Transformation Manager

(Description of component functionality)

Last modification date: 12.7.2013
Version: 0.7

Change History				
Date	**Version**	**Status**	**Author**	**Details**
25.02.2013	0.1	draft	Milada Kovárová	Revision of functional specification document from epSOS1 (WP3.9 – Specification of the common component_TM.doc)
27.02.2013	0.2	draft	Milada Kovárová	Adoptions necessary for epSOS 2 services
15.3.2013	0.3	draft	Milada Kovárová	Insert of coded element list XML Append implementation tasks
15.4.2013	0.4	draft	Milada Kovárová	Warnings in validation failure
11.5.2013	0.5	draft	Milada Kovárová	Model based validation added.
11.7.2013	0.6	draft	Milada Kovárová	Document review and consolidation
12.7.2013	0.7	draft	Milada Kovárová	Comments to validation process and configuration

1 Transformation Manager use cases

Transformation Manager is responsible for data transformation from a national language to the epSOS Reference Terminology and from the epSOS Reference Terminology to a national language. It deals mainly with syntax of original content and epSOS pivot content making use of TSAM component for transcoding/translation of particular coded elements.
Two steps made by transformation are distinguished by name. First processing is called **transcoding**, because national codes are changed to epSOS codes (with English designations). Second step is called **translation**, because designations in target language are added into document.

1.1 Use case 1 – Transcoding of original content

Context:
NCP transcoding an Original content to a epSOS pivot content. For each coded element in the Original content corresponding epSOS concept has to be found.
Original content can be either original CDA level 3 (fully structured content) or original CDA level 1 (content embedding PDF).

Typical Scenario:

1. Component receives the Original content.
2. TM validates this data whether they are provided in format compliant to epSOS CDA syntax (epSOS friendly document).
3. If data are valid, TM looks for all coded elements and transcode them into the epSOS Reference Terminology (calling TSAM component).
4. TM compiles the epSOS pivot content containing both original codes and display name and codes and display name from epSOS Reference Terminology.
5. TM validates finalized epSOS pivot content.
6. After validation the epSOS pivot content will be provided as output.

Alternative Scenario:
In case of original CDA level 1 (embedding PDF), only header coded elements will be transcoded, unstructured body will be preserved.

1.2 Use case 2 – Translation of epSOS pivot content

Context:
NCP receives epSOS pivot content and its task is to translate it to target language.

Typical Scenario:

1. Component receives the epSOS pivot content.
2. TM validates this data to be valid epSOS pivot content.
3. TM looks for all coded elements and translate their display names to specified language (calling TSAM component)
4. TM compiles the epSOS pivot content by adding display name in target language to both original codes and display name (of country of origin) and codes and display name from epSOS Reference Terminology.
5. TM validates finalized content.
6. After validation the Translated pivot content will be provided as output.

Alternative Scenario:

In case of pivot CDA level 1 content (embedding PDF), only header coded elements will be translated, unstructured body will be preserved.

2 Usage of Transformation Manager

Transformation Manager is used as a component of National Contact Point (NCP), both in country A (country of patient origin) and country B (country of healthcare providing).

For exchange of

- Patient Summary (PS)
- ePrescription (eP)
- Medication Related Overview (MRO)

transcoding of original content (epSOS friendly format) is made in country A and translation of epSOS pivot content in country B.

For exchange of

- eDispensation (eD)
- Healthcare Encounter Report (HCER)

transcoding of original content (epSOS friendly format) is made in country B and translation of epSOS pivot content in country A.

In case of Patient Access (PAC) both transcoding and translation are made in NCP of country B.

3 Document transformation process

This section briefly describes the transformation process of CDA structures during transformation and translation.
In both cases, the subject of transformation is either original CDA level 3 fully structured content or original CDA level 1 content embedding PDF.

The basic difference between two transformation performed by TM, namely transcoding and translation, is in usage of concepts from Local Terminology Repository (LTR).
In case of transcoding, each code from source document is from local code system of document originating country. This code system can be either the same as epSOS uses in epSOS Master Valueset Catalogue (MVC) or it can be of any local code system for which there is a mapping to a code system used in MVC. Then, there are two possibilities of processing. First, if code is not from MVC code systems, mapping in LTR is used and this code is replaced by the “equivalent” code from epSOS MVC with all attributes of this code. Second, codes from epSOS code systems are not changed, just their designations are translated to English as epSOS central language.
The translation case is simpler. As it is always applied to codes from epSOS MVC code systems, codes are never changed, just designations related to these codes in target language are appended.

Transcoding process

Each MS has to provide its original data in format of epSOS friendly CDA level 3 (fully structured content) and/or epSOS friendly CDA level 1 content embedding PDF.
In case both CDA level 3 and CDA level 1 documents are provided, transformation is made in two steps, because each type of content is processed separately. But the process is the same differing only in expected data structure and set of coded elements to be transcoded.

Input: Original content:
either original CDA level 3 fully structured content
or original CDA level 1 content embedding PDF

Output: epSOS Pivot content with all coded elements transcoded into code systems used by epSOS and translated into English (as epSOS central language)
either pivot CDA level 3 fully structured content
or pivot CDA level 1 content embedding PDF

Process:

1. Receive original content (epSOS friendly document) provided by national infrastructure to NCP
2. Validate provided original content using schema to confirm it is compliant with epSOS format
3. Parse provided data and for each coded element
   3.1. extract data about concept (code, code system OID, code system name and if provided code system version)
   3.2. use TSAM to transcode the concept
   3.3. receive response from TSAM
   3.4. create translation element and add it to a proper element
4. Validate final Pivot content using schematron
5. Provide final Pivot content

Translation process

Task of translation process is to add translations (i.e. target language designations) of all coded elements in the pivot content without any other change of the document structure.

Input: epSOS pivot content
either pivot CDA level 3 fully structured content
or pivot CDA level 1 content embedding PDF

Output: translated epSOS pivot content with all coded elements translated into target language
either pivot CDA level 3 fully structured content
or pivot CDA level 1 content embedding PDF

Process:

1. Receive epSOS pivot content
2. Validate provided epSOS pivot content using schema to confirm it is compliant with epSOS format
3. Validate provided epSOS pivot content using schematron to confirm relations to reference to terminology are correct
4. Parse provided epSOS pivot content and for each coded element
   4.1. extract data about concept (code, code system OID, code system name and if provided code system version)
   4.2. use TSAM to translate the concept
   4.3. receive response from TSAM
   4.4. create translation element and add it to a proper element
5. Validate final translated pivot content using schematron
6. Provide final translated pivot content

3.1 Impact on CDA documents

Format and content of epSOS compliant CDA document is defined in documents

- D3.5.2 Semantic Services Definition and its appendices B and C
- Work Package 3.9 epSOS Semantic Implementation Guide
- epSOS Architecture and Design, EED DESIGN – Specification HCER
- epSOS Architecture and Design, EED DESIGN – Specification MRO

The process of translation/transcoding of particular coded element is presented in following example:

PosAm, spol. s r. o., Odborárska 21, 831 02 Bratislava, Slovenská republika
tel.: +421-2-49 23 91 11 ● fax: +421-2-49 23 98 88 ● posam@posam.sk ● www.posam.sk
Bratislava ● Praha ● Banská Bystrica ● Košice ● Levice ● Žilina
IČO: 31 365 078, Obchodný register vedený Okresným súdom Bratislava I, Oddiel: Sro, Vložka č.: 6342/B

	Country A (Slovakia)	NCP A -> NCPB	Country B (Austria)
Document	Document A	Document E	Document B
Code System	SNOMED CT	ICD10	ICD10
Code	230291001	G20	G20
Language	Slovak	English	German
DisplayName	juvenilná Parkinsonova choroba	Parkinson's disease	Primäres Parkinson-Syndrom
CDA schema	<value xsi:type="CE" code="230291001" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED CT" displayName="juvenilná Parkinsonova choroba"> </value>	<value xsi:type="CE" code="**G20**" codeSystem="**2.16.840.1.113883.6.90**" codeSystemName="**ICD10**" displayName="**Parkinson's disease**"> <translation code="230291001" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED CT" displayName=" juvenilná Parkinsonova choroba"/> </value>	<value xsi:type="CE" code="G20" codeSystem="2.16.840.1.113883.6.90" codeSystemName=" ICD10" displayName="**Primäres Parkinson-Syndrom**"> <translation displayName="Parkinson's disease"> <translation code="230291001" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED CT" displayName=" juvenilná Parkinsonova choroba"/> </translation> </value>

Element <translation>

As a result of transformation, for each coded element, nested translation elements will be created. These elements are necessary so that the original code from original code system in which the document was coded is preserved. Usually, it keeps also English translation of epSOS coded concept. This will enable the receiver to see the document in its own language, even if at a higher granular level. At the same time, the original code and English version of epSOS code are kept, enabling the receiver to look it up if needed.

Translation element contains the same set of information as the original coded element, i.e. code, codeSystem (OID), codeSystemName, displayName.

```
<translation
        code="230291001"
        codeSystem="2.16.840.1.113883.6.96”
        codeSystemName="SNOMED CT"
        displayName="Juvenile Parkinson’s disease"/>
```

As a result of transcoding, translation element will be nested into original coded element (first nested level inside original coded element).

```
<value xsi:type="CE"
        code="G20"
        codeSystem="2.16.840.1.113883.6.90"
        codeSystemName="ICD10"
        displayName="Parkinson’s disease"">
        <translation
                code="230291001"
                codeSystem="2.16.840.1.113883.6.96"
                codeSystemName="SNOMED CT"
                displayName="juvenilná Parkinsonova choroba"/>
</value>
```

As a result of translation, another translation element will be created and nested below original coded element (first nested level inside original coded element). Already existing translation element created by transcoding in previous step, will be shifted one level below and nested to the new translation element (second nested level inside original coded element).

```
<value xsi:type="CE"
        code="G20"
        codeSystem="2.16.840.1.113883.6.90"
         codeSystemName="ICD10"
        displayName="Primäres Parkinson-Syndrom">
        <translation
                code="G20"
                codeSystem="2.16.840.1.113883.6.90"
                codeSystemName=" ICD10"
                displayName="Parkinson’s disease">
                 <translation
                        code="230291001"
                        codeSystem="2.16.840.1.113883.6.96"
                        codeSystemName="SNOMED CT"
                        displayName=" juvenilná Parkinsonova choroba"/>
                </translation>
</value>
```

If some of the data is the same as in the element having nested translation element, these should not be repeated again.

```
<value xsi:type="CE"
        code="230291001"
        codeSystem="2.16.840.1.113883.6.96"
        codeSystemName="SNOMED CT"
        displayName="Juvenile Parkinson's disease">
        <translation
                displayName="juvenilná Parkinsonova choroba"/>
</value>
```

If original coded element contains other nested elements, these will be kept without any change in transformed document.

Note:
Except from described changes in coded elements, no other changes in the original data are performed.
It was decided, that even transformation adds new data in target language and in English, element languageCode of the document will remain the original language of country of origin in which the data were provided.

Example of transformation process
This section shows on example of one coded element, how transformation in Country A change its structure and content.

Country A provides epSOS compliant CDA document with original data containing coded element:

```
<value xsi:type='CE'
        code="230291001"
        codeSystem="2.16.840.1.113883.6.96"
        codeSystemName="SNOMED CT"
        codeSystemVersion="July2009"
        displayName="juvenilná Parkinsonova choroba">
        <originalText>
                <reference value="#a1"/>
        </originalText>
</value>
```

Transformation in Country A
As a result of transformation in country A, new <translation> element will be added to coded element. Data obtained from TSAM as transcoding of coded concept replace original data of the element and the original data are stored in <translation> element. All other elements related to transformed coded element remains unchanged.

```
<value xsi:type='CE'
        code="G20"
        codeSystem="2.16.840.1.113883.6.90"
        codeSystemName="ICD10"
        codeSystemVersion="2007"
        displayName="Parkinson's disease">
        <originalText>
                <reference value="#a1"/>
        </originalText>
```

```
        <translation
                code="230291001"
                codeSystem="2.16.840.1.113883.6.96"
                codeSystemName="SNOMED CT"
                codeSystemVersion=”July2009”
                displayName="juvenilná Parkinsonova choroba"/>
    </value>
```

Transformation in Country B

As a result of transformation in country B, new <translation> element will be added to coded element. Data obtained from TSAM as transcoding of coded concept replace original data of the element and the original data are stored in another <translation> element. Already existing <translation> will be nested into the new one.

```
    <value xsi:type='CE'
        code="G20"
        codeSystem="2.16.840.1.113883.6.90"
        codeSystemName="ICD10"
        codeSystemVersion=”2007”
        displayName="Primäres Parkinson-Syndrom">
        <originalText>
            <reference value="#a1"/>
        </originalText>
        <translation
            code="G20"
            codeSystem="2.16.840.1.113883.6.90"
            codeSystemName="ICD10"
            codeSystemVersion=”2007”
            displayName="Parkinson’s disease">
            <translation
                code="230291001"
                codeSystem="2.16.840.1.113883.6.96"
                codeSystemName="SNOMED CT"
                codeSystemVersion=”July2009”
                displayName="juvenilná Parkinsonova choroba"/>
        </translation>
    </value>
```

Information about code, code system and its version does not have to be repeated in added <translation> element, because it is the same as in original data.

```
    <value xsi:type='CE'
        code="G20"
        codeSystem="2.16.840.1.113883.6.90"
        codeSystemName="ICD10"
        codeSystemVersion=”2007”
        displayName="Primäres Parkinson-Syndrom">
        <originalText>
            <reference value="#a1"/>
        </originalText>
        <translation
            displayName="Parkinson’s disease">
            <translation
                code="230291001"
```

```
                codeSystem="2.16.840.1.113883.6.96"
                codeSystemName="SNOMED CT"
                codeSystemVersion="July2009"
                displayName="juvenilná Parkinsonova choroba"/>
        </translation>
    </value>
```

Reference coded system used in Country A

There is a special case, when Country A uses the same code system as epSOS chosen for reference system. In that case, there is no need for transcoding to another code system in country A. Transformation will just add English display name in such case.

Then, format of resulting CDA element could look like:

```
    <value xsi:type='CE'
        code="43116000"
        codeSystem="2.16.840.1.113883.6.96"
        codeSystemName="SNOMED CT"
        codeSystemVersion="July2009"
        displayName="Eczema">
        <translation
            displayName=" vyrážka "/>
    </value>
```

As a result of transformation in country B, new <translation> element will be added to coded element. Data obtained from TSAM as transcoding of coded concept replace original data of the element and the original data are stored in another <translation> element. Already existing <translation> will be nested into the new one.

```
    <value xsi:type='CE'
        code="43116000"
        codeSystem="2.16.840.1.113883.6.96"
        codeSystemName="SNOMED CT"
        codeSystemVersion="July2009"
        displayName="Ekzem">
        <translation
            displayName="Eczema">
            <translation
                displayName="vyrážka"/>
        </translation>
    </value>
```

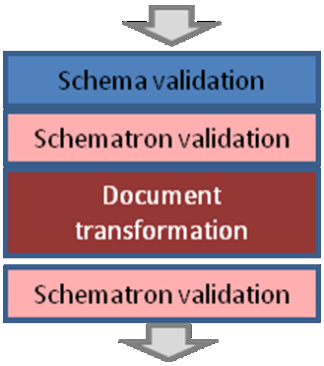
3.2 CDA validation

Transformation Manager validates CDA documents using

- schema – structure of the document,
- schematron – semantic bindings,
- model based validator – model specification (cover both structure and semantics).

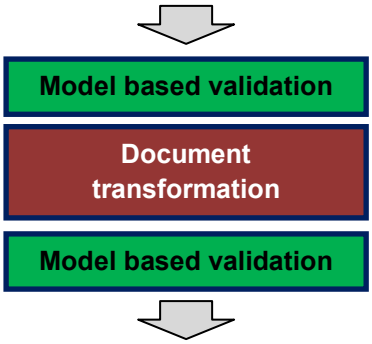
As latter model based validation is full-value replacement of both schema and schematron validations, there are two possibilities how to validate CDA documents.

A. In case of both transcoding and translation, input document will be validated using schema and schematron, and output document just using schematron.



Validation will be based on schemas and schematrons provided by epSOS semantic group and IHE.
Schemas and sample documents are published on
https://gforge.inria.fr/scm/viewvc.php/branches/epSOS/?root=gazelle

B. In case of both transcoding and translation, CDA document will be validated using model based validator on input and output as well.



Model based validator is provided by IHE (http://gazelle.ihe.net/epSOS/validator/).

3.2.1 Validation based on schema

Schemas for validation of input documents will be provided by WP3.9 and IHE.
It will validate structure of

- Patient Summary document (CDA level 3)
- Patient Summary document (CDA level 1 embedding PDF)
- ePrescription document (CDA level 3)
- ePrescription document (CDA level 1 embedding PDF)
- eDispensation document (CDA level 3)
- eDispensation document (CDA level 1 embedding PDF)
- Healthcare Encounter Report document (CDA level 3)
- Healthcare Encounter Report document (CDA level 1 embedding PDF)
- Medication Related Overview document (CDA level 3)
- Medication Related Overview document (CDA level 1 embedding PDF)

If validation of document fails transformation continues, just a validation warning is reported.

For schema validation javax.xml.validation.Validator for CDA parsed as org.w3c.dom.Document is used.

3.2.2 Validation using schematrom

Schematrons for validation of documents will be provided by WP3.9 and IHE.

It will validate

- Patient Summary document (CDA level 3)
- Patient Summary document (CDA level 1 embedding PDF)
- ePrescription document (CDA level 3)
- ePrescription document (CDA level 1 embedding PDF)
- eDispensation document (CDA level 3)
- eDispensation document (CDA level 1 embedding PDF)
- Healthcare Encounter Report document (CDA level 3)
- Healthcare Encounter Report document (CDA level 1 embedding PDF)
- Medication Related Overview document (CDA level 3)
- Medication Related Overview document (CDA level 1 embedding PDF)

All documents will be validated with respect to both epSOS reference terminology and local terminology bindings.

If validation of document fails transformation continues, just a validation warning is reported.

In relation to transcoding and translation, when CDA level 3 document is validated for each type of document there is always schematron for epSOS friendly document and for epSOS pivot document. In case of CDA level 1 containing PDF schematrons for scanned documents are used regardless the document type. Validation is performed twice, on input document and after transformation on output document.
Before start of transcoding input document is validated with schematron for epSOS friendly document. When transcoding finishes output document is validated with schematron for epSOS pivot document.
Before start of transcoding translation, input document is validated with schematron for epSOS pivot document. At the end of transformation process output document is validated with schematron for epSOS friendly document.

3.2.3 Validation using Model based validator

Model based validator validates the same set of documents as schematron validator in previous chapter.

All documents will be validated with respect to both epSOS reference terminology and local terminology bindings.

If validation of document fails transformation continues, just a validation warning is reported.

In relation to transcoding and translation, for each type of document (separately for CDA level 3 and CDA level 1 containing PDF), input parameter of validator is set. Validation is performed twice, on input document and after transformation on output document.
In case of CDA level 3 documents validator is chosen based on document type:

- By transcoding, input document is validated with validator for epSOS friendly document (e.g. “epSOS - eDispensation Friendly”). Output document is validated with validator for epSOS pivot document (e.g. “epSOS - eDispensation Pivot”).
- By translation, input document is validated with validator for epSOS pivot document. Output document is validated with validator for epSOS friendly document.

In case of CDA level 1 documents validator for scanned documents is used:

- By transcoding, input document is validated with validator “epSOS - Scanned Document Friendly”. Output document is validated with validator “epSOS - Scanned Document Pivot”.
- By translation, input document is validated with validator “epSOS - Scanned Document Pivot”. Output document is validated with validator “epSOS - Scanned Document Friendly”.

4 Detailed function specification

Transformation Manager deals with two main functions:

- transcoding of original data to epSOS CDA pivot document
- translation of epSOS CDA pivot document

This section describes functional specification of how these functions will be implemented.

Transformation to epSOS CDA pivot document is covered by API method:

- toEpSOSPivot(EpSOSOriginalData; TargetLanguageCode):ResponseStructure

(it is used when NCP is in role of Country A for PS, eP, MRO and of Country B for eD and HCER)

Transformation to translated epSOS CDA pivot document is covered by API method:

- translate(EpSOSCDA; TargetLanguageCode):ResponseStructure

(it is used when NCP is in role of Country B for PS, eP, MRO and of Country A for eD and HCER)

Both methods can be used for transformation of structured CDA level 3 document and also CDA document embedding PDF.

4.1 Identification of coded elements

Parsing provided CDA document, for each transformation coded elements have to be identified, which will be sent to TSAM for transcoding or translation.
Any element can be can be considered as a coded element when it has:

- code and codeSystem attributes,
- with type CD, CE,
- corresponding to valueset from Master ValueSet Catalogue.

To be able to handle all possibilities and produce comprehensive transformation, identification of possible coded elements will consider all these aspects.
To follow relation between coded element and valueset, configuration file called CodedElementList will be used. It will contain list of all coded elements, and for each of them:

- name of coded element (xpath),
- indication, in which pivot document type coded element is used (Patient Summary, ePrescription, eDispensation, HCER, MRO and for each of them if it is in CDA level 3 or CDA level 1 embedding pdf and whether element is required or optional),
- related value set (and value set version) [optional],
- language to which element should be translated [optional].

The process of coded element identification will be as follows:

1. Check for document type: element ClinicalDocument/code with value from epSOSDocumentCode valueset

57833-6	ePrescription
60593-1	eDispensation
60591-5	Patient Summary
34133-9	Healthcare Encounter Report
56445-0	Medication Related Overview

2. Check content of the document (whether document has structured or unstructured body)
3. For each entry in CodedElementList, check if it is applicable for a particular document type and content (element <usage>)

4. If applicable, find corresponding element in CDA document which is considered to be CodedElement and contains data about concept, which should be transcoded / translated (if not found, report error or warning)
5. CodedElement has to have attributes “code” and “codeSystem” (if not, and CodedElement is required, report error; if not, and CodedElement is optional, report warning)
6. For CodedElement, check whether its type is one of {CD, CE} – if not, report warning
7. For each CodedElement TSAM can be called for transcoding or translation.

Structure of coded element information kept for coded element identification:

Name	Type	Optionality	Description
codedElement	element	mandatory	Main XML element of codedElement
elementPath	element	mandatory	Identification of coded element in form of xpath
usage	element	mandatory	Indication, in which pivot document types coded element is used. Each subelement contains optionality of coded element in a particular document type {R, RNFA, O, NA}
patientSummaryCDAl3	element	optional	Element is contained in Patient Summary CDA document (level 3)
patientSummaryCDAl1pdf	element	optional	Element is contained in header of Patient Summary CDA document (level 1 with pdf)
ePrescriptionCDAl3	element	optional	Element is contained in ePrescription CDA document (level 3)
ePrescriptionCDAl1pdf	element	optional	Element is contained in header of ePrescription CDA document (level 1 with pdf)
eDispensationCDAl3	element	optional	Element is contained in eDispensation CDA document (level 3)
eDispensationCDAl1pdf	element	optional	Element is contained in header of eDispensation CDA document (level 1 with pdf)
HCERDocCDAl3	element	optional	Element is contained in HCER CDA document (level 3)
HCERDocCDAl1pdf	element	optional	Element is contained in header of HCER CDA document (level 1 with pdf)
MRODocCDAl3	element	optional	Element is contained in MRO CDA document (level 3)
MRODocCDAl1pdf	element	optional	Element is contained in header of MRO CDA document (level 1 with pdf)
valueSet	element	optional	Relation of coded element to a particular ValueSet from MVC
valueSetVersion	element	optional	Version of ValueSet
targetLanguageCode	element	optional	When translating coded element and there is an exception in translation of a particular coded element, target language code may be specified.

<codedElement>
<elementPath>

```
            entry/procedure[templateId/@root= '1.3.6.1.4.1.19376.1.5.3.1.4.19']/code/@code
        </elementPath>
        <usage>
            <patientSummaryCDAI3>R</patientSummaryCDAI3>
            <patientSummaryCDAI1pdf>NA</patientSummaryCDAI1pdf>
            <ePrescriptionCDAI3>RNFA</ePrescriptionCDAI3/>
            <ePrescriptionCDAI1pdf>NA</ePrescriptionCDAI1pdf/>
            <eDispensationCDAI3>O</eDispensationCDAI3/>
            <eDispensationCDAI1pdf>NA</eDispensationCDAI1pdf/>
            <HCERDocCDAI3>O</HCERDocCDAI3/>
            <HCERDocCDAI1pdf>NA</HCERDocCDAI1pdf/>
            <MRODocCDAI3>O</MRODocCDAI3/>
            <MRODocCDAI1pdf>NA</MRODocCDAI1pdf/>
        </usage>
        <valueSet>1.3.6.1.4.1.12559.11.10.1.3.1.42.10</valueSet>
        <valueSetVersion>1.3</valueSetVersion>
        <targetLanguageCode>de-AT</targetLanguageCode>
</codedElement>
```

Configuration file CodedElementList will be stored as XML file. Property file of component Transformation Manager (tm.properties) contains reference to this XML file.

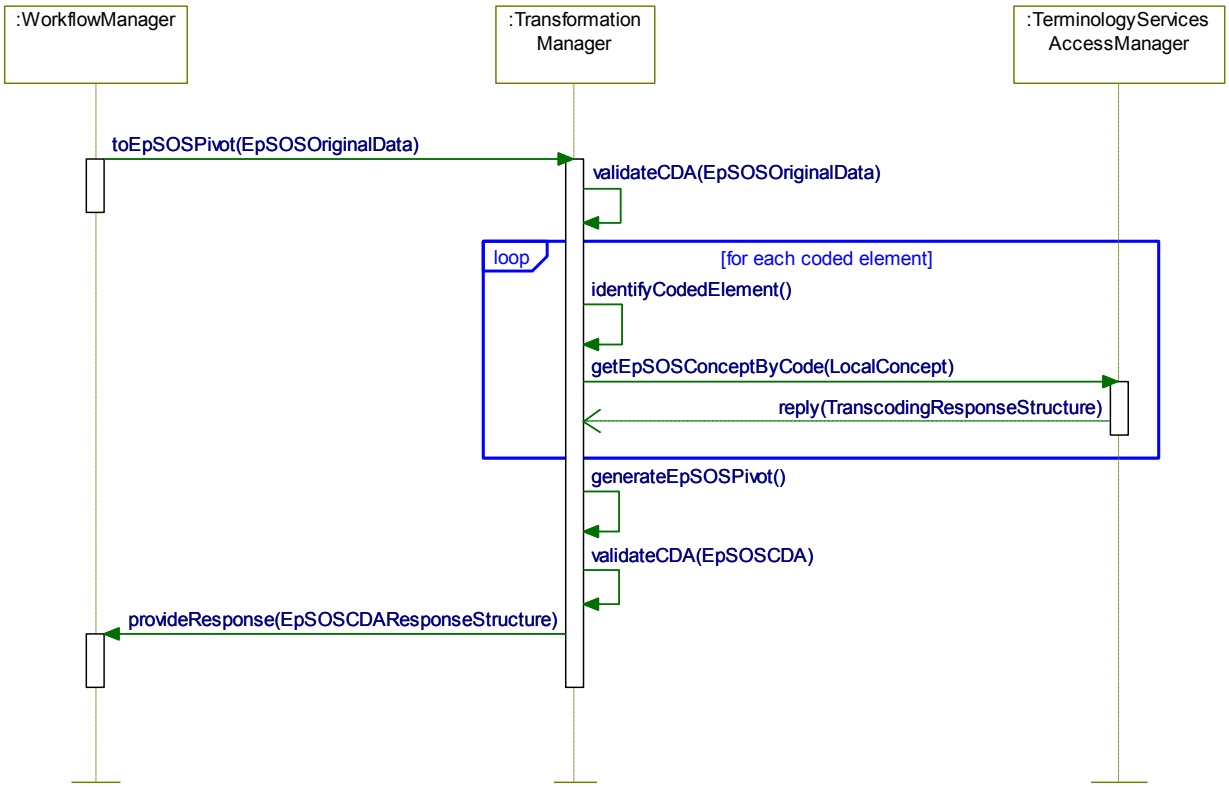
Note:
If identification of coded elements based on CodedElementList should be used, configuration parameter codedelementlist.enabled **has to be set to True and XML file containing CodedElementList has to be available.**
In other cases, all elements having attribute "code" will be considered as coded elements.

4.2 Transformation to epSOS pivot document

Transcoding process is described in section 1.1.

Transformation of Original content to epSOS pivot content is handled by API method ***toEpSOSPivot().***

4.2.1 Sequence diagram



4.2.2 Procedures and Algorithms

As an input for transcoding, Original content will be provided. It contains code elements in local code system of providing country and these have to be transcoded into epSOS reference terminology.

1. Check for document type: element ClinicalDocument/code with value from epSOSDocumentCode valueset:

Code	Document type	Property name
57833-6	ePrescription	tm.documenttype.eprescription
60593-1	eDispensation	tm.documenttype.edispensation
60591-5	Patient Summary	tm.documenttype.patientsummary
34133-9	Healthcare Encounter Report	tm.documenttype.hcer
56445-0	Medication Related Overview	tm.documenttype.mro

   Note:

   Current setting of codes is taken from tm.properties file following Property name in table above.
2. Check content, whether it has structured or unstructured body
3. Provided Original content will be validated using schema and schematron to check, that it has the same structure as epSOS CDA pivot document (i.e. contains all required elements - R, RNFA) and correct relations to local terminology

   Note:

   Schema and schematron validation is also configured via tm.properties file using following parameters:

Property name	Description
tm.schema.validation.enabled	True if validation using XSD schema is enabled, false otherwise (general for all

	transformation services)
tm.schemafilepath	Current path to XSD schema file for epSOS documents
tm.schematron.validation.enabled	True if validation using schematron is enabled, false otherwise (general for all transformation services)
tm.schematron.path.xsldir	Current path to schematron XSL directory
tm.schematron.path.patientsummary.friendly	Current path to a schematron file for original (epSOS friendly) content of Patient Summary
tm.schematron.path.eprescription.friendly	Current path to a schematron file for original (epSOS friendly) content of ePrescription
tm.schematron.path.edispensation.friendly	Current path to a schematron file for original (epSOS friendly) content of eDispensation
tm.schematron.path.hcer.friendly	Current path to a schematron file for original (epSOS friendly) content of Health care encounter report
tm.schematron.path.mro.friendly	Current path to a schematron file for original (epSOS friendly) content of Medication related overview

4. Based on document type and content, appropriate identification of coded elements should be used for the Original content (see section 4.1)
5. For each coded element,
   5.1. extract from Original content
      - Code
      - Code System OID
      - Code System Name (optional)
      - Code System Version (optional)
   5.2. find in CodedElementList
      - ValueSet
      - ValueSet Version (optional)
   5.3. create SourceConcept structure
   5.4. call TSAM: ***getEpSOSConceptByCode()*** using SourceConcept as input parameter
   5.5. receive from TSAM ResponseStructure
   5.6. from ResponseStructure, check result
   5.7. if result is “success”, extract content of ResponseElement
   5.8. check for duplicate information in original data and data from ResponseElement
   5.9. create <translation> element containing non duplicated data (if any) and append it to the coded element
6. Prepare epSOS pivot content
7. Validate epSOS pivot content using schematron to check correct relations to epSOS reference terminology

   Note:

   Schematron validation is configured via tm.properties file using following parameters:

Property name	**Description**
tm.schematron.path.xsldir	Current path to schematron XSL directory
tm.schematron.path.patientsummary.pivot	Current path to a schematron file for pivot content of Patient Summary
tm.schematron.path.eprescription.pivot	Current path to a schematron file for pivot content of ePrescription
tm.schematron.path.edispensation.pivot	Current path to a schematron file for pivot

	content of eDispensation
tm.schematron.path.hcer.pivot	Current path to a schematron file for pivot content of Health care encounter report
tm.schematron.path.mro.pivot	Current path to a schematron file for pivot content of Medication related overview

8. Generate response structure

4.2.3 Errors and Warnings

Exception is considered to be an error, if

- required coded element (R, RNFA) was not transcoded (failure result from TSAM)

Exception will raise a warning, if

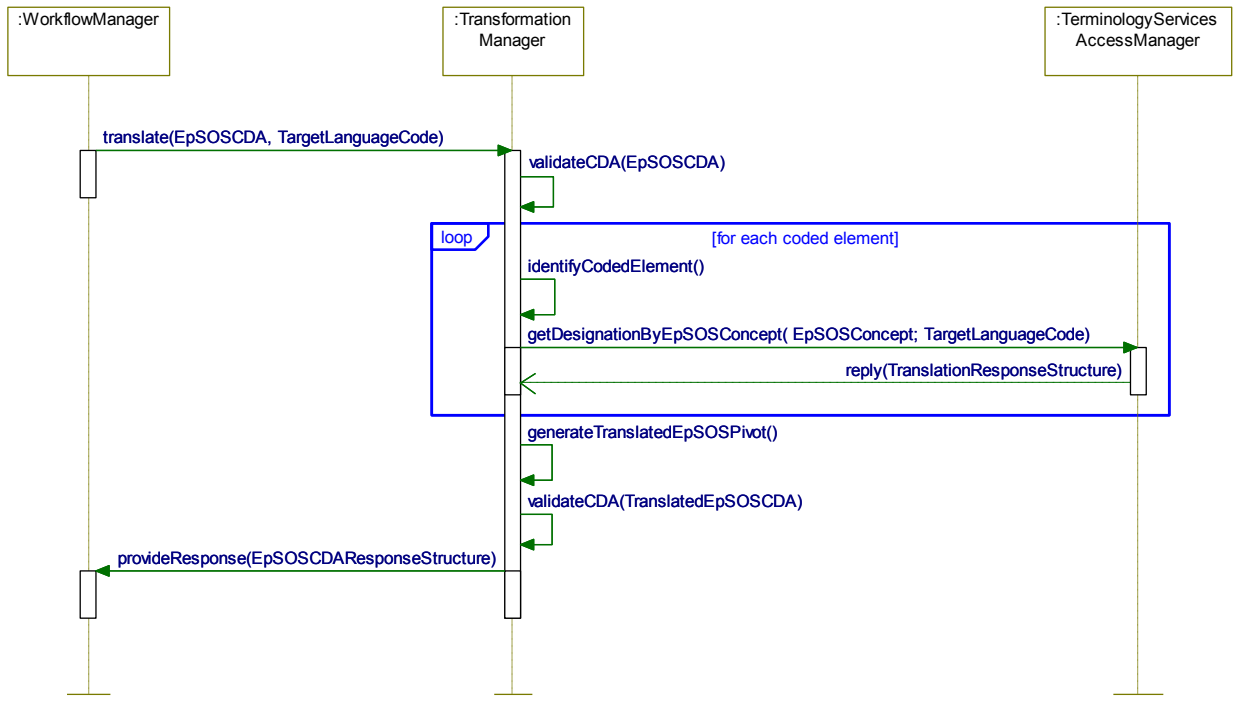
- Original content validation fails, or due to external conditions it is impossible to run validation (e.g. missing schematron)
- Pivot content validation fails, or due to external conditions it is impossible to run validation (e.g. missing schematron)
- By coded element identification, element does not have a proper type
- By coded element identification, element is not in CodedElementList
- optional coded element (O, NS) was not transcoded (failure result from TSAM)
- TSAM reported warnings

4.3 Translation of epSOS pivot document

Translation process is described in section **Error! Reference source not found.**1.2.

Transformation of epSOS pivot content to its translated version is handled by API method ***translate().***

4.3.1 Sequence diagram



4.3.2 Procedures and Algorithms

As an input for transformation, epSOS pivot content will be provided. Language code of target language can be also provided (if not, default TranslationLanguage set in properties file will be used). This content contains code elements in epSOS code system and their English display names. These have to be translated to required target language.

1. Check for document type: element ClinicalDocument/code with value from epSOSDocumentCode valueset

Code	Document type	Property name
57833-6	ePrescription	tm.documenttype.eprescription
60593-1	eDispensation	tm.documenttype.edispensation
60591-5	Patient Summary	tm.documenttype.patientsummary
34133-9	Healthcare Encounter Report	tm.documenttype.hcer
56445-0	Medication Related Overview	tm.documenttype.mro

Note:

Current setting of codes is taken from tm.properties file following Property name in table above.

2. Check content, whether it has structured or unstructured body
3. Provided pivot content will be validated using schema and schematron to check, that it has the structure of epSOS CDA pivot document (i.e. contains all required elements - R, RNFA) and correct relations to epSOS reference terminology

Note:

Schema and schematron validation is also configured via tm.properties file using following parameters:

Property name	Description
tm.schema.validation.enabled	True if validation using XSD schema is enabled, false otherwise (general for all transformation services)
tm.schemafilepath	Current path to XSD schema file for epSOS documents
tm.schematron.validation.enabled	True if validation using schematron is enabled, false otherwise (general for all transformation services)
tm.schematron.path.xsldir	Current path to schematron XSL directory
tm.schematron.path.patientsummary.pivot	Current path to a schematron file for pivot content of Patient Summary
tm.schematron.path.eprescription.pivot	Current path to a schematron file for pivot content of ePrescription
tm.schematron.path.edispensation.pivot	Current path to a schematron file for pivot content of eDispensation
tm.schematron.path.hcer.pivot	Current path to a schematron file for pivot content of Health care encounter report
tm.schematron.path.mro.pivot	Current path to a schematron file for pivot content of Medication related overview

4. Based on document type and content, appropriate identification of coded elements should be used for in the document (see section 4.1)
5. For each coded element,
   5.1. extract from pivot content
   - Code
   - Code System OID

- Code System Name (optional)
- Code System Version (optional)

5.2. find in CodedElementList
- Value Set
- Value Set Version (optional)

5.3. create SourceConcept structure
5.4. if there is a target language set for particular coded element in CodedElementList, use it as TargetLanguageCode for TSAM call
5.5. if no target language is set for particular element, but there is a language code provided from Workflow manager (input parameter of ***translate()***), use this language code as TargetLanguageCode for TSAM call
5.6. call TSAM: ***getDesignationByEpSOSConcept()*** using SourceConcept and if provided TargetLanguageCode as input parameters
5.7. receive from TSAM ResponseStructure
5.8. from ResponseStructure, check result
5.9. if result is "success", extract content of ResponseElement
5.10. check for duplicate information in original data and data from ResponseElement
5.11. create <translation> element containing non duplicated data (if any) and append it to coded element

6. Prepare translated epSOS pivot content
7. Validate epSOS pivot content using schematron to check correct relations to epSOS reference terminology

Note:
Schematron validation is configured via tm.properties file using following parameters:

Property name	Description
tm.schematron.path.xsldir	Current path to schematron XSL directory
tm.schematron.path.patientsummary.friendly	Current path to a schematron file for original (epSOS friendly) content of Patient Summary
tm.schematron.path.eprescription.friendly	Current path to a schematron file for original (epSOS friendly) content of ePrescription
tm.schematron.path.edispensation.friendly	Current path to a schematron file for original (epSOS friendly) content of eDispensation
tm.schematron.path.hcer.friendly	Current path to a schematron file for original (epSOS friendly) content of Health care encounter report
tm.schematron.path.mro.friendly	Current path to a schematron file for original (epSOS friendly) content of Medication related overview

8. Generate response structure

4.3.3 Errors and Warnings

Exception is considered to be an error, if
- Required coded element (R, RNFA) was not translated (failure result from TSAM)

Exception will raise a warning, if
- Pivot content validation fails, or due to external condition it is impossible to run validation (e.g. missing schematron)
- Translated pivot content validation fails, or due to external condition it is impossible to run validation (e.g. missing schematron)

- By coded element identification, element does not have a proper type
- By coded element identification, element is not in CodedElementList
- Optional coded element (O, NS) was not transcoded (failure result from TSAM)
- TSAM reported warnings

4.4 Input parameters

Definition of input parameters for both public and private methods.

4.4.1 epSOSOriginalData

Original content based on document compliant with HL7 CDA standard and epSOS pivot document specification with arbitrary coding used.
As epSOSOriginalData two types of document can be provided:

- original CDA level 3 fully structured content (compliant to epSOS format)
- original CDA level 1 content embedding PDF (compliant to epSOS format)

4.4.2 epSOSCDA

Pivot content compliant with HL7 CDA standard and epSOS pivot document specification with all coded elements represented using epSOS reference terminology.
As epSOSOriginalData two types of document can be provided:

- pivot CDA level 3 fully structured content
- pivot CDA level 1 content embedding PDF

4.4.3 TargetLanguageCode

Target language expressed in form of IETF language tags containing language and country abbreviation.

Name	Data Type	Optionality	Description
LanguageCode	String	mandatory	Target language code

Note: In MVC, language code and country code are in two separate valuesets based on two ISO standards.

Country: ISO 3166-1 (epSOSCountry)
Language: ISO 639-1 (epSOSLanguage)

4.5 Output parameters

Definition of output parameters for public methods.

4.5.1 ResponseStructure

It is a response structure including: the epSOS CDA document and response status structure providing information about operation result, including possible errors and warning.

ResponseStructure is java object.

It provides methods:

- getResponseCDA() : Document
  Output is response CDA Document, which can be:
  - epSOS CDA Pivot Document (result of transcoding in country A)
  - Translated epSOS CDA Pivot Document (result of translation in country B)

- getErrors() : List<TSAMError>
  Output is List of Errors

- getWarnings() : List<TSAMWarning>
  Output is List of Warnings

- getStatus() : String
  Output is String – status of response (success, failure…)

- getDocument() : Document
  The response structure in XML Document presentation and it looks like:

```
<responseStructure>
    <responseElement>
        <!--One of these:-->
        <!-- epSOS CDA Pivot Document (result of transcoding in country A)-->
        <!-- Translated epSOS CDA Pivot Document (result of translation in country B)-->
    </responseElement>
    <responseStatus>
        <status result="success/failure"/>
        <!-- optional -->
        <errors>
            <error code="..." description=".."/>
            <error code="..." description=".."/>
        </errors>
        <!-- optional -->
        <warnings>
            <warning code="..." description=".."/>
            <warning code="..." description=".."/>
        </warnings>
    </responseStatus>
</responseStructure>
```

Name	Type	Optionality	Description
responseStructure	element	mandatory	Main XML element of response structure
responseElement	element	mandatory	Main result of called method – contains CDA document (either epSOSCDA, translated epSOSCDA or epSOSPDF)
responseStatus	element	mandatory	Information about status of called method
status	element	mandatory	Element containing status
result	attribute	mandatory	Result of method can be success or failure
errors	element	optional	List of errors occurred during call processing
error	element	optional	Element containing error
code	attribute	mandatory	Code of error
description	attribute	mandatory	Description of error
warning	element	optional	Element containing warning
code	attribute	mandatory	Code of warning
description	attribute	mandatory	Description of warning

4.6 Internal parameters

Definition of internal parameters for communication with Terminology Services Access Manager.

4.6.1 SourceConcept

SourceConcept is a structure used to convey the concept derived from the epSOS Pivot Document. It includes all information necessary to locate concept in LTR, i.e. at least the concept code and the concept code system.
It is used for representation of a concept from local code system and also an epSOS concept.

Name	Data Type	Optionality	Description
Code	String	mandatory	Local code taken from CDA document
CodeSystem	String	mandatory	Identification (OID) of local code system from CDA document
CodeSystemVersion	String	optional	Name of a version of code system (if available in CDA document)
ValueSet	String	optional	Identification (OID) of value set
ValueSetVersion	String	optional	Name of value set version (if provided, identification of ValueSet is required)

4.6.2 TargetLanguageCode

See section 4.4.3.

4.6.3 ResponseStructure (TSAM ResponseStructure)

It is a response structure including: the epSOS Reference Concept and response status structure providing information about operation result, including possible errors and warning.

ResponseStructure is java object.

Object attributes:

Name	Type	Optionality	Description
code	String	mandatory	Code of a concept
codeSystem	String	mandatory	Identification (OID) of a code system
codeSystemName	String	optional	Name of a code system
codeSystemVersion	String	optional	Name of a version of code system (if not current)
displayName	String	mandatory	Designation of a concept
status	element	mandatory	Element containing status
errors	element	optional	List of errors occurred during call processing
warnings	element	optional	List of warnings occurred during call processing

Response structure object provides methods:

- getResponseElement() : Element
  Output is response Element (transcoded or translated)

Direct acces to parts of response Element is possible also with methods:

- getCode() : String;
- getCodeSystem() : String;
- getDesignation() : String

- getErrors() : List<TSAMError>
  Output is List of Errors

- getWarnings() : List<TSAMWarning>
  Output is List of Warnings

- getStatus() : String
  Output is String – status of response (success, failure…)

- getDocument() : Document

The response structure in XML Document presentation and it looks like:

```
<responseStructure>
        <responseElement>
                <!--One of these:-->
                <!-- EpSOS Reference Concept (result of transcoding in country A)-->
                <translation
                        code=""
                        codeSystem=""
                        codeSystemName=""
                        codeSystemVersion=""
                        displayName=""/>
                <!—Translated epSOS Reference Concept  (result of translation in country B)-->
                <translation
                        displayName=""/>
        </responseElement>
        <responseStatus>
                <status result="success/failure"/>
                <!-- optional -->
                <errors>
                        <error code="..." description=".."/>
                        <error code="..." description=".."/>
                </errors>
                <!-- optional -->
                <warnings>
                        <warning code="..." description=".."/>
                        <warning code="..." description=".."/>
                </warnings>
        </responseStatus>
</responseStructure>
```

Name	Type	Optionality	Description
responseStructure	element	mandatory	Main XML element of response structure
responseElement	element	mandatory	Main result of called method – contains epSOS concept representation
translation	element	mandatory	Element containing result of transcoding / translation (in a form compliant with epSOS CDA format)
code	attribute	mandatory / optional	Code of epSOS concept
codeSystem	attribute	mandatory / optional	Identification (OID) of epSOS code system

codeSystemName	attribute	optional	Name of epSOS code system
codeSystemVersion	attribute	optional	Name of a version of code system (if not current)
displayName	attribute	mandatory	DisplayName of epSOS concept
responseStatus	element	mandatory	Information about status of called method
status	element	mandatory	Element containing status
result	attribute	mandatory	Result of method can be success or failure
errors	element	optional	List of errors occurred during call processing
error	element	optional	Element containing error
code	attribute	mandatory	Code of error
description	attribute	mandatory	Description of error
warning	element	optional	Element containing warning
code	attribute	mandatory	Code of warning
description	attribute	mandatory	Description of warning

4.6.4 RetrievedConcept

It is a response structure of function getValueSetConcepts() representing concept and its designation.

Object attributes:

Name	Type	Optionality	Description
id	String	mandatory	Identifier of concept
code	String	mandatory	Concept code
status	String	mandatory	Status of concept
designation	String	optional	Designation of concept in requested language
languageCode	String	optional	Code of language for designation

4.7 Public API

Public interface used by WorkflowManager.

Transformation to epSOS CDA pivot document: ***toEpSOSPivot()***

Method:	***toEpSOSPivot()***
Description:	Transformation of an original CDA document to epSOS pivot document.
Input parameters:	EpSOSOriginalData [mandatory] TargetLanguageCode [optional]
Output parameter:	ResponseStructure

Translation of epSOS CDA pivot document: **translate()**

Method:	***translate()***
Description:	Transformation of an epSOS pivot document by adding translation to target language.
Input parameters:	EpSOSCDA [mandatory] TargetLanguageCode [optional]
Output parameter:	ResponseStructure

4.8 Configuration parameters

List of configuration parameters of TM component stored in file tm.properties:

Property Name	Type	Initial value	Description
tm.documenttype.patientsummary	String	60591-5	epSOS code for Patient Summary CDA document
tm.documenttype.eprescription	String	57833-6	epSOS code for ePrescription CDA document
tm.documenttype.edispensation	String	60593-1	epSOS code for eDispensation CDA document
tm.documenttype.hcer	String	34133-9	epSOS code for HCER CDA document
tm.documenttype.mro	String	56445-0	epSOS code for MRO CDA document
tm.codedelementlist.enabled	boolean	true	Flag indicating that for identification of coded elements configurable list of elements should be used.
tm.codedelementlist.path	path		Path to the XML file (coded_element_list.xml): List of all coded elements which can be found in epSOS CDA pivot document. For each coded element contains: - xpath - usage (document type and optionality) - related value set - value set version - language_code (to which element should be translated, if TargetLanguageCode should not be used)
tm.schema.validation.enabled	boolean	false	Flag indicating that schema validation is enabled or disabled.
tm.schemafilepath	path		Current path to schema file (CDA_extended.xsd)
tm.schematron.validation.enabled	boolean	false	Flag indicating that schematron validation is enabled or disabled.
tm.schematron.path.patientsummary.friendly	path		Current path to schematron file for Patient Summary - Friendly.
tm.schematron.path.eprescription.friendly	path		Current path to schematron file for ePrescription - Friendly.
tm.schematron.path.edispensation.friendly	path		Current path to schematron file for eDispensation - Friendly.
tm.schematron.path.hcer.friendly	path		Current path to schematron file for HCER - Friendly.
tm.schematron.path.mro.friendly	path		Current path to schematron file for MRO - Friendly.
tm.schematron.path.scannedDocument.friendly	path		Current path to schematron file for ScannedDocument - Friendly.
tm.schematron.path.patientsummary.pivot	path		Current path to schematron file for Patient Summary - Pivot.
tm.schematron.path.eprescription.pivot	path		Current path to schematron file for ePrescription - Pivot.
tm.schematron.path.edispensation.pivot	path		Current path to schematron file for eDispensation - Pivot.
tm.schematron.path.hcer.pivot	path		Current path to schematron file for

			HCER - Pivot.
tm.schematron.path.mro.pivot	path		Current path to schematron file for MRO - Pivot.
tm.schematron.path.scannedDocument.pivot	path		Current path to schematron file for ScannedDocument - Pivot.
tm.schematron.path.xsldir	path		Current path to schematron XSL directory
tm.mda.validation.enabled	boolean	true	Flag indicating that IHE model based validation is enabled or disabled.
tm.mda.cda_xsd_path	path		Current paths to IHE model based validator configuration files
tm.mda.cda_epsos_xsd_path	path		
tm.mda.cda_xsl_transformer_path	path		
tm.mda.value_set_repository_path	path		
tm.audittrail.enabled	boolean	true	Flag indicating that audit trail is enabled or disabled.
tm.audittrail.transactionnumber	String		Audit Trail Event Log - The number of transaction including the epsos- prefix
tm.audittrail.targetip	String		Audit Trail Event Log - The IP Address of the target Gateway
tm.audittrail.facility	String		Audit Trail Audit Service - The facility number according to log4j
tm.audittrail.severity	String		Audit Trail Audit Service - The severity of the message

Annex A – Coded element list

coded_element_list_
epSOS2.xml

Annex B – Implementation platform

List of tools used in PosAm implementation.

Tools

- Eclipse IDE (Eclipse Galileo)
- Maven (project managment)
- SVN (revision control)

Libraries

- Java 5.0 – implementation
- Logback-classic (0.9.24) – basic logging
- Jcl-over-slf4j (1.6.0) basic logging
- Junit – (junit-4.7.jar) – testing
- Hibernate – (hibernate-core-3.3.1.GA.jar) - Mapping DB – Java Objects
- Spring framework (spring-2.5.6.jar)
- Saxon 9 HE (saxon9.jar) – XSLT processor used in process of schematron validation
- Commons-codec (1.4) – Base64 encoding of CDA embedded pdf file
- Commons-io (1.4) – File manipulation (File to byte[])

Schema validation

Using Java 5.0 javax.xml.validation.Validator for CDA parsed as org.w3c.dom.Document

Schematron validation

epSOS/TM own implementation of ISO schematron using XSLT.
http://schematron.com/implementation.html and SAXON 9 XSLT2 Engine. (Saxon-HE (home edition) is an open source product available under the Mozilla Public License.)