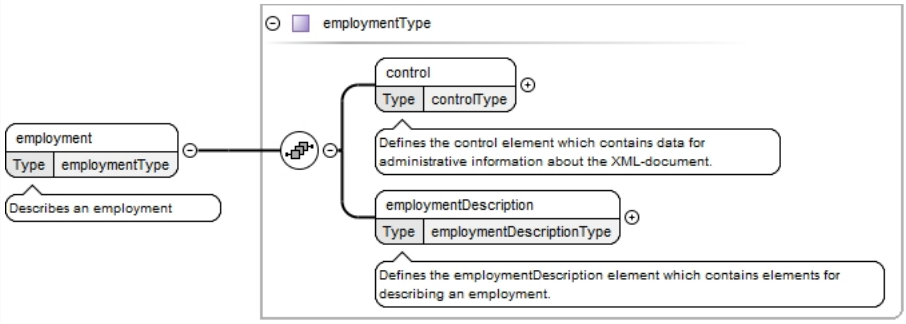
Schema(s)

Main schema `Employment.xsd`

Namespace	employment
Properties	attribute form default: `unqualified` element form default: `qualified`

Element(s)

Element `employment`

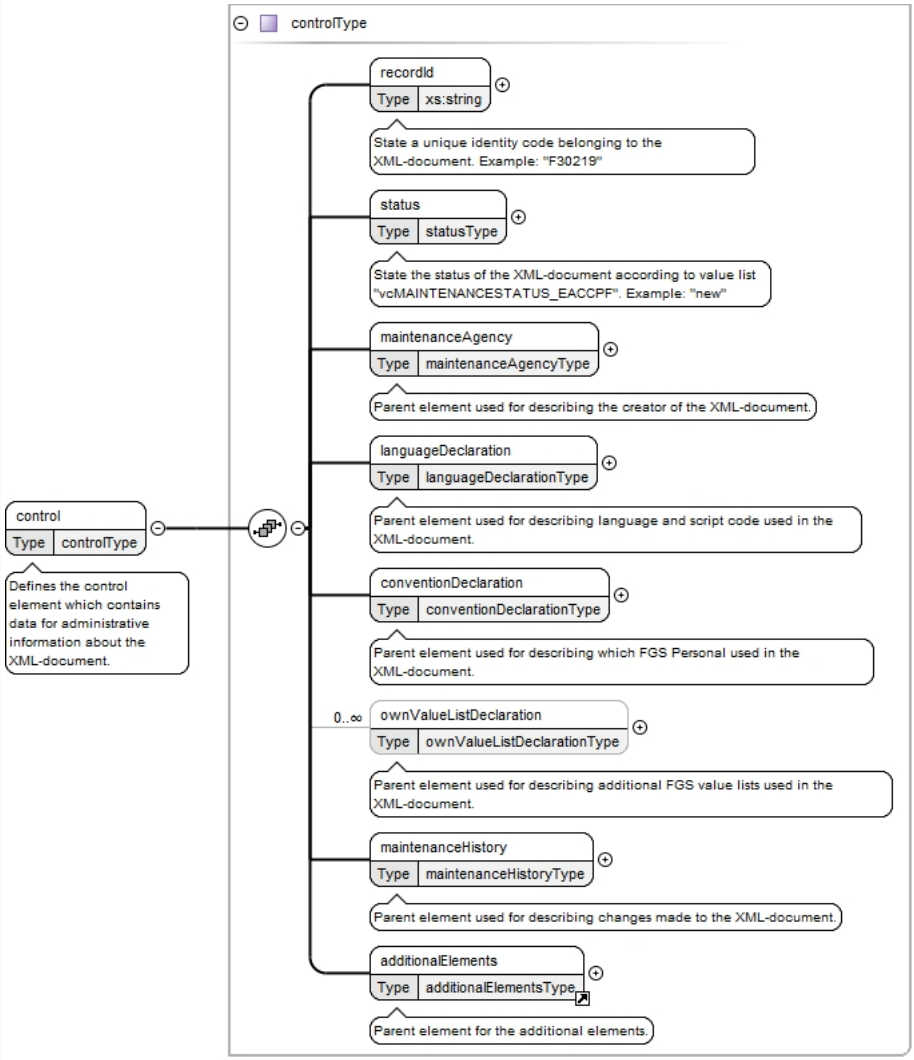
Namespace	employment
Annotations	`Describes an employment`
Diagram	

Type	employmentType
Properties	content: complex
Model	control , employmentDescription
Children	control, employmentDescription

Instance	`<employment xmlns="employment">`
	`  <control>{1,1}</control>`
	`  <employmentDescription>{1,1}</employmentDescription>`
	`</employment>`

Source	`<xs:element name="employment" type="employmentType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describes an employment</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

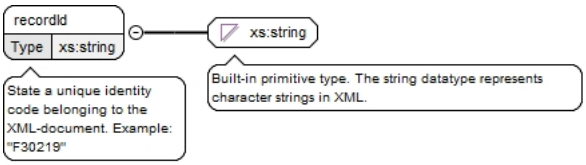
Element employmentType / control

Namespace	employment
Annotations	Defines the control element which contains data for administrative information about the XML-document.
Diagram	
Type	controlType
Properties	content: complex
Model	recordId , status , maintenanceAgency , languageDeclaration , conventionDeclaration , ownValueListDeclaration* , maintenanceHistory , additionalElements
Children	additionalElements, conventionDeclaration, languageDeclaration, maintenanceAgency, maintenanceHistory, ownValueListDeclaration, recordId, status

Instance	`<control xmlns="employment">`
	`  <recordId>{1,1}</recordId>`
	`  <status>{1,1}</status>`
	`  <maintenanceAgency>{1,1}</maintenanceAgency>`
	`  <languageDeclaration>{1,1}</languageDeclaration>`
	`  <conventionDeclaration>{1,1}</conventionDeclaration>`
	`  <ownValueListDeclaration>{0,unbounded}</ownValueListDeclaration>`
	`  <maintenanceHistory>{1,1}</maintenanceHistory>`
	`  <additionalElements>{1,1}</additionalElements>`
	`</control>`

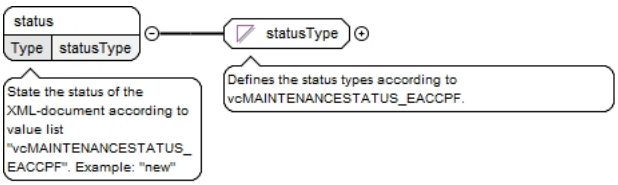
Source	`<xs:element name="control" type="controlType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Defines the control element which contains data for administrative information about the XML-document.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `controlType / recordId`

Namespace	employment
Annotations	`State a unique identity code belonging to the XML-document. Example: "F30219"`
Diagram	
Type	xs:string
Properties	content: `simple`

Source	`<xs:element name="recordId" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State a unique identity code belonging to the XML-document. Example: "F30219"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

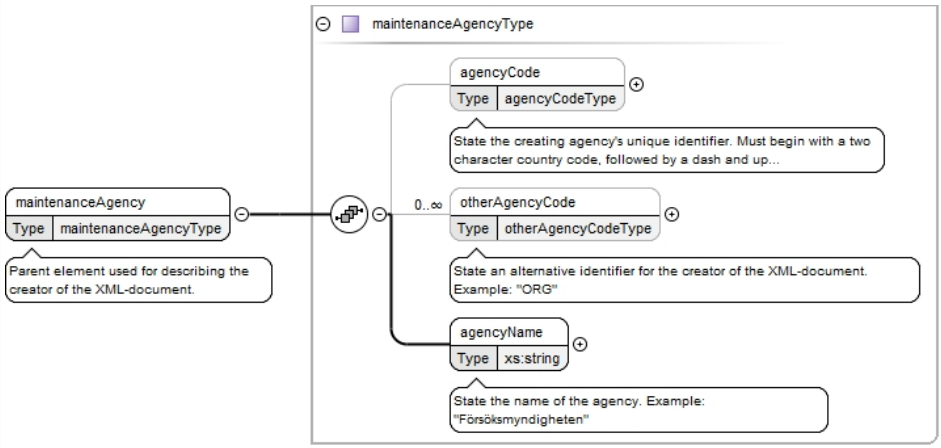
Element `controlType / status`

Namespace	employment
Annotations	`State the status of the XML-document according to value list "vcMAINTENANCESTATUS_EACCPF". Example: "new"`
Diagram	
Type	statusType
Properties	content: `simple`
Facets	enumeration `cancelled` enumeration `deleted` enumeration `deletedReplaced` enumeration `derived` enumeration `new` enumeration `revised`

Source	`<xs:element name="status" type="statusType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the status of the XML-document according to value list "vcMAINTENANCESTATUS_EACCPF". Example: "new"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `controlType / maintenanceAgency`

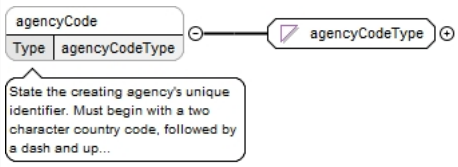
Namespace	employment

Annotations	Parent element used for describing the creator of the XML-document.
Diagram	
Type	maintenanceAgencyType
Properties	content: complex
Model	agencyCode{0,1} , otherAgencyCode* , agencyName
Children	agencyCode, agencyName, otherAgencyCode

Instance	`<maintenanceAgency xmlns="employment">`
	`  <agencyCode>{0,1}</agencyCode>`
	`  <otherAgencyCode agencyCodePrefix="" otherAgencyCodePrefix="">{0,unbounded}</otherAgencyCode>`
	`  <agencyName>{1,1}</agencyName>`
	`</maintenanceAgency>`

Source	`<xs:element name="maintenanceAgency" type="maintenanceAgencyType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element used for describing the creator of the XML-document.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

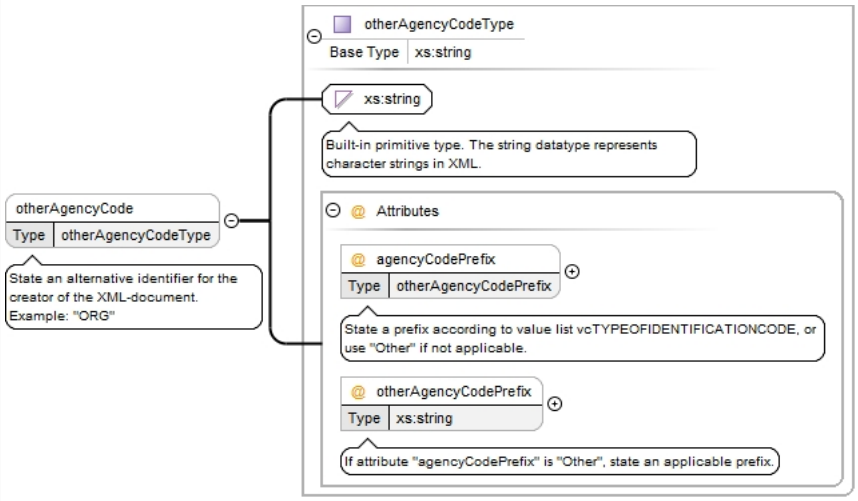
Element maintenanceAgencyType / agencyCode

Namespace	employment
Annotations	State the creating agency's unique identifier. Must begin with a two character country code, followed by a dash and up to 10 characters. Example: "SE-A:ARA/123
Diagram	
Type	agencyCodeType
Properties	content: simple minOccurs: 0
Facets	pattern [A-Z]{2}\-[a-zA-Z0-9/:\-]{1,10}

Source	`<xs:element name="agencyCode" type="agencyCodeType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the creating agency's unique identifier. Must begin with a two character country code, followed by a dash and up to 10 characters. Example: "SE-A:ARA/123</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element maintenanceAgencyType / otherAgencyCode

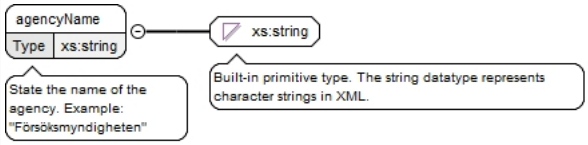
Namespace	employment
Annotations	State an alternative identifier for the creator of the XML-document. Example: "ORG"

Diagram	
Type	otherAgencyCodeType
Properties	content: `complex` minOccurs: `0` maxOccurs: `unbounded`

Attributes	**QName** / **Type** / **Use**
	agencyCodePrefix / otherAgencyCodePrefix / optional
	`State a prefix according to value list vcTYPEOFIDENTIFICATIONCODE, or use "Other" if not applicable.`
	otherAgencyCodePrefix / xs:string / optional
	`If attribute "agencyCodePrefix" is "Other", state an applicable prefix.`

Source	`<xs:element name="otherAgencyCode" minOccurs="0" maxOccurs="unbounded" type="otherAgencyCodeType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State an alternative identifier for the creator of the XML-document. Example: "ORG"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

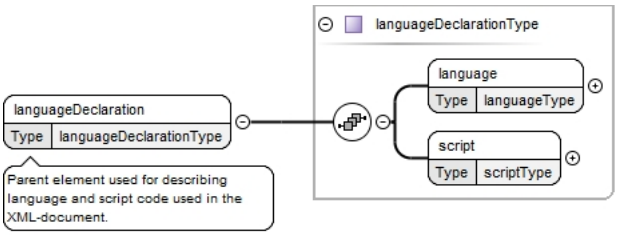
Element `maintenanceAgencyType / agencyName`

Namespace	employment
Annotations	`State the name of the agency. Example: "Försöksmyndigheten"`
Diagram	
Type	xs:string
Properties	content: `simple`

Source	`<xs:element name="agencyName" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the name of the agency. Example: "Försöksmyndigheten"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `controlType / languageDeclaration`

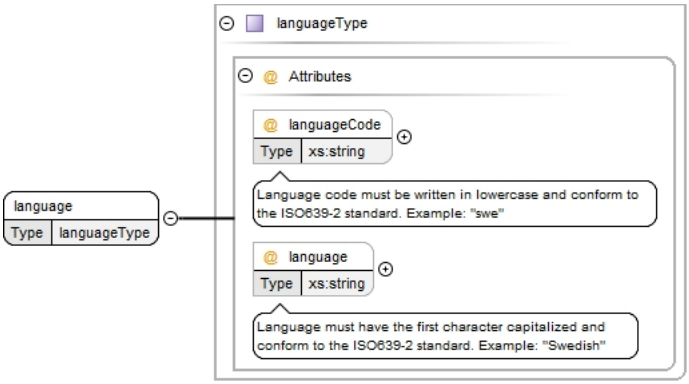
Namespace	employment
Annotations	`Parent element used for describing language and script code used in the XML-document.`

Diagram	
Type	languageDeclarationType
Properties	content: complex
Model	language , script
Children	language, script

Instance	`<languageDeclaration xmlns="employment">`
	`  <language language="" languageCode="">{1,1}</language>`
	`  <script script="" scriptCode="">{1,1}</script>`
	`</languageDeclaration>`

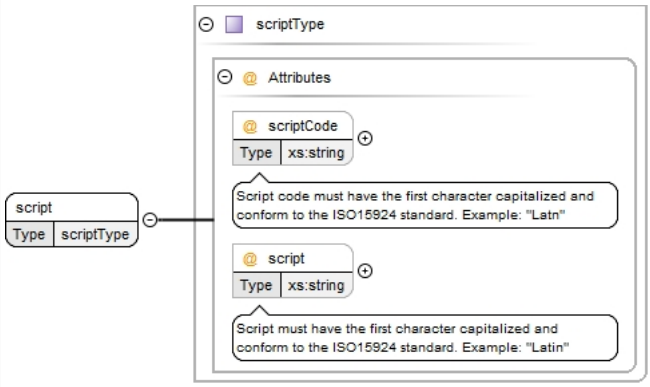
Source	`<xs:element name="languageDeclaration" type="languageDeclarationType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element used for describing language and script code used in the XML-document.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element languageDeclarationType / language

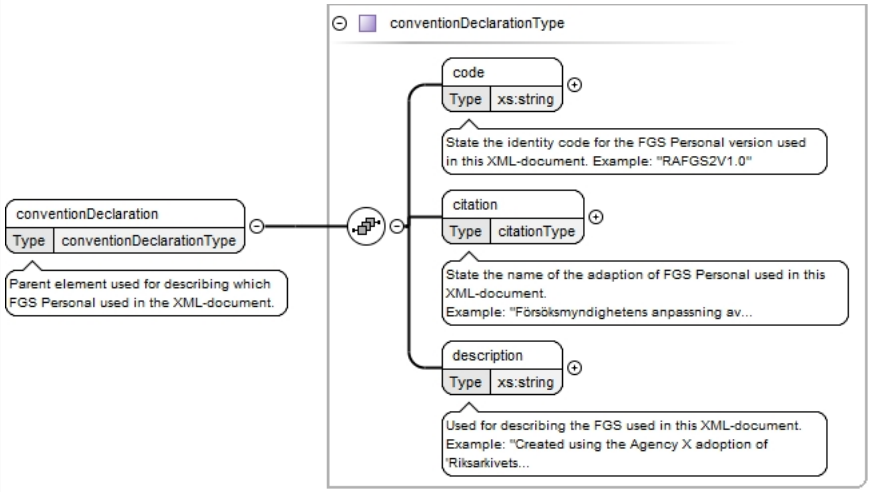
Namespace	employment
Diagram	
Type	languageType
Properties	content: complex
Attributes	**QName** / **Type** / **Use** **language** / xs:string / optional Language must have the first character capitalized and conform to the ISO639-2 standard. Example: "Swedish" **languageCode** / xs:string / optional Language code must be written in lowercase and conform to the ISO639-2 standard. Example: "swe"
Source	`<xs:element name="language" type="languageType"/>`

Element languageDeclarationType / script

Namespace	employment

Diagram	
Type	scriptType
Properties	content: complex
Attributes	**QName** / **Type** / **Use** **script** / xs:string / optional Script must have the first character capitalized and conform to the ISO15924 standard. Example: "Latin" **scriptCode** / xs:string / optional Script code must have the first character capitalized and conform to the ISO15924 standard. Example: "Latn"
Source	`<xs:element name="script" type="scriptType"/>`

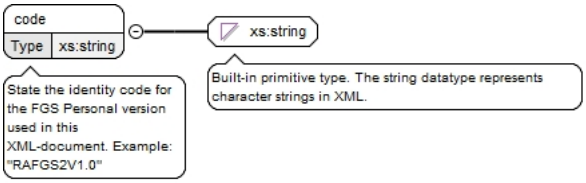
Element `controlType / conventionDeclaration`

Namespace	employment
Annotations	Parent element used for describing which FGS Personal used in the XML-document.
Diagram	
Type	conventionDeclarationType
Properties	content: complex
Model	code , citation , description
Children	citation, code, description

Instance	`<conventionDeclaration xmlns="employment">`
	`  <code>{1,1}</code>`
	`  <citation href="">{1,1}</citation>`
	`  <description>{1,1}</description>`
	`</conventionDeclaration>`

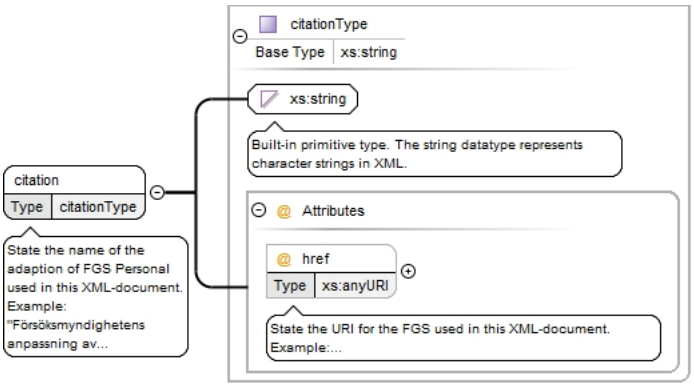
Source	`<xs:element name="conventionDeclaration" type="conventionDeclarationType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element used for describing which FGS Personal used in the XML-document.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element conventionDeclarationType / code

Namespace	employment
Annotations	State the identity code for the FGS Personal version used in this XML-document. Example: "RAFGS2V1.0"
Diagram	
Type	xs:string
Properties	content: simple

Source	`<xs:element name="code" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the identity code for the FGS Personal version used in this XML-document. Example: "RAFGS2V1.0"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

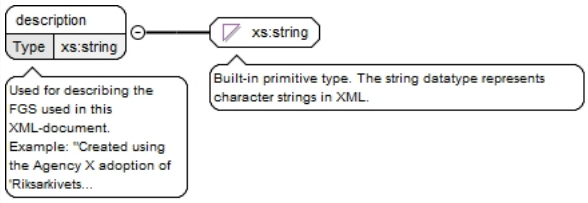
Element conventionDeclarationType / citation

Namespace	employment
Annotations	State the name of the adaption of FGS Personal used in this XML-document. Example: "Försöksmyndighetens anpassning av FGS Personal för överföring av personalinformation till det egna E-arktivet"
Diagram	
Type	citationType
Properties	content: complex
Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional State the URI for the FGS used in this XML-document. Example: "https://myndighet.se/Försöksmyndighetens_FGS_Personal_V_20181224.pdf"

Source	`<xs:element name="citation" type="citationType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the name of the adaption of FGS Personal used in this XML-document.</xs:documentation>`
	`  <xs:documentation xml:lang="en">Example: "Försöksmyndighetens anpassning av FGS Personal för överföring av personalinformation till det egna E-arktivet"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

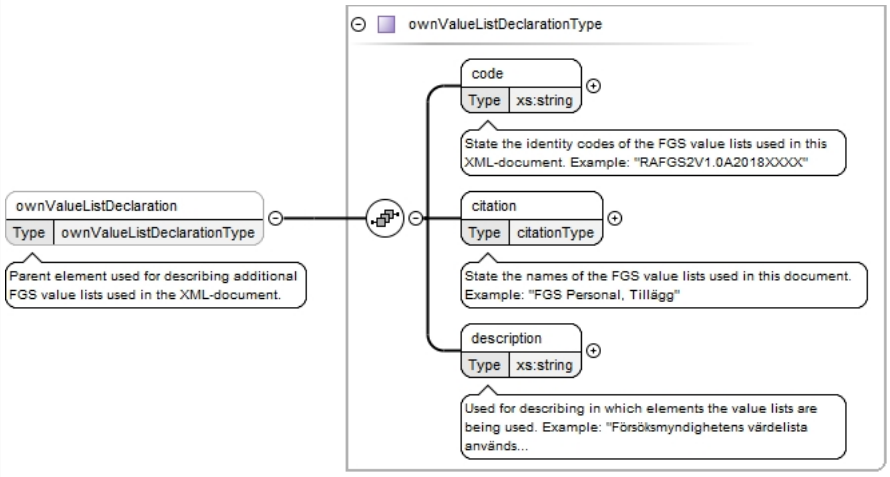
Element conventionDeclarationType / description

Namespace	employment
Annotations	Used for describing the FGS used in this XML-document. Example: "Created using the Agency X adoption of 'Riksarkivets anpassning av FGS Personal'"

Diagram	
Type	xs:string
Properties	content: `simple`

Source	`<xs:element name="description" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for describing the FGS used in this XML-document.</xs:documentation>`
	`  <xs:documentation xml:lang="en">Example: "Created using the Agency X adoption of 'Riksarkivets anpassning av FGS Personal'"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `controlType / ownValueListDeclaration`

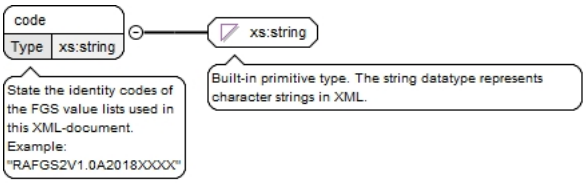
Namespace	employment
Annotations	`Parent element used for describing additional FGS value lists used in the XML-document.`
Diagram	
Type	ownValueListDeclarationType
Properties	content: `complex` minOccurs: `0` maxOccurs: `unbounded`
Model	code , citation , description
Children	citation, code, description

Instance	`<ownValueListDeclaration xmlns="employment">`
	`  <code>{1,1}</code>`
	`  <citation href="">{1,1}</citation>`
	`  <description>{1,1}</description>`
	`</ownValueListDeclaration>`

Source	`<xs:element name="ownValueListDeclaration" type="ownValueListDeclarationType" minOccurs="0" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element used for describing additional FGS value lists used in the XML-document.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

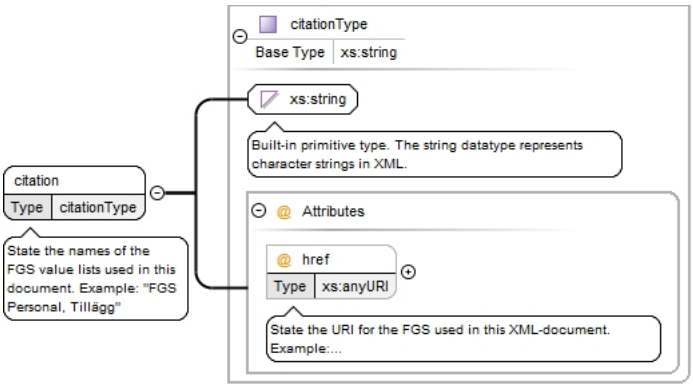
Element `ownValueListDeclarationType / code`

Namespace	employment
Annotations	`State the identity codes of the FGS value lists used in this XML-document. Example: "RAFGS2V1.0A2018XXXX"`

Diagram	
Type	xs:string
Properties	content: `simple`

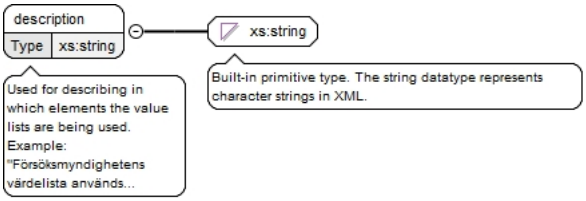
Source	`<xs:element name="code" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the identity codes of the FGS value lists used in this XML-document. Example: "RAFGS2V1.0A2018XXXX"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element ownValueListDeclarationType / citation

Namespace	employment
Annotations	`State the names of the FGS value lists used in this document. Example: "FGS Personal, Tillägg"`
Diagram	
Type	citationType
Properties	content: `complex`
Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional `State the URI for the FGS used in this XML-document. Example: "https://myndighet.se/Försöksmyndighetens_FGS_Personal_V_20181224.pdf"`

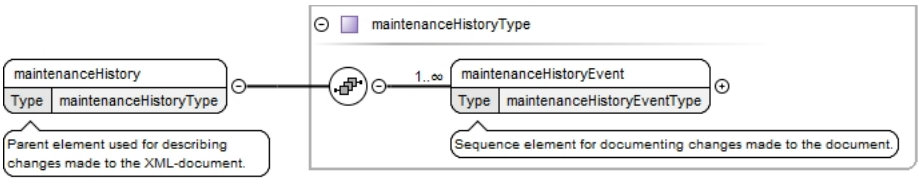
Source	`<xs:element name="citation" type="citationType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the names of the FGS value lists used in this document. Example: "FGS Personal, Tillägg"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element ownValueListDeclarationType / description

Namespace	employment
Annotations	`Used for describing in which elements the value lists are being used. Example: "Försöksmyndighetens värdelista används i elementet AnhörigFält"`
Diagram	
Type	xs:string
Properties	content: `simple`

Source	`<xs:element name="description" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for describing in which elements the value lists are being used. Example: "Försöksmyndighetens värdelista används i elementet AnhörigFält"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `controlType / maintenanceHistory`

Namespace	employment
Annotations	Parent element used for describing changes made to the XML-document.
Diagram	
Type	maintenanceHistoryType
Properties	content: complex
Model	maintenanceHistoryEvent+
Children	maintenanceHistoryEvent

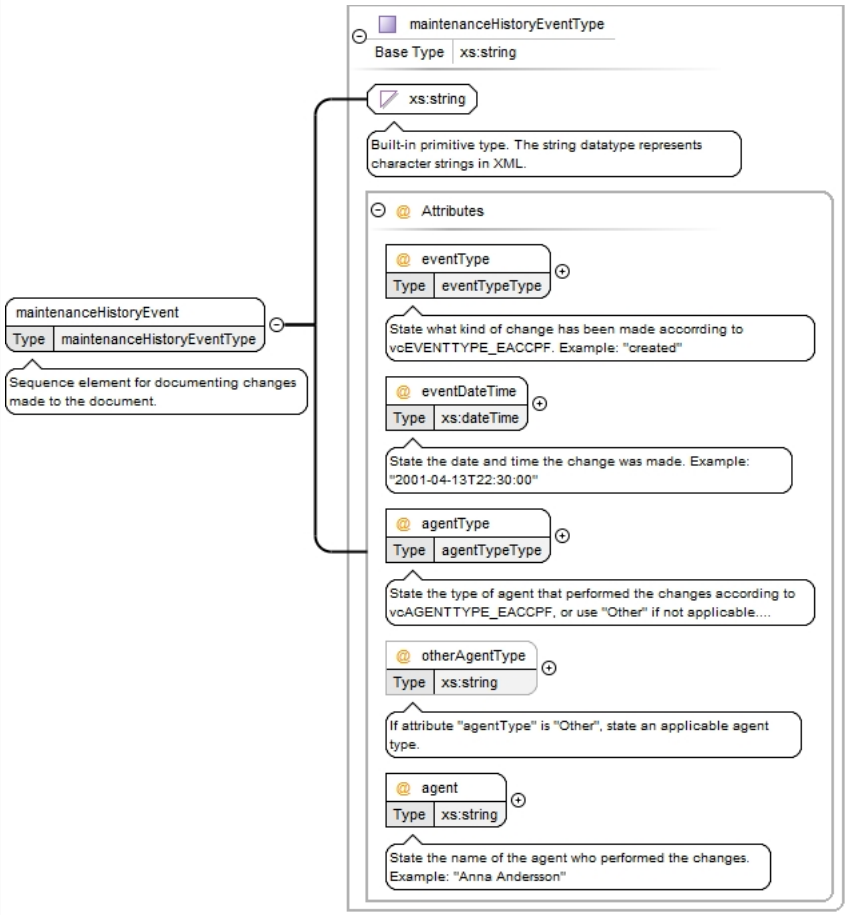
Instance	`<maintenanceHistory xmlns="employment">`
	`  <maintenanceHistoryEvent agent="" agentType="" eventDateTime="" eventType="" otherAgentType="">{1,unbounded}</maintenanceHistoryEvent>`
	`</maintenanceHistory>`

Source	`<xs:element name="maintenanceHistory" type="maintenanceHistoryType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element used for describing changes made to the XML-document.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `maintenanceHistoryType / maintenanceHistoryEvent`

Namespace	employment
Annotations	Sequence element for documenting changes made to the document.

Diagram



Type	maintenanceHistoryEventType
Properties	content: `complex` minOccurs: `1` maxOccurs: `unbounded`

Attributes

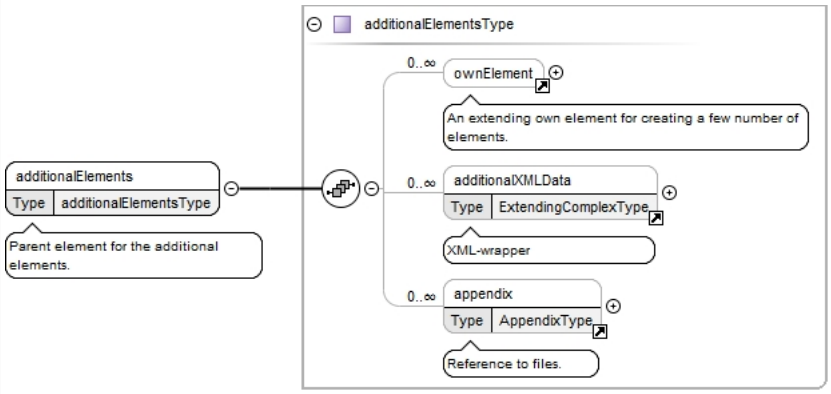
QName	Type	Use	
agent	xs:string	required	
	State the name of the agent who performed the changes. Example: "Anna Andersson"		
agentType	agentTypeType	required	
	State the type of agent that performed the changes according to vcAGENTTYPE_EACCPF, or use "Other" if not applicable. Example: "human"		
eventDateTime	xs:dateTime	required	
	State the date and time the change was made. Example: "2001-04-13T22:30:00"		
eventType	eventTypeType	required	
	State what kind of change has been made accorrding to vcEVENTTYPE_EACCPF. Example: "created"		
otherAgentType	xs:string	optional	
	If attribute "agentType" is "Other", state an applicable agent type.		

Source

```
<xs:element name="maintenanceHistoryEvent" type="maintenanceHistoryEventType" minOccurs="1"
 maxOccurs="unbounded">
  <xs:annotation>
    <xs:documentation xml:lang="en">Sequence element for documenting changes made to the document.</
xs:documentation>
  </xs:annotation>
</xs:element>
```

Element `additionalElements`

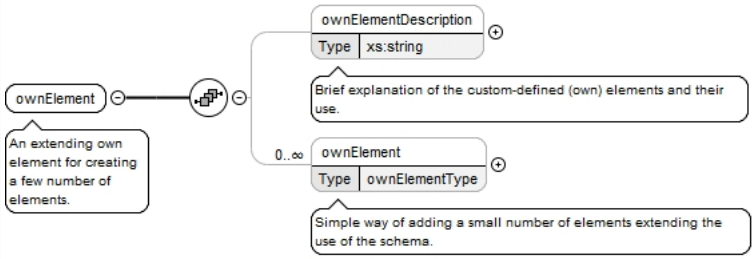
Namespace	employment

Annotations	Parent element for the additional elements.
Diagram	
Type	additionalElementsType
Properties	content: complex
Used by	Complex Types: agreementType, applicationDocumentType, applicationDocumentsType, approvedOccupationType, contractType, controlType, employmentDescriptionType, employmentTerminationType, fringeBenefitType, jobLocationType, positionType, referenceType, salaryAgreementType, terminationApplicationType, terminationDecisionType, testimonialType, weeklyWorkingTimeType, workplaceType
Model	ownElement* , additionalXMLData* , appendix*
Children	additionalXMLData, appendix, ownElement

Instance	`<additionalElements xmlns="employment">`
	`  <ownElement>{0,unbounded}</ownElement>`
	`  <additionalXMLData>{0,unbounded}</additionalXMLData>`
	`  <appendix description="" disposable="" eSignatureHaveExisted="" fileFormat="" name="" originalFileFormat="" path=`
	`appendix>`
	`</additionalElements>`

Source	`<xs:element name="additionalElements" type="additionalElementsType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for the additional elements.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element ownElement

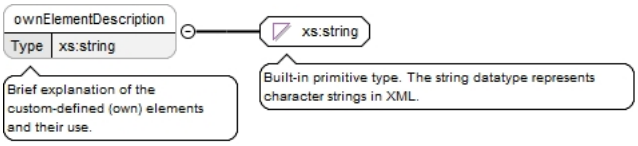
Namespace	employment
Annotations	An extending own element for creating a few number of elements.
Diagram	
Properties	content: complex
Used by	Complex Type: additionalElementsType
Model	ownElementDescription{0,1} , ownElement*
Children	ownElement, ownElementDescription

Instance	`<ownElement xmlns="employment">`
	`  <ownElementDescription>{0,1}</ownElementDescription>`
	`  <ownElement dataType="" format="" name="">{0,unbounded}</ownElement>`
	`</ownElement>`

Source	`<xs:element name="ownElement">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">An extending own element for creating a few number of elements.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:complexType>`
	`  <xs:sequence>`

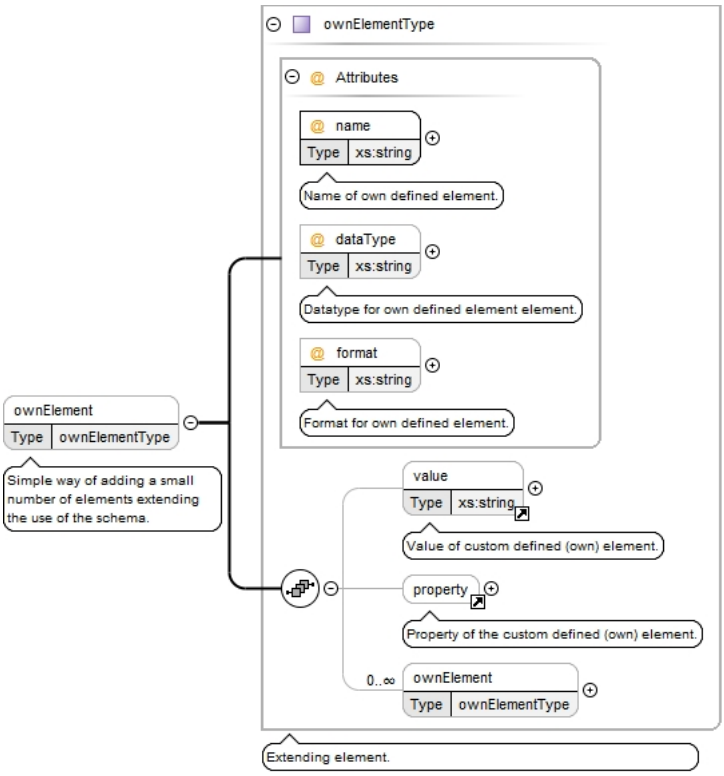
```
      <xs:element name="ownElementDescription" minOccurs="0" type="xs:string">
        <xs:annotation>
          <xs:documentation xml:lang="en">Brief explanation of the custom-defined (own) elements and
 their use.</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="ownElement" type="ownElementType" minOccurs="0" maxOccurs="unbounded">
        <xs:annotation>
          <xs:documentation xml:lang="en">Simple way of adding a small number of elements extending
 the use of the schema.</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

Element ownElement / ownElementDescription

Namespace	employment
Annotations	Brief explanation of the custom-defined (own) elements and their use.
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 0

Source	`<xs:element name="ownElementDescription" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Brief explanation of the custom-defined (own) elements and their use.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element ownElement / ownElement

Namespace	employment
Annotations	Simple way of adding a small number of elements extending the use of the schema.
Diagram	

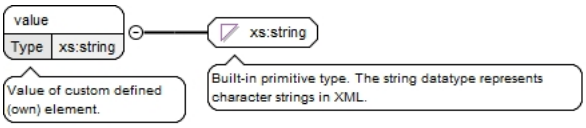
Type	ownElementType
Properties	content: complex minOccurs: 0 maxOccurs: unbounded
Model	value{0,1} , property{0,1} , ownElement*
Children	ownElement, property, value

Instance	`<ownElement dataType="" format="" name="" xmlns="employment">`
	`  <value>{0,1}</value>`
	`  <property>{0,1}</property>`
	`  <ownElement dataType="" format="" name="">{0,unbounded}</ownElement>`
	`</ownElement>`

Attributes	**QName** / **Type** / **Use** **dataType** / xs:string / optional Datatype for own defined element element. **format** / xs:string / optional Format for own defined element. **name** / xs:string / required Name of own defined element.

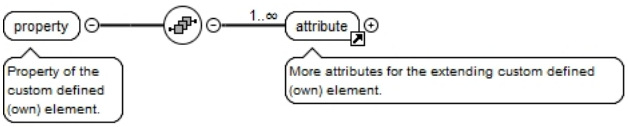
Source	`<xs:element name="ownElement" type="ownElementType" minOccurs="0" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Simple way of adding a small number of elements extending the use of the schema.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element value

Namespace	employment
Annotations	Value of custom defined (own) element.
Diagram	
Type	xs:string
Properties	content: simple
Used by	Element: attribute Complex Type: ownElementType

Source	`<xs:element name="value" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Value of custom defined (own) element.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

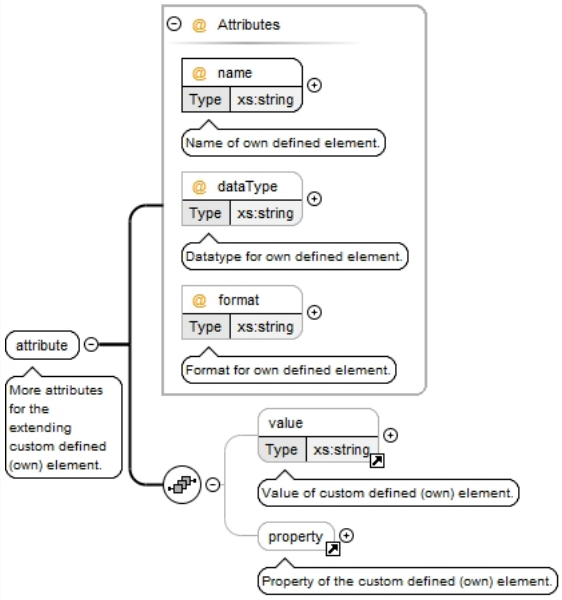
Element property

Namespace	employment
Annotations	Property of the custom defined (own) element.
Diagram	
Properties	content: complex
Used by	Element: attribute Complex Type: ownElementType
Model	attribute+
Children	attribute

Instance	`<property xmlns="employment">`
	`  <attribute dataType="" format="" name="">{1,unbounded}</attribute>`
	`</property>`

Source	`<xs:element name="property">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Property of the custom defined (own) element.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:complexType>`
	`  <xs:sequence>`
	`  <xs:element ref="attribute" maxOccurs="unbounded"/>`
	`  </xs:sequence>`
	`  </xs:complexType>`
	`</xs:element>`

Element `attribute`

Namespace	employment
Annotations	More attributes for the extending custom defined (own) element.
Diagram	
Properties	content: complex
Used by	Element property
Model	value{0,1} , property{0,1}
Children	property, value

Instance	`<attribute dataType="" format="" name="" xmlns="employment">`
	`  <value>{0,1}</value>`
	`  <property>{0,1}</property>`
	`</attribute>`

Attributes

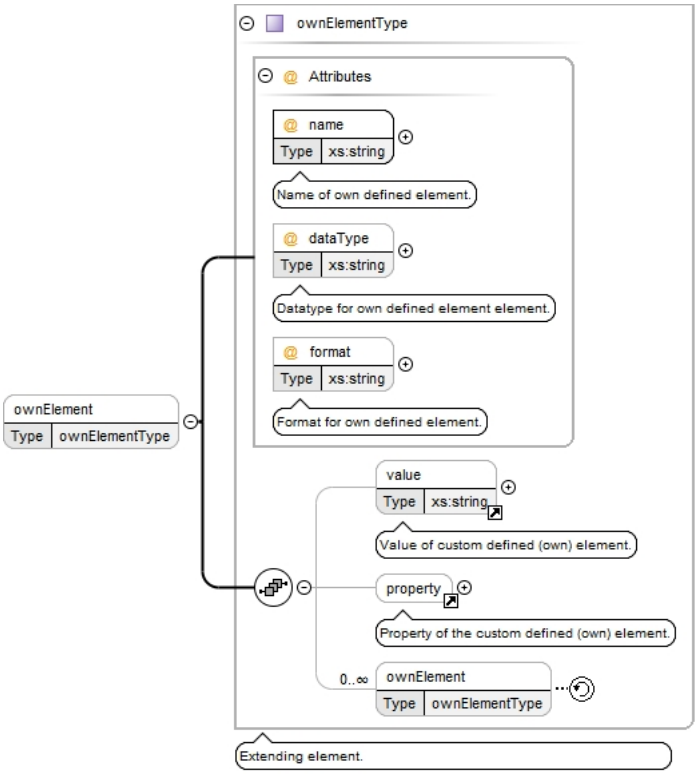
QName	Type	Use	
dataType	xs:string	optional	
	Datatype for own defined element.		
format	xs:string	optional	
	Format for own defined element.		
name	xs:string	required	
	Name of own defined element.		

Source

```
<xs:element name="attribute">
  <xs:annotation>
    <xs:documentation xml:lang="sv">More attributes for the extending custom defined (own)
 element.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="value" minOccurs="0"/>
      <xs:element ref="property" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required">
```

```
      <xs:annotation>
        <xs:documentation xml:lang="en">Name of own defined element.</xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="dataType" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">Datatype for own defined element.</xs:documentation>
      </xs:annotation>
    </xs:attribute>
    <xs:attribute name="format" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">Format for own defined element.</xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
</xs:element>
```

Element `ownElementType / ownElement`

Namespace	employment
Diagram	
Type	ownElementType
Properties	content: `complex` minOccurs: `0` maxOccurs: `unbounded`
Model	value{0,1} , property{0,1} , ownElement*
Children	ownElement, property, value

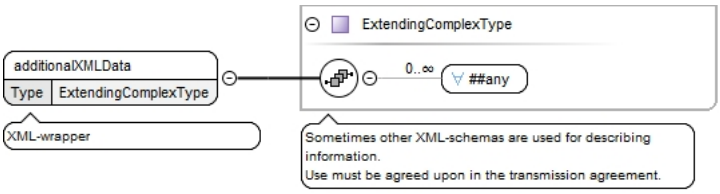
Instance	`<ownElement dataType="" format="" name="" xmlns="employment">`
	`  <value>{0,1}</value>`
	`  <property>{0,1}</property>`
	`  <ownElement dataType="" format="" name="">{0,unbounded}</ownElement>`
	`</ownElement>`

Attributes	

QName	Type	Use	
dataType	xs:string	optional	
	Datatype for own defined element element.		
format	xs:string	optional	
	Format for own defined element.		
name	xs:string	required	

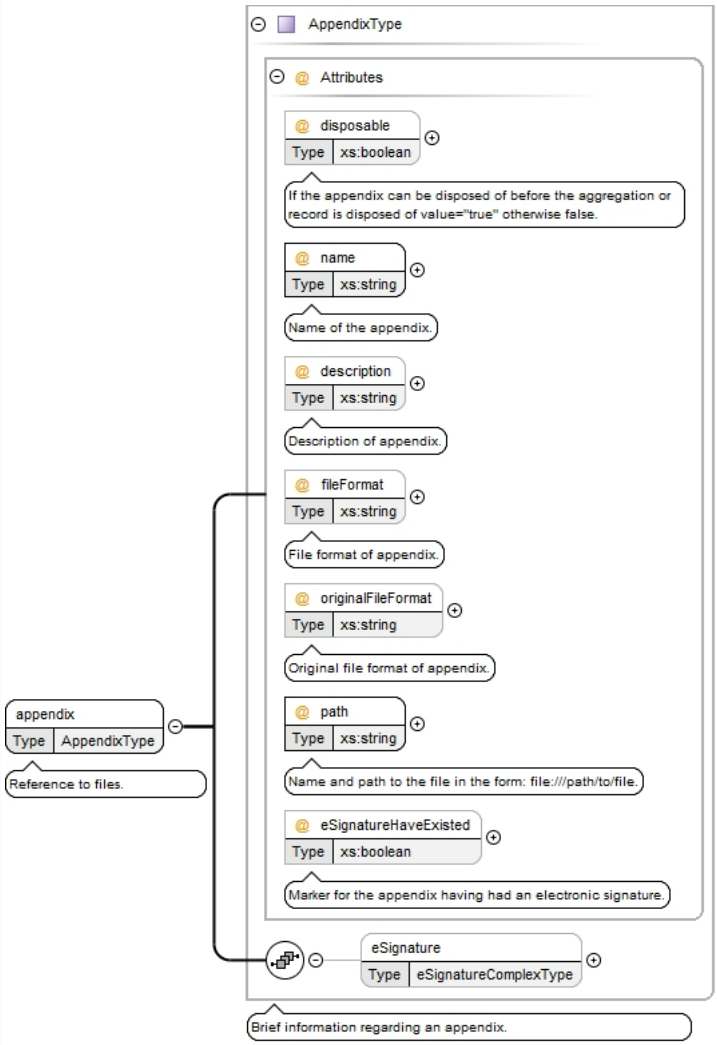
	QName	Type	Use	
		Name of own defined element.		
Source	<xs:element name="ownElement" type="ownElementType" minOccurs="0" maxOccurs="unbounded"/>			

Element additionalXMLData

Namespace	employment
Annotations	XML-wrapper
Diagram	
Type	ExtendingComplexType
Properties	content: complex
Used by	Complex Type additionalElementsType
Model	ANY element from ANY namespace
Source	<xs:element name="additionalXMLData" type="ExtendingComplexType"> <xs:annotation> <xs:documentation xml:lang="en">XML-wrapper</xs:documentation> </xs:annotation> </xs:element>

Element appendix

Namespace	employment
Annotations	Reference to files.

Diagram	
Type	AppendixType
Properties	content: complex
Used by	Complex Type additionalElementsType
Model	eSignature{0,1}
Children	eSignature
Instance	`<appendix description="" disposable="" eSignatureHaveExisted="" fileFormat="" name="" originalFileFormat="" path="" >   <eSignature dateESignatureIsVerified="" present="">{0,1}</eSignature> </appendix>`

Attributes

QName	Type	Use	
description	xs:string	optional	
	Description of appendix.		
disposable	xs:boolean	optional	
	If the appendix can be disposed of before the aggregation or record is disposed of value="true" otherwise false.		
eSignatureHaveExisted	xs:boolean	optional	
	Marker for the appendix having had an electronic signature.		
fileFormat	xs:string	optional	
	File format of appendix.		
name	xs:string	required	
	Name of the appendix.		
originalFileFormat	xs:string	optional	
	Original file format of appendix.		

	QName	Type	Use	
	path	xs:string	required	
		Name and path to the file in the form: file:///path/to/file.		

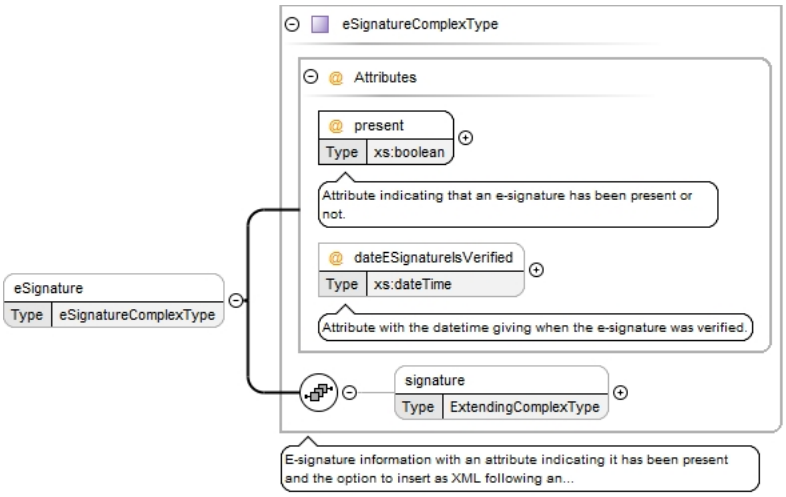
Source

```
<xs:element name="appendix" type="AppendixType">
  <xs:annotation>
    <xs:documentation xml:lang="en">Reference to files.</xs:documentation>
  </xs:annotation>
</xs:element>
```

Element AppendixType / eSignature

Namespace: employment

Diagram



Type: eSignatureComplexType

Properties

content:	complex
minOccurs:	0

Model: signature{0,1}

Children: signature

Instance

```
<eSignature dateESignatureIsVerified="" present="" xmlns="employment">
  <signature>{0,1}</signature>
</eSignature>
```

Attributes

QName	Type	Use	
dateESignatureIsVerified	xs:dateTime	optional	
	Attribute with the datetime giving when the e-signature was verified.		
present	xs:boolean	required	
	Attribute indicating that an e-signature has been present or not.		

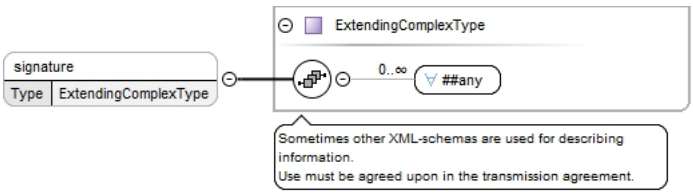
Source

```
<xs:element name="eSignature" type="eSignatureComplexType" minOccurs="0"/>
```

Element eSignatureComplexType / signature

Namespace: employment

Diagram



Type: ExtendingComplexType

Properties

content:	complex
minOccurs:	0

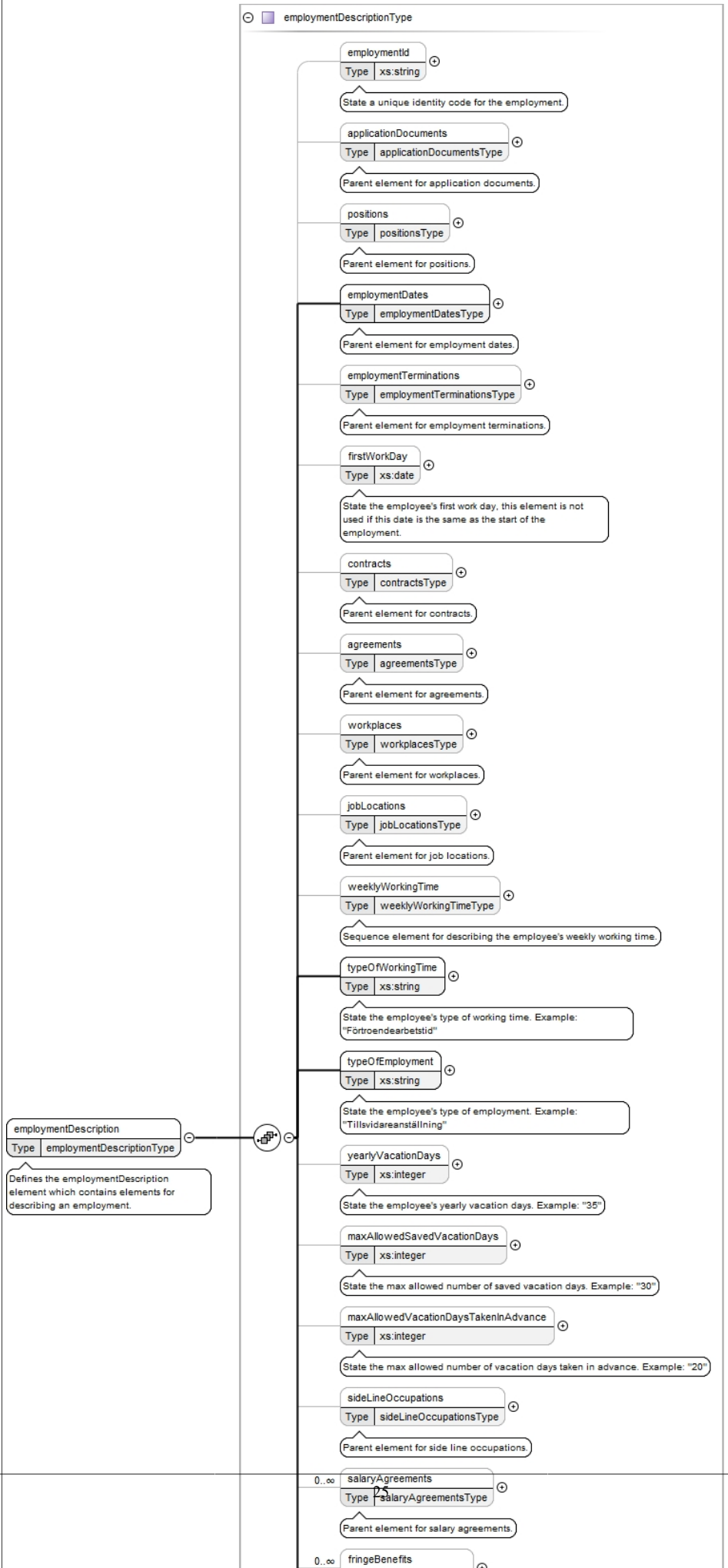
Model: ANY element from ANY namespace

Source	`<xs:element name="signature" type="ExtendingComplexType" minOccurs="0"/>`

Element `employmentType / employmentDescription`

Namespace	employment
Annotations	`Defines the employmentDescription element which contains elements for describing an employment.`

Diagram

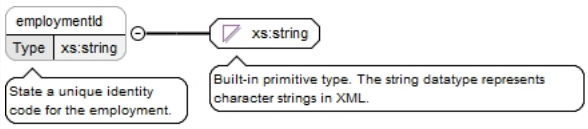

employmentDescriptionType
employmentId
Type xs:string
State a unique identity code for the employment.
applicationDocuments
Type applicationDocumentsType
Parent element for application documents.
positions
Type positionsType
Parent element for positions.
employmentDates
Type employmentDatesType
Parent element for employment dates.
employmentTerminations
Type employmentTerminationsType
Parent element for employment terminations.
firstWorkDay
Type xs:date
State the employee's first work day, this element is not used if this date is the same as the start of the employment.
contracts
Type contractsType
Parent element for contracts.
agreements
Type agreementsType
Parent element for agreements.
workplaces
Type workplacesType
Parent element for workplaces.
jobLocations
Type jobLocationsType
Parent element for job locations.
weeklyWorkingTime
Type weeklyWorkingTimeType
Sequence element for describing the employee's weekly working time.
typeOfWorkingTime
Type xs:string
State the employee's type of working time. Example: "Förtroendearbetstid"
typeOfEmployment
Type xs:string
State the employee's type of employment. Example: "Tillsvidareanställning"
yearlyVacationDays
Type xs:integer
State the employee's yearly vacation days. Example: "35"
maxAllowedSavedVacationDays
Type xs:integer
State the max allowed number of saved vacation days. Example: "30"
maxAllowedVacationDaysTakenInAdvance
Type xs:integer
State the max allowed number of vacation days taken in advance. Example: "20"
sideLineOccupations
Type sideLineOccupationsType
Parent element for side line occupations.
0..∞
salaryAgreements
Type salaryAgreementsType
Parent element for salary agreements.
0..∞
fringeBenefits
employmentDescription
Type employmentDescriptionType
Defines the employmentDescription element which contains elements for describing an employment.

Type	employmentDescriptionType
Properties	content: complex
Model	employmentId{0,1} , applicationDocuments{0,1} , positions{0,1} , employmentDates , employmentTerminations{0,1} , firstWorkDay{0,1} , contracts{0,1} , agreements{0,1} , workplaces{0,1} , jobLocations{0,1} , weeklyWorkingTime{0,1} , typeOfWorkingTime , typeOfEmployment , yearlyVacationDays{0,1} , maxAllowedSavedVacationDays{0,1} , maxAllowedVacationDaysTakenInAdvance{0,1} , sideLineOccupations{0,1} , salaryAgreements* , fringeBenefits* , paydays{0,1} , retirementRegulations{0,1} , retirementAge{0,1} , otherTerms* , testimonials{0,1} , references{0,1} , additionalElements{0,1} , relations
Children	additionalElements, agreements, applicationDocuments, contracts, employmentDates, employmentId, employmentTerminations, firstWorkDay, fringeBenefits, jobLocations, maxAllowedSavedVacationDays, maxAllowedVacationDaysTakenInAdvance, otherTerms, paydays, positions, references, relations, retirementAge, retirementRegulations, salaryAgreements, sideLineOccupations, testimonials, typeOfEmployment, typeOfWorkingTime, weeklyWorkingTime, workplaces, yearlyVacationDays

Instance	`<employmentDescription xmlns="employment">`
	`  <employmentId>{0,1}</employmentId>`
	`  <applicationDocuments>{0,1}</applicationDocuments>`
	`  <positions>{0,1}</positions>`
	`  <employmentDates>{1,1}</employmentDates>`
	`  <employmentTerminations>{0,1}</employmentTerminations>`
	`  <firstWorkDay>{0,1}</firstWorkDay>`
	`  <contracts>{0,1}</contracts>`
	`  <agreements>{0,1}</agreements>`
	`  <workplaces>{0,1}</workplaces>`
	`  <jobLocations>{0,1}</jobLocations>`
	`  <weeklyWorkingTime numberOfUnits="" unit="">{0,1}</weeklyWorkingTime>`
	`  <typeOfWorkingTime>{1,1}</typeOfWorkingTime>`
	`  <typeOfEmployment>{1,1}</typeOfEmployment>`
	`  <yearlyVacationDays>{0,1}</yearlyVacationDays>`
	`  <maxAllowedSavedVacationDays>{0,1}</maxAllowedSavedVacationDays>`
	`  <maxAllowedVacationDaysTakenInAdvance>{0,1}</maxAllowedVacationDaysTakenInAdvance>`
	`  <sideLineOccupations>{0,1}</sideLineOccupations>`
	`  <salaryAgreements>{0,unbounded}</salaryAgreements>`
	`  <fringeBenefits>{0,unbounded}</fringeBenefits>`
	`  <paydays>{0,1}</paydays>`
	`  <retirementRegulations>{0,1}</retirementRegulations>`
	`  <retirementAge>{0,1}</retirementAge>`
	`  <otherTerms>{0,unbounded}</otherTerms>`
	`  <testimonials>{0,1}</testimonials>`
	`  <references>{0,1}</references>`
	`  <additionalElements>{0,1}</additionalElements>`
	`  <relations>{1,1}</relations>`
	`</employmentDescription>`

Source	`<xs:element name="employmentDescription" type="employmentDescriptionType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Defines the employmentDescription element which contains elements for describing an employment.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

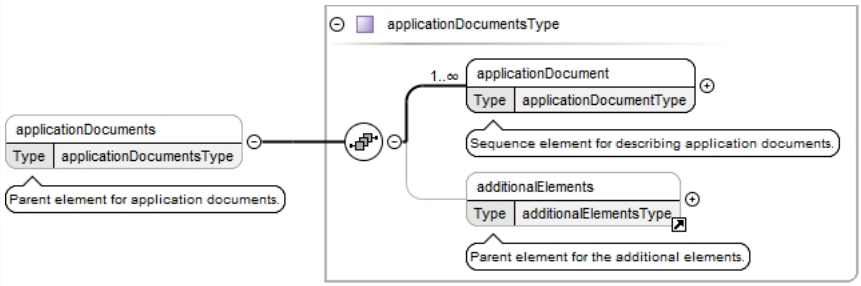
Element employmentDescriptionType / employmentId

Namespace	employment
Annotations	State a unique identity code for the employment.
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 0

Source	`<xs:element name="employmentId" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State a unique identity code for the employment.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / applicationDocuments

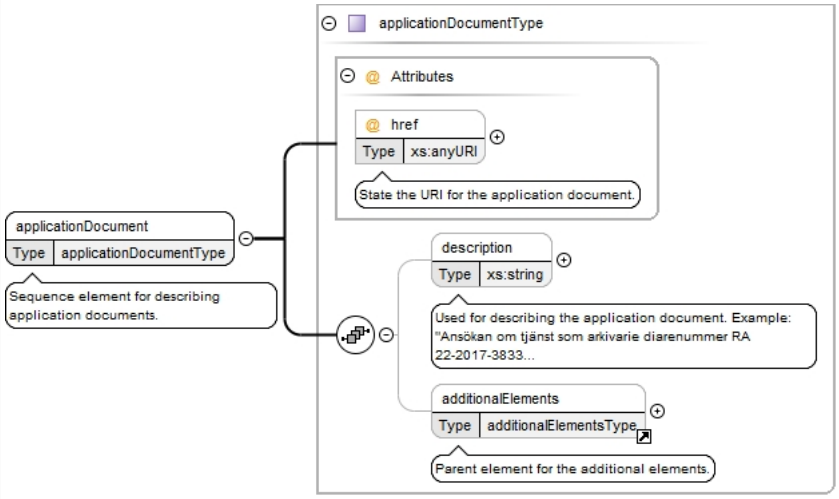
Namespace	employment

Annotations	Parent element for application documents.
Diagram	
Type	applicationDocumentsType
Properties	content: complex minOccurs: 0
Model	applicationDocument+ , additionalElements{0,1}
Children	additionalElements, applicationDocument

Instance	`<applicationDocuments xmlns="employment">`
	`  <applicationDocument href="">{1,unbounded}</applicationDocument>`
	`  <additionalElements>{0,1}</additionalElements>`
	`</applicationDocuments>`

Source	`<xs:element name="applicationDocuments" type="applicationDocumentsType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for application documents.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element applicationDocumentsType / applicationDocument

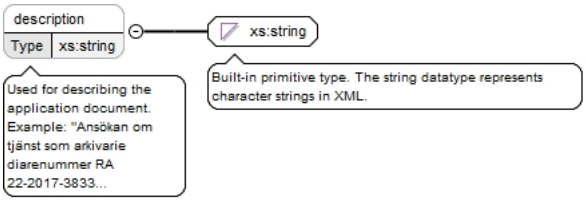
Namespace	employment
Annotations	Sequence element for describing application documents.
Diagram	
Type	applicationDocumentType
Properties	content: complex maxOccurs: unbounded
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description

Instance	`<applicationDocument href="" xmlns="employment">`
	`  <description>{0,1}</description>`
	`  <additionalElements>{0,1}</additionalElements>`
	`</applicationDocument>`

Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional State the URI for the application document.

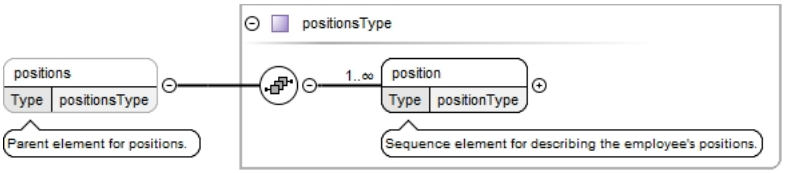
Source	`<xs:element name="applicationDocument" type="applicationDocumentType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing application documents.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element applicationDocumentType / description

Namespace	employment
Annotations	Used for describing the application document. Example: "Ansökan om tjänst som arkivarie diarenummer RA 22-2017-3833 Anna Andersson"
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 0

Source	`<xs:element name="description" type="xs:string" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for describing the application document. Example: "Ansökan om tjänst som arkivarie diarenummer RA 22-2017-3833 Anna Andersson"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / positions

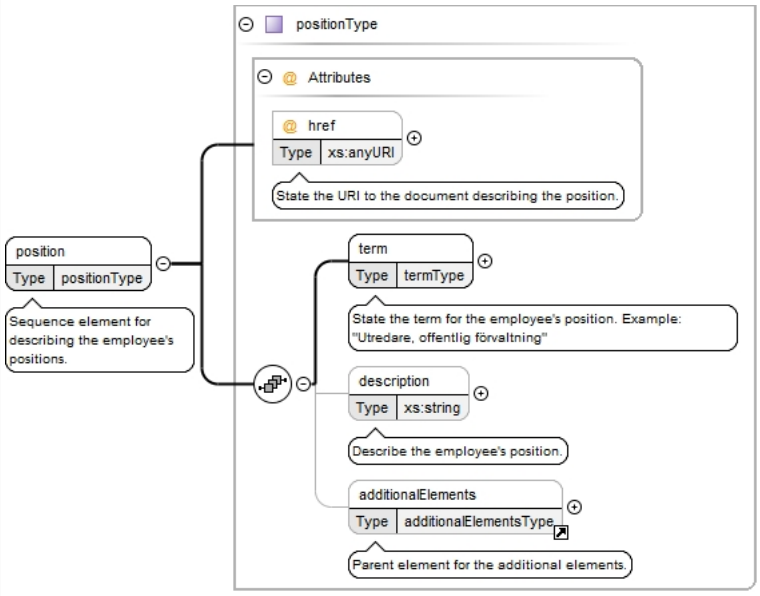
Namespace	employment
Annotations	Parent element for positions.
Diagram	
Type	positionsType
Properties	content: complex minOccurs: 0
Model	position+
Children	position

Instance	`<positions xmlns="employment">`
	`  <position href="">{1,unbounded}</position>`
	`</positions>`

Source	`<xs:element name="positions" type="positionsType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for positions.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element positionsType / position

Namespace	employment
Annotations	Sequence element for describing the employee's positions.

Diagram	
Type	positionType
Properties	content: complex maxOccurs: unbounded
Model	term , description{0,1} , additionalElements{0,1}
Children	additionalElements, description, term

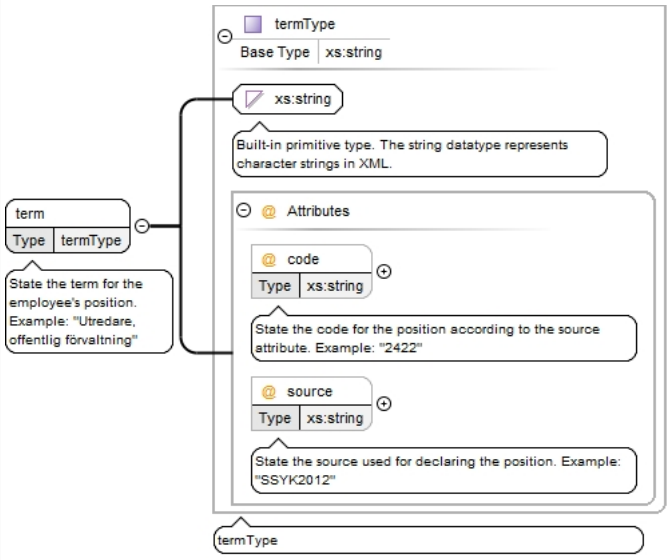
Instance	`<position href="" xmlns="employment">`
	`  <term code="" source="">{1,1}</term>`
	`  <description>{0,1}</description>`
	`  <additionalElements>{0,1}</additionalElements>`
	`</position>`

Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional State the URI to the document describing the position.

Source	`<xs:element name="position" type="positionType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing the employee's positions.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element positionType / term

Namespace	employment
Annotations	State the term for the employee's position. Example: "Utredare, offentlig förvaltning"

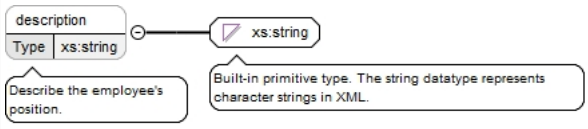
Diagram	
Type	termType
Properties	content: `complex`
Attributes	

QName	Type	Use	
code	xs:string	optional	
	State the code for the position according to the source attribute. Example: "2422"		
source	xs:string	optional	
	State the source used for declaring the position. Example: "SSYK2012"		

Source

```
<xs:element name="term" type="termType">
  <xs:annotation>
    <xs:documentation xml:lang="en">State the term for the employee's position. Example: "Utredare,
 offentlig förvaltning"</xs:documentation>
  </xs:annotation>
</xs:element>
```

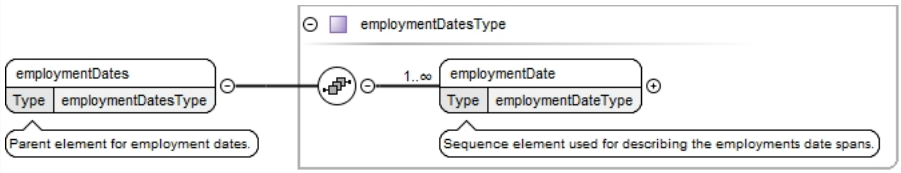
Element `positionType / description`

Namespace	employment
Annotations	Describe the employee's position.
Diagram	
Type	xs:string
Properties	content: `simple` minOccurs: `0`
Source	

```
<xs:element name="description" minOccurs="0" type="xs:string">
  <xs:annotation>
    <xs:documentation xml:lang="en">Describe the employee's position.</xs:documentation>
  </xs:annotation>
</xs:element>
```

Element `employmentDescriptionType / employmentDates`

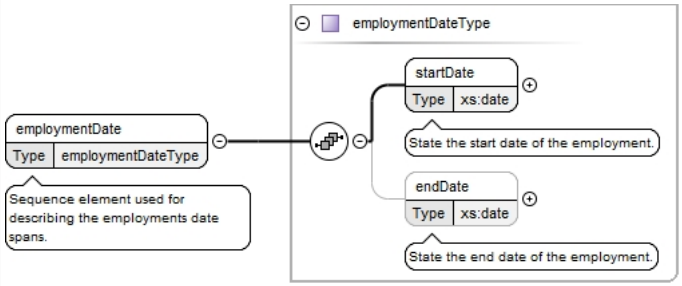
Namespace	employment
Annotations	Parent element for employment dates.

Diagram	
Type	employmentDatesType
Properties	content: complex
Model	employmentDate+
Children	employmentDate

Instance	`<employmentDates xmlns="employment">`
	`  <employmentDate>{1,unbounded}</employmentDate>`
	`</employmentDates>`

Source	`<xs:element name="employmentDates" type="employmentDatesType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for employment dates.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

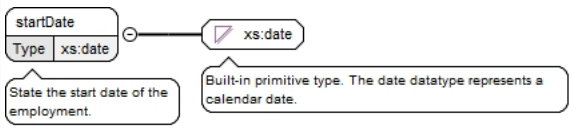
Element employmentDatesType / employmentDate

Namespace	employment
Annotations	Sequence element used for describing the employments date spans.
Diagram	
Type	employmentDateType
Properties	content: complex maxOccurs: unbounded
Model	startDate , endDate{0,1}
Children	endDate, startDate

Instance	`<employmentDate xmlns="employment">`
	`  <startDate>{1,1}</startDate>`
	`  <endDate>{0,1}</endDate>`
	`</employmentDate>`

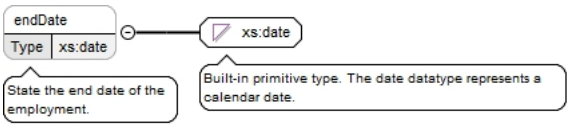
Source	`<xs:element name="employmentDate" type="employmentDateType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element used for describing the employments date`
	` spans.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDateType / startDate

Namespace	employment
Annotations	State the start date of the employment.
Diagram	
Type	xs:date
Properties	content: simple

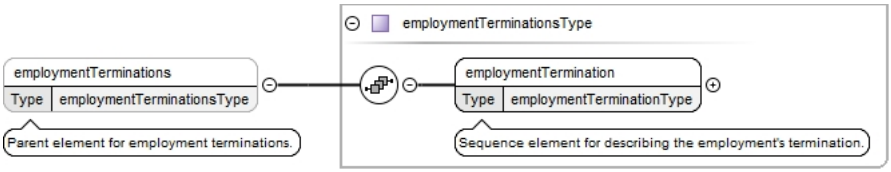
Source	`<xs:element name="startDate" type="xs:date">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the start date of the employment.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `employmentDateType / endDate`

Namespace	employment
Annotations	`State the end date of the employment.`
Diagram	
Type	xs:date
Properties	content: `simple` minOccurs: `0`

Source	`<xs:element name="endDate" minOccurs="0" type="xs:date">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the end date of the employment.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `employmentDescriptionType / employmentTerminations`

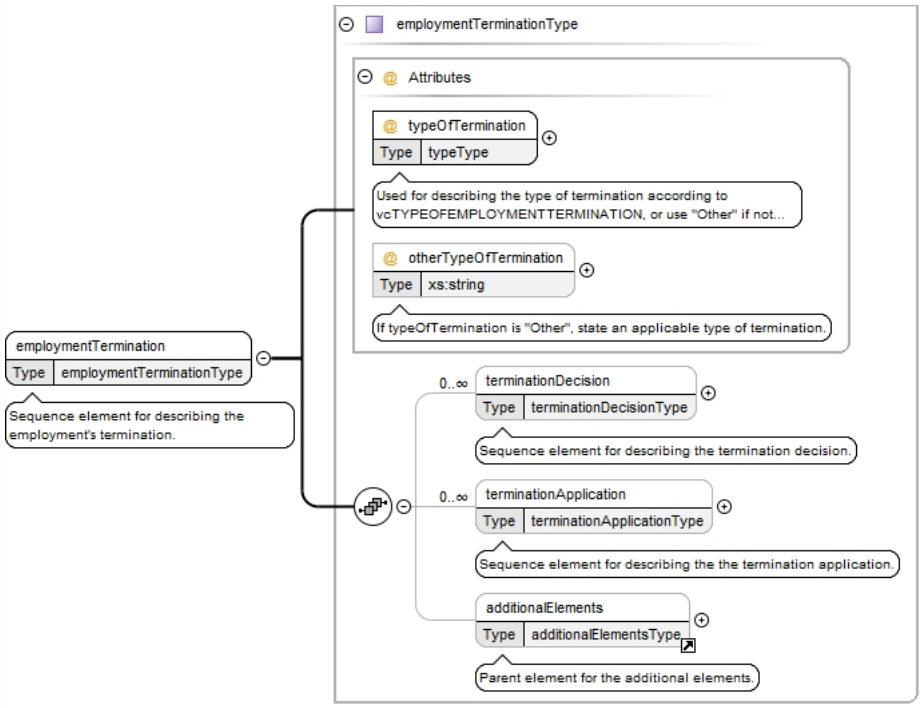
Namespace	employment
Annotations	`Parent element for employment terminations.`
Diagram	
Type	employmentTerminationsType
Properties	content: `complex` minOccurs: `0`
Model	employmentTermination
Children	employmentTermination

Instance	`<employmentTerminations xmlns="employment">`
	`  <employmentTermination otherTypeOfTermination="" typeOfTermination="">{1,1}</`
	`employmentTermination>`
	`</employmentTerminations>`

Source	`<xs:element name="employmentTerminations" type="employmentTerminationsType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for employment terminations.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `employmentTerminationsType / employmentTermination`

Namespace	employment
Annotations	`Sequence element for describing the employment's termination.`

Diagram	
Type	employmentTerminationType
Properties	content: complex
Model	terminationDecision* , terminationApplication* , additionalElements{0,1}
Children	additionalElements, terminationApplication, terminationDecision

Instance	`<employmentTermination otherTypeOfTermination="" typeOfTermination="" xmlns="employment">`
	`  <terminationDecision href="">{0,unbounded}</terminationDecision>`
	`  <terminationApplication href="">{0,unbounded}</terminationApplication>`
	`  <additionalElements>{0,1}</additionalElements>`
	`</employmentTermination>`

Attributes

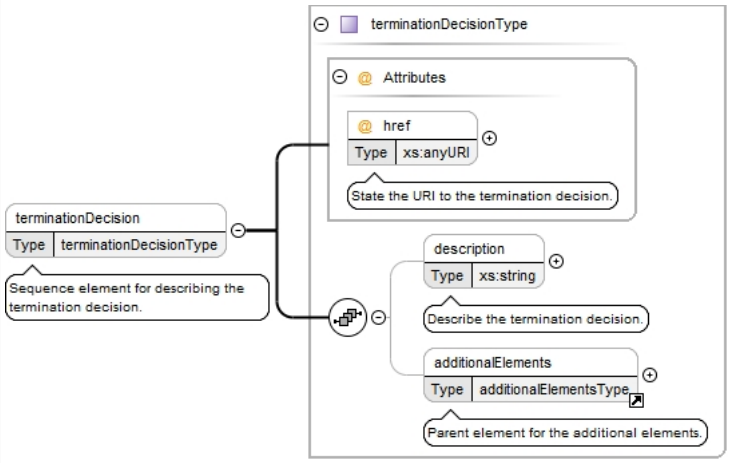
QName	Type	Use	
otherTypeOfTermination	xs:string	optional	
	If typeOfTermination is "Other", state an applicable type of termination.		
typeOfTermination	typeType	required	
	Used for describing the type of termination according to vcTYPEOFEMPLOYMENTTERMINATION, or use "Other" if not applicable. Example: "LackOfWork"		

Source

```
<xs:element name="employmentTermination" type="employmentTerminationType">
  <xs:annotation>
    <xs:documentation xml:lang="en">Sequence element for describing the employment's termination.</
xs:documentation>
  </xs:annotation>
</xs:element>
```

Element employmentTerminationType / terminationDecision

Namespace	employment
Annotations	Sequence element for describing the termination decision.

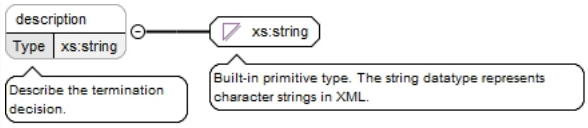
Diagram	
Type	terminationDecisionType
Properties	content: complex minOccurs: 0 maxOccurs: unbounded
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description

Instance	`<terminationDecision href="" xmlns="employment">`
	`  <description>{0,1}</description>`
	`  <additionalElements>{0,1}</additionalElements>`
	`</terminationDecision>`

Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional State the URI to the termination decision.

Source	`<xs:element name="terminationDecision" minOccurs="0" maxOccurs="unbounded"`
	` type="terminationDecisionType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing the termination decision.</`
	`xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

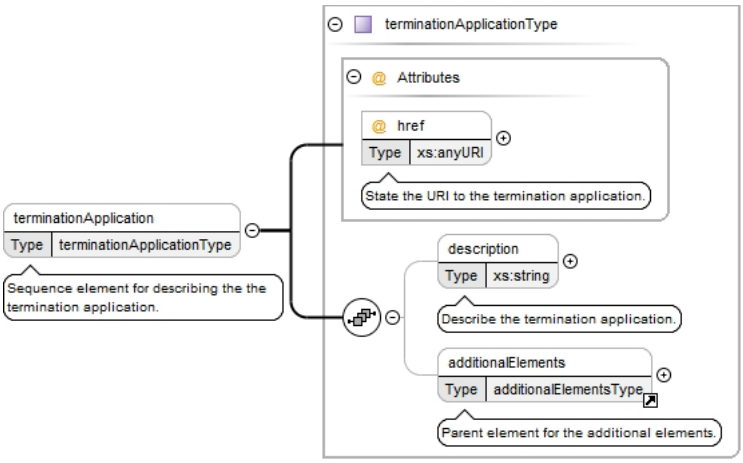
Element terminationDecisionType / description

Namespace	employment
Annotations	Describe the termination decision.
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 0

Source	`<xs:element name="description" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the termination decision.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentTerminationType / terminationApplication

Namespace	employment
Annotations	Sequence element for describing the the termination application.

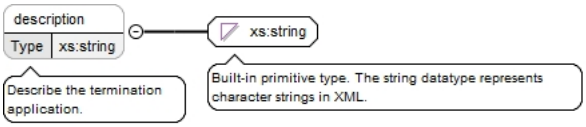
Diagram	
Type	terminationApplicationType
Properties	content: `complex` minOccurs: `0` maxOccurs: `unbounded`
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description

Instance	`<terminationApplication href="" xmlns="employment">`
	`  <description>{0,1}</description>`
	`  <additionalElements>{0,1}</additionalElements>`
	`</terminationApplication>`

Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional `State the URI to the termination application.`

Source	`<xs:element name="terminationApplication" type="terminationApplicationType" minOccurs="0" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing the the termination application.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

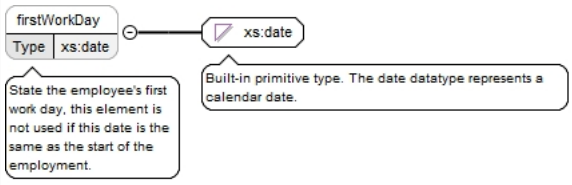
Element `terminationApplicationType / description`

Namespace	employment
Annotations	`Describe the termination application.`
Diagram	
Type	xs:string
Properties	content: `simple` minOccurs: `0`

Source	`<xs:element name="description" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the termination application.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

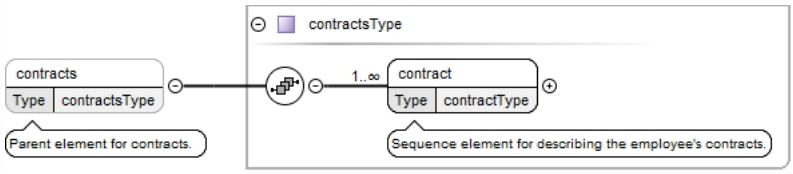
Element `employmentDescriptionType / firstWorkDay`

Namespace	employment
Annotations	`State the employee's first work day, this element is not used if this date is the same as the start of the employment.`

Diagram	
Type	xs:date
Properties	content: simple minOccurs: 0

Source	`<xs:element name="firstWorkDay" minOccurs="0" type="xs:date">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the employee's first work day, this element is not used if this date is the same as the start of the employment.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

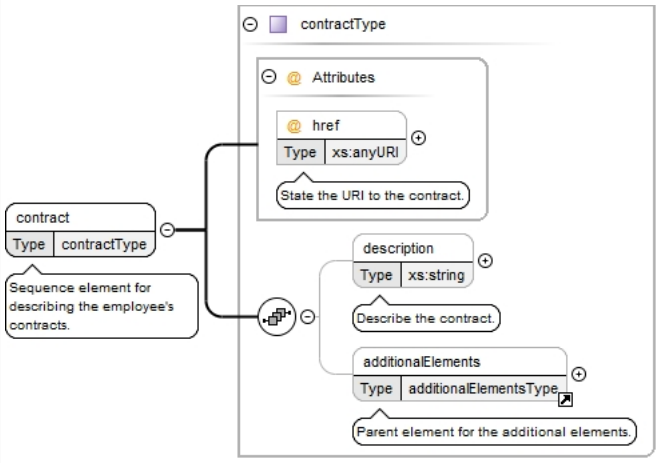
Element employmentDescriptionType / contracts

Namespace	employment
Annotations	Parent element for contracts.
Diagram	
Type	contractsType
Properties	content: complex minOccurs: 0
Model	contract+
Children	contract

Instance	`<contracts xmlns="employment">`
	`  <contract href="">{1,unbounded}</contract>`
	`</contracts>`

Source	`<xs:element name="contracts" type="contractsType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for contracts.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element contractsType / contract

Namespace	employment
Annotations	Sequence element for describing the employee's contracts.
Diagram	

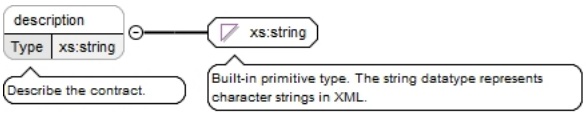
Type	contractType
Properties	content: complex maxOccurs: unbounded
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description

Instance	`<contract href="" xmlns="employment">`
	`  <description>{0,1}</description>`
	`  <additionalElements>{0,1}</additionalElements>`
	`</contract>`

Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional State the URI to the contract.

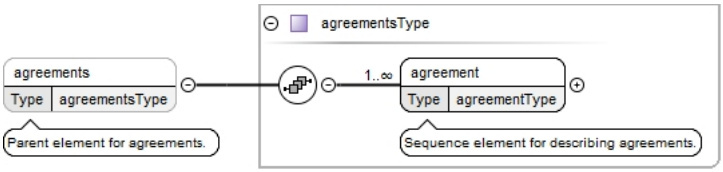
Source	`<xs:element name="contract" type="contractType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing the employee's contracts.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element contractType / description

Namespace	employment
Annotations	Describe the contract.
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 0

Source	`<xs:element name="description" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the contract.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

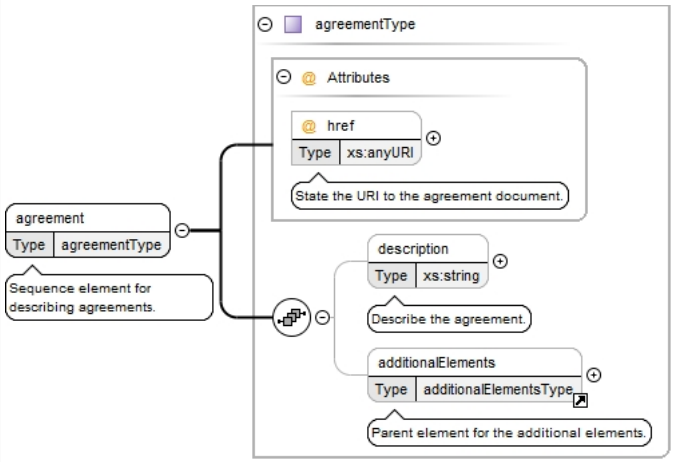
Element employmentDescriptionType / agreements

Namespace	employment
Annotations	Parent element for agreements.
Diagram	
Type	agreementsType
Properties	content: complex minOccurs: 0
Model	agreement+
Children	agreement

Instance	`<agreements xmlns="employment">`
	`  <agreement href="">{1,unbounded}</agreement>`
	`</agreements>`

Source	`<xs:element name="agreements" type="agreementsType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for agreements.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element agreementsType / agreement

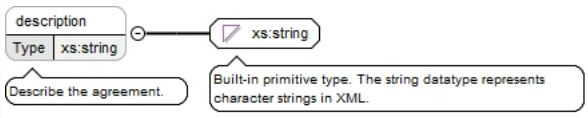
Namespace	employment
Annotations	Sequence element for describing agreements.
Diagram	
Type	agreementType
Properties	content: complex maxOccurs: unbounded
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description

Instance	`<agreement href="" xmlns="employment">`
	`  <description>{0,1}</description>`
	`  <additionalElements>{0,1}</additionalElements>`
	`</agreement>`

Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional State the URI to the agreement document.

Source	`<xs:element name="agreement" type="agreementType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing agreements.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

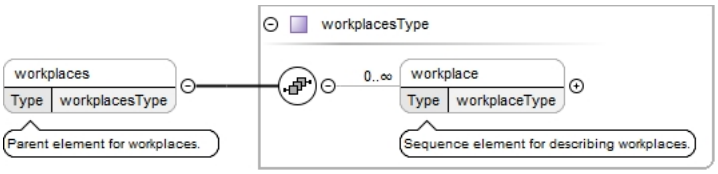
Element agreementType / description

Namespace	employment
Annotations	Describe the agreement.
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 0

Source	`<xs:element name="description" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the agreement.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / workplaces

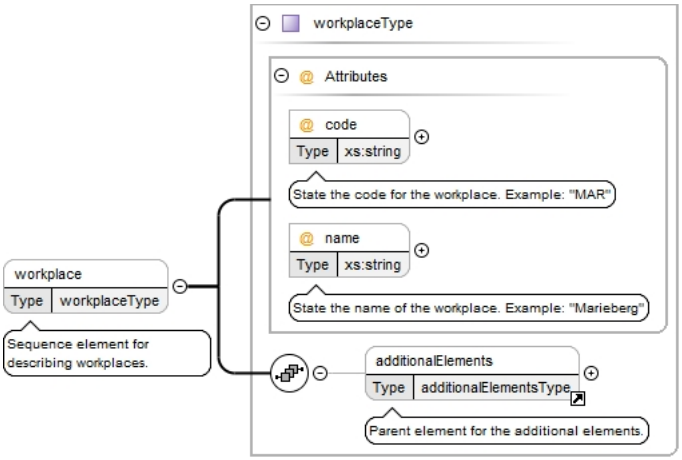
Namespace	employment

Annotations	Parent element for workplaces.
Diagram	
Type	workplacesType
Properties	content: complex minOccurs: 0
Model	workplace*
Children	workplace

Instance	`<workplaces xmlns="employment">`
	`  <workplace code="" name="">{0,unbounded}</workplace>`
	`</workplaces>`

Source	`<xs:element name="workplaces" type="workplacesType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for workplaces.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element workplacesType / workplace

Namespace	employment
Annotations	Sequence element for describing workplaces.
Diagram	
Type	workplaceType
Properties	content: complex minOccurs: 0 maxOccurs: unbounded
Model	additionalElements{0,1}
Children	additionalElements

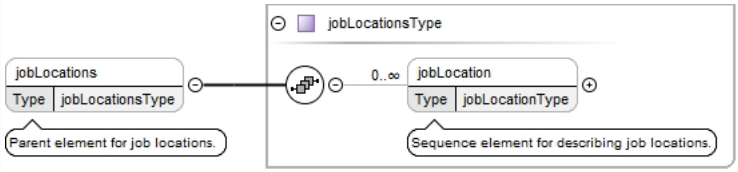
Instance	`<workplace code="" name="" xmlns="employment">`
	`  <additionalElements>{0,1}</additionalElements>`
	`</workplace>`

Attributes	**QName** / **Type** / **Use** **code** / xs:string / optional State the code for the workplace. Example: "MAR" **name** / xs:string / optional State the name of the workplace. Example: "Marieberg"

Source	`<xs:element name="workplace" type="workplaceType" minOccurs="0" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing workplaces.</xs:documentation>`
	`  </xs:annotation>`

```
</xs:element>
```

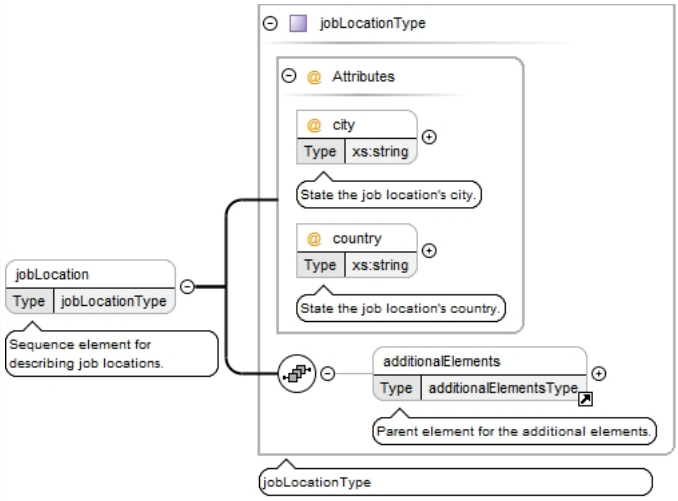
Element employmentDescriptionType / jobLocations

Namespace	employment
Annotations	Parent element for job locations.
Diagram	
Type	jobLocationsType
Properties	content: complex minOccurs: 0
Model	jobLocation*
Children	jobLocation

Instance	`<jobLocations xmlns="employment">`
	`  <jobLocation city="" country="">{0,unbounded}</jobLocation>`
	`</jobLocations>`

Source	`<xs:element name="jobLocations" type="jobLocationsType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for job locations.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element jobLocationsType / jobLocation

Namespace	employment
Annotations	Sequence element for describing job locations.
Diagram	
Type	jobLocationType
Properties	content: complex minOccurs: 0 maxOccurs: unbounded
Model	additionalElements{0,1}
Children	additionalElements

Instance	`<jobLocation city="" country="" xmlns="employment">`
	`  <additionalElements>{0,1}</additionalElements>`
	`</jobLocation>`

Attributes	

QName	Type	Use
city	xs:string	optional

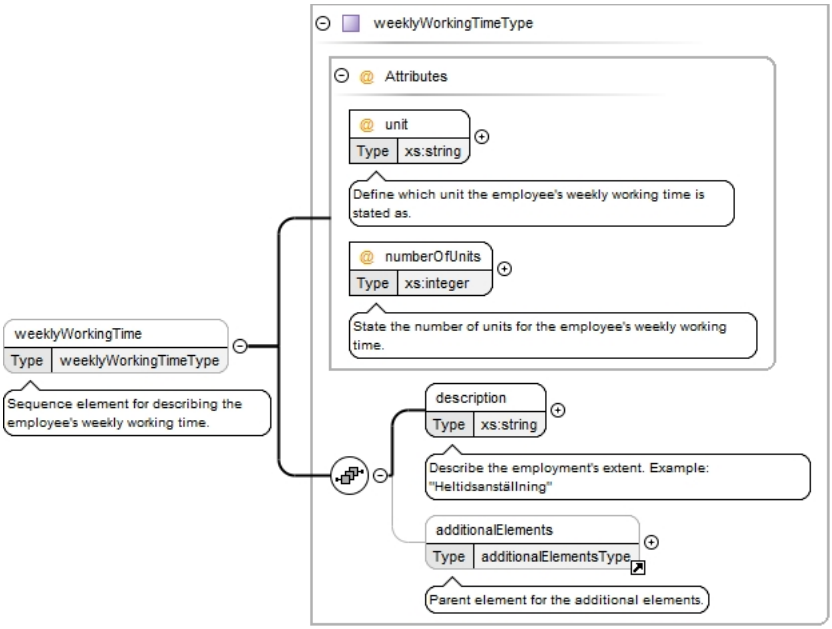
	QName	Type	Use	
		State the job location's city.		
	country	xs:string	optional	
		State the job location's country.		

Source
```
<xs:element name="jobLocation" type="jobLocationType" minOccurs="0" maxOccurs="unbounded">
  <xs:annotation>
    <xs:documentation>Sequence element for describing job locations.</xs:documentation>
  </xs:annotation>
</xs:element>
```

Element employmentDescriptionType / weeklyWorkingTime

Namespace	employment
Annotations	Sequence element for describing the employee's weekly working time.

Diagram



Type	weeklyWorkingTimeType
Properties	content: complex minOccurs: 0
Model	description , additionalElements{0,1}
Children	additionalElements, description

Instance
```
<weeklyWorkingTime numberOfUnits="" unit="" xmlns="employment">
  <description>{1,1}</description>
  <additionalElements>{0,1}</additionalElements>
</weeklyWorkingTime>
```

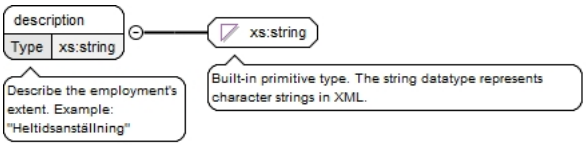
Attributes

QName	Type	Use	
numberOfUnits	xs:integer	required	
	State the number of units for the employee's weekly working time.		
unit	xs:string	required	
	Define which unit the employee's weekly working time is stated as.		

Source
```
<xs:element name="weeklyWorkingTime" type="weeklyWorkingTimeType" minOccurs="0">
  <xs:annotation>
    <xs:documentation xml:lang="en">Sequence element for describing the employee's weekly working
 time.</xs:documentation>
  </xs:annotation>
</xs:element>
```

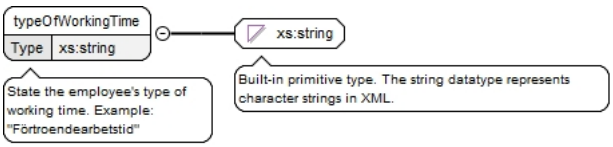
Element weeklyWorkingTimeType / description

Namespace	employment
Annotations	Describe the employment's extent. Example: "Heltidsanställning"

Diagram	
Type	xs:string
Properties	content: simple

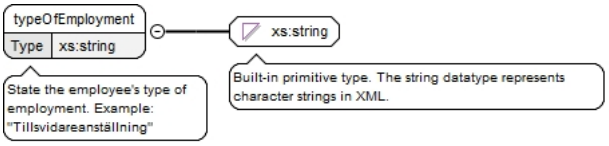
Source	`<xs:element name="description" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the employment's extent. Example: "Heltidsanställning"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / typeOfWorkingTime

Namespace	employment
Annotations	State the employee's type of working time. Example: "Förtroendearbetstid"
Diagram	
Type	xs:string
Properties	content: simple

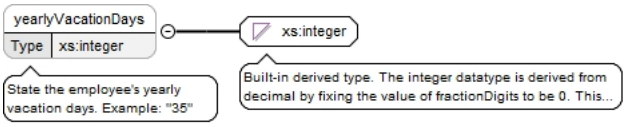
Source	`<xs:element name="typeOfWorkingTime" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the employee's type of working time. Example: "Förtroendearbetstid"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / typeOfEmployment

Namespace	employment
Annotations	State the employee's type of employment. Example: "Tillsvidareanställning"
Diagram	
Type	xs:string
Properties	content: simple

Source	`<xs:element name="typeOfEmployment" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the employee's type of employment. Example: "Tillsvidareanställning"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

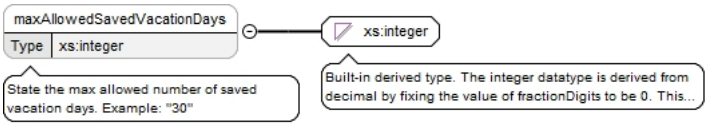
Element employmentDescriptionType / yearlyVacationDays

Namespace	employment
Annotations	State the employee's yearly vacation days. Example: "35"
Diagram	
Type	xs:integer

Properties	content: simple minOccurs: 0

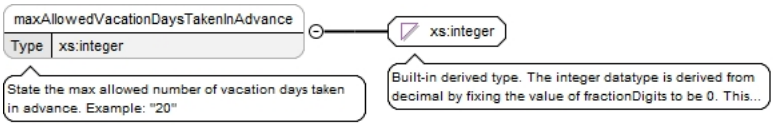
Source	`<xs:element name="yearlyVacationDays" minOccurs="0" type="xs:integer">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the employee's yearly vacation days. Example: "35"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / maxAllowedSavedVacationDays

Namespace	employment
Annotations	State the max allowed number of saved vacation days. Example: "30"
Diagram	
Type	xs:integer
Properties	content: simple minOccurs: 0

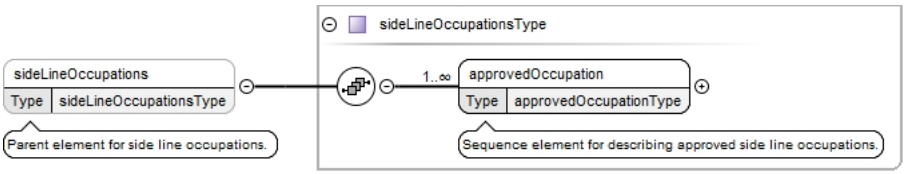
Source	`<xs:element name="maxAllowedSavedVacationDays" minOccurs="0" type="xs:integer">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the max allowed number of saved vacation days. Example: "30"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / maxAllowedVacationDaysTakenInAdvance

Namespace	employment
Annotations	State the max allowed number of vacation days taken in advance. Example: "20"
Diagram	
Type	xs:integer
Properties	content: simple minOccurs: 0

Source	`<xs:element name="maxAllowedVacationDaysTakenInAdvance" minOccurs="0" type="xs:integer">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the max allowed number of vacation days taken in advance. Example: "20"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / sideLineOccupations

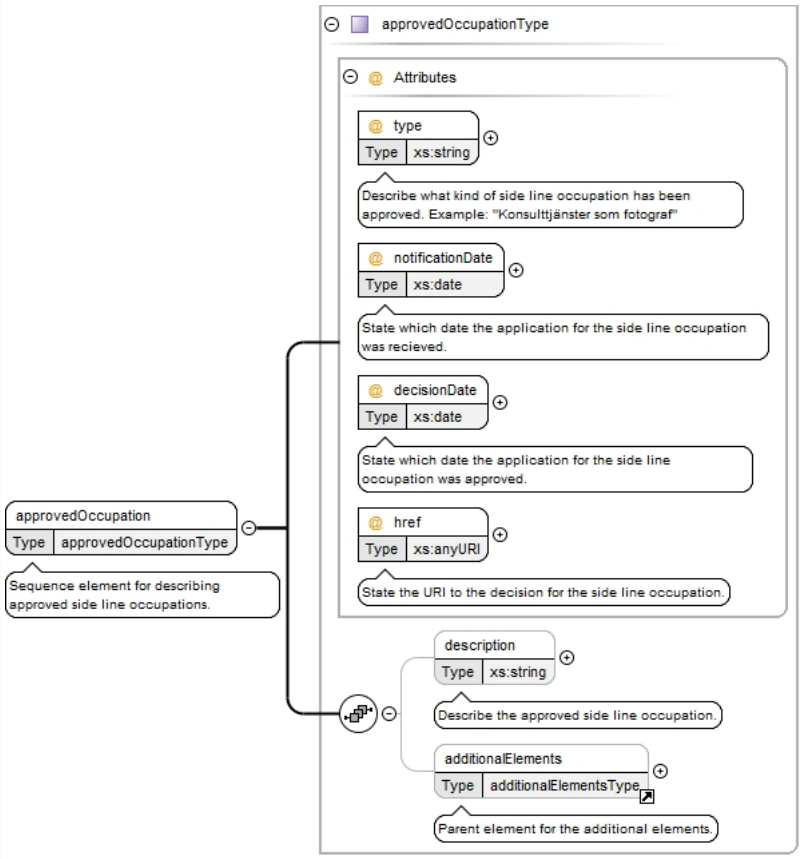
Namespace	employment
Annotations	Parent element for side line occupations.
Diagram	
Type	sideLineOccupationsType
Properties	content: complex minOccurs: 0

Model	approvedOccupation+
Children	approvedOccupation

Instance	`<sideLineOccupations xmlns="employment">`
	`  <approvedOccupation decisionDate="" href="" notificationDate="" type="">{1,unbounded}</approvedOccupation>`
	`</sideLineOccupations>`

Source	`<xs:element name="sideLineOccupations" type="sideLineOccupationsType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for side line occupations.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `sideLineOccupationsType / approvedOccupation`

Namespace	employment
Annotations	Sequence element for describing approved side line occupations.
Diagram	
Type	approvedOccupationType
Properties	content: complex maxOccurs: unbounded
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description

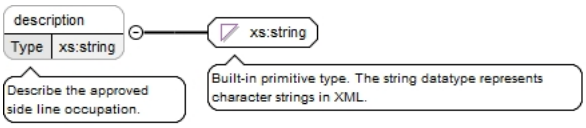
Instance	`<approvedOccupation decisionDate="" href="" notificationDate="" type="" xmlns="employment">`
	`  <description>{0,1}</description>`
	`  <additionalElements>{0,1}</additionalElements>`
	`</approvedOccupation>`

Attributes	

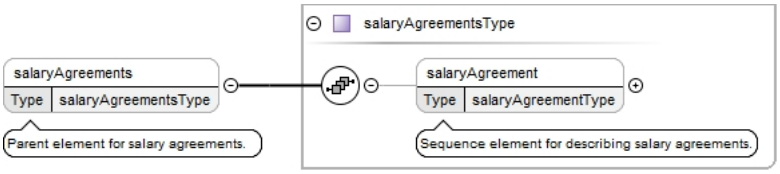
QName	Type	Use	
decisionDate	xs:date	required	
	State which date the application for the side line occupation was approved.		
href	xs:anyURI	required	
	State the URI to the decision for the side line occupation.		
notificationDate	xs:date	required	

	QName	Type	Use	
		State which date the application for the side line occupation was recieved.		
	type	xs:string	required	
		Describe what kind of side line occupation has been approved. Example: "Konsulttjänster som fotograf"		
Source	<xs:element name="approvedOccupation" type="approvedOccupationType" maxOccurs="unbounded"> <xs:annotation> <xs:documentation xml:lang="en">Sequence element for describing approved side line occupations.</xs:documentation> </xs:annotation> </xs:element>			

Element approvedOccupationType / description

Namespace	employment
Annotations	Describe the approved side line occupation.
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 0
Source	<xs:element name="description" minOccurs="0" type="xs:string"> <xs:annotation> <xs:documentation xml:lang="en">Describe the approved side line occupation.</xs:documentation> </xs:annotation> </xs:element>

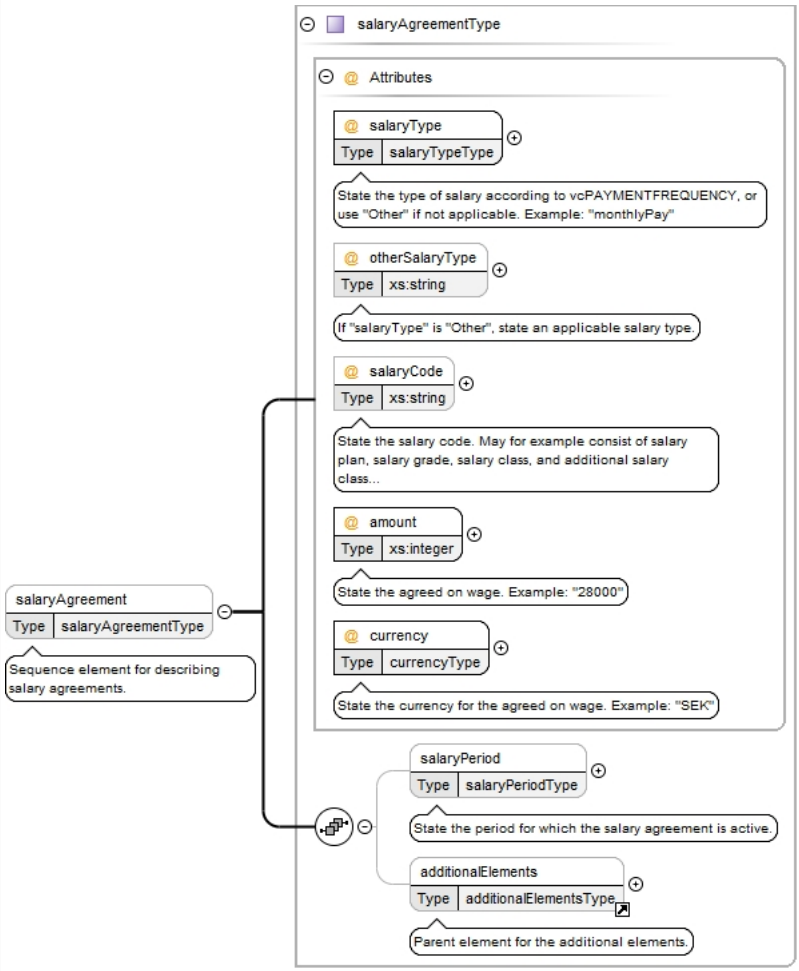
Element employmentDescriptionType / salaryAgreements

Namespace	employment
Annotations	Parent element for salary agreements.
Diagram	
Type	salaryAgreementsType
Properties	content: complex minOccurs: 0 maxOccurs: unbounded
Model	salaryAgreement{0,1}
Children	salaryAgreement
Instance	<salaryAgreements xmlns="employment"> <salaryAgreement amount="" currency="" otherSalaryType="" salaryCode="" salaryType="">{0,1}</salaryAgreement> </salaryAgreements>
Source	<xs:element name="salaryAgreements" type="salaryAgreementsType" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation xml:lang="en">Parent element for salary agreements.</xs:documentation> </xs:annotation> </xs:element>

Element salaryAgreementsType / salaryAgreement

Namespace	employment
Annotations	Sequence element for describing salary agreements.

Diagram



Type: salaryAgreementType

Properties

content:	complex
minOccurs:	0

Model: salaryPeriod{0,1} , additionalElements{0,1}

Children: additionalElements, salaryPeriod

Instance

```
<salaryAgreement amount="" currency="" otherSalaryType="" salaryCode="" salaryType="" xmlns="employment">
  <salaryPeriod>{0,1}</salaryPeriod>
  <additionalElements>{0,1}</additionalElements>
</salaryAgreement>
```

Attributes

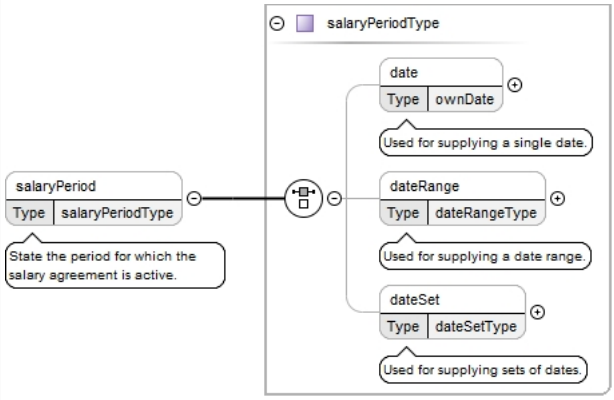
QName	Type	Use	
amount	xs:integer	required	
	State the agreed on wage. Example: "28000"		
currency	currencyType	required	
	State the currency for the agreed on wage. Example: "SEK"		
otherSalaryType	xs:string	optional	
	If "salaryType" is "Other", state an applicable salary type.		
salaryCode	xs:string	optional	
	State the salary code. May for example consist of salary plan, salary grade, salary class, and additional salary class separated by ":". Example: "KA"		
salaryType	salaryTypeType	required	
	State the type of salary according to vcPAYMENTFREQUENCY, or use "Other" if not applicable. Example: "monthlyPay"		

Source

```
<xs:element name="salaryAgreement" type="salaryAgreementType" minOccurs="0">
  <xs:annotation>
    <xs:documentation xml:lang="en">Sequence element for describing salary agreements.</
xs:documentation>
```

```
  </xs:annotation>
</xs:element>
```

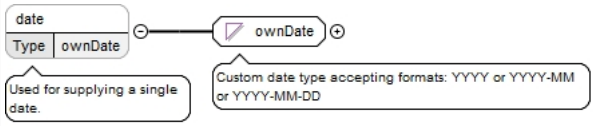
Element salaryAgreementType / salaryPeriod

Namespace	employment
Annotations	State the period for which the salary agreement is active.
Diagram	
Type	salaryPeriodType
Properties	content: complex minOccurs: 0
Model	date{0,1} | dateRange{0,1} | dateSet{0,1}
Children	date, dateRange, dateSet

Instance	`<salaryPeriod xmlns="employment">`
	`  <date>{0,1}</date>`
	`  <dateRange>{0,1}</dateRange>`
	`  <dateSet>{0,1}</dateSet>`
	`</salaryPeriod>`

Source	`<xs:element name="salaryPeriod" type="salaryPeriodType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the period for which the salary agreement is active.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

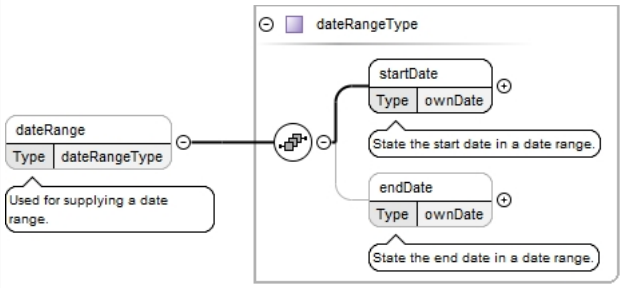
Element salaryPeriodType / date

Namespace	employment
Annotations	Used for supplying a single date.
Diagram	
Type	ownDate
Properties	content: simple minOccurs: 0
Facets	pattern: [\d]{4}\-[\d]{2}\-[\d]{2}|[\d]{4}\-[\d]{2}|[\d]{4}

Source	`<xs:element name="date" minOccurs="0" type="ownDate">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for supplying a single date.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element salaryPeriodType / dateRange

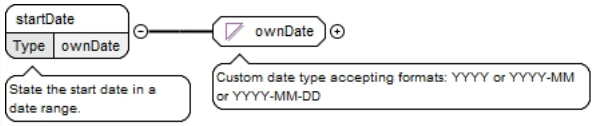
Namespace	employment
Annotations	Used for supplying a date range.

Diagram	
Type	dateRangeType
Properties	content: `complex` minOccurs: `0`
Model	startDate , endDate{0,1}
Children	endDate, startDate

Instance	`<dateRange xmlns="employment">`
	`  <startDate>{1,1}</startDate>`
	`  <endDate>{0,1}</endDate>`
	`</dateRange>`

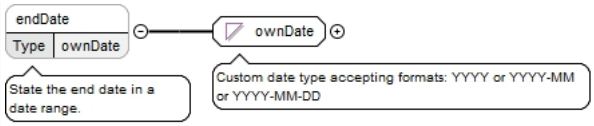
Source	`<xs:element name="dateRange" type="dateRangeType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for supplying a date range.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element dateRangeType / startDate

Namespace	employment
Annotations	`State the start date in a date range.`
Diagram	
Type	ownDate
Properties	content: `simple`
Facets	pattern: `[\d]{4}\-[\d]{2}\-[\d]{2}|[\d]{4}\-[\d]{2}|[\d]{4}`

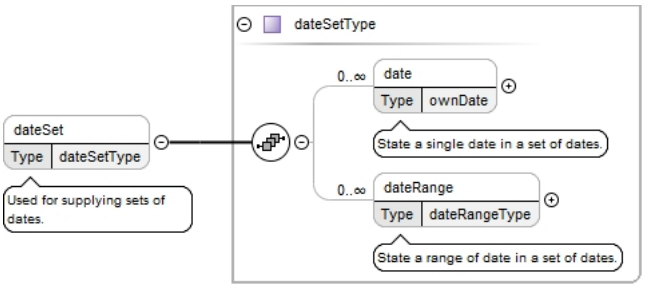
Source	`<xs:element name="startDate" type="ownDate">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the start date in a date range.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element dateRangeType / endDate

Namespace	employment
Annotations	`State the end date in a date range.`
Diagram	
Type	ownDate
Properties	content: `simple` minOccurs: `0`
Facets	pattern: `[\d]{4}\-[\d]{2}\-[\d]{2}|[\d]{4}\-[\d]{2}|[\d]{4}`

Source	`<xs:element name="endDate" minOccurs="0" type="ownDate">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the end date in a date range.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

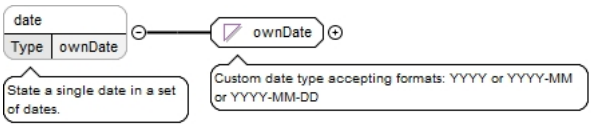
Element `salaryPeriodType / dateSet`

Namespace	employment
Annotations	`Used for supplying sets of dates.`
Diagram	
Type	dateSetType
Properties	content: `complex` minOccurs: `0`
Model	date* , dateRange*
Children	date, dateRange

Instance	`<dateSet xmlns="employment">`
	`  <date>{0,unbounded}</date>`
	`  <dateRange>{0,unbounded}</dateRange>`
	`</dateSet>`

Source	`<xs:element name="dateSet" type="dateSetType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for supplying sets of dates.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

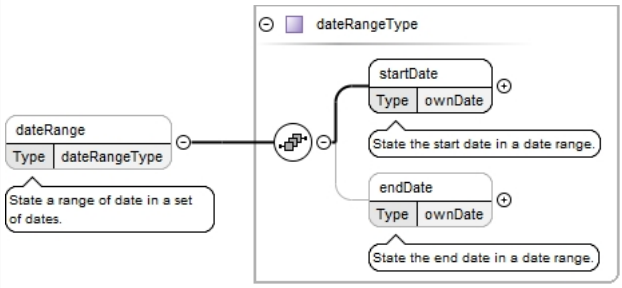
Element `dateSetType / date`

Namespace	employment
Annotations	`State a single date in a set of dates.`
Diagram	
Type	ownDate
Properties	content: `simple` minOccurs: `0` maxOccurs: `unbounded`
Facets	pattern: `[\d]{4}\-[\d]{2}\-[\d]{2}|[\d]{4}\-[\d]{2}|[\d]{4}`

Source	`<xs:element name="date" minOccurs="0" maxOccurs="unbounded" type="ownDate">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State a single date in a set of dates.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `dateSetType / dateRange`

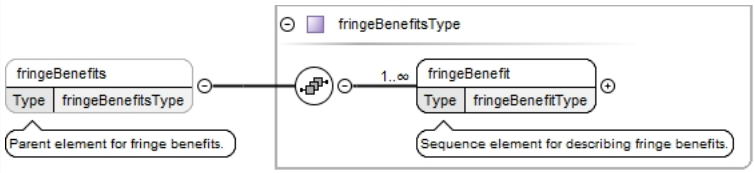
Namespace	employment
Annotations	`State a range of date in a set of dates.`

Diagram	
Type	dateRangeType
Properties	content: complex minOccurs: 0 maxOccurs: unbounded
Model	startDate , endDate{0,1}
Children	endDate, startDate

Instance	`<dateRange xmlns="employment">`
	`  <startDate>{1,1}</startDate>`
	`  <endDate>{0,1}</endDate>`
	`</dateRange>`

Source	`<xs:element name="dateRange" type="dateRangeType" minOccurs="0" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State a range of date in a set of dates.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / fringeBenefits

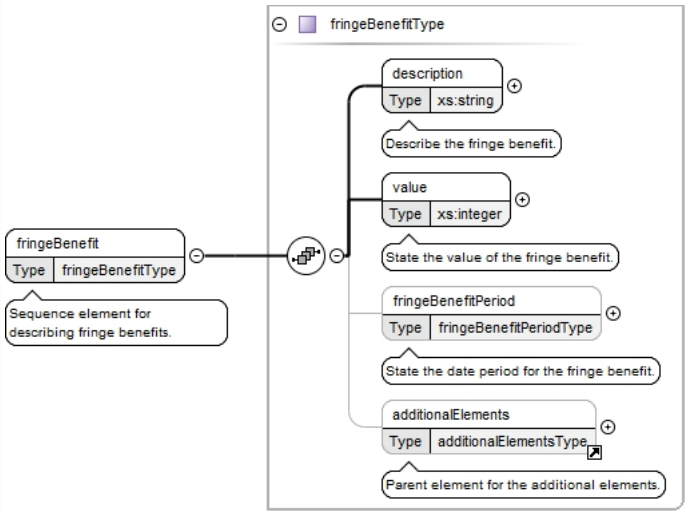
Namespace	employment
Annotations	Parent element for fringe benefits.
Diagram	
Type	fringeBenefitsType
Properties	content: complex minOccurs: 0 maxOccurs: unbounded
Model	fringeBenefit+
Children	fringeBenefit

Instance	`<fringeBenefits xmlns="employment">`
	`  <fringeBenefit>{1,unbounded}</fringeBenefit>`
	`</fringeBenefits>`

Source	`<xs:element name="fringeBenefits" type="fringeBenefitsType" minOccurs="0" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for fringe benefits.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element fringeBenefitsType / fringeBenefit

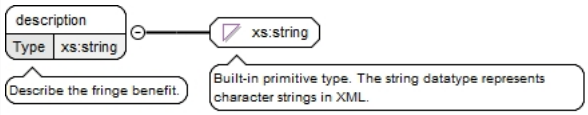
Namespace	employment
Annotations	Sequence element for describing fringe benefits.

Diagram	
Type	fringeBenefitType
Properties	content: complex maxOccurs: unbounded
Model	description , value , fringeBenefitPeriod{0,1} , additionalElements{0,1}
Children	additionalElements, description, fringeBenefitPeriod, value

Instance	`<fringeBenefit xmlns="employment">`
	`  <description>{1,1}</description>`
	`  <value>{1,1}</value>`
	`  <fringeBenefitPeriod>{0,1}</fringeBenefitPeriod>`
	`  <additionalElements>{0,1}</additionalElements>`
	`</fringeBenefit>`

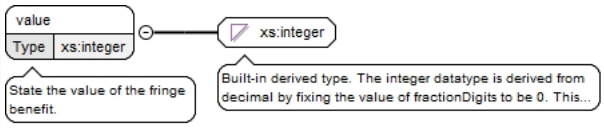
Source	`<xs:element name="fringeBenefit" type="fringeBenefitType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing fringe benefits.</`
	`xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element fringeBenefitType / description

Namespace	employment
Annotations	Describe the fringe benefit.
Diagram	
Type	xs:string
Properties	content: simple

Source	`<xs:element name="description" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the fringe benefit.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

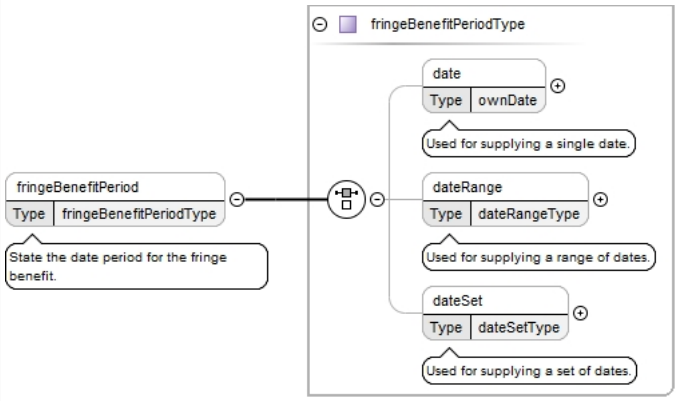
Element fringeBenefitType / value

Namespace	employment
Annotations	State the value of the fringe benefit.
Diagram	
Type	xs:integer

Properties	content: simple

Source	`<xs:element name="value" type="xs:integer">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the value of the fringe benefit.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

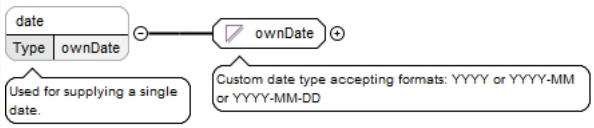
Element fringeBenefitType / fringeBenefitPeriod

Namespace	employment
Annotations	State the date period for the fringe benefit.
Diagram	
Type	fringeBenefitPeriodType
Properties	content: complex minOccurs: 0
Model	date{0,1} | dateRange{0,1} | dateSet{0,1}
Children	date, dateRange, dateSet

Instance	`<fringeBenefitPeriod xmlns="employment">`
	`  <date>{0,1}</date>`
	`  <dateRange>{0,1}</dateRange>`
	`  <dateSet>{0,1}</dateSet>`
	`</fringeBenefitPeriod>`

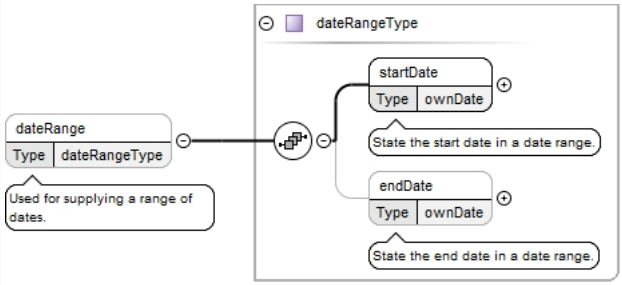
Source	`<xs:element name="fringeBenefitPeriod" type="fringeBenefitPeriodType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the date period for the fringe benefit.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element fringeBenefitPeriodType / date

Namespace	employment
Annotations	Used for supplying a single date.
Diagram	
Type	ownDate
Properties	content: simple minOccurs: 0
Facets	pattern [\d]{4}\-[\d]{2}\-[\d]{2}|[\d]{4}\-[\d]{2}|[\d]{4}

Source	`<xs:element name="date" minOccurs="0" type="ownDate">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for supplying a single date.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

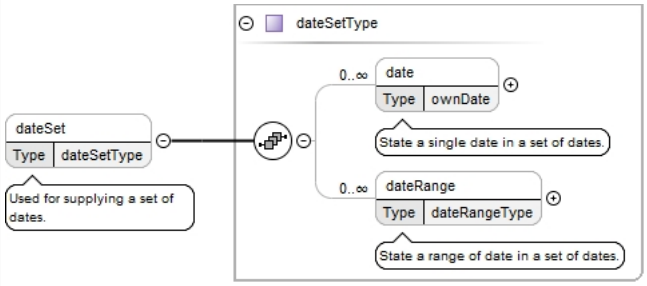
Element fringeBenefitPeriodType / dateRange

Namespace	employment
Annotations	Used for supplying a range of dates.
Diagram	
Type	dateRangeType
Properties	content: complex minOccurs: 0
Model	startDate , endDate{0,1}
Children	endDate, startDate

Instance	`<dateRange xmlns="employment">`
	`  <startDate>{1,1}</startDate>`
	`  <endDate>{0,1}</endDate>`
	`</dateRange>`

Source	`<xs:element name="dateRange" type="dateRangeType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for supplying a range of dates.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element fringeBenefitPeriodType / dateSet

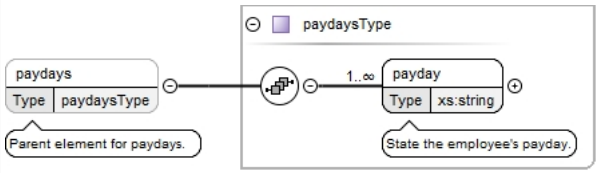
Namespace	employment
Annotations	Used for supplying a set of dates.
Diagram	
Type	dateSetType
Properties	content: complex minOccurs: 0
Model	date* , dateRange*
Children	date, dateRange

Instance	`<dateSet xmlns="employment">`
	`  <date>{0,unbounded}</date>`
	`  <dateRange>{0,unbounded}</dateRange>`
	`</dateSet>`

Source	`<xs:element name="dateSet" type="dateSetType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for supplying a set of dates.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / paydays

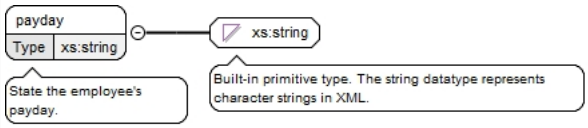
Namespace	employment
Annotations	Parent element for paydays.

Diagram	
Type	paydaysType
Properties	content: complex minOccurs: 0
Model	payday+
Children	payday

Instance	`<paydays xmlns="employment">`
	`  <payday>{1,unbounded}</payday>`
	`</paydays>`

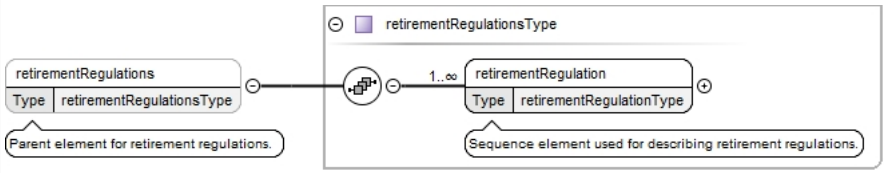
Source	`<xs:element name="paydays" type="paydaysType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for paydays.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element paydaysType / payday

Namespace	employment
Annotations	State the employee's payday.
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 1 maxOccurs: unbounded

Source	`<xs:element name="payday" minOccurs="1" maxOccurs="unbounded" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the employee's payday.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / retirementRegulations

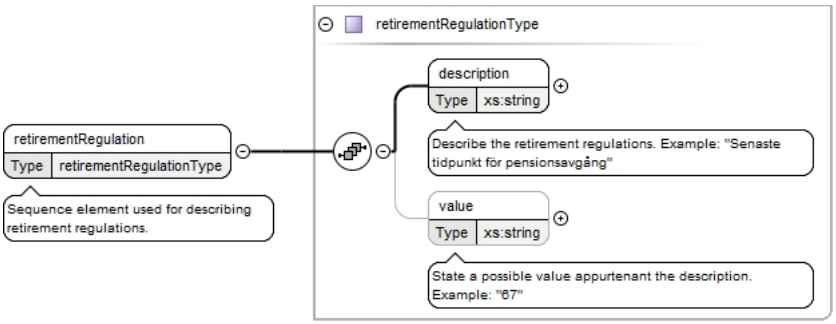
Namespace	employment
Annotations	Parent element for retirement regulations.
Diagram	
Type	retirementRegulationsType
Properties	content: complex minOccurs: 0
Model	retirementRegulation+
Children	retirementRegulation

Instance	`<retirementRegulations xmlns="employment">`
	`  <retirementRegulation>{1,unbounded}</retirementRegulation>`
	`</retirementRegulations>`

Source	`<xs:element name="retirementRegulations" type="retirementRegulationsType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for retirement regulations.</xs:documentation>`
	`  </xs:annotation>`

```
</xs:element>
```

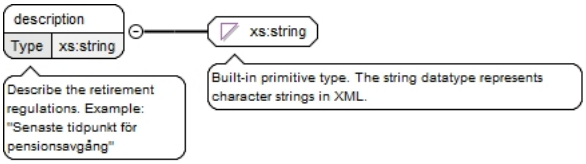
Element retirementRegulationsType / retirementRegulation

Namespace	employment
Annotations	Sequence element used for describing retirement regulations.
Diagram	
Type	retirementRegulationType
Properties	content: complex minOccurs: 1 maxOccurs: unbounded
Model	description , value{0,1}
Children	description, value

Instance	`<retirementRegulation xmlns="employment">`
	`  <description>{1,1}</description>`
	`  <value>{0,1}</value>`
	`</retirementRegulation>`

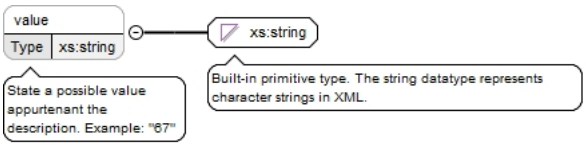
Source	`<xs:element name="retirementRegulation" type="retirementRegulationType" minOccurs="1"`
	` maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element used for describing retirement regulations.</`
	`xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element retirementRegulationType / description

Namespace	employment
Annotations	Describe the retirement regulations. Example: "Senaste tidpunkt för pensionsavgång"
Diagram	
Type	xs:string
Properties	content: simple

Source	`<xs:element name="description" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the retirement regulations. Example: "Senaste tidpunkt`
	` för pensionsavgång"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

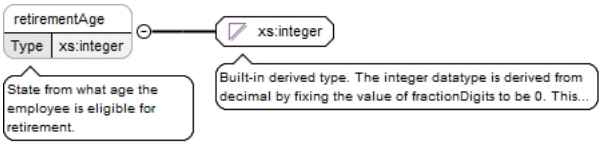
Element retirementRegulationType / value

Namespace	employment
Annotations	State a possible value appurtenant the description. Example: "67"
Diagram	

Type	xs:string
Properties	content: simple minOccurs: 0

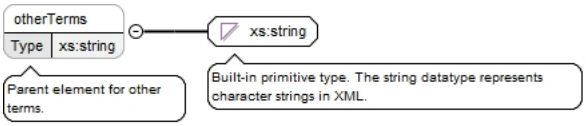
Source	`<xs:element name="value" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State a possible value appurtenant the description. Example: "67"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / retirementAge

Namespace	employment
Annotations	State from what age the employee is eligible for retirement.
Diagram	
Type	xs:integer
Properties	content: simple minOccurs: 0

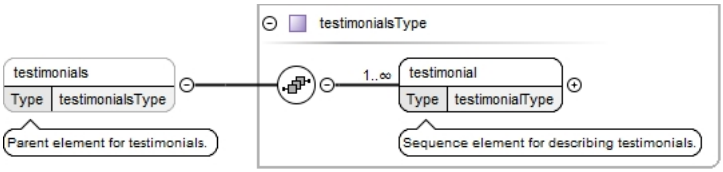
Source	`<xs:element name="retirementAge" minOccurs="0" type="xs:integer">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State from what age the employee is eligible for retirement.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / otherTerms

Namespace	employment
Annotations	Parent element for other terms.
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 0 maxOccurs: unbounded

Source	`<xs:element name="otherTerms" minOccurs="0" maxOccurs="unbounded" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for other terms.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element employmentDescriptionType / testimonials

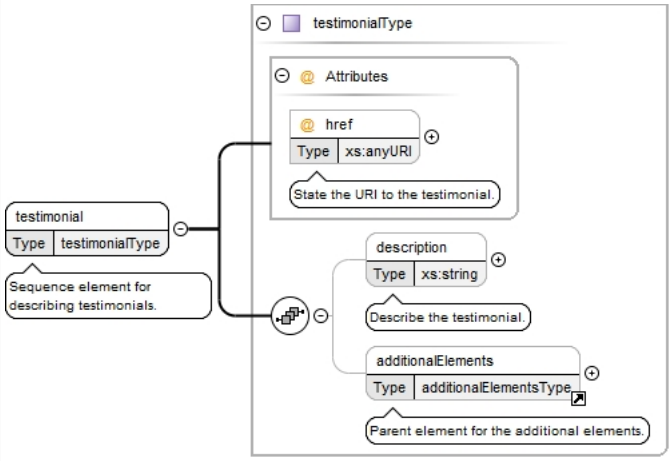
Namespace	employment
Annotations	Parent element for testimonials.
Diagram	
Type	testimonialsType

Properties	content: complex minOccurs: 0
Model	testimonial+
Children	testimonial

Instance	`<testimonials xmlns="employment">`
	`  <testimonial href="">{1,unbounded}</testimonial>`
	`</testimonials>`

Source	`<xs:element name="testimonials" type="testimonialsType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for testimonials.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element testimonialsType / testimonial

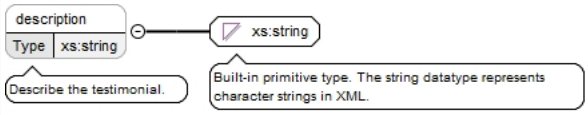
Namespace	employment
Annotations	Sequence element for describing testimonials.
Diagram	
Type	testimonialType
Properties	content: complex maxOccurs: unbounded
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description

Instance	`<testimonial href="" xmlns="employment">`
	`  <description>{0,1}</description>`
	`  <additionalElements>{0,1}</additionalElements>`
	`</testimonial>`

Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional State the URI to the testimonial.

Source	`<xs:element name="testimonial" type="testimonialType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing testimonials.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

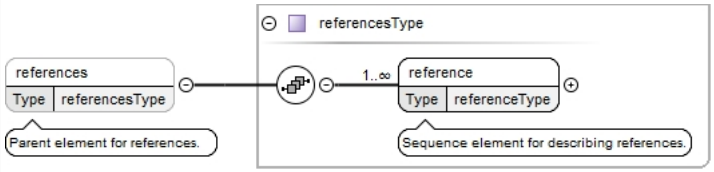
Element testimonialType / description

Namespace	employment
Annotations	Describe the testimonial.
Diagram	
Type	xs:string

Properties	content: `simple` minOccurs: `0`

Source	`<xs:element name="description" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the testimonial.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

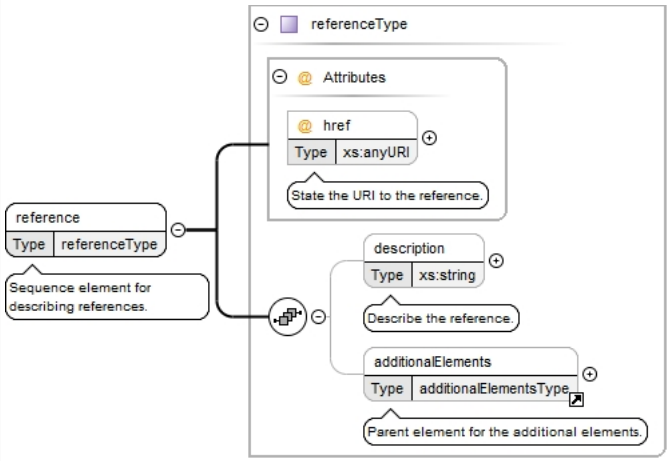
Element `employmentDescriptionType / references`

Namespace	employment
Annotations	`Parent element for references.`
Diagram	
Type	referencesType
Properties	content: `complex` minOccurs: `0`
Model	reference+
Children	reference

Instance	`<references xmlns="employment">`
	`  <reference href="">{1,unbounded}</reference>`
	`</references>`

Source	`<xs:element name="references" type="referencesType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Parent element for references.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Element `referencesType / reference`

Namespace	employment
Annotations	`Sequence element for describing references.`
Diagram	
Type	referenceType
Properties	content: `complex` maxOccurs: `unbounded`
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description

Instance	`<reference href="" xmlns="employment">`
	`  <description>{0,1}</description>`
	`  <additionalElements>{0,1}</additionalElements>`

```
</reference>
```

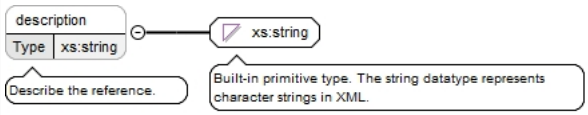
Attributes

QName	Type	Use
href	xs:anyURI	optional
	State the URI to the reference.	

Source

```
<xs:element name="reference" type="referenceType" maxOccurs="unbounded">
  <xs:annotation>
    <xs:documentation xml:lang="en">Sequence element for describing references.</xs:documentation>
  </xs:annotation>
</xs:element>
```

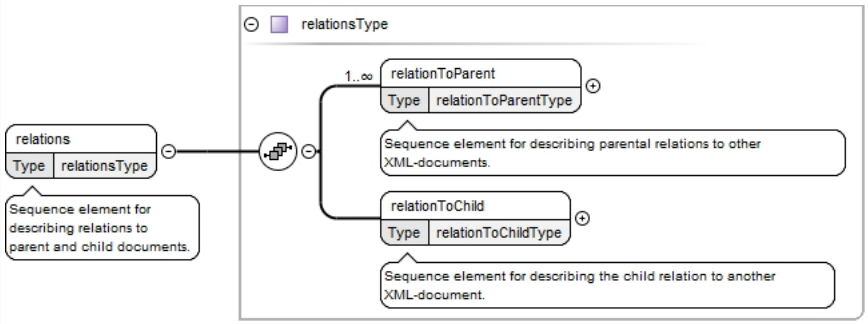
Element referenceType / description

Namespace	employment
Annotations	Describe the reference.
Diagram	
Type	xs:string
Properties	content: simple minOccurs: 0

Source

```
<xs:element name="description" minOccurs="0" type="xs:string">
  <xs:annotation>
    <xs:documentation xml:lang="en">Describe the reference.</xs:documentation>
  </xs:annotation>
</xs:element>
```

Element employmentDescriptionType / relations

Namespace	employment
Annotations	Sequence element for describing relations to parent and child documents.
Diagram	
Type	relationsType
Properties	content: complex
Model	relationToParent+ , relationToChild
Children	relationToChild, relationToParent
Instance	

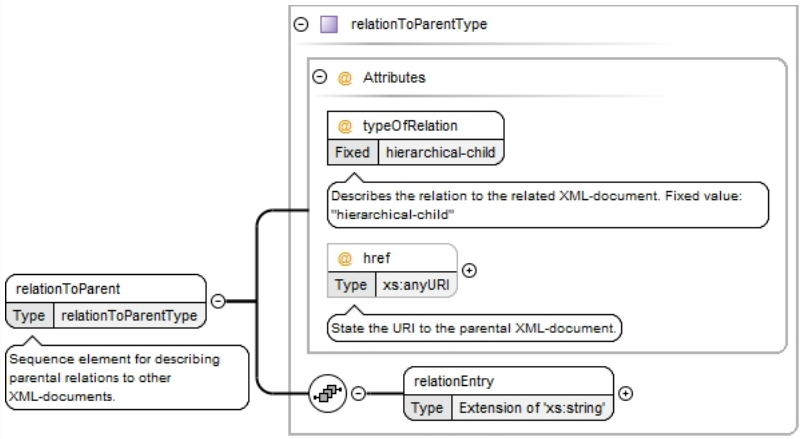
```
<relations xmlns="employment">
  <relationToParent href="" typeOfRelation="hierarchical-child">{1,unbounded}</relationToParent>
  <relationToChild href="" typeOfRelation="hierarchical-parent">{1,1}</relationToChild>
</relations>
```

Source

```
<xs:element name="relations" type="relationsType">
  <xs:annotation>
    <xs:documentation xml:lang="en">Sequence element for describing relations to parent and child
 documents.</xs:documentation>
  </xs:annotation>
</xs:element>
```

Element relationsType / relationToParent

Namespace	employment

Annotations	Sequence element for describing parental relations to other XML-documents.
Diagram	
Type	relationToParentType
Properties	content: complex maxOccurs: unbounded
Model	relationEntry
Children	relationEntry

Instance	`<relationToParent href="" typeOfRelation="hierarchical-child" xmlns="employment">`
	`  <relationEntry parentOrganisationReferenceCode="">{1,1}</relationEntry>`
	`</relationToParent>`

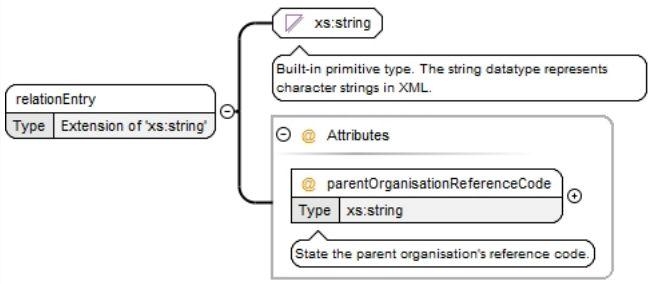
Attributes

QName	Type	Fixed	Use	
href	xs:anyURI		optional	
	State the URI to the parental XML-document.			
typeOfRelation		hierarchical-child	required	
	Describes the relation to the related XML-document. Fixed value: "hierarchical-child"			

Source

```
<xs:element name="relationToParent" type="relationToParentType" maxOccurs="unbounded">
  <xs:annotation>
    <xs:documentation xml:lang="en">Sequence element for describing parental relations to other XML-
documents.</xs:documentation>
  </xs:annotation>
</xs:element>
```

Element relationToParentType / relationEntry

Namespace	employment
Diagram	
Type	extension of xs:string
Properties	content: complex

Attributes

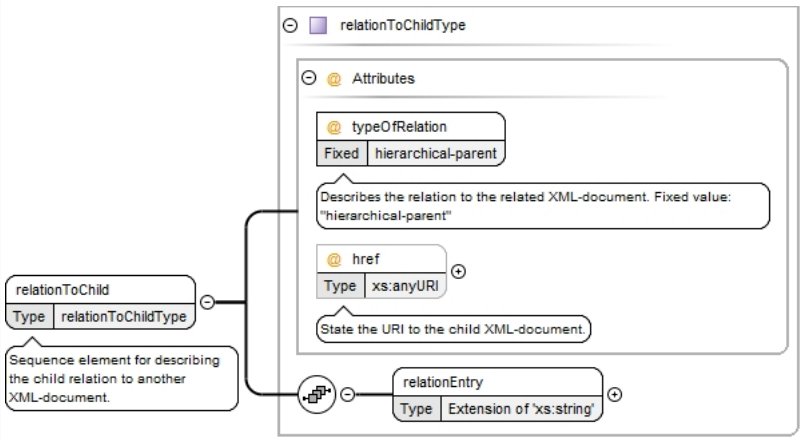
QName	Type	Use	
parentOrganisationReferenceCode	xs:string	required	
	State the parent organisation's reference code.		

Source

```
<xs:element name="relationEntry">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:string">
```

```
        <xs:attribute name="parentOrganisationReferenceCode" type="xs:string" use="required">
          <xs:annotation>
            <xs:documentation xml:lang="en">State the parent organisation's reference code.</
xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
```

Element relationsType / relationToChild

Namespace	employment
Annotations	Sequence element for describing the child relation to another XML-document.
Diagram	
Type	relationToChildType
Properties	content: complex
Model	relationEntry
Children	relationEntry

Instance	`<relationToChild href="" typeOfRelation="hierarchical-parent" xmlns="employment">`
	`  <relationEntry childOrganisationReferenceCode="">{1,1}</relationEntry>`
	`</relationToChild>`

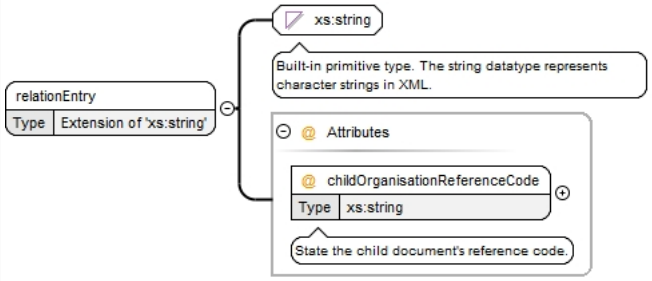
Attributes	

QName	Type	Fixed	Use	
href	xs:anyURI		optional	
	State the URI to the child XML-document.			
typeOfRelation		hierarchical-parent	required	
	Describes the relation to the related XML-document. Fixed value: "hierarchical-parent"			

Source

```
<xs:element name="relationToChild" type="relationToChildType">
  <xs:annotation>
    <xs:documentation xml:lang="en">Sequence element for describing the child relation to another
 XML-document.</xs:documentation>
  </xs:annotation>
</xs:element>
```

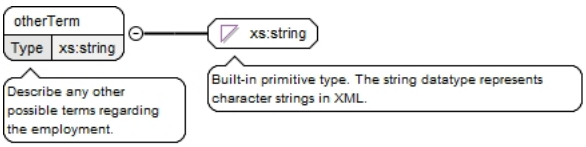
Element relationToChildType / relationEntry

Namespace	employment
Diagram	

Type	extension of xs:string
Properties	content: complex
Attributes	**QName** / **Type** / **Use** **childOrganisationReference-Code** / xs:string / required State the child document's reference code.

Source	`<xs:element name="relationEntry">`
	`  <xs:complexType>`
	`  <xs:simpleContent>`
	`  <xs:extension base="xs:string">`
	`  <xs:attribute name="childOrganisationReferenceCode" type="xs:string" use="required">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the child document's reference code.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:attribute>`
	`  </xs:extension>`
	`  </xs:simpleContent>`
	`  </xs:complexType>`
	`</xs:element>`

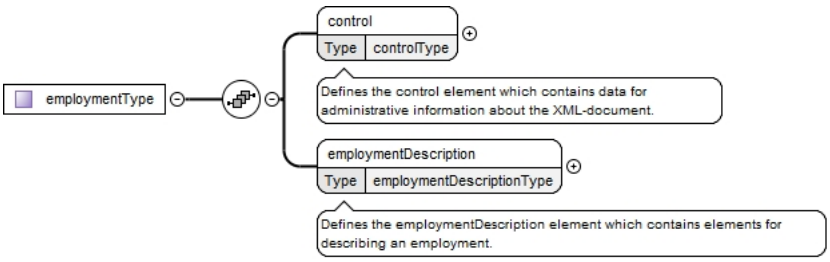
Element otherTermsType / otherTerm

Namespace	employment
Annotations	Describe any other possible terms regarding the employment.
Diagram	
Type	xs:string
Properties	content: simple maxOccurs: unbounded

Source	`<xs:element name="otherTerm" type="xs:string" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe any other possible terms regarding the employment.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:element>`

Complex Type(s)

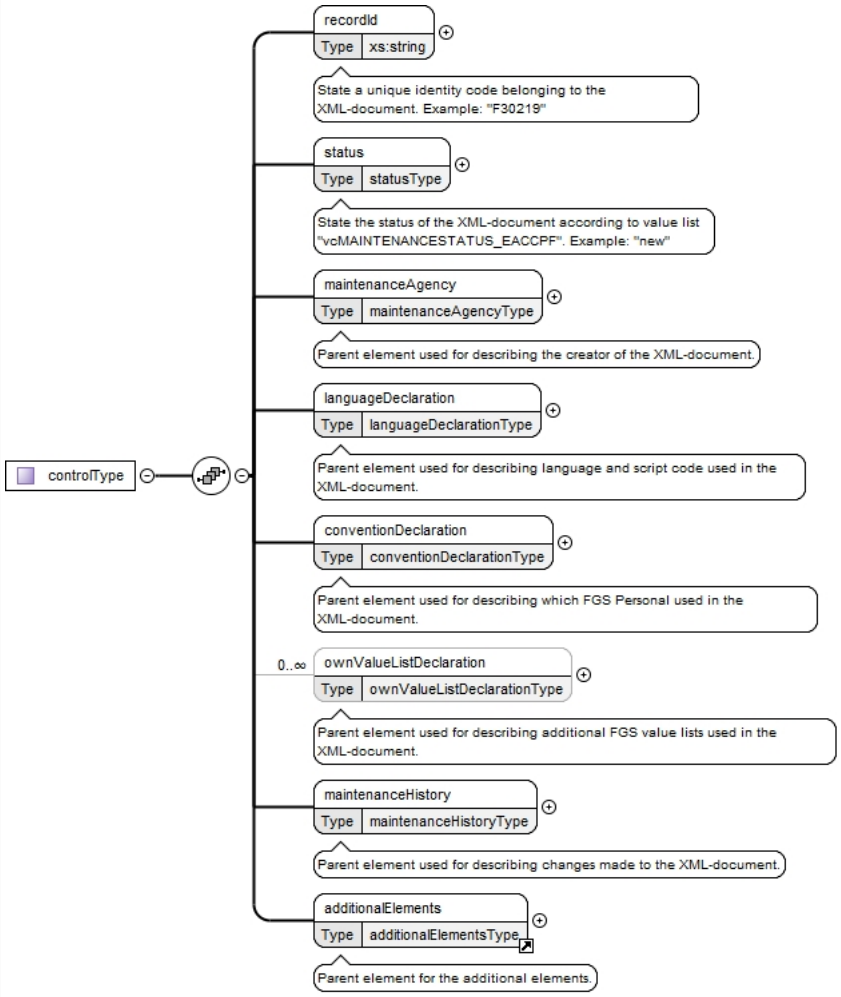
Complex Type employmentType

Namespace	employment
Diagram	
Used by	Element employment
Model	control , employmentDescription
Children	control, employmentDescription

Source	`<xs:complexType name="employmentType">`
	`  <xs:sequence>`
	`  <xs:element name="control" type="controlType">`
	`  <xs:annotation>`

```
        <xs:documentation xml:lang="en">Defines the control element which contains data for
 administrative information about the XML-document.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="employmentDescription" type="employmentDescriptionType">
      <xs:annotation>
        <xs:documentation xml:lang="en">Defines the employmentDescription element which contains
 elements for describing an employment.</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

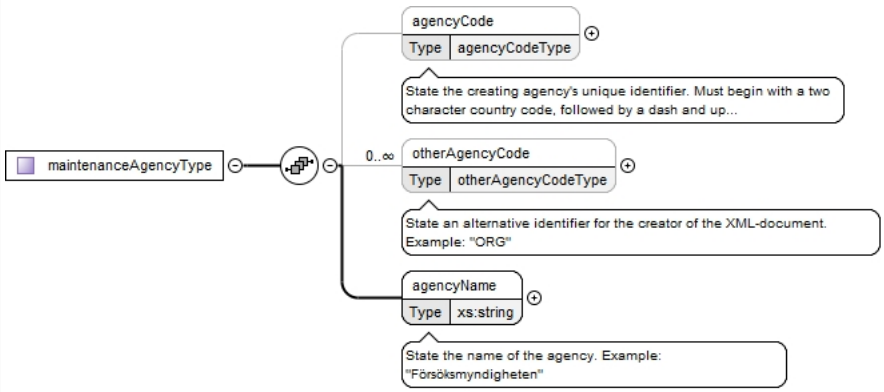
Complex Type `controlType`

Namespace	employment
Diagram	
Used by	Element employmentType/control
Model	recordId , status , maintenanceAgency , languageDeclaration , conventionDeclaration , ownValueListDeclaration* , maintenance-History , additionalElements
Children	additionalElements, conventionDeclaration, languageDeclaration, maintenanceAgency, maintenanceHistory, ownValueListDeclaration, recordId, status
Source	

```
<xs:complexType name="controlType">
  <xs:sequence>
    <xs:element name="recordId" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">State a unique identity code belonging to the XML-document.
 Example: "F30219"</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="status" type="statusType">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the status of the XML-document according to value list
 "vcMAINTENANCESTATUS_EACCPF". Example: "new"</xs:documentation>
```

```
      </xs:annotation>
    </xs:element>
    <xs:element name="maintenanceAgency" type="maintenanceAgencyType">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element used for describing the creator of the XML-
document.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="languageDeclaration" type="languageDeclarationType">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element used for describing language and script code
 used in the XML-document.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="conventionDeclaration" type="conventionDeclarationType">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element used for describing which FGS Personal used
 in the XML-document.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="ownValueListDeclaration" type="ownValueListDeclarationType" minOccurs="0"
 maxOccurs="unbounded">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element used for describing additional FGS value
 lists used in the XML-document.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="maintenanceHistory" type="maintenanceHistoryType">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element used for describing changes made to the XML-
document.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element ref="additionalElements"/>
  </xs:sequence>
</xs:complexType>
```

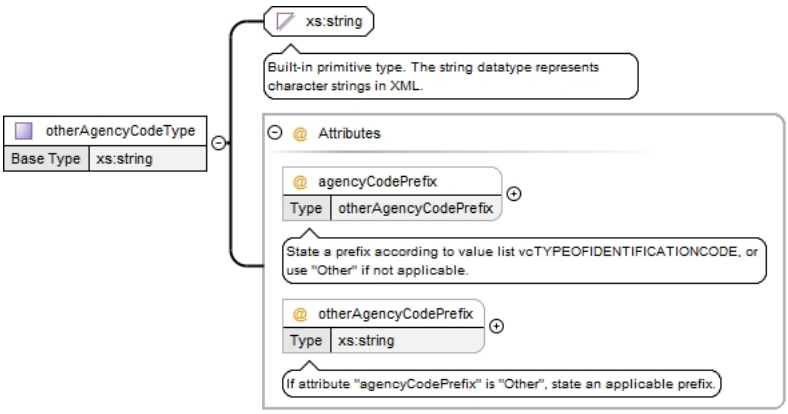
Complex Type `maintenanceAgencyType`

Namespace	employment
Diagram	
Used by	Element controlType/maintenanceAgency
Model	agencyCode{0,1} , otherAgencyCode* , agencyName
Children	agencyCode, agencyName, otherAgencyCode
Source	

```
<xs:complexType name="maintenanceAgencyType">
  <xs:sequence>
    <xs:element name="agencyCode" type="agencyCodeType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the creating agency's unique identifier. Must begin
 with a two character country code, followed by a dash and up to 10 characters. Example: "SE-
A:ARA/123</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="otherAgencyCode" minOccurs="0" maxOccurs="unbounded"
 type="otherAgencyCodeType">
      <xs:annotation>
        <xs:documentation xml:lang="en">State an alternative identifier for the creator of the XML-
document. Example: "ORG"</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="agencyName" type="xs:string">
```

```
      <xs:annotation>
        <xs:documentation xml:lang="en">State the name of the agency. Example:
 "Försöksmyndigheten"</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

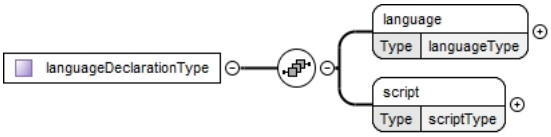
Complex Type `otherAgencyCodeType`

Namespace	employment
Diagram	
Type	extension of xs:string
Used by	Element maintenanceAgencyType/otherAgencyCode
Attributes	**QName** / **Type** / **Use** **agencyCodePrefix** / otherAgencyCodePrefix / optional State a prefix according to value list vcTYPEOFIDENTIFICATIONCODE, or use "Other" if not applicable. **otherAgencyCodePrefix** / xs:string / optional If attribute "agencyCodePrefix" is "Other", state an applicable prefix.

Source

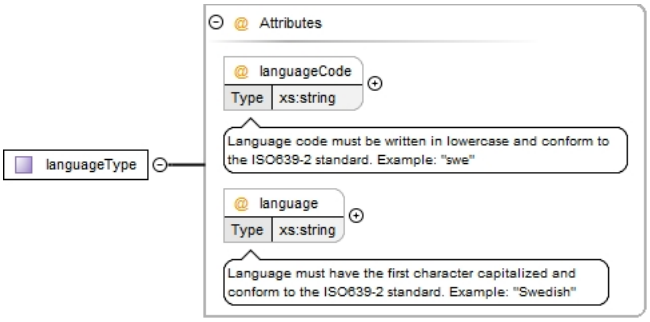
```
<xs:complexType name="otherAgencyCodeType">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="agencyCodePrefix" type="otherAgencyCodePrefix" use="optional">
        <xs:annotation>
          <xs:documentation xml:lang="en">State a prefix according to value list
 vcTYPEOFIDENTIFICATIONCODE, or use "Other" if not applicable.</xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="otherAgencyCodePrefix" type="xs:string" use="optional">
        <xs:annotation>
          <xs:documentation xml:lang="en">If attribute "agencyCodePrefix" is "Other", state an
 applicable prefix.</xs:documentation>
        </xs:annotation>
      </xs:attribute>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Complex Type `languageDeclarationType`

Namespace	employment
Diagram	
Used by	Element controlType/languageDeclaration
Model	language , script
Children	language, script
Source	`<xs:complexType name="languageDeclarationType">`

```
  <xs:sequence>
    <xs:element name="language" type="languageType"/>
    <xs:element name="script" type="scriptType"/>
  </xs:sequence>
</xs:complexType>
```

Complex Type `languageType`

Namespace	employment
Diagram	
Used by	Element languageDeclarationType/language

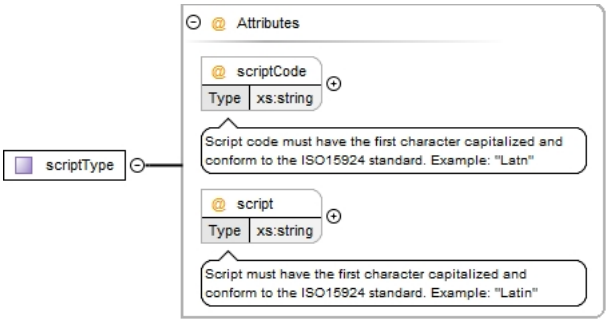
Attributes

QName	Type	Use	
language	xs:string	optional	
	Language must have the first character capitalized and conform to the ISO639-2 standard. Example: "Swedish"		
languageCode	xs:string	optional	
	Language code must be written in lowercase and conform to the ISO639-2 standard. Example: "swe"		

Source

```
<xs:complexType name="languageType">
  <xs:attribute name="languageCode" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">Language code must be written in lowercase and conform to the
 ISO639-2 standard. Example: "swe"</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="language" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">Language must have the first character capitalized and conform
 to the ISO639-2 standard. Example: "Swedish"</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type `scriptType`

Namespace	employment
Diagram	
Used by	Element languageDeclarationType/script

Attributes

QName	Type	Use	
script	xs:string	optional	
	Script must have the first character capitalized and conform to the ISO15924 standard. Example: "Latin"		

	QName	Type	Use	
	scriptCode	xs:string	optional	
		Script code must have the first character capitalized and conform to the ISO15924 standard. Example: "Latn"		

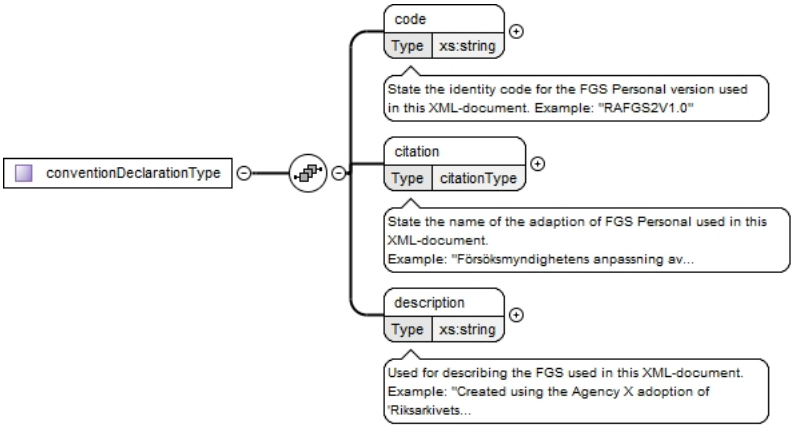
Source

```
<xs:complexType name="scriptType">
  <xs:attribute name="scriptCode" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">Script code must have the first character capitalized and
 conform to the ISO15924 standard. Example: "Latn"</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="script" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">Script must have the first character capitalized and conform
 to the ISO15924 standard. Example: "Latin"</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type `conventionDeclarationType`

Namespace: employment

Diagram



Used by: Element controlType/conventionDeclaration

Model: code , citation , description

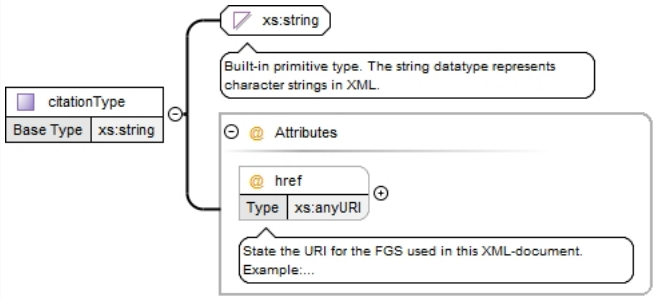
Children: citation, code, description

Source

```
<xs:complexType name="conventionDeclarationType">
  <xs:sequence>
    <xs:element name="code" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the identity code for the FGS Personal version used in
 this XML-document. Example: "RAFGS2V1.0"</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="citation" type="citationType">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the name of the adaption of FGS Personal used in this
 XML-document.</xs:documentation>
        <xs:documentation xml:lang="en">Example: "Försöksmyndighetens anpassning av FGS Personal för
 överföring av personalinformation till det egna E-arktivet"</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="description" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">Used for describing the FGS used in this XML-document.</
xs:documentation>
        <xs:documentation xml:lang="en">Example: "Created using the Agency X adoption of
 'Riksarkivets anpassning av FGS Personal'"</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

Complex Type `citationType`

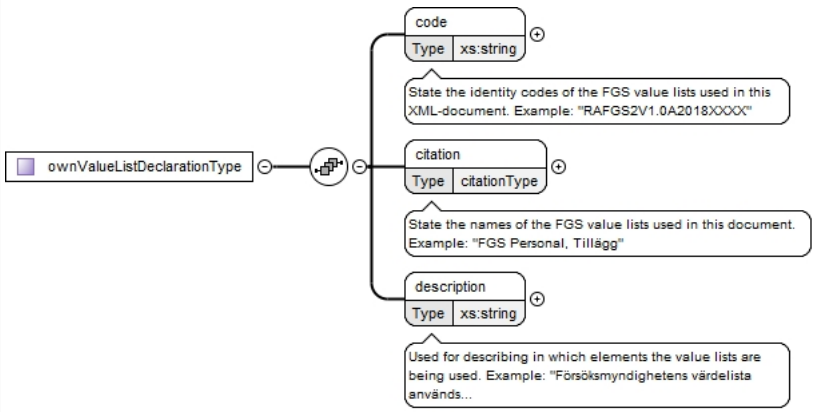
Namespace: employment

Diagram	
Type	extension of xs:string
Used by	Elements conventionDeclarationType/citation, ownValueListDeclarationType/citation
Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional State the URI for the FGS used in this XML-document. Example: "https://myndighet.se/Försöksmyndighetens_FGS_Personal_V_20181224.pdf"

Source

```
<xs:complexType name="citationType">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="href" type="xs:anyURI">
        <xs:annotation>
          <xs:documentation xml:lang="en">State the URI for the FGS used in this XML-document.
 Example: "https://myndighet.se/Försöksmyndighetens_FGS_Personal_V_20181224.pdf"</xs:documentation>
        </xs:annotation>
      </xs:attribute>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Complex Type `ownValueListDeclarationType`

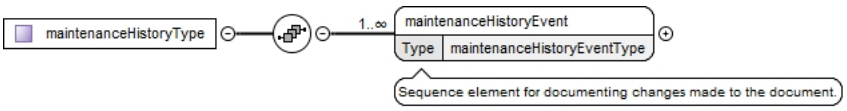
Namespace	employment
Diagram	
Used by	Element controlType/ownValueListDeclaration
Model	code , citation , description
Children	citation, code, description

Source

```
<xs:complexType name="ownValueListDeclarationType">
  <xs:sequence>
    <xs:element name="code" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the identity codes of the FGS value lists used in this
 XML-document. Example: "RAFGS2V1.0A2018XXXX"</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="citation" type="citationType">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the names of the FGS value lists used in this
 document. Example: "FGS Personal, Tillägg"</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="description" type="xs:string">
      <xs:annotation>
```

```
        <xs:documentation xml:lang="en">Used for describing in which elements the value lists
 are being used. Example: "Försöksmyndighetens värdelista används i elementet AnhörigFält"</
xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

Complex Type `maintenanceHistoryType`

Namespace	employment
Diagram	
Used by	Element controlType/maintenanceHistory
Model	maintenanceHistoryEvent+
Children	maintenanceHistoryEvent

Source	`<xs:complexType name="maintenanceHistoryType">`
	`  <xs:sequence>`
	`  <xs:element name="maintenanceHistoryEvent" type="maintenanceHistoryEventType" minOccurs="1" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for documenting changes made to the document.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

Complex Type `maintenanceHistoryEventType`

Namespace	employment
Diagram	
Type	extension of xs:string
Used by	Element maintenanceHistoryType/maintenanceHistoryEvent

Attributes

QName	Type	Use	
agent	xs:string	required	
	State the name of the agent who performed the changes. Example: "Anna Andersson"		
agentType	agentTypeType	required	
	State the type of agent that performed the changes according to vcAGENTTYPE_EACCPF, or use "Other" if not applicable. Example: "human"		
eventDateTime	xs:dateTime	required	
	State the date and time the change was made. Example: "2001-04-13T22:30:00"		
eventType	eventTypeType	required	
	State what kind of change has been made accorrding to vcEVENTTYPE_EACCPF. Example: "created"		
otherAgentType	xs:string	optional	
	If attribute "agentType" is "Other", state an applicable agent type.		

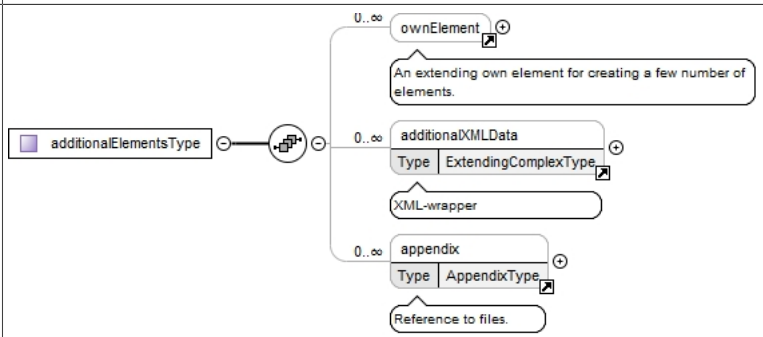
Source

```
<xs:complexType name="maintenanceHistoryEventType">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="eventType" type="eventTypeType" use="required">
        <xs:annotation>
          <xs:documentation xml:lang="en">State what kind of change has been made accorrding to
 vcEVENTTYPE_EACCPF. Example: "created"</xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="eventDateTime" type="xs:dateTime" use="required">
        <xs:annotation>
          <xs:documentation xml:lang="en">State the date and time the change was made. Example:
 "2001-04-13T22:30:00"</xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="agentType" type="agentTypeType" use="required">
        <xs:annotation>
          <xs:documentation xml:lang="en">State the type of agent that performed the changes
 according to vcAGENTTYPE_EACCPF, or use "Other" if not applicable. Example: "human"</
xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="otherAgentType" type="xs:string" use="optional">
        <xs:annotation>
          <xs:documentation xml:lang="en">If attribute "agentType" is "Other", state an applicable
 agent type.</xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="agent" type="xs:string" use="required">
        <xs:annotation>
          <xs:documentation xml:lang="en">State the name of the agent who performed the changes.
 Example: "Anna Andersson"</xs:documentation>
        </xs:annotation>
      </xs:attribute>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Complex Type `additionalElementsType`

Namespace: employment

Diagram



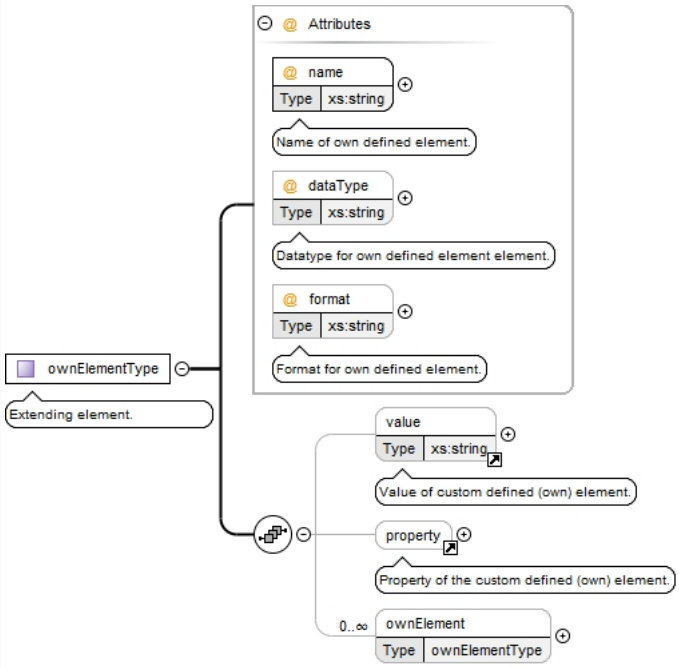
Used by

Element	additionalElements

Model	ownElement* , additionalXMLData* , appendix*
Children	additionalXMLData, appendix, ownElement

Source	`<xs:complexType name="additionalElementsType">`
	`  <xs:sequence>`
	`  <xs:element ref="ownElement" minOccurs="0" maxOccurs="unbounded"/>`
	`  <xs:element ref="additionalXMLData" minOccurs="0" maxOccurs="unbounded"/>`
	`  <xs:element ref="appendix" minOccurs="0" maxOccurs="unbounded"/>`
	`  </xs:sequence>`
	`</xs:complexType>`

Complex Type `ownElementType`

Namespace	employment
Annotations	`Extending element.`
Diagram	
Used by	Elements ownElement/ownElement, ownElementType/ownElement
Model	value{0,1} , property{0,1} , ownElement*
Children	ownElement, property, value

Attributes

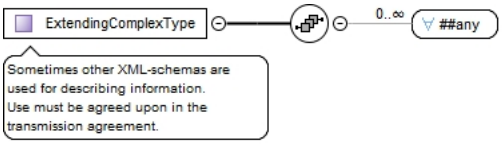
QName	Type	Use	
dataType	xs:string	optional	
	`Datatype for own defined element element.`		
format	xs:string	optional	
	`Format for own defined element.`		
name	xs:string	required	
	`Name of own defined element.`		

Source

```
<xs:complexType name="ownElementType">
  <xs:annotation>
    <xs:documentation xml:lang="en">Extending element.</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element ref="value" minOccurs="0"/>
    <xs:element ref="property" minOccurs="0"/>
    <xs:element name="ownElement" type="ownElementType" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="name" use="required" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">Name of own defined element.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="dataType" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">Datatype for own defined element element.</xs:documentation>
```

```
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="format" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">Format for own defined element.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type `ExtendingComplexType`

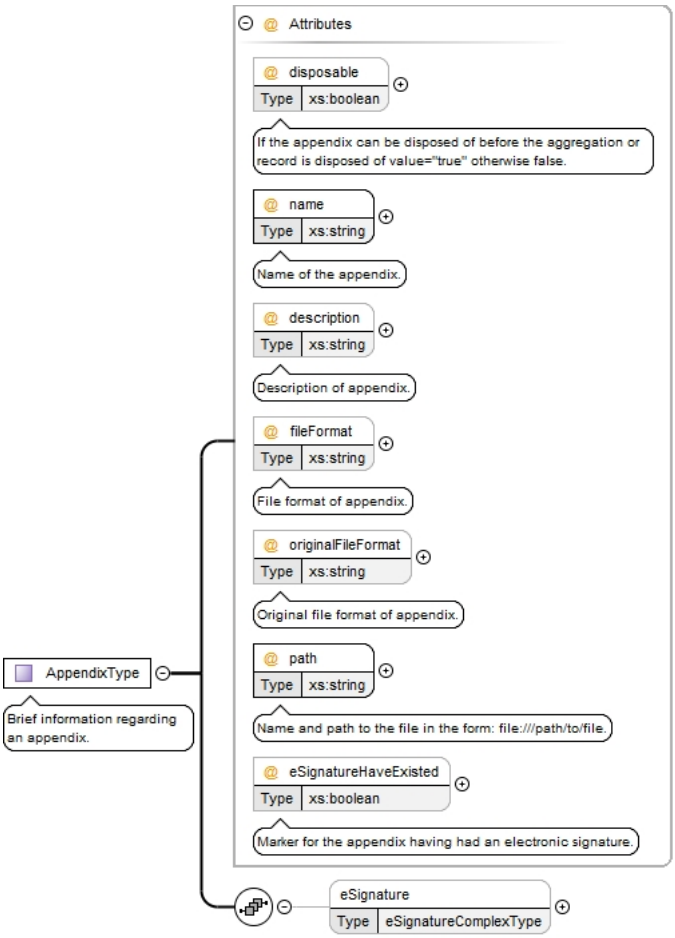
Namespace	employment
Annotations	Sometimes other XML-schemas are used for describing information. Use must be agreed upon in the transmission agreement.
Diagram	
Used by	Elements additionalXMLData, eSignatureComplexType/signature
Model	ANY element from ANY namespace

Source	`<xs:complexType name="ExtendingComplexType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sometimes other XML-schemas are used for describing information.</xs:documentation>`
	`  <xs:documentation xml:lang="en">Use must be agreed upon in the transmission agreement.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:sequence>`
	`  <xs:any namespace="##any" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>`
	`  </xs:sequence>`
	`</xs:complexType>`

Complex Type `AppendixType`

Namespace	employment
Annotations	Brief information regarding an appendix.

Diagram



Used by

Element appendix

Model

eSignature{0,1}

Children

eSignature

Attributes

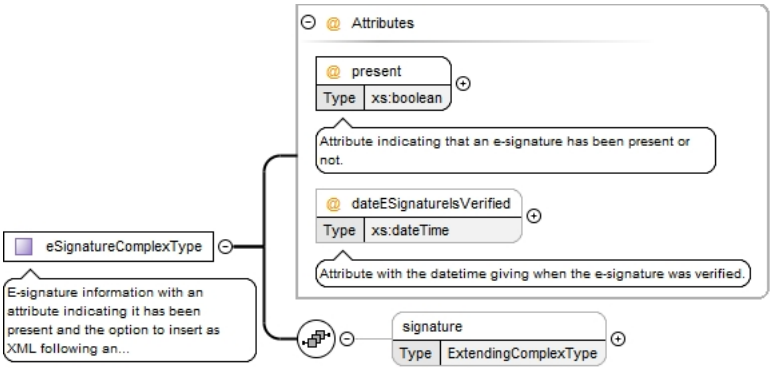
QName	Type	Use	
description	xs:string	optional	
	Description of appendix.		
disposable	xs:boolean	optional	
	If the appendix can be disposed of before the aggregation or record is disposed of value="true" otherwise false.		
eSignatureHaveExisted	xs:boolean	optional	
	Marker for the appendix having had an electronic signature.		
fileFormat	xs:string	optional	
	File format of appendix.		
name	xs:string	required	
	Name of the appendix.		
originalFileFormat	xs:string	optional	
	Original file format of appendix.		
path	xs:string	required	
	Name and path to the file in the form: file:///path/to/file.		

Source

```
<xs:complexType name="AppendixType">
  <xs:annotation>
    <xs:documentation xml:lang="en">Brief information regarding an appendix.</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="eSignature" type="eSignatureComplexType" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="disposable" type="xs:boolean" use="optional">
```

```
    <xs:annotation>
      <xs:documentation xml:lang="en">If the appendix can be disposed of before the aggregation or
 record is disposed of value="true" otherwise false.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="name" type="xs:string" use="required">
    <xs:annotation>
      <xs:documentation xml:lang="en">Name of the appendix.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="description" type="xs:string" use="optional">
    <xs:annotation>
      <xs:documentation xml:lang="en">Description of appendix.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="fileFormat" type="xs:string" use="optional">
    <xs:annotation>
      <xs:documentation xml:lang="en">File format of appendix.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="originalFileFormat" type="xs:string" use="optional">
    <xs:annotation>
      <xs:documentation xml:lang="en">Original file format of appendix.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="path" type="xs:string" use="required">
    <xs:annotation>
      <xs:documentation xml:lang="en">Name and path to the file in the form: file:///path/to/file.</
xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="eSignatureHaveExisted" type="xs:boolean" use="optional">
    <xs:annotation>
      <xs:documentation xml:lang="en">Marker for the appendix having had an electronic signature.</
xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type eSignatureComplexType

Namespace	employment
Annotations	E-signature information with an attribute indicating it has been present and the option to insert as XML following an e-signature XML-schema.
Diagram	
Used by	Element AppendixType/eSignature
Model	signature{0,1}
Children	signature

Attributes

QName	Type	Use	
dateESignatureIsVerified	xs:dateTime	optional	
	Attribute with the datetime giving when the e-signature was verified.		
present	xs:boolean	required	
	Attribute indicating that an e-signature has been present or not.		

Source

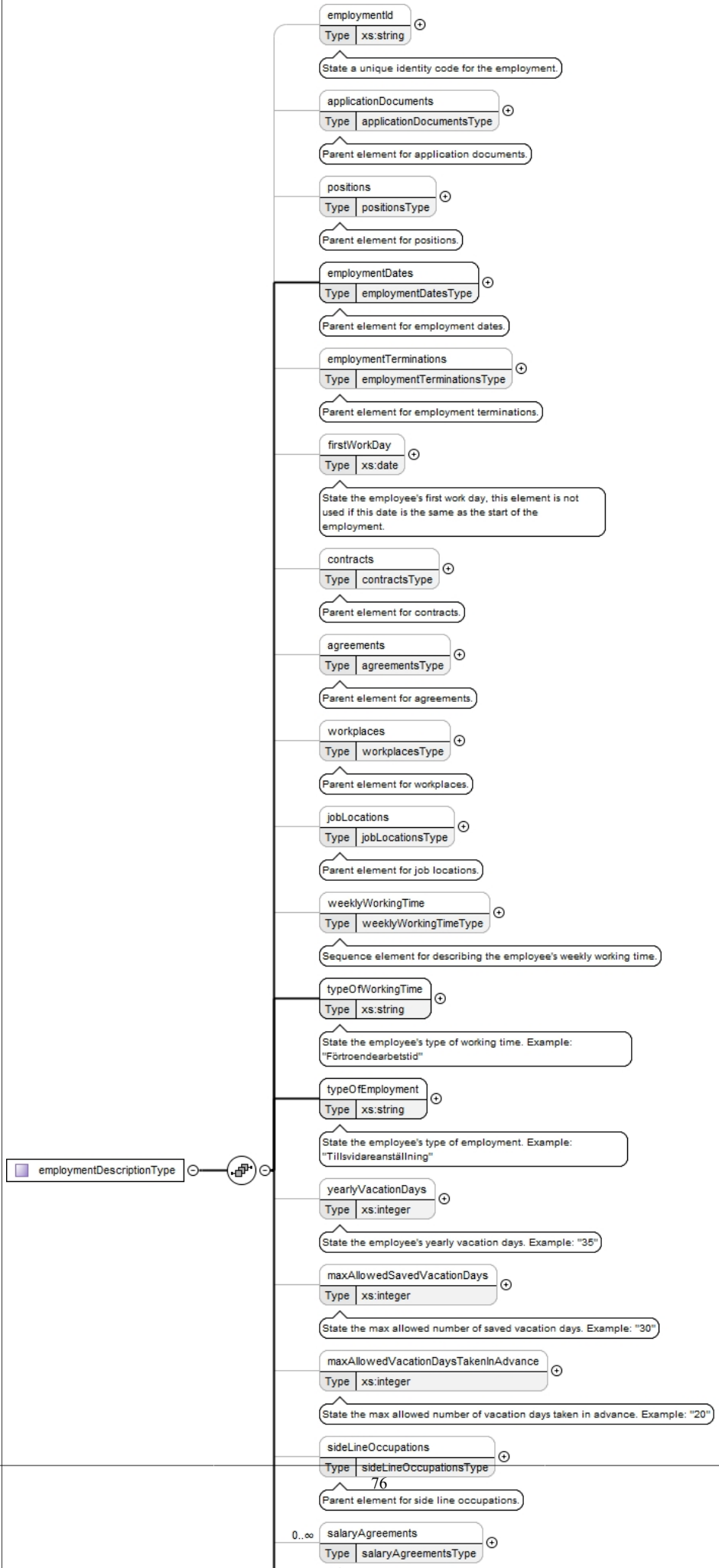
```
<xs:complexType name="eSignatureComplexType">
  <xs:annotation>
    <xs:documentation xml:lang="en">E-signature information with an attribute indicating it has been
 present and the option to insert as XML following an e-signature XML-schema.</xs:documentation>
  </xs:annotation>
```

```
  <xs:sequence>
    <xs:element name="signature" type="ExtendingComplexType" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="present" type="xs:boolean" use="required">
    <xs:annotation>
      <xs:documentation xml:lang="en">Attribute indicating that an e-signature has been present or
 not.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="dateESignatureIsVerified" type="xs:dateTime" use="optional">
    <xs:annotation>
      <xs:documentation xml:lang="en">Attribute with the datetime giving when the e-signature was
 verified.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type `employmentDescriptionType`

Namespace	employment

Diagram


employmentDescriptionType
employmentId
Type xs:string
State a unique identity code for the employment.
applicationDocuments
Type applicationDocumentsType
Parent element for application documents.
positions
Type positionsType
Parent element for positions.
employmentDates
Type employmentDatesType
Parent element for employment dates.
employmentTerminations
Type employmentTerminationsType
Parent element for employment terminations.
firstWorkDay
Type xs:date
State the employee's first work day, this element is not used if this date is the same as the start of the employment.
contracts
Type contractsType
Parent element for contracts.
agreements
Type agreementsType
Parent element for agreements.
workplaces
Type workplacesType
Parent element for workplaces.
jobLocations
Type jobLocationsType
Parent element for job locations.
weeklyWorkingTime
Type weeklyWorkingTimeType
Sequence element for describing the employee's weekly working time.
typeOfWorkingTime
Type xs:string
State the employee's type of working time. Example: "Förtroendearbetstid"
typeOfEmployment
Type xs:string
State the employee's type of employment. Example: "Tillsvidareanställning"
yearlyVacationDays
Type xs:integer
State the employee's yearly vacation days. Example: "35"
maxAllowedSavedVacationDays
Type xs:integer
State the max allowed number of saved vacation days. Example: "30"
maxAllowedVacationDaysTakenInAdvance
Type xs:integer
State the max allowed number of vacation days taken in advance. Example: "20"
sideLineOccupations
Type sideLineOccupationsType
Parent element for side line occupations.
0..∞
salaryAgreements
Type salaryAgreementsType

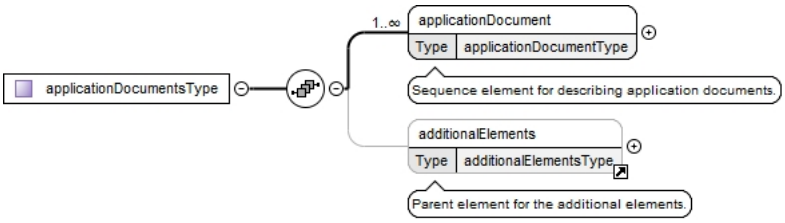
Used by	Element employmentType/employmentDescription
Model	employmentId{0,1} , applicationDocuments{0,1} , positions{0,1} , employmentDates , employmentTerminations{0,1} , firstWorkDay{0,1} , contracts{0,1} , agreements{0,1} , workplaces{0,1} , jobLocations{0,1} , weeklyWorkingTime{0,1} , typeOfWorkingTime , typeOfEmployment , yearlyVacationDays{0,1} , maxAllowedSavedVacationDays{0,1} , maxAllowedVacationDaysTakenInAdvance{0,1} , sideLineOccupations{0,1} , salaryAgreements* , fringeBenefits* , paydays{0,1} , retirementRegulations{0,1} , retirementAge{0,1} , otherTerms* , testimonials{0,1} , references{0,1} , additionalElements{0,1} , relations
Children	additionalElements, agreements, applicationDocuments, contracts, employmentDates, employmentId, employmentTerminations, firstWorkDay, fringeBenefits, jobLocations, maxAllowedSavedVacationDays, maxAllowedVacationDaysTakenInAdvance, otherTerms, paydays, positions, references, relations, retirementAge, retirementRegulations, salaryAgreements, sideLineOccupations, testimonials, typeOfEmployment, typeOfWorkingTime, weeklyWorkingTime, workplaces, yearlyVacationDays

Source

```
<xs:complexType name="employmentDescriptionType">
  <xs:sequence>
    <xs:element name="employmentId" minOccurs="0" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">State a unique identity code for the employment.</
xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="applicationDocuments" type="applicationDocumentsType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for application documents.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="positions" type="positionsType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for positions.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="employmentDates" type="employmentDatesType">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for employment dates.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="employmentTerminations" type="employmentTerminationsType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for employment terminations.</
xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="firstWorkDay" minOccurs="0" type="xs:date">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the employee's first work day, this element is not
 used if this date is the same as the start of the employment.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="contracts" type="contractsType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for contracts.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="agreements" type="agreementsType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for agreements.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="workplaces" type="workplacesType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for workplaces.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="jobLocations" type="jobLocationsType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for job locations.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="weeklyWorkingTime" type="weeklyWorkingTimeType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Sequence element for describing the employee's weekly
 working time.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="typeOfWorkingTime" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the employee's type of working time. Example:
 "Förtroendearbetstid"</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="typeOfEmployment" type="xs:string">
      <xs:annotation>
```

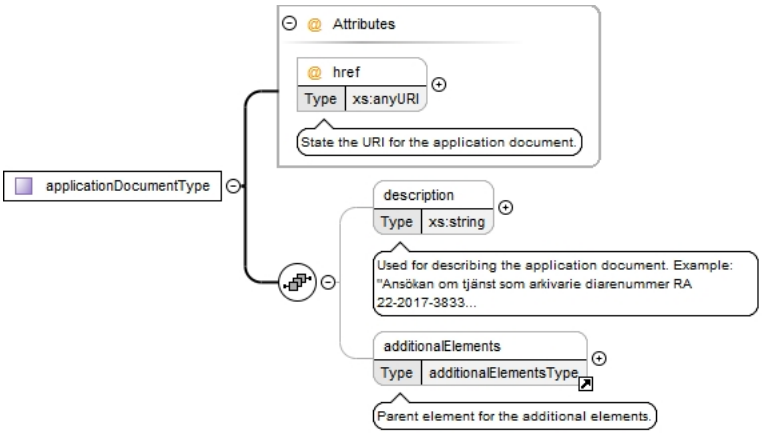
```
        <xs:documentation xml:lang="en">State the employee's type of employment. Example:
 "Tillsvidareanställning"</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="yearlyVacationDays" minOccurs="0" type="xs:integer">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the employee's yearly vacation days. Example: "35"</
xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="maxAllowedSavedVacationDays" minOccurs="0" type="xs:integer">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the max allowed number of saved vacation days.
 Example: "30"</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="maxAllowedVacationDaysTakenInAdvance" minOccurs="0" type="xs:integer">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the max allowed number of vacation days taken in
 advance. Example: "20"</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="sideLineOccupations" type="sideLineOccupationsType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for side line occupations.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="salaryAgreements" type="salaryAgreementsType" minOccurs="0"
 maxOccurs="unbounded">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for salary agreements.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="fringeBenefits" type="fringeBenefitsType" minOccurs="0" maxOccurs="unbounded">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for fringe benefits.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="paydays" type="paydaysType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for paydays.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="retirementRegulations" type="retirementRegulationsType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for retirement regulations.</
xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="retirementAge" minOccurs="0" type="xs:integer">
      <xs:annotation>
        <xs:documentation xml:lang="en">State from what age the employee is eligible for
 retirement.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="otherTerms" minOccurs="0" maxOccurs="unbounded" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for other terms.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="testimonials" type="testimonialsType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for testimonials.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="references" type="referencesType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Parent element for references.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element ref="additionalElements" minOccurs="0"/>
    <xs:element name="relations" type="relationsType">
      <xs:annotation>
        <xs:documentation xml:lang="en">Sequence element for describing relations to parent and
 child documents.</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

Complex Type `applicationDocumentsType`

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/applicationDocuments
Model	applicationDocument+ , additionalElements{0,1}
Children	additionalElements, applicationDocument

Source	`<xs:complexType name="applicationDocumentsType">`
	`  <xs:sequence>`
	`  <xs:element name="applicationDocument" type="applicationDocumentType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing application documents.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element ref="additionalElements" minOccurs="0"/>`
	`  </xs:sequence>`
	`</xs:complexType>`

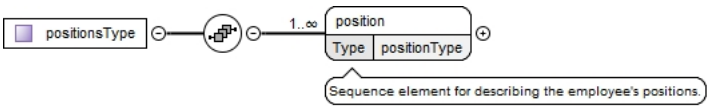
Complex Type `applicationDocumentType`

Namespace	employment
Diagram	
Used by	Element applicationDocumentsType/applicationDocument
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description
Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional `State the URI for the application document.`

Source	`<xs:complexType name="applicationDocumentType">`
	`  <xs:sequence>`
	`  <xs:element name="description" type="xs:string" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for describing the application document. Example: "Ansökan om tjänst som arkivarie diarenummer RA 22-2017-3833 Anna Andersson"</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element ref="additionalElements" minOccurs="0"/>`
	`  </xs:sequence>`
	`  <xs:attribute name="href" type="xs:anyURI" use="optional">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI for the application document.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:attribute>`

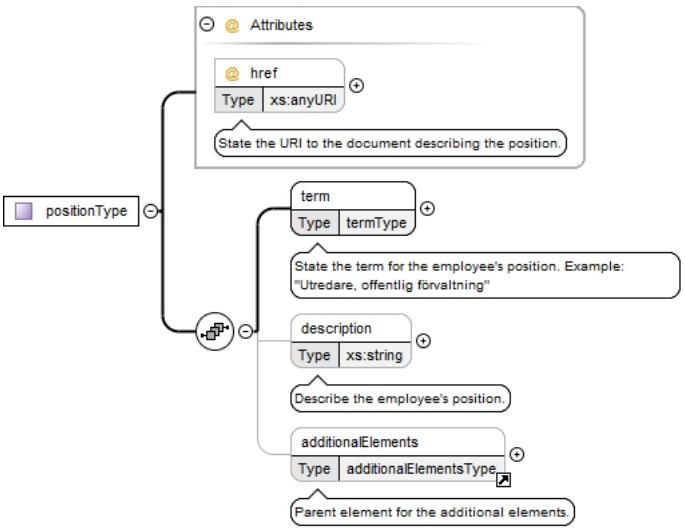
```
</xs:complexType>
```

Complex Type positionsType

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/positions
Model	position+
Children	position

Source	`<xs:complexType name="positionsType">`
	`  <xs:sequence>`
	`  <xs:element name="position" type="positionType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing the employee's positions.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

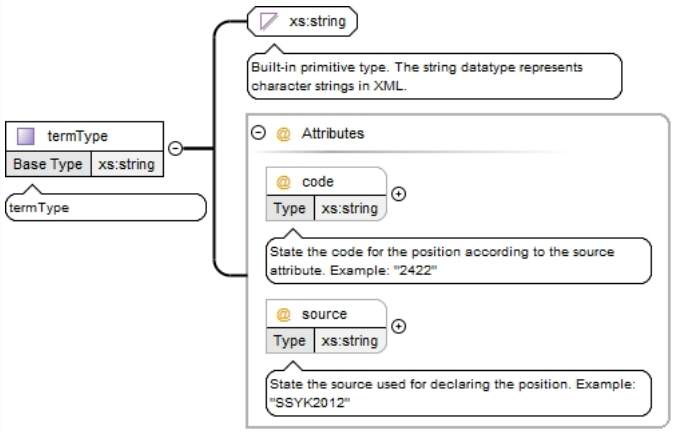
Complex Type positionType

Namespace	employment
Diagram	
Used by	Element positionsType/position
Model	term , description{0,1} , additionalElements{0,1}
Children	additionalElements, description, term
Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional State the URI to the document describing the position.

Source	`<xs:complexType name="positionType">`
	`  <xs:sequence>`
	`  <xs:element name="term" type="termType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the term for the employee's position. Example: "Utredare, offentlig förvaltning"</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element name="description" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the employee's position.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element ref="additionalElements" minOccurs="0"/>`

```
  </xs:sequence>
  <xs:attribute name="href" use="optional" type="xs:anyURI">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the URI to the document describing the position.</
xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type `termType`

Namespace	employment
Annotations	`termType`
Diagram	
Type	extension of xs:string
Used by	Element positionType/term

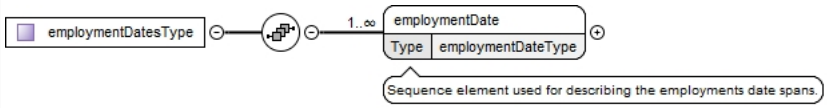
Attributes

QName	Type	Use	
code	xs:string	optional	
	State the code for the position according to the source attribute. Example: "2422"		
source	xs:string	optional	
	State the source used for declaring the position. Example: "SSYK2012"		

Source

```
<xs:complexType name="termType">
  <xs:annotation>
    <xs:documentation xml:lang="en">termType</xs:documentation>
  </xs:annotation>
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="code" type="xs:string" use="optional">
        <xs:annotation>
          <xs:documentation xml:lang="en">State the code for the position according to the source
 attribute. Example: "2422"</xs:documentation>
        </xs:annotation>
      </xs:attribute>
      <xs:attribute name="source" type="xs:string" use="optional">
        <xs:annotation>
          <xs:documentation xml:lang="en">State the source used for declaring the position. Example:
 "SSYK2012"</xs:documentation>
        </xs:annotation>
      </xs:attribute>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

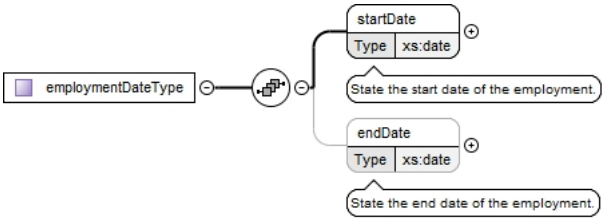
Complex Type `employmentDatesType`

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/employmentDates

Model	employmentDate+
Children	employmentDate

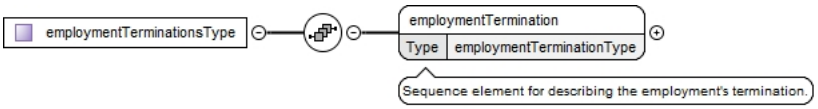
Source	`<xs:complexType name="employmentDatesType">`
	`  <xs:sequence>`
	`  <xs:element name="employmentDate" type="employmentDateType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element used for describing the employments date spans.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

Complex Type `employmentDateType`

Namespace	employment
Diagram	
Used by	Element employmentDatesType/employmentDate
Model	startDate , endDate{0,1}
Children	endDate, startDate

Source	`<xs:complexType name="employmentDateType">`
	`  <xs:sequence>`
	`  <xs:element name="startDate" type="xs:date">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the start date of the employment.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element name="endDate" minOccurs="0" type="xs:date">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the end date of the employment.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

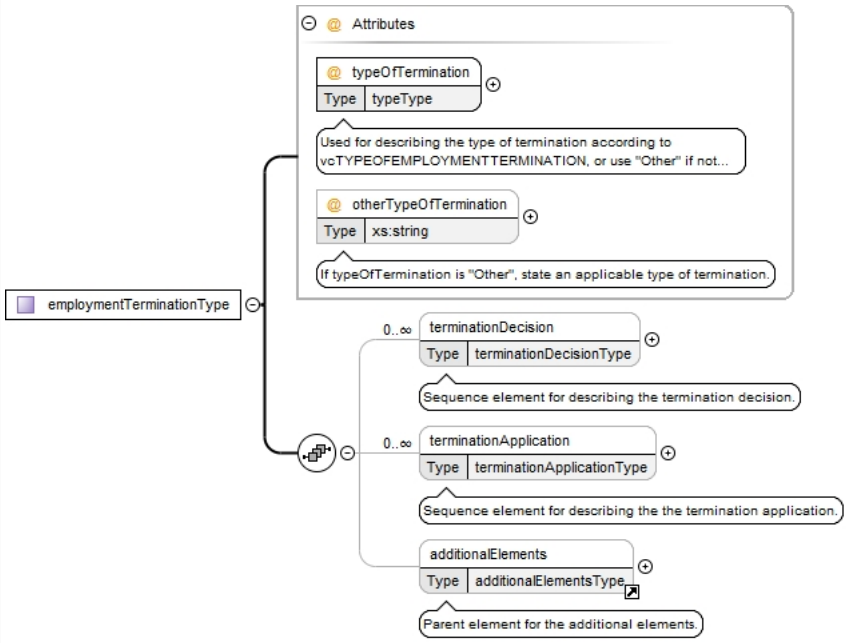
Complex Type `employmentTerminationsType`

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/employmentTerminations
Model	employmentTermination
Children	employmentTermination

Source	`<xs:complexType name="employmentTerminationsType">`
	`  <xs:sequence>`
	`  <xs:element name="employmentTermination" type="employmentTerminationType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing the employment's termination.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

Complex Type `employmentTerminationType`

Namespace	employment

Diagram	
Used by	Element employmentTerminationsType/employmentTermination
Model	terminationDecision* , terminationApplication* , additionalElements{0,1}
Children	additionalElements, terminationApplication, terminationDecision

Attributes

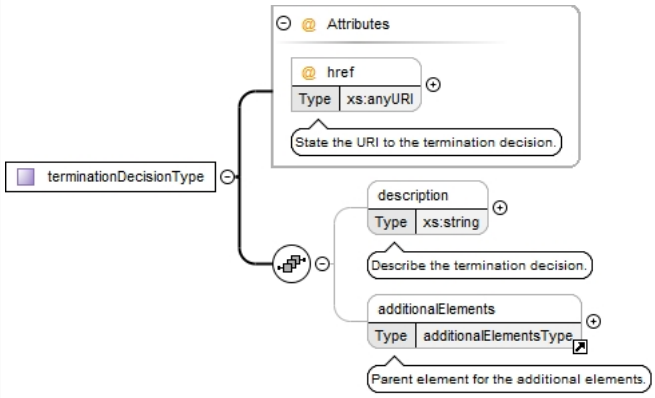
QName	Type	Use	
otherTypeOfTermination	xs:string	optional	
	`If typeOfTermination is "Other", state an applicable type of termination.`		
typeOfTermination	typeType	required	
	`Used for describing the type of termination according to vcTYPEOFEMPLOYMENTTERMINATION, or use "Other" if not applicable. Example: "LackOfWork"`		

Source

```
<xs:complexType name="employmentTerminationType">
  <xs:sequence>
    <xs:element name="terminationDecision" minOccurs="0" maxOccurs="unbounded"
 type="terminationDecisionType">
      <xs:annotation>
        <xs:documentation xml:lang="en">Sequence element for describing the termination decision.</
xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="terminationApplication" type="terminationApplicationType" minOccurs="0"
 maxOccurs="unbounded">
      <xs:annotation>
        <xs:documentation xml:lang="en">Sequence element for describing the the termination
 application.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element ref="additionalElements" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="typeOfTermination" use="required" type="typeType">
    <xs:annotation>
      <xs:documentation xml:lang="en">Used for describing the type of termination according
 to vcTYPEOFEMPLOYMENTTERMINATION, or use "Other" if not applicable. Example: "LackOfWork"</
xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="otherTypeOfTermination" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">If typeOfTermination is "Other", state an applicable type of
 termination.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

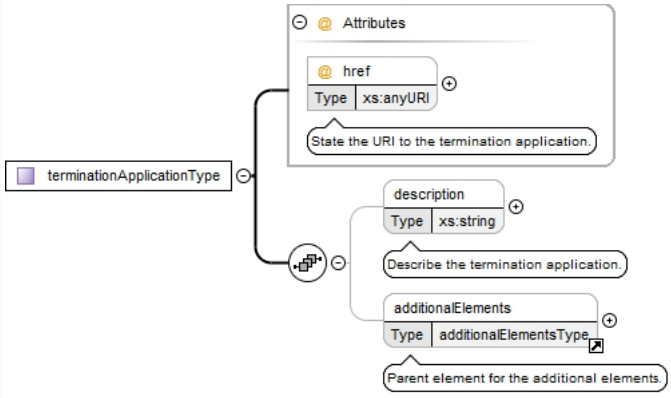
Complex Type `terminationDecisionType`

Namespace	employment

Diagram	
Used by	Element employmentTerminationType/terminationDecision
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description
Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional State the URI to the termination decision.

Source	`<xs:complexType name="terminationDecisionType">`
	`  <xs:sequence>`
	`  <xs:element name="description" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the termination decision.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element ref="additionalElements" minOccurs="0"/>`
	`  </xs:sequence>`
	`  <xs:attribute name="href" use="optional" type="xs:anyURI">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI to the termination decision.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:attribute>`
	`</xs:complexType>`

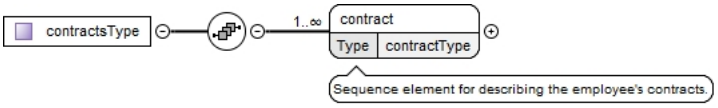
Complex Type `terminationApplicationType`

Namespace	employment
Diagram	
Used by	Element employmentTerminationType/terminationApplication
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description
Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional State the URI to the termination application.

Source	`<xs:complexType name="terminationApplicationType">`
	`  <xs:sequence>`
	`  <xs:element name="description" minOccurs="0" type="xs:string">`

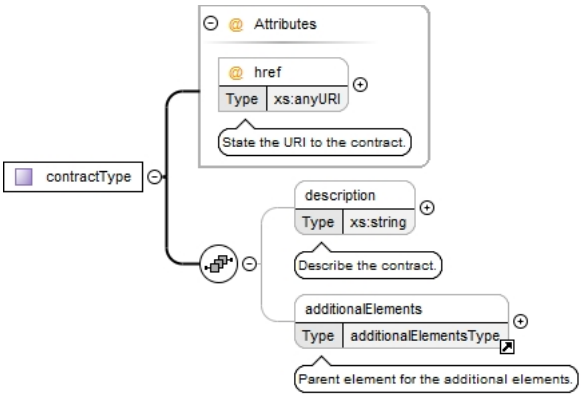
```
      <xs:annotation>
        <xs:documentation xml:lang="en">Describe the termination application.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element ref="additionalElements" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="href" type="xs:anyURI" use="optional">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the URI to the termination application.</
xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type `contractsType`

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/contracts
Model	contract+
Children	contract

Source	`<xs:complexType name="contractsType">`
	`  <xs:sequence>`
	`  <xs:element name="contract" type="contractType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing the employee's contracts.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

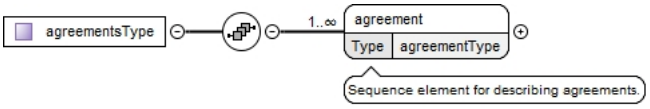
Complex Type `contractType`

Namespace	employment
Diagram	
Used by	Element contractsType/contract
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description
Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional `State the URI to the contract.`

Source	`<xs:complexType name="contractType">`
	`  <xs:sequence>`
	`  <xs:element name="description" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the contract.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`

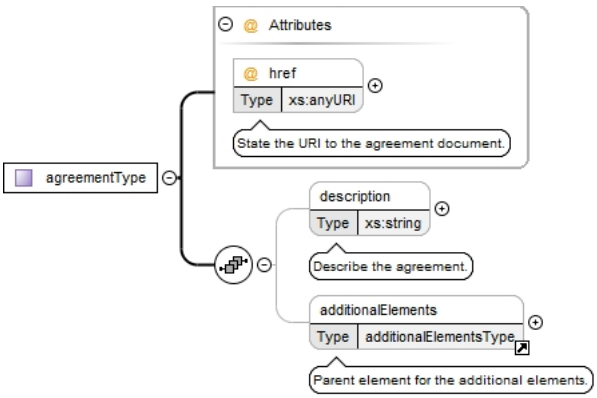
```
    <xs:element ref="additionalElements" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="href" use="optional" type="xs:anyURI">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the URI to the contract.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type `agreementsType`

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/agreements
Model	agreement+
Children	agreement

Source	`<xs:complexType name="agreementsType">`
	`  <xs:sequence>`
	`  <xs:element name="agreement" type="agreementType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing agreements.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

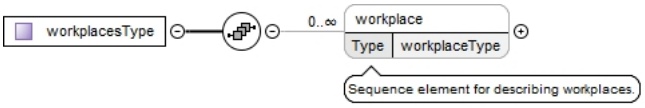
Complex Type `agreementType`

Namespace	employment
Diagram	
Used by	Element agreementsType/agreement
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description
Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional State the URI to the agreement document.

Source	`<xs:complexType name="agreementType">`
	`  <xs:sequence>`
	`  <xs:element name="description" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the agreement.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element ref="additionalElements" minOccurs="0"/>`
	`  </xs:sequence>`
	`  <xs:attribute name="href" use="optional" type="xs:anyURI">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI to the agreement document.</xs:documentation>`
	`  </xs:annotation>`

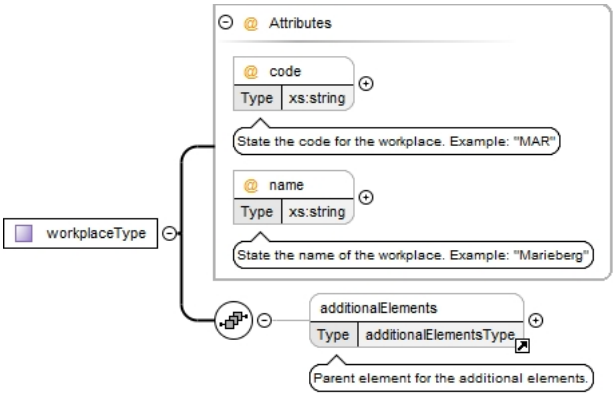
```
  </xs:attribute>
</xs:complexType>
```

Complex Type workplacesType

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/workplaces
Model	workplace*
Children	workplace

Source	`<xs:complexType name="workplacesType">`
	`  <xs:sequence>`
	`  <xs:element name="workplace" type="workplaceType" minOccurs="0" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing workplaces.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

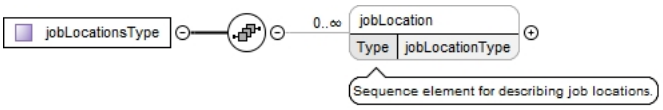
Complex Type workplaceType

Namespace	employment
Diagram	
Used by	Element workplacesType/workplace
Model	additionalElements{0,1}
Children	additionalElements
Attributes	**QName** / **Type** / **Use** **code** / xs:string / optional State the code for the workplace. Example: "MAR" **name** / xs:string / optional State the name of the workplace. Example: "Marieberg"

Source	`<xs:complexType name="workplaceType">`
	`  <xs:sequence>`
	`  <xs:element ref="additionalElements" minOccurs="0"/>`
	`  </xs:sequence>`
	`  <xs:attribute name="code" use="optional" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the code for the workplace. Example: "MAR"</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:attribute>`
	`  <xs:attribute name="name" use="optional" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the name of the workplace. Example: "Marieberg"</xs:documentation>`
	`  </xs:annotation>`

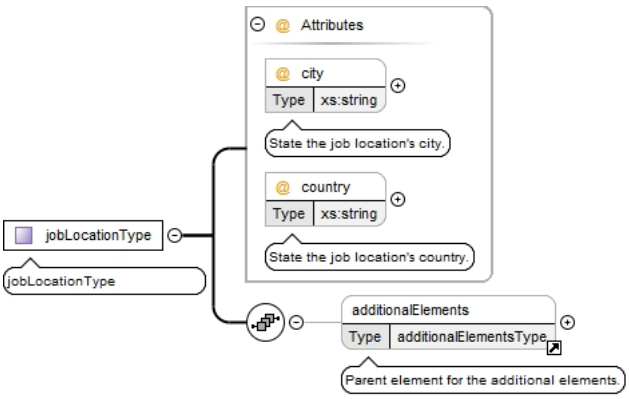
```
  </xs:attribute>
</xs:complexType>
```

Complex Type `jobLocationsType`

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/jobLocations
Model	jobLocation*
Children	jobLocation

Source	`<xs:complexType name="jobLocationsType">`
	`  <xs:sequence>`
	`  <xs:element name="jobLocation" type="jobLocationType" minOccurs="0" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation>Sequence element for describing job locations.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

Complex Type `jobLocationType`

Namespace	employment
Annotations	`jobLocationType`
Diagram	
Used by	Element jobLocationsType/jobLocation
Model	additionalElements{0,1}
Children	additionalElements
Attributes	

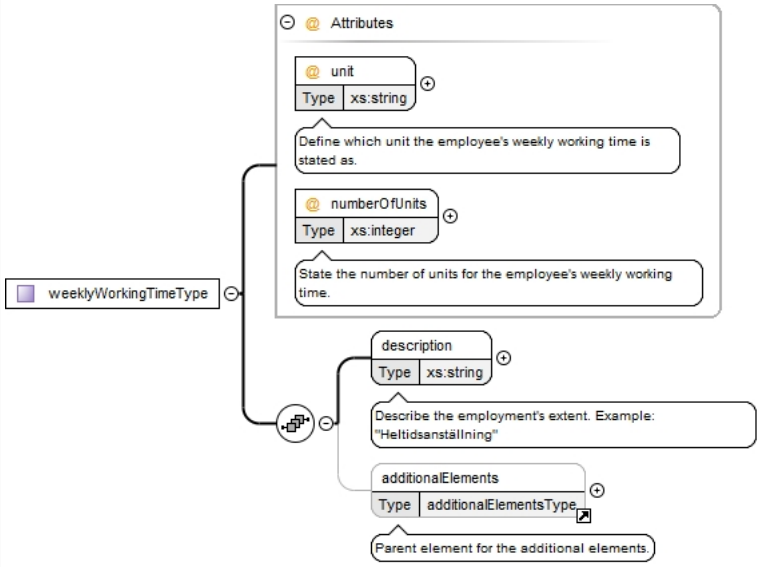
QName	Type	Use	
city	xs:string	optional	
	State the job location's city.		
country	xs:string	optional	
	State the job location's country.		

Source

```
<xs:complexType name="jobLocationType">
  <xs:annotation>
    <xs:documentation xml:lang="en">jobLocationType</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element ref="additionalElements" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="city" use="optional" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the job location's city.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="country" use="optional" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the job location's country.</xs:documentation>
```

```
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type `weeklyWorkingTimeType`

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/weeklyWorkingTime
Model	description , additionalElements{0,1}
Children	additionalElements, description

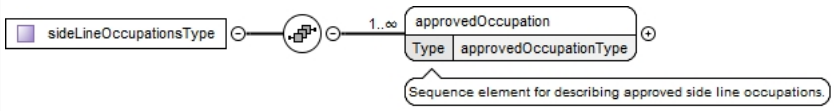
Attributes

QName	Type	Use	
numberOfUnits	xs:integer	required	
	State the number of units for the employee's weekly working time.		
unit	xs:string	required	
	Define which unit the employee's weekly working time is stated as.		

Source

```
<xs:complexType name="weeklyWorkingTimeType">
  <xs:sequence>
    <xs:element name="description" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">Describe the employment's extent. Example:
 "Heltidsanställning"</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element ref="additionalElements" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="unit" use="required" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">Define which unit the employee's weekly working time is stated
 as.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="numberOfUnits" use="required" type="xs:integer">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the number of units for the employee's weekly working
 time.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

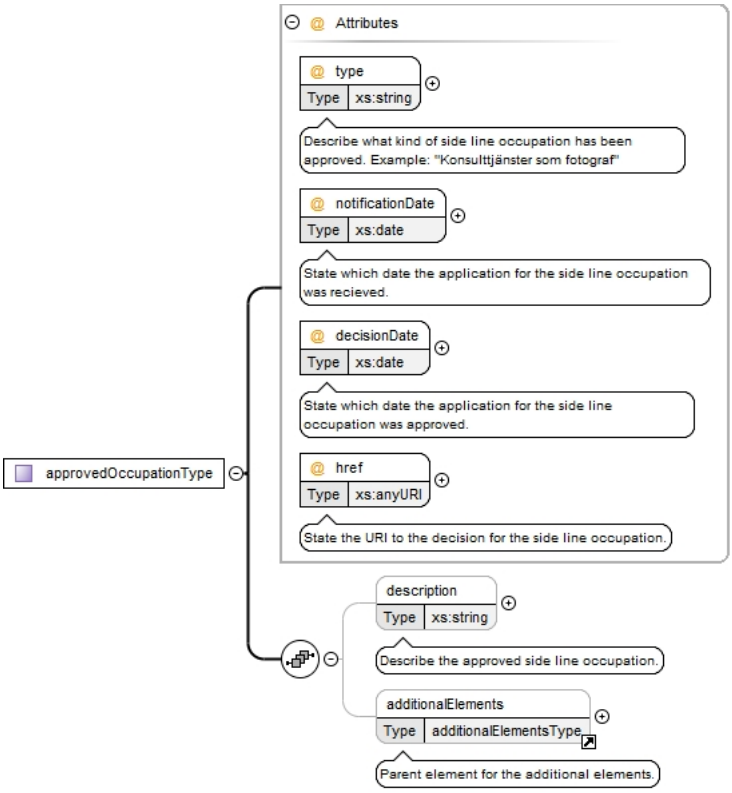
Complex Type `sideLineOccupationsType`

Namespace	employment
Diagram	

Used by	Element employmentDescriptionType/sideLineOccupations
Model	approvedOccupation+
Children	approvedOccupation

Source	`<xs:complexType name="sideLineOccupationsType">`
	`  <xs:sequence>`
	`  <xs:element name="approvedOccupation" type="approvedOccupationType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing approved side line occupations.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

Complex Type `approvedOccupationType`

Namespace	employment
Diagram	
Used by	Element sideLineOccupationsType/approvedOccupation
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description

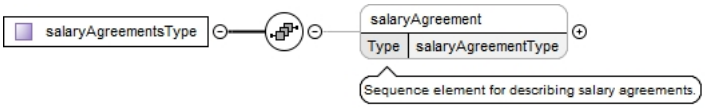
Attributes

QName	Type	Use	
decisionDate	xs:date	required	
	State which date the application for the side line occupation was approved.		
href	xs:anyURI	required	
	State the URI to the decision for the side line occupation.		
notificationDate	xs:date	required	
	State which date the application for the side line occupation was recieved.		
type	xs:string	required	
	Describe what kind of side line occupation has been approved. Example: "Konsulttjänster som fotograf"		

Source	`<xs:complexType name="approvedOccupationType">`

```
  <xs:sequence>
    <xs:element name="description" minOccurs="0" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">Describe the approved side line occupation.</
xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element ref="additionalElements" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="type" use="required" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">Describe what kind of side line occupation has been approved.
 Example: "Konsulttjänster som fotograf"</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="notificationDate" use="required" type="xs:date">
    <xs:annotation>
      <xs:documentation xml:lang="en">State which date the application for the side line occupation
 was recieved.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="decisionDate" use="required" type="xs:date">
    <xs:annotation>
      <xs:documentation xml:lang="en">State which date the application for the side line occupation
 was approved.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="href" use="required" type="xs:anyURI">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the URI to the decision for the side line occupation.</
xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type `salaryAgreementsType`

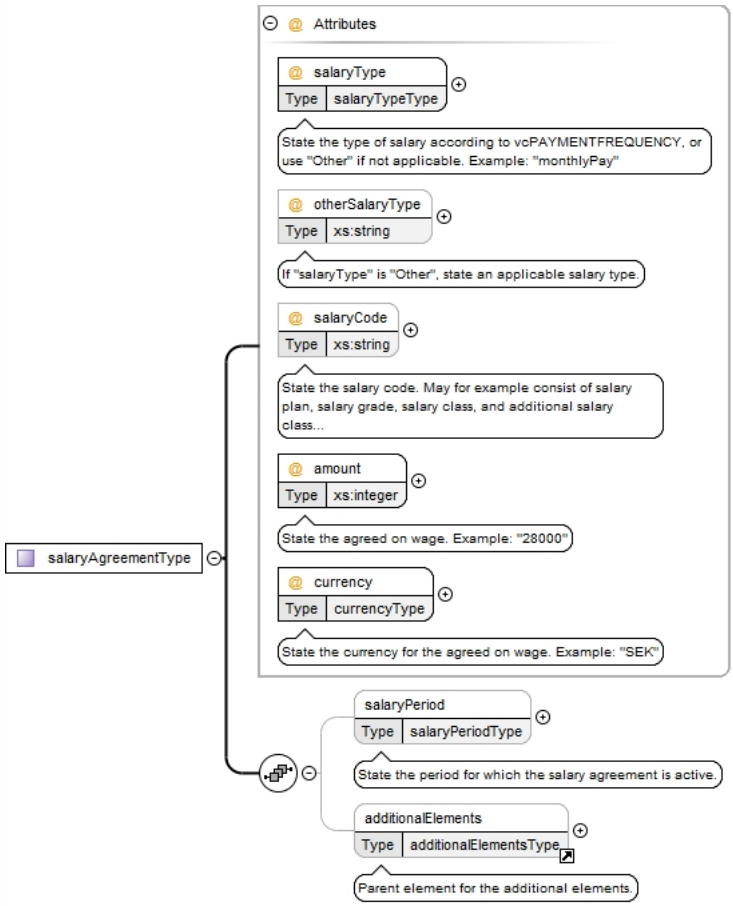
Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/salaryAgreements
Model	salaryAgreement{0,1}
Children	salaryAgreement

Source	`<xs:complexType name="salaryAgreementsType">`
	`  <xs:sequence>`
	`  <xs:element name="salaryAgreement" type="salaryAgreementType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing salary agreements.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

Complex Type `salaryAgreementType`

Namespace	employment

Diagram



Used by

Element salaryAgreementsType/salaryAgreement

Model

salaryPeriod{0,1} , additionalElements{0,1}

Children

additionalElements, salaryPeriod

Attributes

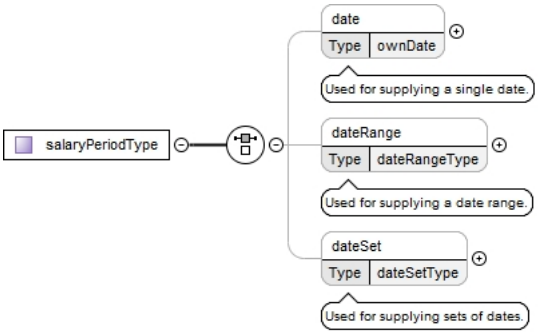
QName	Type	Use	
amount	xs:integer	required	
	State the agreed on wage. Example: "28000"		
currency	currencyType	required	
	State the currency for the agreed on wage. Example: "SEK"		
otherSalaryType	xs:string	optional	
	If "salaryType" is "Other", state an applicable salary type.		
salaryCode	xs:string	optional	
	State the salary code. May for example consist of salary plan, salary grade, salary class, and additional salary class separated by ":". Example: "KA"		
salaryType	salaryTypeType	required	
	State the type of salary according to vcPAYMENTFREQUENCY, or use "Other" if not applicable. Example: "monthlyPay"		

Source

```
<xs:complexType name="salaryAgreementType">
  <xs:sequence>
    <xs:element name="salaryPeriod" type="salaryPeriodType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the period for which the salary agreement is active.</
xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element ref="additionalElements" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="salaryType" use="required" type="salaryTypeType">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the type of salary according to vcPAYMENTFREQUENCY, or
 use "Other" if not applicable. Example: "monthlyPay"</xs:documentation>
```

```
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="otherSalaryType" use="optional" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">If "salaryType" is "Other", state an applicable salary type.</
xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="salaryCode" use="optional" type="xs:string">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the salary code. May for example consist of salary
 plan, salary grade, salary class, and additional salary class separated by ":". Example: "KA"</
xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="amount" use="required" type="xs:integer">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the agreed on wage. Example: "28000"</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="currency" use="required" type="currencyType">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the currency for the agreed on wage. Example: "SEK"</
xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type `salaryPeriodType`

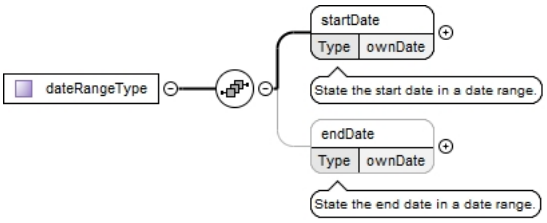
Namespace	employment
Diagram	
Used by	Element salaryAgreementType/salaryPeriod
Model	date{0,1} | dateRange{0,1} | dateSet{0,1}
Children	date, dateRange, dateSet

Source

```
<xs:complexType name="salaryPeriodType">
  <xs:choice>
    <xs:element name="date" minOccurs="0" type="ownDate">
      <xs:annotation>
        <xs:documentation xml:lang="en">Used for supplying a single date.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="dateRange" type="dateRangeType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Used for supplying a date range.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="dateSet" type="dateSetType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">Used for supplying sets of dates.</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:choice>
</xs:complexType>
```

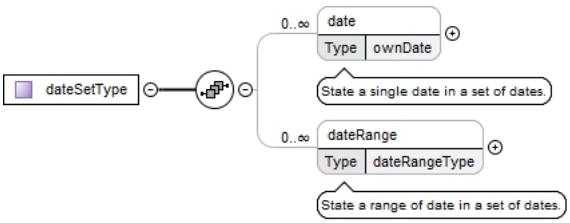
Complex Type `dateRangeType`

Namespace	employment

Diagram	
Used by	Elements dateSetType/dateRange, fringeBenefitPeriodType/dateRange, salaryPeriodType/dateRange
Model	startDate , endDate{0,1}
Children	endDate, startDate

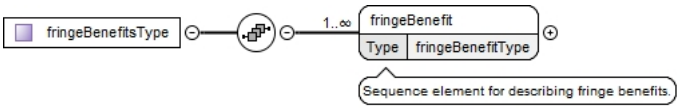
Source	`<xs:complexType name="dateRangeType">`
	`  <xs:sequence>`
	`  <xs:element name="startDate" type="ownDate">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the start date in a date range.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element name="endDate" minOccurs="0" type="ownDate">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the end date in a date range.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

Complex Type `dateSetType`

Namespace	employment
Diagram	
Used by	Elements fringeBenefitPeriodType/dateSet, salaryPeriodType/dateSet
Model	date* , dateRange*
Children	date, dateRange

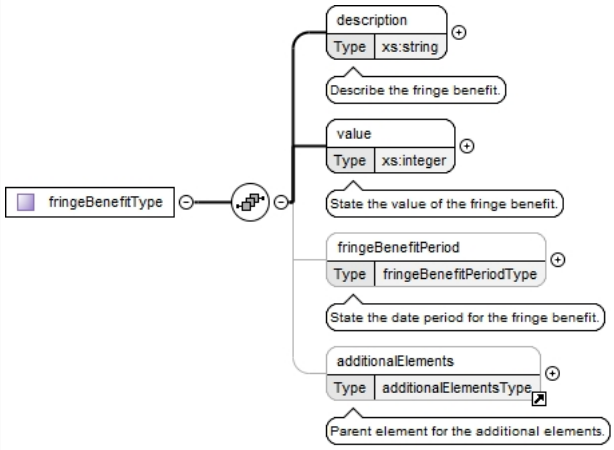
Source	`<xs:complexType name="dateSetType">`
	`  <xs:sequence>`
	`  <xs:element name="date" minOccurs="0" maxOccurs="unbounded" type="ownDate">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State a single date in a set of dates.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element name="dateRange" type="dateRangeType" minOccurs="0" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State a range of date in a set of dates.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

Complex Type `fringeBenefitsType`

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/fringeBenefits
Model	fringeBenefit+
Children	fringeBenefit
Source	`<xs:complexType name="fringeBenefitsType">`

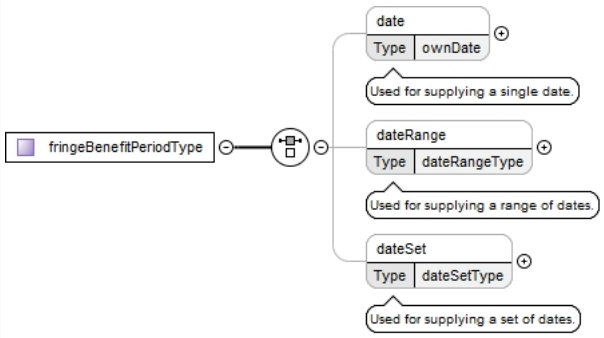
```
  <xs:sequence>
    <xs:element name="fringeBenefit" type="fringeBenefitType" maxOccurs="unbounded">
      <xs:annotation>
        <xs:documentation xml:lang="en">Sequence element for describing fringe benefits.</
xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

Complex Type `fringeBenefitType`

Namespace	employment
Diagram	
Used by	Element fringeBenefitsType/fringeBenefit
Model	description , value , fringeBenefitPeriod{0,1} , additionalElements{0,1}
Children	additionalElements, description, fringeBenefitPeriod, value
Source	

```
<xs:complexType name="fringeBenefitType">
  <xs:sequence>
    <xs:element name="description" type="xs:string">
      <xs:annotation>
        <xs:documentation xml:lang="en">Describe the fringe benefit.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="value" type="xs:integer">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the value of the fringe benefit.</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="fringeBenefitPeriod" type="fringeBenefitPeriodType" minOccurs="0">
      <xs:annotation>
        <xs:documentation xml:lang="en">State the date period for the fringe benefit.</
xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element ref="additionalElements" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
```

Complex Type `fringeBenefitPeriodType`

Namespace	employment
Diagram	

Used by	Element fringeBenefitType/fringeBenefitPeriod
Model	date{0,1} | dateRange{0,1} | dateSet{0,1}
Children	date, dateRange, dateSet

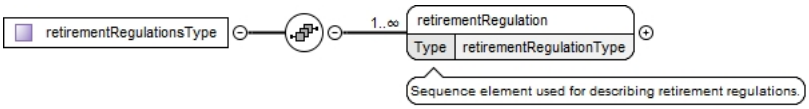
Source	`<xs:complexType name="fringeBenefitPeriodType">`
	`  <xs:choice>`
	`  <xs:element name="date" minOccurs="0" type="ownDate">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for supplying a single date.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element name="dateRange" type="dateRangeType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for supplying a range of dates.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element name="dateSet" type="dateSetType" minOccurs="0">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Used for supplying a set of dates.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:choice>`
	`</xs:complexType>`

Complex Type paydaysType

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/paydays
Model	payday+
Children	payday

Source	`<xs:complexType name="paydaysType">`
	`  <xs:sequence>`
	`  <xs:element name="payday" minOccurs="1" maxOccurs="unbounded" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the employee's payday.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

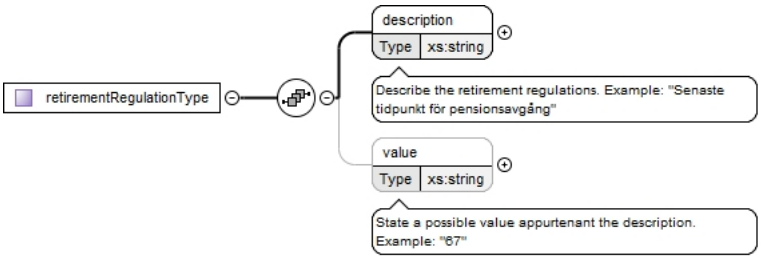
Complex Type retirementRegulationsType

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/retirementRegulations
Model	retirementRegulation+
Children	retirementRegulation

Source	`<xs:complexType name="retirementRegulationsType">`
	`  <xs:sequence>`
	`  <xs:element name="retirementRegulation" type="retirementRegulationType" minOccurs="1"`
	` maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element used for describing retirement`
	` regulations.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

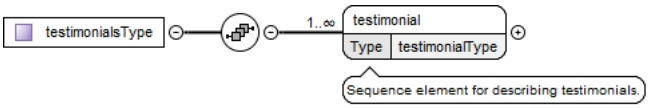
Complex Type retirementRegulationType

Namespace	employment

Diagram	
Used by	Element retirementRegulationsType/retirementRegulation
Model	description , value{0,1}
Children	description, value

Source	`<xs:complexType name="retirementRegulationType">`
	`  <xs:sequence>`
	`  <xs:element name="description" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the retirement regulations. Example: "Senaste tidpunkt för pensionsavgång"</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element name="value" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State a possible value appurtenant the description. Example: "67"</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

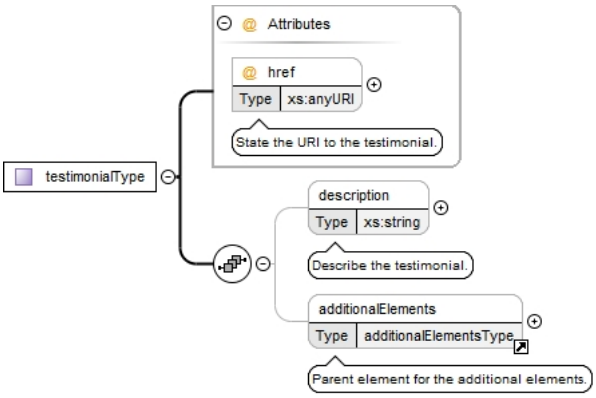
Complex Type `testimonialsType`

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/testimonials
Model	testimonial+
Children	testimonial

Source	`<xs:complexType name="testimonialsType">`
	`  <xs:sequence>`
	`  <xs:element name="testimonial" type="testimonialType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing testimonials.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

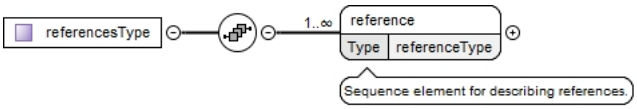
Complex Type `testimonialType`

Namespace	employment

Diagram	
Used by	Element testimonialsType/testimonial
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description
Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional `State the URI to the testimonial.`

Source	`<xs:complexType name="testimonialType">`
	`  <xs:sequence>`
	`  <xs:element name="description" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the testimonial.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element ref="additionalElements" minOccurs="0"/>`
	`  </xs:sequence>`
	`  <xs:attribute name="href" use="optional" type="xs:anyURI">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI to the testimonial.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:attribute>`
	`</xs:complexType>`

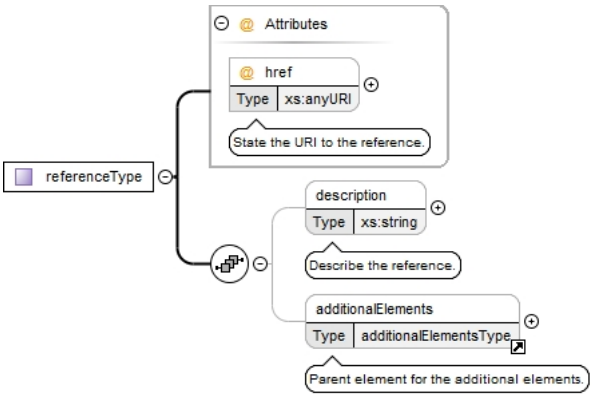
Complex Type `referencesType`

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/references
Model	reference+
Children	reference

Source	`<xs:complexType name="referencesType">`
	`  <xs:sequence>`
	`  <xs:element name="reference" type="referenceType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing references.</`
	`xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`
	`</xs:complexType>`

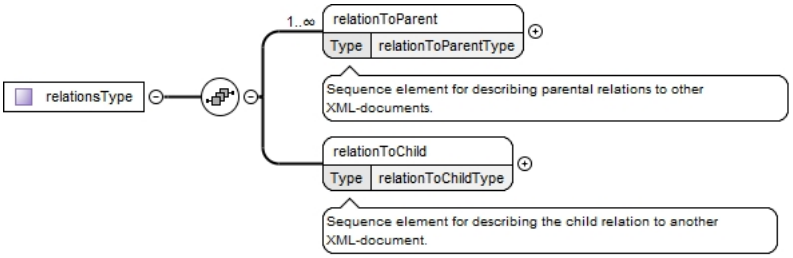
Complex Type `referenceType`

Namespace	employment

Diagram	
Used by	Element referencesType/reference
Model	description{0,1} , additionalElements{0,1}
Children	additionalElements, description
Attributes	**QName** / **Type** / **Use** **href** / xs:anyURI / optional `State the URI to the reference.`

Source	`<xs:complexType name="referenceType">`
	`  <xs:sequence>`
	`  <xs:element name="description" minOccurs="0" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe the reference.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element ref="additionalElements" minOccurs="0"/>`
	`  </xs:sequence>`
	`  <xs:attribute name="href" use="optional" type="xs:anyURI">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI to the reference.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:attribute>`
	`</xs:complexType>`

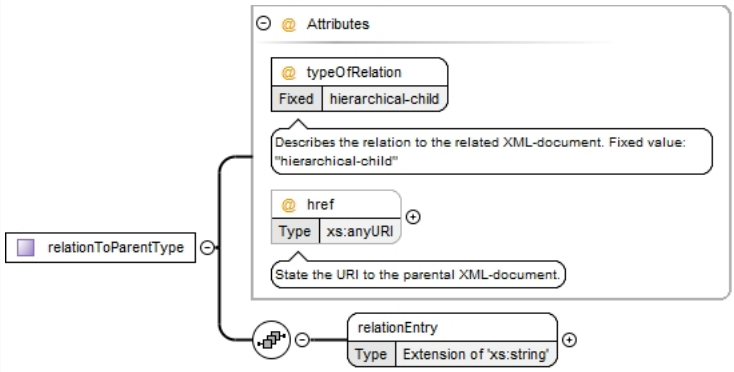
Complex Type `relationsType`

Namespace	employment
Diagram	
Used by	Element employmentDescriptionType/relations
Model	relationToParent+ , relationToChild
Children	relationToChild, relationToParent

Source	`<xs:complexType name="relationsType">`
	`  <xs:sequence>`
	`  <xs:element name="relationToParent" type="relationToParentType" maxOccurs="unbounded">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing parental relations to other`
	` XML-documents.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  <xs:element name="relationToChild" type="relationToChildType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Sequence element for describing the child relation to`
	` another XML-document.</xs:documentation>`
	`  </xs:annotation>`
	`  </xs:element>`
	`  </xs:sequence>`

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

Complex Type `relationToParentType`

Namespace	employment
Diagram	
Used by	Element relationsType/relationToParent
Model	relationEntry
Children	relationEntry

Attributes

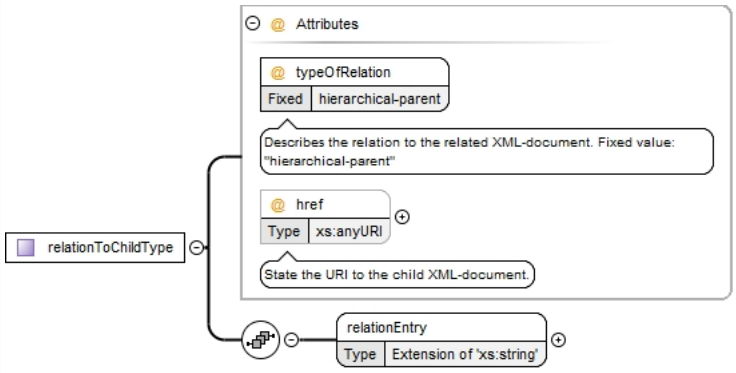
QName	Type	Fixed	Use	
href	xs:anyURI		optional	
	State the URI to the parental XML-document.			
typeOfRelation		hierarchical-child	required	
	Describes the relation to the related XML-document. Fixed value: "hierarchical-child"			

Source

```
<xs:complexType name="relationToParentType">
  <xs:sequence>
    <xs:element name="relationEntry">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="xs:string">
            <xs:attribute name="parentOrganisationReferenceCode" type="xs:string" use="required">
              <xs:annotation>
                <xs:documentation xml:lang="en">State the parent organisation's reference code.</
xs:documentation>
              </xs:annotation>
            </xs:attribute>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="typeOfRelation" use="required" fixed="hierarchical-child">
    <xs:annotation>
      <xs:documentation xml:lang="en">Describes the relation to the related XML-document. Fixed
 value: "hierarchical-child"</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="href" use="optional" type="xs:anyURI">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the URI to the parental XML-document.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type `relationToChildType`

Namespace	employment

Diagram	
Used by	Element relationsType/relationToChild
Model	relationEntry
Children	relationEntry

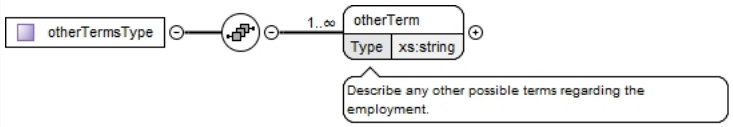
Attributes

QName	Type	Fixed	Use	
href	xs:anyURI		optional	
	State the URI to the child XML-document.			
typeOfRelation		hierarchical-parent	required	
	Describes the relation to the related XML-document. Fixed value: "hierarchical-parent"			

Source

```
<xs:complexType name="relationToChildType">
  <xs:sequence>
    <xs:element name="relationEntry">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="xs:string">
            <xs:attribute name="childOrganisationReferenceCode" type="xs:string" use="required">
              <xs:annotation>
                <xs:documentation xml:lang="en">State the child document's reference code.</
xs:documentation>
              </xs:annotation>
            </xs:attribute>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:attribute name="typeOfRelation" use="required" fixed="hierarchical-parent">
    <xs:annotation>
      <xs:documentation xml:lang="en">Describes the relation to the related XML-document. Fixed
 value: "hierarchical-parent"</xs:documentation>
    </xs:annotation>
  </xs:attribute>
  <xs:attribute name="href" use="optional" type="xs:anyURI">
    <xs:annotation>
      <xs:documentation xml:lang="en">State the URI to the child XML-document.</xs:documentation>
    </xs:annotation>
  </xs:attribute>
</xs:complexType>
```

Complex Type `otherTermsType`

Namespace	employment
Diagram	
Model	otherTerm+
Children	otherTerm

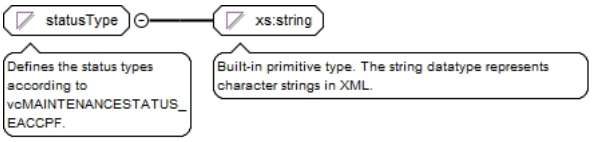
Source

```
<xs:complexType name="otherTermsType">
  <xs:sequence>
    <xs:element name="otherTerm" type="xs:string" maxOccurs="unbounded">
      <xs:annotation>
        <xs:documentation xml:lang="en">Describe any other possible terms regarding the
 employment.</xs:documentation>
```

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

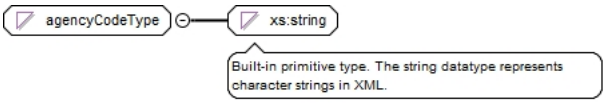
Simple Type(s)

Simple Type statusType

Namespace	employment
Annotations	Defines the status types according to vcMAINTENANCESTATUS_EACCPF.
Diagram	
Type	restriction of xs:string
Facets	enumeration: cancelled enumeration: deleted enumeration: deletedReplaced enumeration: derived enumeration: new enumeration: revised
Used by	Element: controlType/status
Source	

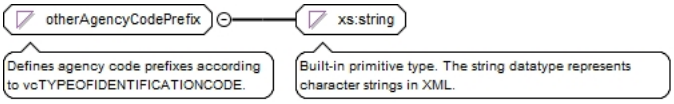
```
<xs:simpleType name="statusType">
  <xs:annotation>
    <xs:documentation xml:lang="en">Defines the status types according to
 vcMAINTENANCESTATUS_EACCPF.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="cancelled"/>
    <xs:enumeration value="deleted"/>
    <xs:enumeration value="deletedReplaced"/>
    <xs:enumeration value="derived"/>
    <xs:enumeration value="new"/>
    <xs:enumeration value="revised"/>
  </xs:restriction>
</xs:simpleType>
```

Simple Type agencyCodeType

Namespace	employment
Diagram	
Type	restriction of xs:string
Facets	pattern: [A-Z]{2}\-[a-zA-Z0-9/:\-]{1,10}
Used by	Element: maintenanceAgencyType/agencyCode
Source	

```
<xs:simpleType name="agencyCodeType">
  <xs:restriction base="xs:string">
    <xs:pattern value="[A-Z]{2}\-[a-zA-Z0-9/:\-]{1,10}"/>
  </xs:restriction>
</xs:simpleType>
```

Simple Type otherAgencyCodePrefix

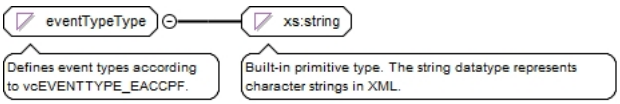
Namespace	employment
Annotations	Defines agency code prefixes according to vcTYPEOFIDENTIFICATIONCODE.
Diagram	

Type	restriction of xs:string	
Facets	enumeration	VAT
	enumeration	DUNS
	enumeration	ORG
	enumeration	HSA
	enumeration	Local
	enumeration	URI
	enumeration	PersonalIdentityNumber
	enumeration	Co-ordinationNumber
	enumeration	Other
Used by	Attribute	otherAgencyCodeType/@agencyCodePrefix

Source

```
<xs:simpleType name="otherAgencyCodePrefix">
  <xs:annotation>
    <xs:documentation xml:lang="en">Defines agency code prefixes according to
 vcTYPEOFIDENTIFICATIONCODE.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="VAT"/>
    <xs:enumeration value="DUNS"/>
    <xs:enumeration value="ORG"/>
    <xs:enumeration value="HSA"/>
    <xs:enumeration value="Local"/>
    <xs:enumeration value="URI"/>
    <xs:enumeration value="PersonalIdentityNumber"/>
    <xs:enumeration value="Co-ordinationNumber"/>
    <xs:enumeration value="Other"/>
  </xs:restriction>
</xs:simpleType>
```

Simple Type eventTypeType

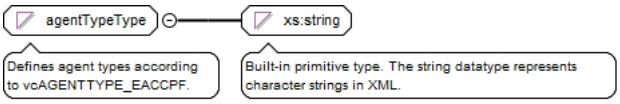
Namespace	employment
Annotations	Defines event types according to vcEVENTTYPE_EACCPF.
Diagram	
Type	restriction of xs:string
Facets	enumeration: cancelled
	enumeration: created
	enumeration: deleted
	enumeration: derived
	enumeration: revised1
	enumeration: updated
Used by	Attribute: maintenanceHistoryEventType/@eventType

Source

```
<xs:simpleType name="eventTypeType">
  <xs:annotation>
    <xs:documentation xml:lang="en">Defines event types according to vcEVENTTYPE_EACCPF.</
xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="cancelled"/>
    <xs:enumeration value="created"/>
    <xs:enumeration value="deleted"/>
    <xs:enumeration value="derived"/>
    <xs:enumeration value="revised1"/>
    <xs:enumeration value="updated"/>
  </xs:restriction>
</xs:simpleType>
```

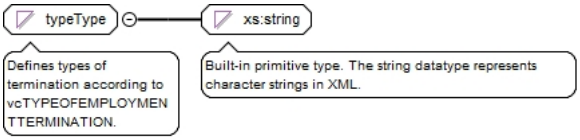
Simple Type agentTypeType

Namespace	employment

Annotations	Defines agent types according to vcAGENTTYPE_EACCPF.
Diagram	
Type	restriction of xs:string
Facets	enumeration: human enumeration: machine enumeration: Other
Used by	Attribute: maintenanceHistoryEventType/@agentType

Source	`<xs:simpleType name="agentTypeType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Defines agent types according to vcAGENTTYPE_EACCPF.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:restriction base="xs:string">`
	`  <xs:enumeration value="human"/>`
	`  <xs:enumeration value="machine"/>`
	`  <xs:enumeration value="Other"/>`
	`  </xs:restriction>`
	`</xs:simpleType>`

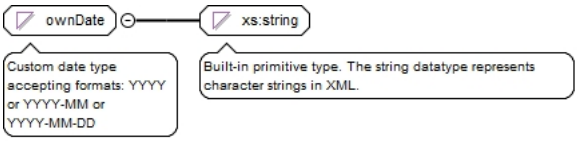
Simple Type `typeType`

Namespace	employment
Annotations	Defines types of termination according to vcTYPEOFEMPLOYMENTTERMINATION.
Diagram	
Type	restriction of xs:string
Facets	enumeration: AtTheirOwnRequest enumeration: Retirement enumeration: SicknessCompensation enumeration: PersonalReasons enumeration: LackOfWork enumeration: EmployeeDied enumeration: Dismissal enumeration: Other
Used by	Attribute: employmentTerminationType/@typeOfTermination

Source	`<xs:simpleType name="typeType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Defines types of termination according to vcTYPEOFEMPLOYMENTTERMINATION.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:restriction base="xs:string">`
	`  <xs:enumeration value="AtTheirOwnRequest"/>`
	`  <xs:enumeration value="Retirement"/>`
	`  <xs:enumeration value="SicknessCompensation"/>`
	`  <xs:enumeration value="PersonalReasons"/>`
	`  <xs:enumeration value="LackOfWork"/>`
	`  <xs:enumeration value="EmployeeDied"/>`
	`  <xs:enumeration value="Dismissal"/>`
	`  <xs:enumeration value="Other"/>`
	`  </xs:restriction>`
	`</xs:simpleType>`

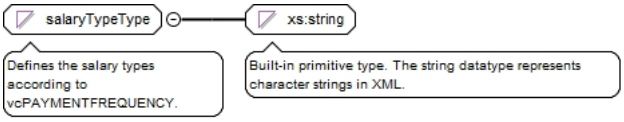
Simple Type `ownDate`

Namespace	employment
Annotations	Custom date type accepting formats: YYYY or YYYY-MM or YYYY-MM-DD

Diagram	
Type	restriction of xs:string
Facets	pattern: `[\d]{4}\-[\d]{2}\-[\d]{2}|[\d]{4}\-[\d]{2}|[\d]{4}`
Used by	Elements: dateRangeType/endDate, dateRangeType/startDate, dateSetType/date, fringeBenefitPeriodType/date, salaryPeriodType/date

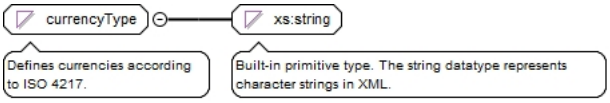
Source	`<xs:simpleType name="ownDate">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Custom date type accepting formats: YYYY or YYYY-MM or YYYY-MM-DD</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:restriction base="xs:string">`
	`  <xs:pattern value="[\d]{4}\-[\d]{2}\-[\d]{2}|[\d]{4}\-[\d]{2}|[\d]{4}"/>`
	`  </xs:restriction>`
	`</xs:simpleType>`

Simple Type `salaryTypeType`

Namespace	employment
Annotations	`Defines the salary types according to vcPAYMENTFREQUENCY.`
Diagram	
Type	restriction of xs:string
Facets	enumeration: `monthlyPay` enumeration: `weeklyPay` enumeration: `fortnightPay` enumeration: `dayRate` enumeration: `hourlyWage` enumeration: `rateOfCommission` enumeration: `Other`
Used by	Attribute: salaryAgreementType/@salaryType

Source	`<xs:simpleType name="salaryTypeType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Defines the salary types according to vcPAYMENTFREQUENCY.</xs:documentation>`
	`  </xs:annotation>`
	`  <xs:restriction base="xs:string">`
	`  <xs:enumeration value="monthlyPay"/>`
	`  <xs:enumeration value="weeklyPay"/>`
	`  <xs:enumeration value="fortnightPay"/>`
	`  <xs:enumeration value="dayRate"/>`
	`  <xs:enumeration value="hourlyWage"/>`
	`  <xs:enumeration value="rateOfCommission"/>`
	`  <xs:enumeration value="Other"/>`
	`  </xs:restriction>`
	`</xs:simpleType>`

Simple Type `currencyType`

Namespace	employment
Annotations	`Defines currencies according to ISO 4217.`
Diagram	
Type	restriction of xs:string

Facets	enumeration	USD
	enumeration	XBC
	enumeration	ARS
	enumeration	CAD
	enumeration	AZN
	enumeration	JPY
	enumeration	MGA
	enumeration	PLN
	enumeration	RWF
	enumeration	BZD
	enumeration	TWD
	enumeration	NIO
	enumeration	SRD
	enumeration	CHW
	enumeration	TTD
	enumeration	BBD
	enumeration	STN
	enumeration	SLL
	enumeration	XBD
	enumeration	TRY
	enumeration	SCR
	enumeration	XDR
	enumeration	EUR
	enumeration	BHD
	enumeration	HRK
	enumeration	HTG
	enumeration	RUB
	enumeration	GYD
	enumeration	BGN
	enumeration	BMD
	enumeration	MXN
	enumeration	COP
	enumeration	BOV
	enumeration	USN
	enumeration	KRW
	enumeration	RON
	enumeration	HUF
	enumeration	PGK
	enumeration	KWD
	enumeration	TJS
	enumeration	XOF
	enumeration	SEK
	enumeration	MZN
	enumeration	HNL
	enumeration	NGN
	enumeration	BYN
	enumeration	CLP
	enumeration	CUP
	enumeration	DOP

enumeration	UGX
enumeration	SBD
enumeration	ISK
enumeration	VUV
enumeration	BAM
enumeration	AED
enumeration	XUA
enumeration	ZMW
enumeration	IDR
enumeration	TMT
enumeration	MUR
enumeration	AOA
enumeration	DJF
enumeration	XPF
enumeration	LSL
enumeration	CZK
enumeration	MVR
enumeration	PKR
enumeration	FJD
enumeration	MMK
enumeration	CHE
enumeration	SDG
enumeration	KGS
enumeration	SSP
enumeration	SHP
enumeration	CRC
enumeration	ETB
enumeration	MOP
enumeration	ANG
enumeration	GBP
enumeration	PHP
enumeration	CNY
enumeration	ERN
enumeration	RSD
enumeration	BSD
enumeration	KHR
enumeration	FKP
enumeration	JMD
enumeration	CUC
enumeration	MRU
enumeration	WST
enumeration	XPT
enumeration	ILS
enumeration	BIF
enumeration	GMD
enumeration	UYI
enumeration	BWP
enumeration	LYD
enumeration	LBP

enumeration	NOK
enumeration	CVE
enumeration	TOP
enumeration	AFN
enumeration	TZS
enumeration	XCD
enumeration	XBB
enumeration	AWG
enumeration	AMD
enumeration	MWK
enumeration	ZWL
enumeration	MXV
enumeration	UYU
enumeration	INR
enumeration	EGP
enumeration	IRR
enumeration	UAH
enumeration	DZD
enumeration	NPR
enumeration	BDT
enumeration	GHS
enumeration	CHF
enumeration	SOS
enumeration	PAB
enumeration	PYG
enumeration	KES
enumeration	UZS
enumeration	QAR
enumeration	VES
enumeration	NAD
enumeration	KZT
enumeration	AUD
enumeration	KYD
enumeration	LAK
enumeration	BTN
enumeration	CDF
enumeration	PEN
enumeration	HKD
enumeration	SYP
enumeration	GEL
enumeration	LKR
enumeration	KPW
enumeration	YER
enumeration	XAF
enumeration	COU
enumeration	MNT
enumeration	MAD
enumeration	SGD
enumeration	UYW

	enumeration	NZD
	enumeration	TND
	enumeration	IQD
	enumeration	GTQ
	enumeration	GNF
	enumeration	LRD
	enumeration	XSU
	enumeration	MDL
	enumeration	XAU
	enumeration	KMF
	enumeration	BND
	enumeration	OMR
	enumeration	XTS
	enumeration	SAR
	enumeration	CLF
	enumeration	BRL
	enumeration	GIP
	enumeration	BOB
	enumeration	JOD
	enumeration	XBA
	enumeration	SZL
	enumeration	VND
	enumeration	XAG
	enumeration	MKD
	enumeration	THB
	enumeration	ALL
	enumeration	SVC
	enumeration	DKK
	enumeration	XXX
	enumeration	XPD
	enumeration	MYR
	enumeration	ZAR
Used by	Attribute	salaryAgreementType/@currency

Source

```
<xs:simpleType name="currencyType">
  <xs:annotation>
    <xs:documentation xml:lang="en">Defines currencies according to ISO 4217.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="USD"/>
    <xs:enumeration value="XBC"/>
    <xs:enumeration value="ARS"/>
    <xs:enumeration value="CAD"/>
    <xs:enumeration value="AZN"/>
    <xs:enumeration value="JPY"/>
    <xs:enumeration value="MGA"/>
    <xs:enumeration value="PLN"/>
    <xs:enumeration value="RWF"/>
    <xs:enumeration value="BZD"/>
    <xs:enumeration value="TWD"/>
    <xs:enumeration value="NIO"/>
    <xs:enumeration value="SRD"/>
    <xs:enumeration value="CHW"/>
    <xs:enumeration value="TTD"/>
    <xs:enumeration value="BBD"/>
    <xs:enumeration value="STN"/>
    <xs:enumeration value="SLL"/>
    <xs:enumeration value="XBD"/>
    <xs:enumeration value="TRY"/>
    <xs:enumeration value="SCR"/>
```

```
<xs:enumeration value="XDR"/>
<xs:enumeration value="EUR"/>
<xs:enumeration value="BHD"/>
<xs:enumeration value="HRK"/>
<xs:enumeration value="HTG"/>
<xs:enumeration value="RUB"/>
<xs:enumeration value="GYD"/>
<xs:enumeration value="BGN"/>
<xs:enumeration value="BMD"/>
<xs:enumeration value="MXN"/>
<xs:enumeration value="COP"/>
<xs:enumeration value="BOV"/>
<xs:enumeration value="USN"/>
<xs:enumeration value="KRW"/>
<xs:enumeration value="RON"/>
<xs:enumeration value="HUF"/>
<xs:enumeration value="PGK"/>
<xs:enumeration value="KWD"/>
<xs:enumeration value="TJS"/>
<xs:enumeration value="XOF"/>
<xs:enumeration value="SEK"/>
<xs:enumeration value="MZN"/>
<xs:enumeration value="HNL"/>
<xs:enumeration value="NGN"/>
<xs:enumeration value="BYN"/>
<xs:enumeration value="CLP"/>
<xs:enumeration value="CUP"/>
<xs:enumeration value="DOP"/>
<xs:enumeration value="UGX"/>
<xs:enumeration value="SBD"/>
<xs:enumeration value="ISK"/>
<xs:enumeration value="VUV"/>
<xs:enumeration value="BAM"/>
<xs:enumeration value="AED"/>
<xs:enumeration value="XUA"/>
<xs:enumeration value="ZMW"/>
<xs:enumeration value="IDR"/>
<xs:enumeration value="TMT"/>
<xs:enumeration value="MUR"/>
<xs:enumeration value="AOA"/>
<xs:enumeration value="DJF"/>
<xs:enumeration value="XPF"/>
<xs:enumeration value="LSL"/>
<xs:enumeration value="CZK"/>
<xs:enumeration value="MVR"/>
<xs:enumeration value="PKR"/>
<xs:enumeration value="FJD"/>
<xs:enumeration value="MMK"/>
<xs:enumeration value="CHE"/>
<xs:enumeration value="SDG"/>
<xs:enumeration value="KGS"/>
<xs:enumeration value="SSP"/>
<xs:enumeration value="SHP"/>
<xs:enumeration value="CRC"/>
<xs:enumeration value="ETB"/>
<xs:enumeration value="MOP"/>
<xs:enumeration value="ANG"/>
<xs:enumeration value="GBP"/>
<xs:enumeration value="PHP"/>
<xs:enumeration value="CNY"/>
<xs:enumeration value="ERN"/>
<xs:enumeration value="RSD"/>
<xs:enumeration value="BSD"/>
<xs:enumeration value="KHR"/>
<xs:enumeration value="FKP"/>
<xs:enumeration value="JMD"/>
<xs:enumeration value="CUC"/>
<xs:enumeration value="MRU"/>
<xs:enumeration value="WST"/>
<xs:enumeration value="XPT"/>
<xs:enumeration value="ILS"/>
<xs:enumeration value="BIF"/>
<xs:enumeration value="GMD"/>
<xs:enumeration value="UYI"/>
<xs:enumeration value="BWP"/>
<xs:enumeration value="LYD"/>
<xs:enumeration value="LBP"/>
<xs:enumeration value="NOK"/>
<xs:enumeration value="CVE"/>
<xs:enumeration value="TOP"/>
<xs:enumeration value="AFN"/>
<xs:enumeration value="TZS"/>
<xs:enumeration value="XCD"/>
```

```
      <xs:enumeration value="XBB"/>
      <xs:enumeration value="AWG"/>
      <xs:enumeration value="AMD"/>
      <xs:enumeration value="MWK"/>
      <xs:enumeration value="ZWL"/>
      <xs:enumeration value="MXV"/>
      <xs:enumeration value="UYU"/>
      <xs:enumeration value="INR"/>
      <xs:enumeration value="EGP"/>
      <xs:enumeration value="IRR"/>
      <xs:enumeration value="UAH"/>
      <xs:enumeration value="DZD"/>
      <xs:enumeration value="NPR"/>
      <xs:enumeration value="BDT"/>
      <xs:enumeration value="GHS"/>
      <xs:enumeration value="CHF"/>
      <xs:enumeration value="SOS"/>
      <xs:enumeration value="PAB"/>
      <xs:enumeration value="PYG"/>
      <xs:enumeration value="KES"/>
      <xs:enumeration value="UZS"/>
      <xs:enumeration value="QAR"/>
      <xs:enumeration value="VES"/>
      <xs:enumeration value="NAD"/>
      <xs:enumeration value="KZT"/>
      <xs:enumeration value="AUD"/>
      <xs:enumeration value="KYD"/>
      <xs:enumeration value="LAK"/>
      <xs:enumeration value="BTN"/>
      <xs:enumeration value="CDF"/>
      <xs:enumeration value="PEN"/>
      <xs:enumeration value="HKD"/>
      <xs:enumeration value="SYP"/>
      <xs:enumeration value="GEL"/>
      <xs:enumeration value="LKR"/>
      <xs:enumeration value="KPW"/>
      <xs:enumeration value="YER"/>
      <xs:enumeration value="XAF"/>
      <xs:enumeration value="COU"/>
      <xs:enumeration value="MNT"/>
      <xs:enumeration value="MAD"/>
      <xs:enumeration value="SGD"/>
      <xs:enumeration value="UYW"/>
      <xs:enumeration value="NZD"/>
      <xs:enumeration value="TND"/>
      <xs:enumeration value="IQD"/>
      <xs:enumeration value="GTQ"/>
      <xs:enumeration value="GNF"/>
      <xs:enumeration value="LRD"/>
      <xs:enumeration value="XSU"/>
      <xs:enumeration value="MDL"/>
      <xs:enumeration value="XAU"/>
      <xs:enumeration value="KMF"/>
      <xs:enumeration value="BND"/>
      <xs:enumeration value="OMR"/>
      <xs:enumeration value="XTS"/>
      <xs:enumeration value="SAR"/>
      <xs:enumeration value="CLF"/>
      <xs:enumeration value="BRL"/>
      <xs:enumeration value="GIP"/>
      <xs:enumeration value="BOB"/>
      <xs:enumeration value="JOD"/>
      <xs:enumeration value="XBA"/>
      <xs:enumeration value="SZL"/>
      <xs:enumeration value="VND"/>
      <xs:enumeration value="XAG"/>
      <xs:enumeration value="MKD"/>
      <xs:enumeration value="THB"/>
      <xs:enumeration value="ALL"/>
      <xs:enumeration value="SVC"/>
      <xs:enumeration value="DKK"/>
      <xs:enumeration value="XXX"/>
      <xs:enumeration value="XPD"/>
      <xs:enumeration value="MYR"/>
      <xs:enumeration value="ZAR"/>
    </xs:restriction>
  </xs:simpleType>
```

Namespace: ""

Attribute(s)

Attribute `otherAgencyCodeType / @agencyCodePrefix`

Namespace	No namespace
Annotations	`State a prefix according to value list vcTYPEOFIDENTIFICATIONCODE, or use "Other" if not applicable.`
Type	otherAgencyCodePrefix
Properties	use: `optional`
Facets	enumeration `VAT` enumeration `DUNS` enumeration `ORG` enumeration `HSA` enumeration `Local` enumeration `URI` enumeration `PersonalIdentityNumber` enumeration `Co-ordinationNumber` enumeration `Other`
Used by	Complex Type otherAgencyCodeType

Source	`<xs:attribute name="agencyCodePrefix" type="otherAgencyCodePrefix" use="optional">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State a prefix according to value list vcTYPEOFIDENTIFICATIONCODE, or use "Other" if not applicable.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `otherAgencyCodeType / @otherAgencyCodePrefix`

Namespace	No namespace
Annotations	`If attribute "agencyCodePrefix" is "Other", state an applicable prefix.`
Type	xs:string
Properties	use: `optional`
Used by	Complex Type otherAgencyCodeType

Source	`<xs:attribute name="otherAgencyCodePrefix" type="xs:string" use="optional">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">If attribute "agencyCodePrefix" is "Other", state an applicable prefix.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `languageType / @languageCode`

Namespace	No namespace
Annotations	`Language code must be written in lowercase and conform to the ISO639-2 standard. Example: "swe"`
Type	xs:string
Properties	content: `simple`
Used by	Complex Type languageType

Source	`<xs:attribute name="languageCode" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Language code must be written in lowercase and conform to the ISO639-2 standard. Example: "swe"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `languageType / @language`

Namespace	No namespace
Annotations	`Language must have the first character capitalized and conform to the ISO639-2 standard. Example: "Swedish"`

Type	xs:string
Properties	content: simple
Used by	Complex Type languageType

Source	`<xs:attribute name="language" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Language must have the first character capitalized and conform to the ISO639-2 standard. Example: "Swedish"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `scriptType / @scriptCode`

Namespace	No namespace
Annotations	Script code must have the first character capitalized and conform to the ISO15924 standard. Example: "Latn"
Type	xs:string
Properties	content: simple
Used by	Complex Type scriptType

Source	`<xs:attribute name="scriptCode" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Script code must have the first character capitalized and conform to the ISO15924 standard. Example: "Latn"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `scriptType / @script`

Namespace	No namespace
Annotations	Script must have the first character capitalized and conform to the ISO15924 standard. Example: "Latin"
Type	xs:string
Properties	content: simple
Used by	Complex Type scriptType

Source	`<xs:attribute name="script" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Script must have the first character capitalized and conform to the ISO15924 standard. Example: "Latin"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `citationType / @href`

Namespace	No namespace
Annotations	State the URI for the FGS used in this XML-document. Example: "https://myndighet.se/Försöksmyndighetens_FGS_Personal_V_20181224.pdf"
Type	xs:anyURI
Properties	content: simple
Used by	Complex Type citationType

Source	`<xs:attribute name="href" type="xs:anyURI">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI for the FGS used in this XML-document. Example: "https://myndighet.se/Försöksmyndighetens_FGS_Personal_V_20181224.pdf"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `maintenanceHistoryEventType / @eventType`

Namespace	No namespace
Annotations	State what kind of change has been made accorrding to vcEVENTTYPE_EACCPF. Example: "created"
Type	eventTypeType
Properties	use: required

Facets	enumeration	cancelled
	enumeration	created
	enumeration	deleted
	enumeration	derived
	enumeration	revised1
	enumeration	updated
Used by	Complex Type	maintenanceHistoryEventType
Source	<xs:attribute name="eventType" type="eventTypeType" use="required">\<xs:annotation>\<xs:documentation xml:lang="en">State what kind of change has been made accorrding to vcEVENTTYPE_EACCPF. Example: "created"</xs:documentation>\</xs:annotation>\</xs:attribute>	

Attribute maintenanceHistoryEventType / @eventDateTime

Namespace	No namespace	
Annotations	State the date and time the change was made. Example: "2001-04-13T22:30:00"	
Type	xs:dateTime	
Properties	use:	required
Used by	Complex Type	maintenanceHistoryEventType
Source	<xs:attribute name="eventDateTime" type="xs:dateTime" use="required">\<xs:annotation>\<xs:documentation xml:lang="en">State the date and time the change was made. Example: "2001-04-13T22:30:00"</xs:documentation>\</xs:annotation>\</xs:attribute>	

Attribute maintenanceHistoryEventType / @agentType

Namespace	No namespace	
Annotations	State the type of agent that performed the changes according to vcAGENTTYPE_EACCPF, or use "Other" if not applicable. Example: "human"	
Type	agentTypeType	
Properties	use:	required
Facets	enumeration	human
	enumeration	machine
	enumeration	Other
Used by	Complex Type	maintenanceHistoryEventType
Source	<xs:attribute name="agentType" type="agentTypeType" use="required">\<xs:annotation>\<xs:documentation xml:lang="en">State the type of agent that performed the changes according to vcAGENTTYPE_EACCPF, or use "Other" if not applicable. Example: "human"</xs:documentation>\</xs:annotation>\</xs:attribute>	

Attribute maintenanceHistoryEventType / @otherAgentType

Namespace	No namespace	
Annotations	If attribute "agentType" is "Other", state an applicable agent type.	
Type	xs:string	
Properties	use:	optional
Used by	Complex Type	maintenanceHistoryEventType
Source	<xs:attribute name="otherAgentType" type="xs:string" use="optional">\<xs:annotation>\<xs:documentation xml:lang="en">If attribute "agentType" is "Other", state an applicable agent type.</xs:documentation>\</xs:annotation>\</xs:attribute>	

Attribute maintenanceHistoryEventType / @agent

Namespace	No namespace
Annotations	State the name of the agent who performed the changes. Example: "Anna Andersson"
Type	xs:string
Properties	use: required
Used by	Complex Type maintenanceHistoryEventType

Source	`<xs:attribute name="agent" type="xs:string" use="required">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the name of the agent who performed the changes. Example: "Anna Andersson"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute attribute / @name

Namespace	No namespace
Annotations	Name of own defined element.
Type	xs:string
Properties	use: required
Used by	Element attribute

Source	`<xs:attribute name="name" type="xs:string" use="required">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Name of own defined element.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute attribute / @dataType

Namespace	No namespace
Annotations	Datatype for own defined element.
Type	xs:string
Properties	content: simple
Used by	Element attribute

Source	`<xs:attribute name="dataType" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Datatype for own defined element.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute attribute / @format

Namespace	No namespace
Annotations	Format for own defined element.
Type	xs:string
Properties	content: simple
Used by	Element attribute

Source	`<xs:attribute name="format" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Format for own defined element.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute ownElementType / @name

Namespace	No namespace
Annotations	Name of own defined element.
Type	xs:string
Properties	use: required
Used by	Complex Type ownElementType

Source	`<xs:attribute name="name" use="required" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Name of own defined element.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute ownElementType / @dataType

Namespace	No namespace
Annotations	Datatype for own defined element element.
Type	xs:string
Properties	content: simple
Used by	Complex Type ownElementType

Source	`<xs:attribute name="dataType" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Datatype for own defined element element.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute ownElementType / @format

Namespace	No namespace
Annotations	Format for own defined element.
Type	xs:string
Properties	content: simple
Used by	Complex Type ownElementType

Source	`<xs:attribute name="format" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Format for own defined element.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute eSignatureComplexType / @present

Namespace	No namespace
Annotations	Attribute indicating that an e-signature has been present or not.
Type	xs:boolean
Properties	use: required
Used by	Complex Type eSignatureComplexType

Source	`<xs:attribute name="present" type="xs:boolean" use="required">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Attribute indicating that an e-signature has been present or`
	` not.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute eSignatureComplexType / @dateESignatureIsVerified

Namespace	No namespace
Annotations	Attribute with the datetime giving when the e-signature was verified.
Type	xs:dateTime
Properties	use: optional
Used by	Complex Type eSignatureComplexType

Source	`<xs:attribute name="dateESignatureIsVerified" type="xs:dateTime" use="optional">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Attribute with the datetime giving when the e-signature was`
	` verified.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute AppendixType / @disposable

Namespace	No namespace

Annotations	If the appendix can be disposed of before the aggregation or record is disposed of value="true" otherwise false.
Type	xs:boolean
Properties	use: optional
Used by	Complex Type AppendixType

Source	`<xs:attribute name="disposable" type="xs:boolean" use="optional">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">If the appendix can be disposed of before the aggregation or record is disposed of value="true" otherwise false.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute AppendixType / @name

Namespace	No namespace
Annotations	Name of the appendix.
Type	xs:string
Properties	use: required
Used by	Complex Type AppendixType

Source	`<xs:attribute name="name" type="xs:string" use="required">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Name of the appendix.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute AppendixType / @description

Namespace	No namespace
Annotations	Description of appendix.
Type	xs:string
Properties	use: optional
Used by	Complex Type AppendixType

Source	`<xs:attribute name="description" type="xs:string" use="optional">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Description of appendix.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute AppendixType / @fileFormat

Namespace	No namespace
Annotations	File format of appendix.
Type	xs:string
Properties	use: optional
Used by	Complex Type AppendixType

Source	`<xs:attribute name="fileFormat" type="xs:string" use="optional">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">File format of appendix.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute AppendixType / @originalFileFormat

Namespace	No namespace
Annotations	Original file format of appendix.
Type	xs:string
Properties	use: optional
Used by	Complex Type AppendixType
Source	`<xs:attribute name="originalFileFormat" type="xs:string" use="optional">` `  <xs:annotation>`

```
    <xs:documentation xml:lang="en">Original file format of appendix.</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

Attribute AppendixType / @path

Namespace	No namespace
Annotations	Name and path to the file in the form: file:///path/to/file.
Type	xs:string
Properties	use: required
Used by	Complex Type AppendixType

Source	`<xs:attribute name="path" type="xs:string" use="required">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Name and path to the file in the form: file:///path/to/file.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute AppendixType / @eSignatureHaveExisted

Namespace	No namespace
Annotations	Marker for the appendix having had an electronic signature.
Type	xs:boolean
Properties	use: optional
Used by	Complex Type AppendixType

Source	`<xs:attribute name="eSignatureHaveExisted" type="xs:boolean" use="optional">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Marker for the appendix having had an electronic signature.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute applicationDocumentType / @href

Namespace	No namespace
Annotations	State the URI for the application document.
Type	xs:anyURI
Properties	use: optional
Used by	Complex Type applicationDocumentType

Source	`<xs:attribute name="href" type="xs:anyURI" use="optional">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI for the application document.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute termType / @code

Namespace	No namespace
Annotations	State the code for the position according to the source attribute. Example: "2422"
Type	xs:string
Properties	use: optional
Used by	Complex Type termType

Source	`<xs:attribute name="code" type="xs:string" use="optional">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the code for the position according to the source`
	` attribute. Example: "2422"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute termType / @source

Namespace	No namespace
Annotations	State the source used for declaring the position. Example: "SSYK2012"

Type	xs:string
Properties	use: optional
Used by	Complex Type termType

Source	`<xs:attribute name="source" type="xs:string" use="optional">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the source used for declaring the position. Example: "SSYK2012"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute positionType / @href

Namespace	No namespace
Annotations	State the URI to the document describing the position.
Type	xs:anyURI
Properties	use: optional
Used by	Complex Type positionType

Source	`<xs:attribute name="href" use="optional" type="xs:anyURI">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI to the document describing the position.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute terminationDecisionType / @href

Namespace	No namespace
Annotations	State the URI to the termination decision.
Type	xs:anyURI
Properties	use: optional
Used by	Complex Type terminationDecisionType

Source	`<xs:attribute name="href" use="optional" type="xs:anyURI">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI to the termination decision.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute terminationApplicationType / @href

Namespace	No namespace
Annotations	State the URI to the termination application.
Type	xs:anyURI
Properties	use: optional
Used by	Complex Type terminationApplicationType

Source	`<xs:attribute name="href" type="xs:anyURI" use="optional">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI to the termination application.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute employmentTerminationType / @typeOfTermination

Namespace	No namespace
Annotations	Used for describing the type of termination according to vcTYPEOFEMPLOYMENTTERMINATION, or use "Other" if not applicable. Example: "LackOfWork"
Type	typeType
Properties	use: required
Facets	enumeration AtTheirOwnRequest
	enumeration Retirement
	enumeration SicknessCompensation

	enumeration	PersonalReasons
	enumeration	LackOfWork
	enumeration	EmployeeDied
	enumeration	Dismissal
	enumeration	Other
Used by	Complex Type	employmentTerminationType

Source

```
<xs:attribute name="typeOfTermination" use="required" type="typeType">
  <xs:annotation>
    <xs:documentation xml:lang="en">Used for describing the type of termination according
 to vcTYPEOFEMPLOYMENTTERMINATION, or use "Other" if not applicable. Example: "LackOfWork"</
xs:documentation>
  </xs:annotation>
</xs:attribute>
```

Attribute employmentTerminationType / @otherTypeOfTermination

Namespace	No namespace	
Annotations	If typeOfTermination is "Other", state an applicable type of termination.	
Type	xs:string	
Properties	content:	simple
Used by	Complex Type	employmentTerminationType

Source

```
<xs:attribute name="otherTypeOfTermination" type="xs:string">
  <xs:annotation>
    <xs:documentation xml:lang="en">If typeOfTermination is "Other", state an applicable type of
 termination.</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

Attribute contractType / @href

Namespace	No namespace	
Annotations	State the URI to the contract.	
Type	xs:anyURI	
Properties	use:	optional
Used by	Complex Type	contractType

Source

```
<xs:attribute name="href" use="optional" type="xs:anyURI">
  <xs:annotation>
    <xs:documentation xml:lang="en">State the URI to the contract.</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

Attribute agreementType / @href

Namespace	No namespace	
Annotations	State the URI to the agreement document.	
Type	xs:anyURI	
Properties	use:	optional
Used by	Complex Type	agreementType

Source

```
<xs:attribute name="href" use="optional" type="xs:anyURI">
  <xs:annotation>
    <xs:documentation xml:lang="en">State the URI to the agreement document.</xs:documentation>
  </xs:annotation>
</xs:attribute>
```

Attribute workplaceType / @code

Namespace	No namespace	
Annotations	State the code for the workplace. Example: "MAR"	
Type	xs:string	
Properties	use:	optional

Used by	Complex Type workplaceType

Source	`<xs:attribute name="code" use="optional" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the code for the workplace. Example: "MAR"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `workplaceType / @name`

Namespace	No namespace
Annotations	State the name of the workplace. Example: "Marieberg"
Type	xs:string
Properties	use: optional
Used by	Complex Type workplaceType

Source	`<xs:attribute name="name" use="optional" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the name of the workplace. Example: "Marieberg"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `jobLocationType / @city`

Namespace	No namespace
Annotations	State the job location's city.
Type	xs:string
Properties	use: optional
Used by	Complex Type jobLocationType

Source	`<xs:attribute name="city" use="optional" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the job location's city.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `jobLocationType / @country`

Namespace	No namespace
Annotations	State the job location's country.
Type	xs:string
Properties	use: optional
Used by	Complex Type jobLocationType

Source	`<xs:attribute name="country" use="optional" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the job location's country.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `weeklyWorkingTimeType / @unit`

Namespace	No namespace
Annotations	Define which unit the employee's weekly working time is stated as.
Type	xs:string
Properties	use: required
Used by	Complex Type weeklyWorkingTimeType

Source	`<xs:attribute name="unit" use="required" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Define which unit the employee's weekly working time is stated as.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute weeklyWorkingTimeType / @numberOfUnits

Namespace	No namespace
Annotations	State the number of units for the employee's weekly working time.
Type	xs:integer
Properties	use: required
Used by	Complex Type weeklyWorkingTimeType

Source	`<xs:attribute name="numberOfUnits" use="required" type="xs:integer">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the number of units for the employee's weekly working time.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute approvedOccupationType / @type

Namespace	No namespace
Annotations	Describe what kind of side line occupation has been approved. Example: "Konsulttjänster som fotograf"
Type	xs:string
Properties	use: required
Used by	Complex Type approvedOccupationType

Source	`<xs:attribute name="type" use="required" type="xs:string">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describe what kind of side line occupation has been approved. Example: "Konsulttjänster som fotograf"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute approvedOccupationType / @notificationDate

Namespace	No namespace
Annotations	State which date the application for the side line occupation was recieved.
Type	xs:date
Properties	use: required
Used by	Complex Type approvedOccupationType

Source	`<xs:attribute name="notificationDate" use="required" type="xs:date">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State which date the application for the side line occupation was recieved.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute approvedOccupationType / @decisionDate

Namespace	No namespace
Annotations	State which date the application for the side line occupation was approved.
Type	xs:date
Properties	use: required
Used by	Complex Type approvedOccupationType

Source	`<xs:attribute name="decisionDate" use="required" type="xs:date">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State which date the application for the side line occupation was approved.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute approvedOccupationType / @href

Namespace	No namespace
Annotations	State the URI to the decision for the side line occupation.
Type	xs:anyURI

Properties	use:	required
Used by	Complex Type	approvedOccupationType
Source	<xs:attribute name="href" use="required" type="xs:anyURI"> <xs:annotation> <xs:documentation xml:lang="en">State the URI to the decision for the side line occupation.</ xs:documentation> </xs:annotation> </xs:attribute>	

Attribute salaryAgreementType / @salaryType

Namespace	No namespace	
Annotations	State the type of salary according to vcPAYMENTFREQUENCY, or use "Other" if not applicable. Example: "monthlyPay"	
Type	salaryTypeType	
Properties	use:	required
Facets	enumeration	monthlyPay
	enumeration	weeklyPay
	enumeration	fortnightPay
	enumeration	dayRate
	enumeration	hourlyWage
	enumeration	rateOfCommission
	enumeration	Other
Used by	Complex Type	salaryAgreementType
Source	<xs:attribute name="salaryType" use="required" type="salaryTypeType"> <xs:annotation> <xs:documentation xml:lang="en">State the type of salary according to vcPAYMENTFREQUENCY, or use "Other" if not applicable. Example: "monthlyPay"</xs:documentation> </xs:annotation> </xs:attribute>	

Attribute salaryAgreementType / @otherSalaryType

Namespace	No namespace	
Annotations	If "salaryType" is "Other", state an applicable salary type.	
Type	xs:string	
Properties	use:	optional
Used by	Complex Type	salaryAgreementType
Source	<xs:attribute name="otherSalaryType" use="optional" type="xs:string"> <xs:annotation> <xs:documentation xml:lang="en">If "salaryType" is "Other", state an applicable salary type.</ xs:documentation> </xs:annotation> </xs:attribute>	

Attribute salaryAgreementType / @salaryCode

Namespace	No namespace	
Annotations	State the salary code. May for example consist of salary plan, salary grade, salary class, and additional salary class separated by ":". Example: "KA"	
Type	xs:string	
Properties	use:	optional
Used by	Complex Type	salaryAgreementType
Source	<xs:attribute name="salaryCode" use="optional" type="xs:string"> <xs:annotation> <xs:documentation xml:lang="en">State the salary code. May for example consist of salary plan, salary grade, salary class, and additional salary class separated by ":". Example: "KA"</ xs:documentation> </xs:annotation> </xs:attribute>	

Attribute salaryAgreementType / @amount

Namespace	No namespace
Annotations	State the agreed on wage. Example: "28000"
Type	xs:integer
Properties	use: required
Used by	Complex Type salaryAgreementType

Source	`<xs:attribute name="amount" use="required" type="xs:integer">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the agreed on wage. Example: "28000"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute salaryAgreementType / @currency

Namespace	No namespace
Annotations	State the currency for the agreed on wage. Example: "SEK"
Type	currencyType
Properties	use: required
Facets	enumeration USD
	enumeration XBC
	enumeration ARS
	enumeration CAD
	enumeration AZN
	enumeration JPY
	enumeration MGA
	enumeration PLN
	enumeration RWF
	enumeration BZD
	enumeration TWD
	enumeration NIO
	enumeration SRD
	enumeration CHW
	enumeration TTD
	enumeration BBD
	enumeration STN
	enumeration SLL
	enumeration XBD
	enumeration TRY
	enumeration SCR
	enumeration XDR
	enumeration EUR
	enumeration BHD
	enumeration HRK
	enumeration HTG
	enumeration RUB
	enumeration GYD
	enumeration BGN
	enumeration BMD
	enumeration MXN
	enumeration COP
	enumeration BOV

enumeration	USN
enumeration	KRW
enumeration	RON
enumeration	HUF
enumeration	PGK
enumeration	KWD
enumeration	TJS
enumeration	XOF
enumeration	SEK
enumeration	MZN
enumeration	HNL
enumeration	NGN
enumeration	BYN
enumeration	CLP
enumeration	CUP
enumeration	DOP
enumeration	UGX
enumeration	SBD
enumeration	ISK
enumeration	VUV
enumeration	BAM
enumeration	AED
enumeration	XUA
enumeration	ZMW
enumeration	IDR
enumeration	TMT
enumeration	MUR
enumeration	AOA
enumeration	DJF
enumeration	XPF
enumeration	LSL
enumeration	CZK
enumeration	MVR
enumeration	PKR
enumeration	FJD
enumeration	MMK
enumeration	CHE
enumeration	SDG
enumeration	KGS
enumeration	SSP
enumeration	SHP
enumeration	CRC
enumeration	ETB
enumeration	MOP
enumeration	ANG
enumeration	GBP
enumeration	PHP
enumeration	CNY
enumeration	ERN

enumeration	RSD
enumeration	BSD
enumeration	KHR
enumeration	FKP
enumeration	JMD
enumeration	CUC
enumeration	MRU
enumeration	WST
enumeration	XPT
enumeration	ILS
enumeration	BIF
enumeration	GMD
enumeration	UYI
enumeration	BWP
enumeration	LYD
enumeration	LBP
enumeration	NOK
enumeration	CVE
enumeration	TOP
enumeration	AFN
enumeration	TZS
enumeration	XCD
enumeration	XBB
enumeration	AWG
enumeration	AMD
enumeration	MWK
enumeration	ZWL
enumeration	MXV
enumeration	UYU
enumeration	INR
enumeration	EGP
enumeration	IRR
enumeration	UAH
enumeration	DZD
enumeration	NPR
enumeration	BDT
enumeration	GHS
enumeration	CHF
enumeration	SOS
enumeration	PAB
enumeration	PYG
enumeration	KES
enumeration	UZS
enumeration	QAR
enumeration	VES
enumeration	NAD
enumeration	KZT
enumeration	AUD
enumeration	KYD

	enumeration	LAK
	enumeration	BTN
	enumeration	CDF
	enumeration	PEN
	enumeration	HKD
	enumeration	SYP
	enumeration	GEL
	enumeration	LKR
	enumeration	KPW
	enumeration	YER
	enumeration	XAF
	enumeration	COU
	enumeration	MNT
	enumeration	MAD
	enumeration	SGD
	enumeration	UYW
	enumeration	NZD
	enumeration	TND
	enumeration	IQD
	enumeration	GTQ
	enumeration	GNF
	enumeration	LRD
	enumeration	XSU
	enumeration	MDL
	enumeration	XAU
	enumeration	KMF
	enumeration	BND
	enumeration	OMR
	enumeration	XTS
	enumeration	SAR
	enumeration	CLF
	enumeration	BRL
	enumeration	GIP
	enumeration	BOB
	enumeration	JOD
	enumeration	XBA
	enumeration	SZL
	enumeration	VND
	enumeration	XAG
	enumeration	MKD
	enumeration	THB
	enumeration	ALL
	enumeration	SVC
	enumeration	DKK
	enumeration	XXX
	enumeration	XPD
	enumeration	MYR
	enumeration	ZAR
Used by	Complex Type	salaryAgreementType

Source	`<xs:attribute name="currency" use="required" type="currencyType">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the currency for the agreed on wage. Example: "SEK"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `testimonialType / @href`

Namespace	No namespace
Annotations	`State the URI to the testimonial.`
Type	xs:anyURI
Properties	use: `optional`
Used by	Complex Type testimonialType

Source	`<xs:attribute name="href" use="optional" type="xs:anyURI">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI to the testimonial.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `referenceType / @href`

Namespace	No namespace
Annotations	`State the URI to the reference.`
Type	xs:anyURI
Properties	use: `optional`
Used by	Complex Type referenceType

Source	`<xs:attribute name="href" use="optional" type="xs:anyURI">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI to the reference.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `relationToParentType / relationEntry / @parentOrganisationReference-Code`

Namespace	No namespace
Annotations	`State the parent organisation's reference code.`
Type	xs:string
Properties	use: `required`
Used by	Element relationToParentType/relationEntry

Source	`<xs:attribute name="parentOrganisationReferenceCode" type="xs:string" use="required">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the parent organisation's reference code.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute `relationToParentType / @typeOfRelation`

Namespace	No namespace
Annotations	`Describes the relation to the related XML-document. Fixed value: "hierarchical-child"`
Properties	use: `required` fixed: `hierarchical-child`
Used by	Complex Type relationToParentType

Source	`<xs:attribute name="typeOfRelation" use="required" fixed="hierarchical-child">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describes the relation to the related XML-document. Fixed value: "hierarchical-child"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute relationToParentType / @href

Namespace	No namespace
Annotations	State the URI to the parental XML-document.
Type	xs:anyURI
Properties	use: optional
Used by	Complex Type relationToParentType

Source	`<xs:attribute name="href" use="optional" type="xs:anyURI">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI to the parental XML-document.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute relationToChildType / relationEntry / @childOrganisationReferenceCode

Namespace	No namespace
Annotations	State the child document's reference code.
Type	xs:string
Properties	use: required
Used by	Element relationToChildType/relationEntry

Source	`<xs:attribute name="childOrganisationReferenceCode" type="xs:string" use="required">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the child document's reference code.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute relationToChildType / @typeOfRelation

Namespace	No namespace
Annotations	Describes the relation to the related XML-document. Fixed value: "hierarchical-parent"
Properties	use: required fixed: hierarchical-parent
Used by	Complex Type relationToChildType

Source	`<xs:attribute name="typeOfRelation" use="required" fixed="hierarchical-parent">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">Describes the relation to the related XML-document. Fixed value:`
	` "hierarchical-parent"</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`

Attribute relationToChildType / @href

Namespace	No namespace
Annotations	State the URI to the child XML-document.
Type	xs:anyURI
Properties	use: optional
Used by	Complex Type relationToChildType

Source	`<xs:attribute name="href" use="optional" type="xs:anyURI">`
	`  <xs:annotation>`
	`  <xs:documentation xml:lang="en">State the URI to the child XML-document.</xs:documentation>`
	`  </xs:annotation>`
	`</xs:attribute>`