

Quality Assurance/Quality Assessment of the Sumex1 Implementation of generalFormRequest450

This form contains all semantically well defined addresses and structures and serves primarily as a workpiece for the implementation of generalFormRequest450.

Furthermore the concept of a 'constraint' is demonstrated, which is primarily used to define an address or a structure. For example, an insurance address can be given a further binding as UVG insurance. Or a specific laboratory value like 'HbA1c' can be required.

A complete API documentation can be found at http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/index.html

The two distinct specific addresses: patient address vs other address types

patientS1Address

birthDate	ssn
sex	
salutation	
title	
firstName	
lastName	
subaddressing	
pobox	
street	
zip	city
countryCode	stateCode
email	
url	
fax	phone[0]
phone[1]	phone[2]

AddressRoleType = enAddressRolePatient
 (http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/AddressRoleType_type.html)

The address type iterator *IGeneralFormRequest::GetFirstAddressRole()* / *IGeneralFormRequest::GetNextAddressRole* announces this address as *enAddressRolePatient*

(http://sumex1.tmr.net/en/xmlmanagers/generalformrequest/api_cpp/v450/IGeneralFormRequest/GetFirstAddressRole_method.html)

The name of the fields serve as semantical binding for the data constructor

IGeneralFormRequest::SetPatient()
 (http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/IGeneralFormRequest/SetPatient_method.html)

physicianS1Address

ean	nif
zsr	
salutation	title
firstName	
lastName	
companyName	
departmentName	
subaddressing	
pobox	
street	
zip	city
countryCode	stateCode
email	
url	
fax	phone[0]
phone[1]	phone[2]

AddressRoleType = enAddressRolePhysician
 (http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/AddressRoleType_type.html)

The complete list of AddressRoleTypes:

The address has fields for defining a personal or a company address. The mapping onto the personal or company fields is transparently done based on input.

(http://sumex1.tmr.net/en/xmlmanagers/generalformrequest/api_cpp/v450/IGeneralFormRequest/AddAddress_method.html)

All semantically well defined structures

In the following, all semantically well-defined structures are shown/defined, whereby the semantic binding is defined via the `StructRoleType` variable (cf. right side).

Analogous to the addresses the structures are announced by the `StructType` iterator `IGeneralFormRequest::GetFirstStructRole()` / `IGeneralFormRequest::GetNextStructRole` and set by the corresponding data binding method, e.g. `IGeneralFormRequest::SetLaw()`.

The complete list of `StructRoleTypes`:

cardS1Struct	
cardID	
expiryDate	validationDate
validationID	
validationServer	

StructRoleType = enStructRoleCard
(http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/StructRoleType_type.html)

The form data binding is done by the `IGeneralRequest::SetCard()` method

lawS1Struct	
type	
caseID	caseDate
insuredID	
contractNumber	

StructRoleType = enStructRoleLaw
(http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/StructRoleType_type.html)

The form data binding is done by the `IGeneralRequest::SetLaw()` method

treatmentS1Struct	
treatmentType	
treatmentCanton	
treatmentReason	
beginDate	endDate

StructRoleType = enStructRoleTreatment
(http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/StructRoleType_type.html)

The form data binding is done by the `IGeneralRequest::SetTreatment()` method

unemployabilityS1Struct	
date	
beginDate	endDate
percentageNum	
remark	

StructRoleType = enStructRoleUnemployability
(http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/StructRoleType_type.html)

The form data binding is done by the `IGeneralRequest::AddUnemployability()` method

medicationS1Struct	
type	
code	
name	
beginDate	endDate

StructRoleType = enStructRoleMedication
(http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/StructRoleType_type.html)

The form data binding is done by the `IGeneralRequest::AddMedication()` method

diagnosisS1Struct	
type	
code	
name	
date	active <input type="checkbox"/>
remark	

StructRoleType = enStructRoleDiagnosis
(http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/StructRoleType_type.html)

The form data binding is done by the `IGeneralRequest::AddDiagnosis()` method

allergyS1Struct	
type	
severity	date
code	
name	

StructRoleType = enStructRoleAllergy
(http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/StructRoleType_type.html)

The form data binding is done by the `IGeneralRequest::AddAllergy()` method

All semantically well defined structures... continuation

vaccinationS1Struct

type	
code	
name	
date	lotNumber
remark	

StructRoleType = enStructRoleVaccination
(http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/StructRoleType_type.html)

The form data binding is done by the `IGeneralRequest::AddVaccination()` method

textResultS1Struct

type	
code	
name	
date	
value	
reference	
remark	

StructRoleType = enStructRoleTextResult
(http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/StructRoleType_type.html)

The form data binding is done by the `IGeneralRequest::AddTextResult()` method

numericResultS1Struct

type	
code	
name	
date	
value	unit
lowerReference	upperReference
remark	

StructRoleType = enStructRoleNumericResult
(http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/StructRoleType_type.html)

The form data binding is done by the `IGeneralRequest::AddNumericResult()` method

documentS1Struct

file	
title	
mimeType	

StructRoleType = enStructRoleDocument
(http://sumex1.net/en/xmlmanagers/generalformrequest/api_cpp/v450/types/StructRoleType_type.html)

The form data binding is done by the `IGeneralRequest::AddDocument()` method

Higher cardinality of a specific address or structure

With certain forms, an address or structure can occur several times, e.g. a chronological list of unemployabilities. The cardinality of a specific address or structure is expressed in the announcement via `IGeneralFormRequest::GetFirstAddressRole()` / `IGeneralFormRequest::GetNextAddressRole` resp. `IGeneralFormRequest::GetFirstStructRole()` / `IGeneralFormRequest::GetNextStructRole` via the variable **IMaxOccurs**, while **lIndex** indicates the current array index (0 based!).

unemployabilityS1Struct[1]

beginDate		endDate	
percentageNum	%		<i>lIndex=1; lMaxOccurs=3</i>

unemployabilityS1Struct[2]

beginDate		endDate	
percentageNum	%		<i>lIndex=2; lMaxOccurs=3</i>

Note, that `unemployabilityS1Struct[0]` is the structure fully defined on the previous page!

Higher cardinality with subspecification by using the constraint information

There are forms, where an address or structure appears more than once, but not in the sense of a classical list. E.g. 3 insurance addresses can be required, one per given law or 2 specific laboratory values like 'HbA1C' and 'creatinine' are required. Such subspecified conditions are defined by a constraint.

constraint:

insuranceS1Address[0]

condensedName
condensedAddress

should be a KVG insurance address

constraint:

insuranceS1Address[1]

condensedName
condensedAddress

should be an UVG insurance address

Request of a specific numeric lab result by a constraint

constraint:

numericResultS1Struct[1]

type
code
name
date
value
unit

constraint:

numericResultS1Struct[2]

type
code
name
date
value
unit

Request specific attachments by a constraint

constraint:

documentS1Struct

file

title

mimeType

The attachments required in the 'constraint' are given/requested CRLF-separated, e.g. each line is one requested type. The constraints are formulated in the language of the form.

confirm this as a test scenario with data from a non-existing patient. **Target mail:**

Electronic transmission

Click the «Send» button to send the PDF form to the specified e-mail for test purposes.
Only mail addresses of software implementers are allowed to avoid abuse.