SPECIFICATION OF THE CDA LEVEL 1 BODY

State:	Pre-final
Distribution:	Public
Version:	0.6
Date:	07.03.2014
Reference:	CDA_Level1_Body_Specification_v0.6r.pdf

HISTORY

Version	Date	Author	Description
V0.1	18.10.2013	Samuel Danhardt	Initial creation
V0.2	24.10.2013	Samuel Danhardt	Additional information about required PDF file format
V0.3	18.12.2013	Samuel Danhardt	Page numbers added
V0.4	30.12.2013	Samuel Danhardt	New CONF rules for ValueSet restrictions
V0.5	14.01.2014	Samuel Danhardt	New “optionality” column Rephrasing CONF-BODY1-LU-3 and CONF-BODY1-LU-4
V0.6	07.03.2014	Samuel Danhardt, Heiko Zimmermann	Complete review

COPYRIGHT

LEGAL DISCLAIMER

Table of contents

1. INTRODUCTION

This implementation guide provides the specification of the CDA Level 1 Body for the Luxemburgish healthcare sector.

Goal is to provide a consistent specification which can be used by IT-staff of the healthcare sector or by vendors to create compliant CDA documents for the exchange of medical documents in the Luxemburgish healthcare system and with the national eHealth platform in Luxemburg.

The clinical documents that are exchanged or shared within the DSP (Dossier de Soins Partagé) must be conformed to the specifications of the HL7 Clinical Document Architecture release 2.0 (HL7 CDA R2) standard. This standard allows the exchange of clinical documents in xml format compliant with the CDA.xsd and assures the accessibility and readiness, stewardship and the consistency related to the Healthcare professionals responsibilities.

A CDA document (Clinical Document architecture) is divided in two main parts:

- A header where general information needed to the stewardship of the document; it allows the qualification of the document, describes the participants of the document, the relationship between documents and any other information used for its integration in the care process. The information of the header is generally common to all clinical documents that are shared or exchanged in the DSP system;
- A body where clinical information are presented. The clinical information could be unstructured such as a PDF document or an image or could be structured containing xml data.

This document describes **constraints** that apply to the data required for the CDA Document Body. The CDA Body represents the narrative content of a clinical document. This content can be represented in a structured format (e.g. XML format) or in an unstructured format (e.g. a PDF file). This document concerns CDA Level 1 which means that the content is in an **unstructured format**. For information, a structured format is used in CDA Level 2 and 3.

This document is one of a set of documents that will create the Healthcare Interoperability framework of Luxemburg.

It is assumed that readers of the document have a basic knowledge about HL7 and CDA and the purpose of use as a document format for the exchange and sharing of medical information.

This specification document has to be considered together with the CDA general Header specifications provided by the Agence eSanté (for information about data types, CDA structure, but also the specifications workflow and much more). If different CDA Header specifications exist (e.g. for different medical domains), then this specification has to be considered together with the CDA Header specification applicable to the domain to build the domain specific CDA Level 1 specification.

1.1. AUDIENCE

This document is intended for software developers, consultants and the national agency staffs responsible for the implementation of the DSP system in Luxemburg and the integration of the clinical software to the DSP system and all local or national health information exchange who wish to create/process clinical documents according to this specification.

1.2. REFERENCES

- ISO/HL7 27932 :2008 - HL7 Clinical Document Architecture, Release 2.0 (CDA R2)
- CDA Header Specification document edited by Agence eSanté

1.3. VALIDATION

In addition, we reviewed the testing tools called IHE Gazelle Model Based Validation (http://gazelle.ihe.net/content/cda-model-based-validation) where several existing templates are available including the epSOS templates for Patient Summary, ePrescription and eDispensation.

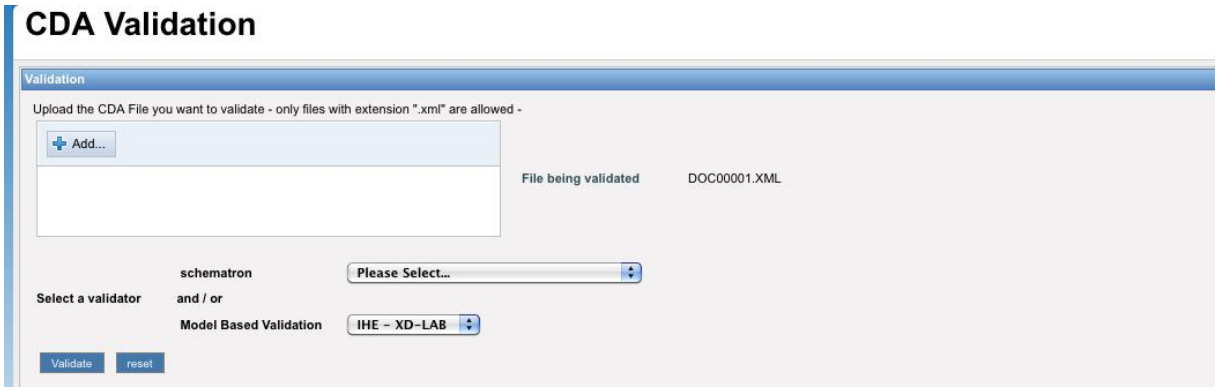

Figure 1: Document upload window for Gazelle CDA Validation

The Agence eSanté Luxemburg provides a test and validation environment for CDA documents[1]. This environment is based on the Gazelle[2] testbed platform tool from IHE Europe and can be used by software developers to check their software compliance.

2. CDA BODY LEVEL 1 CONSTRAINTS FOR LUXEMBURG

2.1. GENERAL CONSIDERATIONS

It is understood that the CDA Body is part of a CDA Document (thus share the same namespace definitions), which itself starts with the CDA Header. The reader can refer to the CDA Header Specification document edited by Agence eSanté for further information about this specific topic, and obviously to the CDA Schemas to respect the general CDA structure.

[1] The CDA validation environment for Luxemburg can be reached following this link: http://gazelle.agence-esante.lu

[2] The Gazelle tool can be accessed using the following link: http://gazelle.ihe.net

2.2. General Definitions

The encoding **SHALL** be UTF-8, and the XML Version 1.0 **SHALL** be used with CDA.

CONF-BODY1-LU-1: The encoding **SHALL be UTF-8**.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
```

The majority of applications handling non XML documents are using the ISO-8859-1 or ISO-8859-15 encoding. These applications must transcode between the two sets of characters ISO-8859 and UTF-8 for CDA documents.
The encapsulated content in base64 in an unstructured body of a CDA document **SHALL** keep their initial set of characters.

2.3. Component Element

The component element is the first element of the CDA Body. It is common to both structured and unstructured narrative, and acts as a container for the actual data.

CONF-BODY1-LU-2: The nonXMLBody **SHALL** be present

Element/Attribute		DT	Card	Opt	Description
component		POCD_MT000040.Component2	[1..1]	M	
	nonXMLBody	POCD_MT000040.NonXMLBody	[1..1]	M	

Table 1: component specification

Example:

```
<component>
        <!-- the nonXMLBody comes here -->
</component>
```

2.4. NONXMLBODY ELEMENT

2.4.1. nonXMLBody

The nonXMLBody element carries the narrative through the text element. Because the confidentiality and language code are already defined in the CDA Header, it would be a duplicate to specify them again here.

The text element is of type ED (Encapsulated Data) which is defined later on in this document.

CONF-BODY1-LU-3: The confidentialityCode is already defined in the CDA Header thus this element **MAY** be omitted

CONF-BODY1-LU-4: The languageCode is already defined in the CDA Header thus this element **MAY** be omitted

Element/Attribute		DT	Card	Opt	Description
nonXMLBody		POCD_MT000040.NonXMLBody	[1..1]	M	
	text	ED	[1..1]	M	The actual narrative
	confidentialityCode	CE	[0..1]	O	Confidentiality of the content. Do not use.
	languageCode	CS	[0..1]	O	Human language used in the narrative. Do not use.

Table 2: nonXMLBody specification

Examples will be provider in the next chapter.

2.4.2. nonXMLBody/text

The text element contains the actual narrative, either as text, or base 64 content for PDF files. It is also possible to compress the data (before the base 64 encoding is done). Two attributes can carry the integrity checks information.

Although PDF documents are used in everyday life, it is not always known that there are different kinds of PDF formats. If the first aim of the PDF format was to produce “as-printed”

documents, it was realized that for the growing purpose of archiving or preserving documents, more features are needed.

For example, it is necessary that a PDF will in several years, render exactly as it renders today. This means that the PDF document should be self-sufficient and contain all information for the rendering (e.g. text, graphics, color information…) and does not rely on an external source. Those constraints are expressed in the PDF/A standard.

This standard evolved from PDF/A-1 (2005) to PDF/A-3 (2012), adding functionalities and taking into account the evolution of the PDF format itself. For compatibility reasons, only PDF/A documents are allowed, but not that more recent version should be backward compatible.

PDF/A-1 itself is separated in 2 compliance levels with different purpose:

- PDF/A-1a : Ensure that document content can be searched and repurposed
- PDF/A-1b : Ensure reliable reproduction of the visual appearance of the document

PDF/A-1a contains and enhances PDF/A-1b, thus for compatibility reasons, only PDF documents conforming to PDF/A-1b are allowed. But, note that conforming to PDF/A-1a implies to conform to PDF/A-1b.

To summarize: it is allowed to use plain text and PDF documents. Accepted PDF documents are PDF/A compliant conforming to PDF/A-1b.

CONF-BODY1-LU-5: The nonStructured body **SHALL** only contain text or PDF data

CONF-BODY1-LU-6: PDF documents **SHALL** be PDF/A-1b compliant

Conforming to PDF/A-1a implies to conform to PDF/A-1b, thus documents creators are encouraged to conform to PDF/A-1a to the maximum extent possible, but a simple document scanner may be unable to fully conform to PDF/A-1a.

The CDA specification allows to use a reference (e.g. an URL) instead of the document itself, and to join a thumbnail. This is dangerous because the document can be changed, altered, or even deleted and the Plateforme can't maintain a correct versioning of documents. This usage is thus forbidden.

CONF-BODY1-LU-7: reference and thumbnail **SHALL NOT** be used

The mediaType and language attributes of the text element are normalized:

CONF-BODY1-LU-8: text@mediaType values **SHALL** come from the Valueset eSanté_MimeTypes

CONF-BODY1-LU-9: text@language values **SHALL** come from the Valueset eSanté_LanguageCode

Note that including XML text inside the text element is against the CDA schemas.

Element/Attribute	DT	Card	Opt	Description
text	ED	[1..1]	M	The actual narrative. The text value or the base 64 representation is to be written inside this tag.
@representation	BinaryDataEncoding	[0..1]	O	Values can be “B64” for base 64 encoding, or the default value “TXT”.
@mediaType	cs	[0..1]	O	Values SHALL come from the Valueset eSanté_MimeTypes
@language	cs	[0..1]	O	Values SHALL come from the Valueset eSanté_LanguageCode
@compression	CompressionAlgorithm	[0..1]	O	Indicates whether the raw byte data is compressed, and what compression algorithm was used. Values can be “DF”, “GZ”, “Z” or “ZL”.
@integrityCheck	bin (based on xs:base64Binary)	[0..1]	O	The integrity check is a short binary value representing a cryptographically strong checksum that is calculated over the binary data. The purpose of this property, when communicated with a reference is for anyone to validate later whether the reference still resolved to the same data that the reference resolved to when the encapsulated data value with reference was created.
@integrityCheckAlgorithm	IntegrityCheckAlgorithm	[0..1]	O	Specifies the algorithm used to compute the integrityCheck value. Values can be “SHA-256” or the default “SHA-1.

Table 3: nonXMLBody/text

Note:

> You can use CDATA sections (e.g. `<![CDATA[my sp&cial sentence]]>`) to escape characters like “<” and “&” that are normally illegal in XML elements.

Example: sending text

```
<component>
  <nonXMLBody>
    <text mediaType='text/plain'>The patient is in perfect health
condition</text>
  </nonXMLBody>
</component>
```

Example: sending a PDF

```
<component>
  <nonXMLBody>
    <text mediaType='application/pdf'
representation='B64'>TWFuIGlzIGRpc3Rpbmd1aXNoZWQsI=</text>
  </nonXMLBody>
</component>
```