On the Evaluation of HL7 CDA R2 Documents Richness and Validation Reliability

Abderrazek Boufahja, Eric Poiseau

IHE-Europe

Abstract

Many test providers and certification programs define test plans in order to test the conformity of CDA documents against implementation guides. Even if the applications and systems tested provide valid CDA documents, it is not easy for test providers to decide if the validated documents are rich enough to have a good reliability on tested tools; providing the coverage of the areas tested is mandatory based on many test framework specifications. Many projects tried to define a way to describe the richness and providing scoring for validated CDA documents. In this paper, we describe a new methodology to identify the richness of CDA documents based on implementation guides specification. We define a way to provide a scoring for the richness of the CDA documents, with some applications on IHE and C-CDA documents.

Keywords

IHL7; CDA; Scoring; Richness; Validation; Reliability; Templates

Correspondence to:

Abderrazek Boufahja
Kereval 4 Rue Hélène Boucher, 35235 Thorigné-Fouillard, France.
E-mail: abderrazek.boufahja@ihe-europe.net

EJBI 2017; 13(1):73-83
received: June 11, 2017
accepted: July 14, 2017
published: October 10, 2017

1 Introduction

Since the publication of HL7 CDA R2 standard [1], evaluating the richness of a CDA document remains a problematic subject, especially in the context of testing and certifying applications creating CDA documents. In order to comply with the requirements of CDA implementation guides, an editor can be tempted to provide documents containing a minimal set of data, in order to comply with the tests requirements. So unless the test description covers the entire tree of templates in the context of a document, the confidence in the tested application remains uncertain [2]. An indicator of the richness of a tested CDA documents can be useful for tester, in order to complete the outcome of conformance checking tools. There is no complete methodology to automatically express the richness of CDA templates. The only way to calculate it is to manually parse the CDA document and to compare it with the original specification. The aim of this paper is to define a new methodology to calculate and interpret the richness of CDA documents. We will first present the state of the art regarding the scoring of the CDA documents. Then we present our analysis regarding the CDA templates richness and scoring. And finally, we describe the implementation and some applications of the richness scoring in some projects.

2 State of the Art

2.1 Richness of CDA Documents

A couple of papers and articles in the literature mentioned the richness of the CDA documents, as a notion describing "how rich are the clinical information in the CDA document" [3, 4, 5]. Technically this is interpreted by how many templates and CDA elements are present, comparing to the list of possible templates and elements (this include both required and optional templates/elements). The richness of CDA documents refers to the 'A' in CDA: Architecture. It is a way to describe the architecture of the provided documents comparing to the specifications' architecture of templates.

2.2 Scoring of C-CDA Documents

Scoring has been introduced by meaningful use with the creation of a scorecard for CDA document [6]. The scorecard provides a score to a CDA document based on the presence of specific components with the analyzed document. Smart C-CDA Scorecard is a tool that promotes best practices for C-CDA [7]. This tool is one of the first clinical checking tools to provide a scoring of information provided in a CDA document. The tool is dedicated to C-CDA usage, and based on users experience with C-CDA specification. The tool provides as output a percentage as a scoring and a grade for the C-CDA document tested, and a description of clinical issues for each CDA section.

3 HL7 CDA R2 Templates Richness Analysis

3.1 Presentation and Notions

Test Data Characteristics

During testing process, tested applications can provide two kinds of test data: minimal test data, and relevant test data. A minimal test data contains the minimal architecture of templates allowing passing the validation process by automated validation tools. This kind of documents contains only the required templates, the required elements, and fits well the rules checked by validation tools; however, it is not enough to confirm that the tested application is able to create valid CDA documents in a real use case. A relevant test data is an expression of a real use case with fake data. Testing using relevant test data increases the reliability on the tested tool. These notions are also treated in the Test Framework specification of eHDSI project [8].

Templates Inheritance

The inheritance between CDA templates is a basic notion for the CDA implementation guides. It allows reusing templates rules, without the need to rewrite them. HL7 Templates Standard defines many types of inheritances between HL7 Templates [9]; the most important one for our study is the specialization (SPEC) between CDA templates. The other kinds of templates inheritance are more to be extensions than specializations of the specification rules [9, 10].

Inheritance between templates can be for multiple levels. Example: in eHDSI project, an epSOS medication [templateId: 1.3.6.1.4.1.12559.11.10.1.3.1.3.4] inherit from IHE medication [templateId: 1.3.6.1.4.1.19376.1.5.3.1.4.7], which inherit from CCD medication activity [templateId: 2.16.840.1.113883.10.20.1.24] [11]. In this case, the more specific template needs to fit all the requirements coming from all the parent templates.

Templates Containment

CDA templates can contain many other sub-templates. In this paper, we are only considering the containment as described in the HL7 Templates Standard [9]. A template can be a header template, a L2 template, or a L3 template. The header templates are related to CDA header elements (like the patient, the participants, etc). The L2 templates are related to the sections containments, and the L3 templates are related to the CDA entries. The richness of CDA templates can be related to the L2 or L3 [12].

3.2 Templates Richness Matrix

There are as explained two kinds of relationship between CDA templates: inheritance and containment. We expressed each of these kinds of relationship by matrices: the matrix of inheritance and the matrix of containment. The work performed here is to merge both matrices and to get a complete richness containment relationship.

Example Analysis

We suppose we have this scenario:

- We have eight CDA templates: A, B, C, D, E, F, G, H
- A inherit from B
- B inherit from C
- A contains two templates D and E
- B contains F
- C contains two templates G and H
- D inherit from G
- F inherit from C

The first step to be executed is to know the real inheritance architecture. From the example, there are an extra inheritance path that we can calculate, which is between A and C Figure 1.

Once we have the final inheritance relationships; we copy all the missing templates from the parents into the more specific templates. Example, A inherits from B, so one possible template for A will be F. Here is the diagram of containment computation Figure 2.

Right now, A can have many sub-templates: D, E, F, G, and H. However, we know that D is a specialization of G. So, we can remove the containment between A and G, as it is redundant: when A includes D, it already includes G Figures 3, 4, 5.

So here we have the final result of containments between templates in our example:

Let's now express these operations using matrix description.

We consider the vector $V = [A\ B\ C\ D\ E\ F\ G\ H]$

The first matrix of containment can be expressed by:

$$A = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

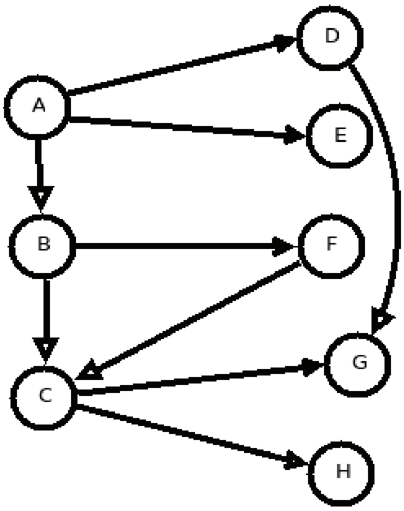

Figure 1: Original containments.

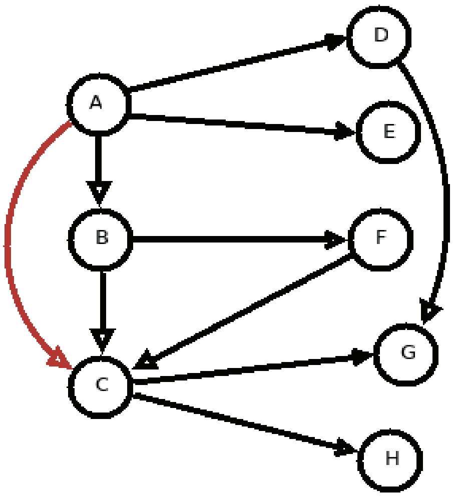

Figure 2: Inheritance computation.

The matrix of inheritance between the templates can be expressed as:

$$B = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

The matrixes A and B are the expression of inheritance and containment relationship. These matrices are square matrixes, having as dimension the number of templates used in the specification. The rows and the columns describe the same vector, the vector V of the templates used. Then, to interpret these matrices, if you have 0, it means there are no

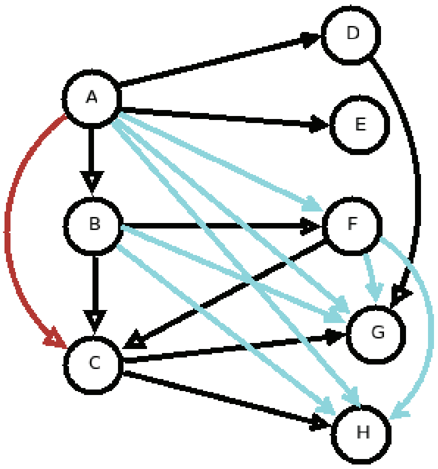

Figure 3: Containment computation.

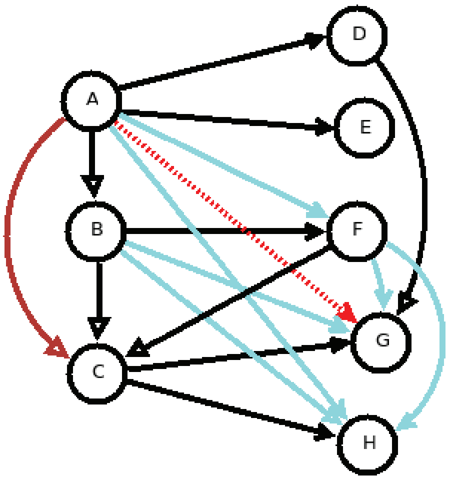

Figure 4: Diagram refinement.

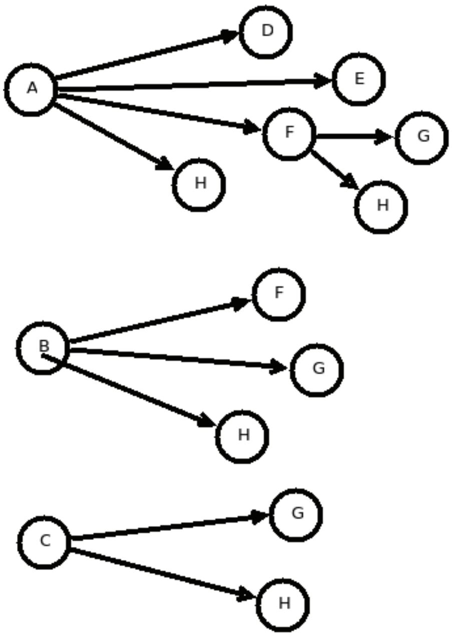

Figure 5: Tree of containments.

containment (or no inheritance) between the two templates (for the row and the column selected), and if we have 1, it means there are a containment (or inheritance). Here for example we have B[0][1] = 1, it means there are inheritance between the template A and B.

The first step was to find the real inheritance architecture, by looking for the parent of the parent of a template. The parent of the parent of the template A is C. To get this mathematically, we need to multiply the matrix B by itself:

$$BxB = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

The complete inheritance for our example can be expressed by the matrix:

$$C = B + B^2 = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

We call the C matrix, the complete inheritance description matrix.

The second step was to identify all the possible missing containment in the CDA templates. For template A, it was F, G and H; for template B it was G and H. To get this information mathematically, we need to multiply the matrix C by A (the containment matrix):

$$CxA = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

The third step is to clean up the redundant containment. To do so, we need first to find what was included redundantly. In our case, for the template A, we included the template G redundantly as we already include the template D. To find this information we need to multiply the matrix A by C:

$$AxC = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Then we need to remove the redundancies found. The final result of our operations is:

$$(CxA + A)\&\overline{(AxC)} = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

And this result fits exactly the output from our analysis of this example.

Richness Computation

From the matrix *A* (for the CDA templates containment) and *B* (for CDA templates inheritance), we are able to extract the possible sub-templates that may appear for a specific templates. The formula allowing computing these containments is described in this equation:

$$\Phi = (CA + A)\&\overline{\mathrm{AC}} = \left(\sum_{n=0}^{\infty} B^n A\right)\&\overline{\sum_{n=1}^{\infty} AB^n} \qquad (1)$$

Where

$C = \sum_{n=1}^{\infty} \mathrm{B}^{\mathrm{n}}$ is the complete inheritance matrix between the different templates.

Explanations

B^n computes the inheritance relationship for the level n.

The matrix C is the complete inheritance architecture for the defined list of templates. As the inheritance between CDA templates is not circular, the matrix *B* can be expressed as a strict triangular matrix, and then it is a nilpotent matrix: $\exists\, n \in \mathbb{N}: B^n = 0$ [13]. And this proves that C is computable.

The matrix *CA* describes the inherited templates containment. *(CA + A)* describes the complete sub-templates architecture. *AC* describes the redundancy in the templates architecture. As *A* describes the templates containment, and *C* describes the templates inheritance, *AC* describes the parent templates that will be contained automatically if we already include a specialization of them. $\overline{AC}$ describes the templates that can be included without redundancy. To create $\overline{AC}$ from AC matrix, we need to replace every “1” by “0” and every “0” by “1”. The matrix ϕ describes then the containment relationship between the different templates, by removing all the redundancies.

Exploitation

Based on the final calculated matrix of containments for all the CDA templates, we are able to refine the matrix of containment, in order to extract only needed templates to express the richness of tested CDA document. For many specifications, there are multiple document header templates (example, for C-CDA 2.1 there are CCD documents, Discharge Summary documents, etc). When analyzing CDA document richness, we are only interested on the specific used document header template, and all its containments. The other document header templates are not useful for the analysis of the document. To extract the matrix of sub-templates containment for a specific template, we need to execute this equation:

$$\Psi = diag\left(\sum_{i=0}^{n} e_k \Phi^i\right) x\Phi \qquad (2)$$

Where

ϕ: the final containment matrix, without refinement

Ψ: describes the final containment matrix, refined for the use case of a specific template.

n: the level of CDA containment we are looking for; if we want to have the complete CDA templates containment architecture, we fix this attribute to the dimension of the matrix ϕ. However if we want for example to analyze only CDA L2, we set n = 1.

k: The index of the selected templateId in the list of rows described by the matrix ϕ

e_k: a linear Boolean vector containing only 0 and 1 values, 0 when the element in the row of the matrix ϕ is different than the selected template, 1 when the row template is equal to the selected template.

In our example, if we want to select all the templates and containment related to template A, we have:

- n: 8
- k: 1
- e_k = [1 0 0 0 0 0 0 0]

$e_k\Phi^i$ is a linear Boolean vector describing the templates included for the ith level of containment between templates, based on the selected CDA template.

From our example: $e_k\Phi^2$ = [1 0 0 0 0 0 0 0] describes the second level of containment (G and H).

$e_k\Phi^i$ is the linear Boolean vector describing all the templates included in all the level of containment, and related to the selected template. In our example,

$$\sum_{i=0}^{n} e_k \Phi^i = [1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]$$

This is interpreted as: in the tree related to template A, we can have A, D, E, F, G, and H.

$diag\left(\sum_{i=0}^{n} e_k \Phi^i\right)$ is the diagonal matrix used to extract useful templates information from the global containment matrix.

For our example

$$diag\left(\sum_{i=0}^{n} e_k \Phi^i\right) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

The final refined matrix provides:

$$diag\left(\sum_{i=0}^{n} e_k \Phi^i\right) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

This matrix allows having only the needed templates containments relationship, and this allows identifying in the validated CDA document the list of present and missing templates for the specific document header templateId. The refined containment matrix allows constructing the tree of sub-templates based on the document header templateId.

From the analyzed CDA document, we extract a matrix containing the same dimensions and row description of the refined containment matrix, called matrix document description (MDD: Λ). When we have a zero in the matrix Ψ we set a zero on Λ. When we have one in the matrix Ψ, we extract the templateId described by this row index, and its related sub-template described by the column index. We look then in the CDA document if the described containment is implemented. Using Λ, we are able to evaluate graphically the list of present and missing templates in the analyzed CDA document. Λ and Ψ are used together to compute the CDA document richness scoring.

3.3 CDA Document Richness Scoring

The aim of scoring a CDA document is to provide a metric to compare CDA documents between themselves. The metric provided is a computation of the richness of templates containment. There are two ways to calculate CDA document richness scoring: basic richness scoring and weighted richness scoring.

Basic Richness Scoring

The calculated scoring is based on existing templates against possible templates.

$$R_s = \frac{D_i}{T_i} = \frac{\textit{Number present templates}}{\textit{Number possible templates}} \qquad (3)$$

For each template, we calculate the tree of containment based on the matrix Ψ. Once we have the complete tree of templates, T_i is equal to the total number of present sub-templates for all levels.

$$T_i = \sum_{l=0}^{\infty} sum_t(l) \qquad (4)$$

Where l is the level of containment and $sum_t(l)$ is the number of possible templates in the level l. T_i is called Template richness indicator.

$$D_i = \sum_{l=0}^{\infty} sum_d(l) \qquad (5)$$

Where l is the level of containment and $sum_d(l)$ is the number of existing templates in the level l included in the CDA document tested. D_i is called Document richness indicator.

From our example, the Template richness indicator for the template A is 6 (we have 6 sub-templates in all the levels included).

D_i is calculated the same way as T_i; however the computation is based on the matrix of existing templates in the CDA document (Λ).

D_i describes the number of existing sub-templates in the CDA document checked, for all the sublevels of the selected template.

For example, if we are missing two sub templates related to the template A from our example (example E and G), we will have D_i= 4 and $R_s = \frac{4}{6} = 0.66$

The problem with this method of scoring computation is the fact that we do not take in consideration the level of containment. This method does not make difference if we are missing a template from a higher level or from a lower level. In CDA, the most important containments are those of the L2 templates (the section). The L3 templates are less important, and the other levels are lesser important. The scoring of the CDA document needs to take in consideration these levels of containments. Missing templates from the CDA L2 shall be more dangerous than missing templates from CDA L3. That's why we need a weighted computation of the richness scoring.

Weighted Richness Scoring

To weight the scoring of the richness of CDA document, we need to weight found templates in each level.

$$R_{ws} = \frac{D_{wi}}{T_{wi}}$$

Where

$$T_{wi} = \sum_{l=0}^{\infty} \frac{sum_t(l)}{f(l)} \qquad (7)$$

And

$$D_{wi} = \sum_{l=0}^{\infty} \frac{sum_d(l)}{f(l)} \qquad (8)$$

Where $f(l)$ is a function based on the level l.

The more $f(l)$ is exponential, the more R_{ws} is reflecting the containment for the first levels of templates and ignoring the other levels.

Examples of $f(l)$:

$$f(l) = l$$

$$f(l) = l!$$

$$f(l) = l!$$

Let's take the example explained in this paper and we score the richness of the templates included in the template A, when we are missing the template G.

$R_s = \frac{5}{6} = 0.83$ (Basic richness scoring computation)

When we score the richness with a weighted level based on $f(l) = l$

$T_{wi} = 5$ and $D_{wi} = 4.5 \Rightarrow R_{ws} = 0.9$

When we score the richness with a weighted level based on $f(l) = l^2$

$T_{wi} = 4.5$ and $D_{wi} = 4.25 \Rightarrow R_{ws} = 0.94$

As we can remark, the weighted scoring better reflect the fact that we are missing a template from the L3 and not from the L2. Making $f(l)$ exponential modify the scoring result by making the accent only on the first levels of templates containment.

It is up to the tests provider to define the strategy regarding the definition of $f(l)$.

For a CDA document template, T_{wi} is always a fixed value for a specific $f(l)$: this number allows describing the complexity of a CDA standard.

When f(l) = 1, $R_{ws} = R_s$.

3.4 Tools Richness Scoring

The richness scoring is about CDA documents; however certification programs are about applications and systems. The evaluation of tools shall be calculated against a bench of CDA documents, and the result of the scoring need to take into account the different trees of the provided test documents. The matrix of richness shall be then a summary of all provided templates, and their containments.

3.5 HL7 CDA R2 Validation Reliability

The automatic validation of CDA documents against validation tools is a way to estimate the correctness of the clinical information provided. However, even if a CDA document is valid, the reliability on the validation result depends on two other parameters:

- The reliability on the tool used for the validation; this reliability is based on the requirements coverage regarding the CDA specification used for the validation [14]
- The richness of the CDA documents provided

A low richness scoring of the CDA documents decreases the reliability on the content creator tool. The validation of the CDA documents cannot confirm if in the future the tool is able to create valid CDA documents, if the CDA documents validated do not cover all the possible kind of clinical information.

Reliability = f(Validation result, Requirements Coverage, Documents richness)

4 Implementation

The CDA richness computation was implemented as a part of Gazelle ObjectsChecker [15]. The input for the CDA richness module is the customer templates design coming from ART-DECOR [16]. This templates design is the formal XML description of the CDA specification, based on HL7 Templates Standard. And any tool able to provide this architecture of requirements may be an input for the CDA richness module. This module takes advantage of Gazelle ObjectsChecker for the parsing and for the information extraction from the custom templates design. The other input for this architecture is the CDA document to be scored. The output from the richness module is the matrix of templates containment. From the custom templates design, we generate the CDA validation tool using Gazelle ObjectsChecker, and then we validate the provided CDA document and generate the report of validation. This report contains the report for all checked rules, and also contains the list of found templates on the validated CDA document. The validation report and the Matrix of templates containment are used as input for the richness scoring (Figure 6).

5 Applications

5.1 HL7 CDA Specifications Richness Comparison

As application, we selected a bunch of standards in order to compare their richness in templates and in depth. The basic richness template indicator provides the number of possible sub-templates included in the parent template of the targeted specification. Weighted richness template indicator is the sum of all possible sub-templates weighted by their

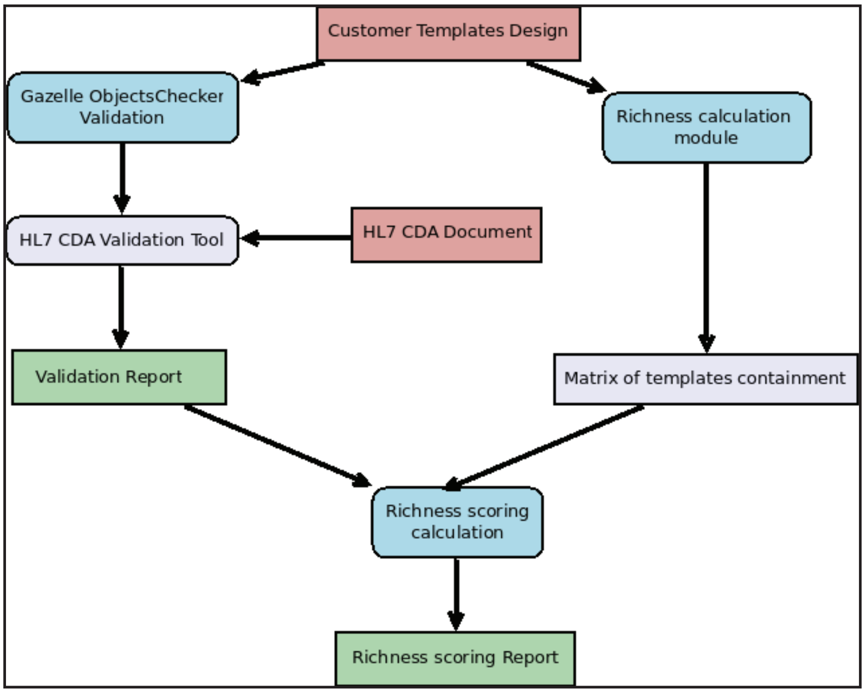
Customer Templates Design
Gazelle ObjectsChecker Validation
Richness calculation module
HL7 CDA Validation Tool
HL7 CDA Document
Validation Report
Matrix of templates containment
Richness scoring calculation
Richness scoring Report

Figure 6: Implementation of richness scoring.

Table 1: CDA implementation guides comparison.

Standards	Basic Richness Template Indicator	Weighted Richness Template Indicator	Number Kind Templates Referenced	Depth
IHE Discharge Summary Specification	1271	52.91	63	10
IHE Immunization Content Specification	796	40.15	68	10
IHE PHR Extract Specification	926	60.86	84	10
IHE Referral Summary Document	862	45.6	66	10
epSOS-Patient Summary	1161	59.67	65	10
C-CDA 2.1 CCD	705	73.77	104	9
C-CDA 2.1 Discharge Summary	645	75.2	101	9
C-CDA 2.1 Diagnostic Imaging Report	42	17.03	17	6

levels (*f(l) = l!*). The depth describes the number of level of templates found in the specification (Table 1).

These metrics provide a big picture of each standard. The weighted template richness indicator provides an indicator for a comparison between the standards. Those with a high T_{wi} are more oriented for the tools processing, and their content is more oriented for CDA L3 content description. Standards with low T_{wi} are more flattened, and then more oriented to human readability. The depth of the standard provides also the same kind of indication. The basic template richness indicator provides a description of the complexity of the architecture of the standard; it describes the number of possible containments between templates. A global remark, nearly all the standards have more than 600 possible containments. This is a huge number and it gives an idea of the complexity of implementing a CDA content creator tool for such specifications.

As example, let's compare the IHE Discharge summary, and the C-CDA Discharge summary characteristics. The number of possible templates for C-CDA DS is bigger than the number of possible templates for IHE-DS (101 against 63). Although the number of possible containments in IHE DS is bigger than C-CDA DS, the weighted comparison describes the C-CDA DS as more flattened than IHE-DS; this is confirmed by the Depth of both standards (10 against 9 levels). We can interpret this by the fact that C-CDA DS is more human readable, with a better granularity of the collected information, and the IHE-DS is more tools interpretation dedicated. Also there are less redundancy of the sub-templates in the tree of containment of C-CDA DS (1271/63 > 645/101); this is interpreted by the fact that C-CDA DS has included more specialized templates than IHE DS, and we have less interpretation to do in C-CDA DS regarding the context of use of a specific sub-template.

5.2 CDA Documents Richness Comparison

In this paragraph, we took a list of CDA documents already validated by Gazelle ObjectsChecker under the database of EVSClient tool [17], and we calculated the average of richness scoring for some available CDA validators (Table 2).

We can remark that all the validated CDA documents are far from covering 100% of the templates containments defined in the CDA implementation guides. Some of the tested documents were used in IHE Connectathon [18], which is not harmful for IHE testing process, but this is dangerous for a certification program; the certification authorities SHALL check the richness of the provided documents [2].

Table 2: Richness scoring analysis of CDA documents.

Standards	Average of Weighted Richness Scoring	Number of tested documents
ePSOS ePrescription	0,229	2447
ePSOS eDispensation	0,201	1322
ePSOS ePatient Summary	0,178	11530
IHE Immunization Content	0,226	276
IHE Referral Summary	0,248	1049
IHE Discharge Summary	0,310	526
IHE PHR Extract	0,139	827
C-CDA Referal Note	0,280	129
C-CDA CCD	0,201	1908

5.3 Interpretation of Richness Scoring

A certification program may define a strategy regarding the accepted CDA documents during a testing process. Based on the C-CDA companion guide, SITE [19] defined 5 score grades regarding the scoring of C-CDA documents [20]: A+, A-, B+, B-, C and D; each of those values describes the quality of the CDA provided. Based on the study of a set of CDA documents related to the implementation guide tested, we can establish a score grade distribution. For example, we take the C-CDA CCD documents tested [20]. Here is the normal distribution of the richness scoring for the 2000 documents tested, coming from EVSClient database (Figure 7).

The distribution of grade is based on the average and the variance of the richness scoring. The average of richness scoring for the C-CDA tested is 0.201, with a variance of 0.07. Here is a possible distribution of grades:

D: $R_{ws} < \mu - 2\sigma$: very low richness

C: $\mu - 2\sigma < R_{ws} < \quad -\sigma$: low richness

B-: $\mu - \sigma < R_{ws} < \mu$: lower than the average

B+: $\mu < R_{ws} < \mu + \sigma$: higher than the average

A-: $\mu + \sigma < R_{ws} < \mu + 2\sigma$: high richness

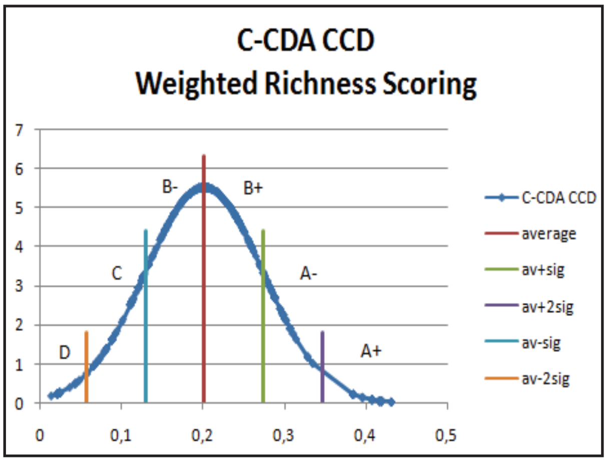

Figure 7: CDA Richness grading.

A+: $R_{ws} > \mu + 2\sigma$: excellent richness

It is up to the testing provider to define the targeted richness grade for the tested CDA tools.

5.4 Combined richness and validation process

IHE-Europe has developed a combined tool for validation and richness analysis of CDA documents. The validation is performed using Gazelle ObjectsChecker. Here is an example of the validation of a C-CDA CCD document, and the GUI provided as validation and richness analysis (Figure 8).

This tree describes all the possible templates that may appear in a CCD document. The green templates in the schema express the fact that such template was found and was valid. The red templates express that such template was found with errors. White templates express that they are missing from the validated CDA document. This output allows having a visible validation and richness report regarding the templates provided.

6 Matrices Glossary

Table 3: Matrices glossary.

Matrix	Name	Description
A	Matrix of inheritance	Describes the inheritance between different templates
B	Matrix of containment	Describes the containment between the different templates as described in the customer templates design
$C = \sum_{n=1}^{\infty} B^n$	Complete matrix of inheritance	Describes all the inheritance between the CDA templates
$\varPhi = (CA + A) \& \overline{\mathrm{AC}}$	Flattened matrix of containment	Describe the final matrix of templates containment
$\varPsi = diag\left(\sum_{i=0}^{n} e_k \varPhi^i\right) x\varPhi$	Refined matrix of containment	This matrix is calculated based on Φ, to describes only the templates related to a specific CDA template
$\mathbf{\Lambda}$	Matrix document description	A matrix to describe the templates containment related to a provided CDA document
$f(l)$	Level heightening	A formula to weight the levels of the CDA templates containment
$T_{wi} = \sum_{l=0}^{\infty} \frac{sum_t(l)}{f(l)}$	Template richness weighted indicator	The richness indicator of possible templates related to a specific root template
$D_{wi} = \sum_{l=0}^{\infty} \frac{sum_d(l)}{f(l)}$	Document richness weighted indicator	The richness indicator of present sub-templates for a CDA document
$R_{ws} = \frac{D_{wi}}{T_{wi}}$	Weighted richness scoring	The weighted scoring computation of a specific template under a CDA document

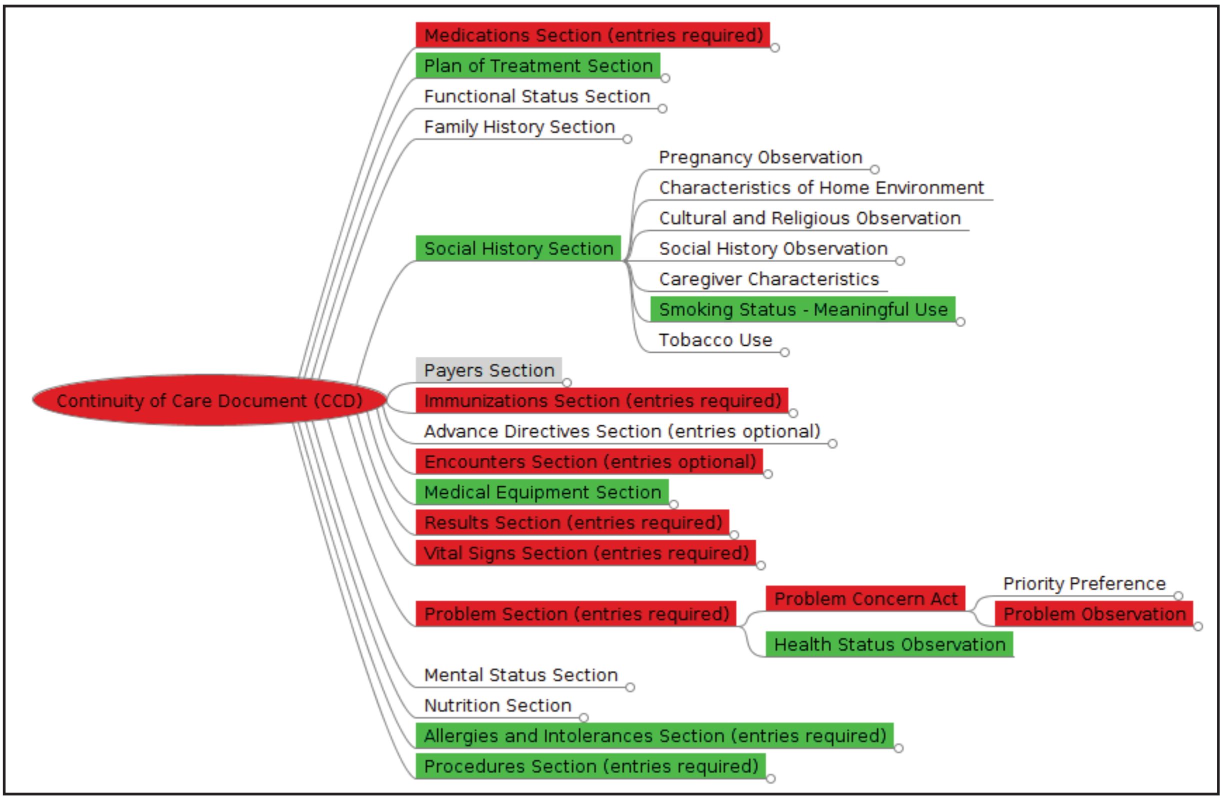
Continuity of Care Document (CCD)
Medications Section (entries required)
Plan of Treatment Section
Functional Status Section
Family History Section
Social History Section
Pregnancy Observation
Characteristics of Home Environment
Cultural and Religious Observation
Social History Observation
Caregiver Characteristics
Smoking Status - Meaningful Use
Tobacco Use
Payers Section
Immunizations Section (entries required)
Advance Directives Section (entries optional)
Encounters Section (entries optional)
Medical Equipment Section
Results Section (entries required)
Vital Signs Section (entries required)
Problem Section (entries required)
Problem Concern Act
Priority Preference
Problem Observation
Health Status Observation
Mental Status Section
Nutrition Section
Allergies and Intolerances Section (entries required)
Procedures Section (entries required)

Figure 8: Combined validation and richness analysis.

7 Conclusion

The richness of CDA documents can be computed based on many criteria. In this paper, we expressed it using the templates containments. Tests providers need to have an indicator of richness to confirm their reliability on the tested applications. Richness study provides reports on areas covered by the testing process.

In this paper, we resolved the main difficulty for scoring the CDA templates, which is the extraction of the complete list of templates containment. Also we defined a methodology to compute the richness of documents tested; this method takes in consideration the specificity of CDA standard, and its architectural levels. Grading of CDA documents is also possible, based on existing test data.

The experimentation of this methodology on some HL7 and IHE specifications confirmed the complexity of these standards, and provided us a way to compare their architectures. The computation of the richness of thousands of CDA documents coming from test tools proved they are in average poor on templates and clinical information. The tests providers need to improve their testing process by taking in consideration the richness of the test data provided.

Many applications of this paper may follow this study, like in conformity assessment accredited testing of IHE CDA documents, and where reporting the coverage of testing area is mandatory during the test procedure.

References

[1] Health Level Seven Internatinoal, Inc. HL7 Clinical Document Architecture, Release 2.0

[2] Internatinoal Organization for Standardization. ISO 17025 General requirements for the competence of testing and calibration laboratories. Geneva: ISO; 2005

[3] Brull R. CDA Levels of Interoperability. Healthcare IT News. June 03, 2011.

[4] D'Amore JD, Mandel JC, Kreda DA, Swain A, Koromia GA, Sundareswaran S. et al. Are Meaningful Use Stage 2 certified EHRs ready for interoperability? Findings from the SMART C-CDA Collaborative. J Am Med Inform Assoc. 2014; 21: 1060-1068.

[5] Boufahja A. Projectathon results and findings on semantic related aspects, eHDSI Project, June 2017

[6] Health Level Seven Internatoinal, Inc. HL7 Implementation Guide: S&I Framework Transitions of Care Companion Guide to Consolidated-CDA for Meaningful Use Stage 2, Release 1 – US Realm. Ann Arbor: HL7; 2014.

[7] SMART. C-CDA Smart Scorecard, http://ccda-scorecard.smartplatforms.org/

[8] eHealth DSI – ePrescription and Patient Summary, Test Framework, version 1.1.0, 01/06/2017

[9] Health Level Seven International, Inc. HL7 Templates Standard: Specification and Use of Reusable Information Constraint Templates, Release 1. Ann Arbor: HL7.

[10] Boone K. A Template Meta-model, http://motorcycleguy.blogspot.fr/2011/04/template-meta-model.html

[11] eHealth DSI CDA Implementation Guides, June 2017.

[12] Benson T. Principles of Health Interoperability HL7 and SNOMED, Section 9.3. Berlin: Springer; 2010.

[13] Horn RA, Johnson CR. Matrix Analysis, 2nd Ed. Cambridge: University Printing House; 2013.

[14] Boufahja A, Poiseau E, Thomazon G, Bergé A. Model-based Analysis of HL7 CDA R2 Conformance and Requirements Coverage. EJBI 2015; 2: en41-en50.

[15] Boufahja A, Heitmann K, Poiseau E. Model-based Validation of HL7 CDA R2 Documents and Implementation Guides Using Gazelle ObjectsChecker and ART-DECOR. EJBI 2016; 1: en62-en69.

[16] Heitmann K. ARTDECOR Basics and User Perspective. Tutorial, IHIC 2015, Prague; 2015.

[17] Gazelle project. EVSClient tool, https://gazelle.ihe.net/EVSClient/

[18] IHE-Europe. The IHE Connectathon: what it is and how it is done, November 2015

[19] The Standards Testing & Implementation Environment (SITE), C-CDA scorecard, https://sitenv.org/scorecard/

[20] Health Level Seven Internatinal, Inc. Consolidated CDA Templates for Clinical Notes (US Realm), DSTU 2.1.