CDA Tutorial

Understanding CDA Templates

Austin Kreisler

Consultant – CDC

Leidos Technical Fellow

CDA Overview

- **What is clinical documentation?**
- **Characteristics of clinical documents**
- **HL7 CDA structure overview**
- **CDA's Templates**
- **Q/A and Discussion**

What is Clinical Documentation?

- ❑ **Documentation of clinical observations and services**
- ❑ **Includes but is not limited to documentation appearing in a patients medical record**
- ❑ **Includes the documentation of any care provided to a patient, regardless of care setting**
- ❑ **Examples:**
  - ▪ History and Physical Notes
  - ▪ Care Summaries
  - ▪ Laboratory Reports
  - ▪ Imaging Reports
 - ▪ Operative Notes
  - ▪ …

Characteristics of a Clinical Document

- **Persistence – A clinical document continues to exist in an unaltered state, for a time period**
- **Stewardship – A clinical document is maintained by an organization entrusted with its care**
- **Potential for authentication - A clinical document is an assemblage of information that is intended to be legally authenticated**
- **Context - A clinical document establishes the default context for its contents**
- **Wholeness - Authentication of a clinical document applies to the whole**
- **Human readability – A clinical document is human readable**

HL7 CDA Structure Overview

CDA Structure - Header

- **Metadata about the document**

- **Focused on data for document**
  - Indexing
  - Document authentication
  - Document context

- **Supports document management**

HL7 CDA Header

Header
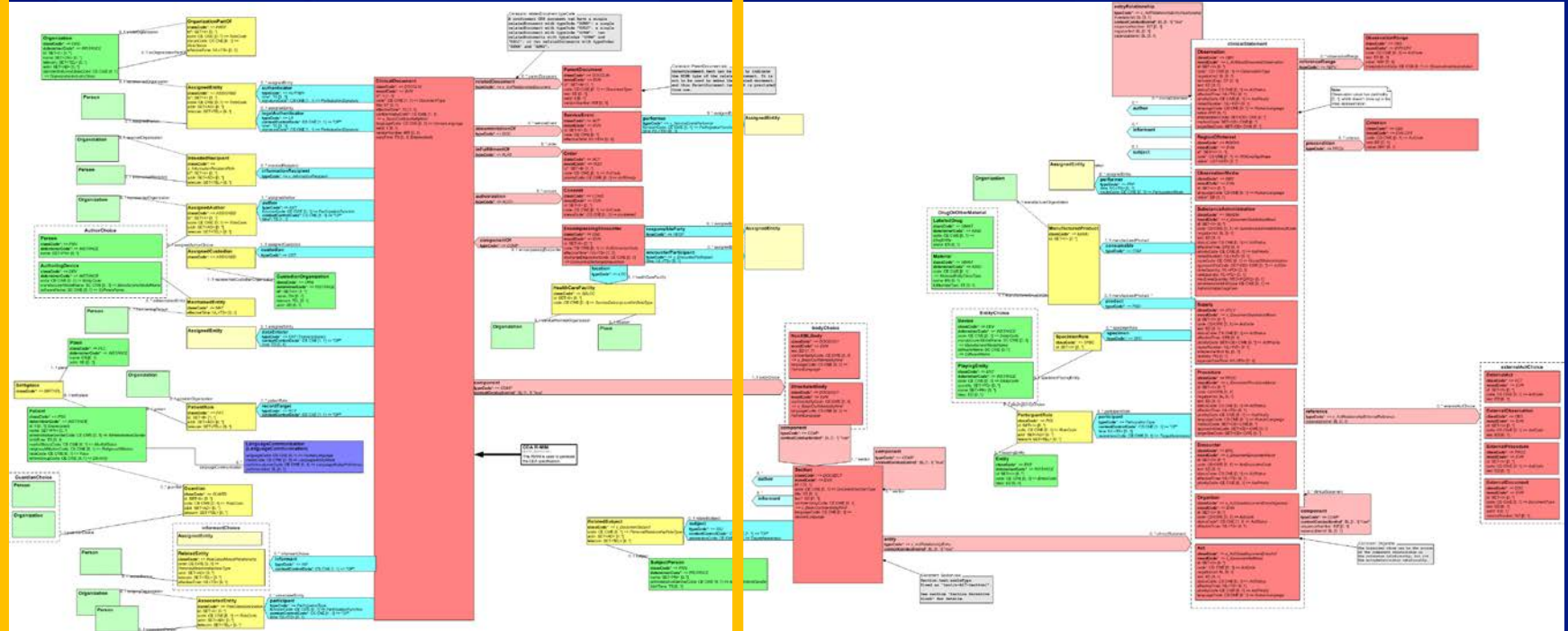
CDA Structure – Body

- **Unstructured Body provides a container for non-XML content**
- **Structured Body that provides both structured human readable narrative as well as machine readable content**
  - Narrative block that provides the human readable content and represents the authenticated content of the document
  - Entries that optionally provide a discrete, machine readable representation of the document content
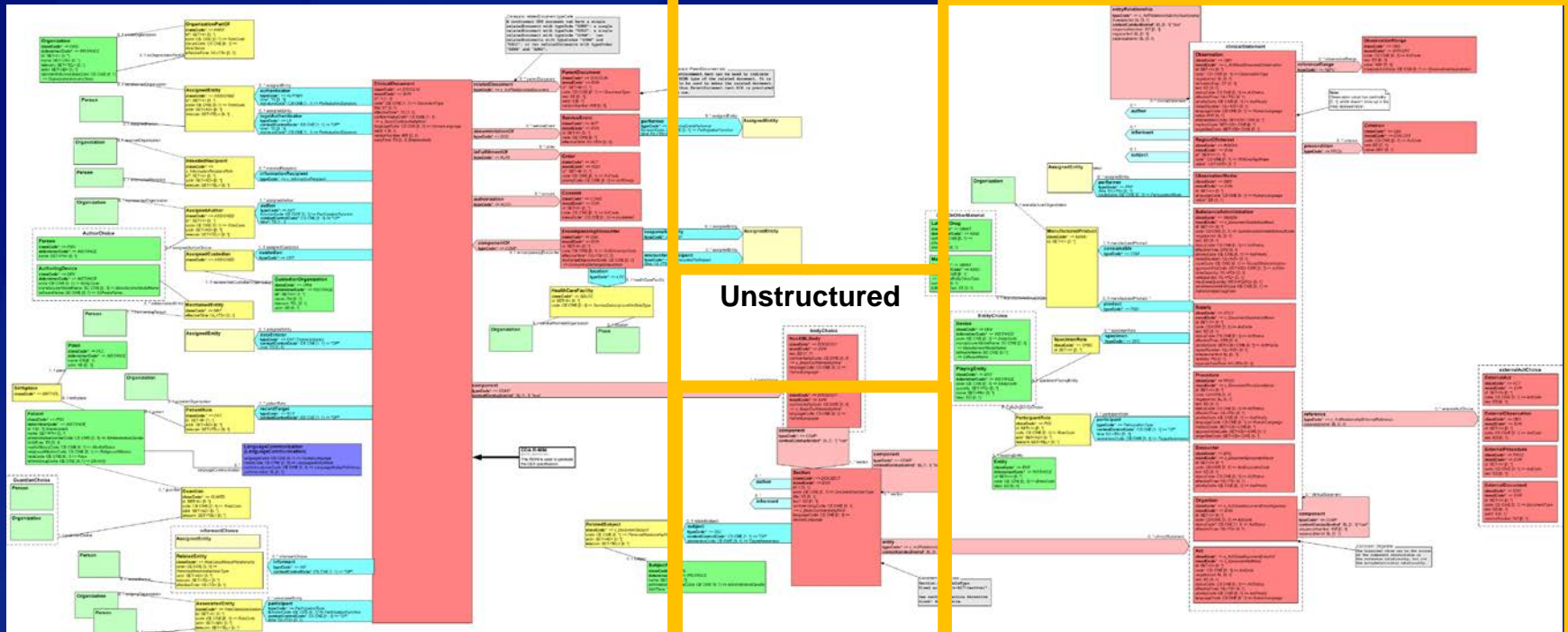
HL7 CDA Body

Body

Unstructured

Structured Narrative

Structured Entries

Structured Narrative & Structured Entries

- **Structured Narrative – Section text**
  - Human readable text that represents the "attested" content of the document
- **Structured Entries – Entries**
  - Machine processable discrete data
- **Two kinds of relationships between the Section text and contained Entries**
  - Derived (DRIV) – The information in Section text is intended to be a faithful and complete rendering of the clinical content of the contained entries
  - Component (COMP) – The entries considered as being contained in the source section and no other meaning is implied.

CDA & XML

- **XML is Extensible Markup Language (www.w3c.org)**
- **In XML, structure and format are conveyed by markup which is embedded into the information**

<markup>text</markup>

```
<section>
<title>Hospital Course</title>
<text> The patient was admitted and started on Lovenox and
       nitroglycerin paste. The patient had serial cardiac
       enzymes and was ruled out for myocardial infarction.
       The patient underwent a dual isotope stress test.
       There was no evidence of reversible ischemia on the
       Cardiolite scan. The patient has been ambulated.
</text>
</section>
```

XML

- **Designed to describe data**
- **Tags (the text between the angle brackets "<Tag>") is not predefined**
- **Intent is that the tags be self descriptive**
  - **Use <name> instead of <x> if you are intending to convey the name of something**
- **XML doesn't do anything on its own**
  - **You need some sort of processor to deal with XML**

XML Example

```
<?xml version="1.0" encoding="UTF-8"?>
<memo>
  <to>Dick</to>
  <from>Jane</from>
  <subject>Taking a walk</subject>
  <body>We should take a <b><i>walk</i></b>
          up the hill</body>
</memo>
```

Closing tag has a leading backslash

An XML Element includes matching opening and closing tags

Required root element

Elements must be properly nested

CDA's Dirty Little Secret

- ❑ It's built on the HL7 Version 3.0 RIM
- ❑ HL7 V3 has a reputation for being too complex
- ❑ CDA's has a secret weapon for dealing with V3 Complexity called -

Templates

Templates
CDA's Secret Weapon

- ❑ **TEMPLATES**
  - ▪ In engineering, complex problems can some times be solved by breaking the problem down into smaller parts and finding solutions for the small parts
  - ▪ CDA Templates do exactly this
- ❑ **CDA Implementation Guides**
  - ▪ Essentially a collection of templates gathered to together for a particular purpose

What is a Template?

- **A template represents a formal definition of a set of constraints on a model**
- **A template has two parts:**
  - Metadata such as an identifier, version, description, etc.
  - Body that contains the actual constraints
- **Templates can be seen as a "set of instructions" for the proper creation of an instance of data for a specific use**

CDA Templates - Levels of Constraint

❑ **The CDA standard describes conformance requirements in terms of three general levels corresponding to three different, incremental types of conformance statements:**

❑ **Level 1 requirements impose constraints upon the CDA Header. The body of a Level 1 document may be XML or an alternate allowed format. If XML, it must be CDA-conformant markup.**

❑ **Level 2 requirements specify constraints at the section level of a CDA XML document: most critically, the section code and the cardinality of the sections themselves, whether optional or required.**

❑ **Level 3 requirements specify constraints at the entry level within a section. A specification is considered "Level 3" if it requires any entry-level templates.**

Types of Templates

❑ **Document-level templates: These templates constrain fields in the CDA header, and define containment relationships to CDA sections.**

❑ **Section-level templates: These templates constrain fields in the CDA section, and define containment relationships to CDA entries.**

❑ **Entry-level templates: These templates constrain the CDA clinical statement model in accordance with real world observations and acts.**

❑ **Other templates: These templates group a common set of constraints for reuse in CDA documents.**

HL7 CDA Templates – Conformance Statements

- **CDA Templates are expressed as a collection of Conformance Statements**
- **Conformance statements constrain some aspect of a CDA class or classes**
- **Example:**
  - **SHALL** contain exactly one [1..1] **@classCode**="OBS" Observation (CodeSystem: 2.16.840.1.113883.5.6 HL7ActClass) **STATIC** (CONF:7345)

Constraint Types

- ❑ **Optionality (aka. Appearance): determine whether a particular element must appear in models or messages derived from the base model, and/or whether the element is precluded from appearing therein.**

- ❑ **Cardinality: define the number of repetitions that may occur for a given element.**

- ❑ **Type: limit the structure (datatype) of the element in question.**

- ❑ **Vocabulary: limit the set of concepts that can be taken as valid values in an instance of a coded attribute or field.**

Optionality Constraints - Conformance Verbs

- *SHALL: an absolute requirement*
- *SHALL NOT: an absolute prohibition against inclusion*
- *SHOULD/SHOULD NOT: best practice or recommendation. There may be valid reasons to ignore an item, but the full implications must be understood and carefully weighed before choosing a different course*
- *MAY/NEED NOT: truly optional; can be included or omitted as the author decides with no implications*

Cardinality Constraints

❑ **Number of times a repeating element may appear in a CDA document instance**

❑ **Expressed as a minimum and a maximum values: [0..1]**

❑ **Examples:**

- ▪ [0..1] as zero to one present
- ▪ [1..1] as one and only one present
- ▪ [1..*] as one or more present
- ▪ [0..*] as zero to many present
- ▪ [0..0] as zero present

Type Constraints

- **All attributes in HL7 V3 models (such as CDA) have an associated data type**
  - Data types are essentially a collection attributes that describe the data being conveyed
  - All the V3 data types derive from a data type called ANY
- **The most common data type constrain in templates is for the Observation.value attribute which uses the ANY data type**
  - In a CDA Observation template, a specific data type needs to be assigned to the Observation.value
- **Example: SHALL contain exactly one [1..1] value with @xsi:type="CD"**
- **xsi:type is the XML schema language mechanism for overriding an element's base data type**

Examples: xsi:type

```
<value xsi:type="CD"
       code="T2"
       codeSystem="2.16.840.1.113883.15.6"
       codeSystemName="TNM 7. Edition"
       displayName="Tumor &gt; 20 mm but &lt;= to
       50 mm in greatest dimension"/>



<value xsi:type="PQ" value="57" unit="a"/>
```

Vocabulary Constraints

❑ **Limit the set of concepts that can be taken as valid values in an instance of a coded attribute or field**

❑ **Single code binding – Limiting the value to a single coded value**

- Example: SHALL contain exactly one [1..1] code/@code="21843-8" Usual Occupation Hx (CodeSystem: LOINC 2.16.840.1.113883.6.1)

❑ **Value Set binding – Limiting the set to a value set of allowed codes**

- SHALL contain exactly one [1..1] value with @xsi:type="CD" to record the occupation of the patient, where the @code SHALL be selected from ValueSet Census Occupation Codes 2.16.840.1.114222.4.11.6036 DYNAMIC.
- Value-set constraints can be STATIC, meaning that they are bound to a specified version of a value set, or DYNAMIC, meaning that they are bound to the most current version of the value set.

Vocabulary Constraints

❑ **Code system constraint narrows the allowed values to a single or multiple code systems**

- where the code SHALL be selected from LOINC (CodeSystem: 2.16.840.1.113883.6.1) or SNOMED CT (CodeSystem: 2.16.840.1.113883.6.96) (CONF:7166).

OIDs – ISO Object Identifier

❑ **A string of numbers and dots that are used to uniquely identify something (an object).**

- ▪ 2.16.840.1.113883.6.1

❑ **HL7 uses OIDs to identify coding systems and value sets**

❑ **CDA Templates use OIDs to uniquely identify the template**

- ▪ SHALL contain exactly one [1..1] templateId such that it
  - a. SHALL contain exactly one [1..1] @root="1.3.6.1.4.1.19376.1.5.3.1.3.28"

Code System vs. Value Set OIDs

SHALL contain exactly one [1..1] targetSiteCode with @xsi:type="CD" to indicate the anatomic location where the primary tumor originated.

a. The targetSiteCode **SHALL** contain exactly one [1..1] @code, where the @code **SHALL** be selected from Value Set Body Site (SNOMED CT) **2.16.840.1.113883.3.88.12.3221.8.9** DYNAMIC

❑ **Wrong:**

▪ ```
<targetSiteCode xsi:type="CD" code="181131000"
codeSystem="2.16.840.1.113883.3.88.12.3221.8.9"
codeSystemName="SNOMED CT" displayName="Entire
breast">
```

❑ **Correct:**

▪ ```
<targetSiteCode xsi:type="CD" code="181131000"
codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Entire
breast">
```

Representing Null Flavors

❑ **Null Flavor – A mechanism for handling the situation where you have incomplete data**

- ▪ The data may be completely or partially missing or it may be present but not valid according to constraints
- ▪ Provides a reason why data is absent

Example:

<birthTime nullFlavor="UNK"/>

Null flavor – Coded Values

- **NI - No information - default null flavor.**
- **NA - Not applicable - Known to have no proper value**
- **UNK – Unknown - Proper value is applicable, but is not known**
- **ASKU - Asked, but not known - Information was sought, but not found**
- **NAV - Temporarily unavailable**
- **NASK - Not asked**
- **MSK - There is information on this item available but it has not been provided by the sender due to security, privacy, or other reasons**
- **OTH – Other - actual value is not a member of the set of permitted data values**

Null Flavors and CDA Templates

- **Templates may require the presence of data even when the document originator has no data**

- **Example:**
  - SHALL contain at least one [1..*] id
  - <id nullFlavor="NI"/>

- Example:
  - SHALL contain exactly one [1..1] code
  - <code nullFlavor="OTH">
    <originalText>New Grading system</originalText>
    </code>

Null Flavor Explicitly Disallowed in a Template

1. SHALL contain exactly one [1..1] effectiveTime

       a. SHALL NOT contain [0..0] nullFlavor

Note: The above use of SHALL NOT for null flavors isn't actually used in the current IG:

❑ **Implementation Guide for Ambulatory Healthcare Provider Reporting to Central Cancer Registries, Release 1.0, August 2012**

Unknown Information

❑ **If the sender doesn't know an attribute of an act, that attribute can be null.**

```
<entry>
   <text>patient was given a medication but I do not know what it
       was</text>
   <substanceAdministration moodCode="EVN" classCode="SBADM">
     <consumable>
       <manufacturedProduct>
         <manufacturedLabeledDrug>
           <code nullFlavor="NI"/>
         </manufacturedLabeledDrug>
       </manufacturedProduct>
     </consumable>
   </substanceAdministration>
</entry>
```

Unknown Information

❑ If the sender doesn't know if an act occurred, the nullFlavor is on the act (detail could include specific allergy, drug, etc.)

```xml
<entry>
  <substanceAdministration moodCode="EVN" classCode="SBADM"
  nullFlavor="NI">
    <text>I do not know whether or not patient received an
          anticoagulant drug</text>
    <consumable>
      <manufacturedProduct>
        <manufacturedLabeledDrug>
          <code code="81839001" displayName="anticoagulant drug«
                codeSystem="2.16.840.1.113883.6.96"
                codeSystemName="SNOMED CT"/>
        …
</entry>
```

Unknown Information

❏ If the sender wants to state 'no known', a negationInd can be used on the corresponding act (substanceAdministration, Procedure, etc.)

```
<entry>
  <substanceAdministration moodCode="EVN" classCode="SBADM"
    negationInd="true">
    <text>No known medications</text>
      <consumable>
        <manufacturedProduct>
          <manufacturedLabeledDrug>
            <code code="410942007" displayName="drug or
                    medication"
                    codeSystem="2.16.840.1.113883.6.96"
                    codeSystemName="SNOMED CT"/>
          </manufacturedLabeledDrug>
          …
</entry>
```

What's Next—Cancer CDA Training Part 2

❑ **Hierarchy of validation tools and appropriate order of use**

❑ **Walk-through of CDA document marked up with instructions for how to find relevant data elements**

Patient's Medical Record Number: (Note, this is a "dummy" OID in the root. The actual OID used here will probably be one set up by the EHR vendor to identify each of their vendors):
```
<id extension="112334-7" root="2.16.840.1.113883.19"/>
```

Patient's Social Security Number: (The OID "2.16.840.1.113883.4.1" in the root tells you that the number in the extension must be a Social Security #. This OID can be found in the HL7 OID registry, http://hl7.amg-hq.net/oid/):
```
<id root="2.16.840.1.113883.4.1" extension="123-45-6789"/>
```

❑ **Case studies—review actual examples from vendor reports to see types of errors and understand their possible causes**

CDA Validation Plus error: The Code System OID 2.16.840.1.113883.3.88.12.3221.8.9 for Primary Site Code System OID is not valid. Valid Code System OIDs for this field are 2.16.840.1.113883.6.103, 2.16.840.1.113883.6.90, 2.16.840.1.113883.6.96
```
<targetSiteCode code="255052006" codeSystem="2.16.840.1.113883.3.88.12.3221.8.9" codeSystemName="SNOMED CT Body Site Value Set" displayName="Malignant tumor of unknown origin">
```

QUESTIONS

For additional questions, please email Lindsay Ryan at VIU3@cdc.gov.

Austin Kreisler

HL7 Subject Matter Expert - Technical Fellow

Leidos Health Solutions

mobile: 706-525-1181

duz1@cdc.gov

austin.j.kreisler@leidos.com