Integrating the Healthcare Enterprise

IHE Quality, Research and Public Health (QRPH)

Technical Framework
Volume 2
Revision 0.1

Trial Implementation

September 2, 2011

Contents

Trial Implementation: Volume 2

This document is published as a Trial Implementation Document. It provides the structure for QRPH TF-2 that will be published as Final Text. QRPH supplements in Trial Implementation as published as changes to this document. When this document is published in its Final Text version, QRPH supplements will then be published as changes to that version of the document.

This section (Trial Implementation: Volume 2) will be removed in the Final Text publication.

1 Introduction

1.1 Intended Audience

The intended audience of this document is:

- Technical staff of vendors planning to participate in the IHE initiative
- IT departments of healthcare institutions
- Experts involved in standards development
- Anyone interested in the technical aspects of integrating healthcare information systems

1.2 Related Information for the Reader

The reader of volume 2 should read or be familiar with the following documents:

- Volume 1 of the Cross-Enterprise Document Sharing (XDS) Integration Profile documented in the ITI Infrastructure Technical Framework (See http://www.ihe.net/Technical_Framework/index.cfm).
- Volume 1 of the Notification of Document Availability (NAV) Integration Profile documented in the ITI Infrastructure Technical Framework (See http://www.ihe.net/Technical_Framework/index.cfm).
- Volume 1 of the Audit Trail and Node Authentication (ATNA) Integration Profile documented in the ITI Infrastructure Technical Framework (See http://www.ihe.net/Technical_Framework/index.cfm).
- HL7 Clinical Document Architecture Release 2: Section 1, CDA Overview.
- Care Record Summary – Implementation Guide for CDA Release 2 (US Realm): Section 1
- Presentations from IHE Workshop: Effective Integration of the Enterprise and the Health System - June 28–29, 2005: http://www.ihe.net/Participation/workshop_2005.cfm, June 2005:
- For a RHIO-3.ppt Leveraging IHE to Build RHIO Interoperability
- Cross-Enterprise Document Sharing (XDS)
- Notification of Document Availability (NAV)

- Use Cases for Medical Summaries
- Ovrw.ppt Patient Care Coordination - Overview of Profiles

1.2.1 How this Document is Organized

Section 1 is the preface, describing the intended audience, related resources, and organizations and conventions used within this document.

Section 2 provides an overview of the concepts of IHE actors and transactions used in IHE to define the functional components of a distributed healthcare environment.

Section 3 defines transactions in detail, specifying the roles for each actor, the standards employed, the information exchanged, and in some cases, implementation options for the transaction.

Section 4 defines a set of payload bindings with transactions.

Section 5 defines the content modules that may be used in transactions.

1.2.2 Conventions Used in this Volume

This document has adopted the following conventions for representing the framework concepts and specifying how the standards upon which the IHE Technical Framework is based should be applied.

1.2.2.1 The Generic IHE Transaction Model

Transaction descriptions are provided in section 4. In each transaction description, the actors, the roles they play, and the transactions between them are presented as use cases.

The generic IHE transaction description includes the following components:

- Scope: a brief description of the transaction.
- Use case roles: textual definitions of the actors and their roles, with a simple diagram relating them, e.g.,:

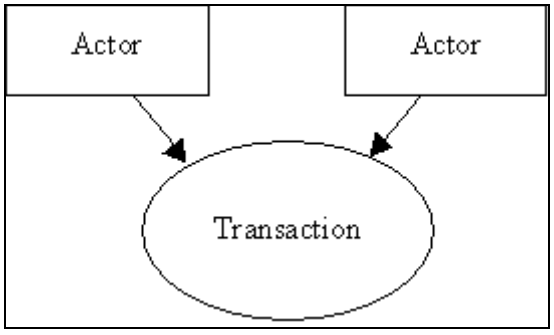

Figure 1.2.2.1-1 Use Case Role Diagram

- *Referenced Standards*: the standards (stating the specific parts, chapters or sections thereof) to be used for the transaction.
- *Interaction Diagram*: a graphical depiction of the actors and transactions, with related processing within an actor shown as a rectangle and time progressing downward, similar to:

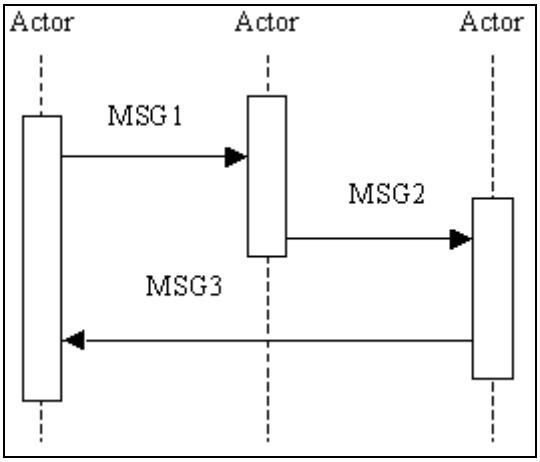

Figure 1.2.2.1-2 Interaction Diagram

The interaction diagrams used in the IHE Technical Framework are modeled after those described in Grady Booch, James Rumbaugh, and Ivar Jacobson, *The Unified Modeling Language User Guide*, ISBN 0-201-57168-4. Simple acknowledgment messages are omitted from the diagrams for brevity.

- *Message definitions*: descriptions of each message involved in the transaction, the events that trigger the message, its semantics, and the actions that the message triggers in the receiver.

1.3 Copyright Permissions

Health Level Seven, Inc., has granted permission to the IHE to reproduce tables from the HL7 standard. The HL7 tables in this document are copyrighted by Health Level Seven, Inc. All rights reserved. Material drawn from these documents is credited where used.

1.4 Comments

IHE International welcomes comments on this document and the IHE initiative. They can be submitted using the Web-based comment form at www.ihe.net/qrph/qrphcomments.cfm or by sending an email to the co-chairs and secretary of the Quality, Research and Public Health domain committees at qrph@ihe.net.

2 Overview of Technical Framework

This document, the IHE Quality, Research and Public Health Technical Framework (IHE QRPH TF-2), defines specific implementations of established standards. These are intended to achieve integration goals that promote appropriate exchange of medical information to coordinate the optimal patient care among care providers in different care settings. It is expanded annually, after a period of public review, and maintained regularly through the identification and correction of errata. The latest version of the document is always available via the Internet at http://www.ihe.net/Technical_Framework/index.cfm, where the technical framework volumes specific to the various healthcare domains addressed by IHE may be found.

The IHE Quality, Research and Public Health Technical Framework identifies a subset of the functional components of the healthcare enterprises and health information networks, called IHE actors, and specifies their interactions in terms of a set of coordinated, standards-based transactions.

The other domains within the IHE initiative also produce Technical Frameworks within their respective areas that together form the IHE Technical Framework. Currently, the following IHE Technical Framework(s) are available:

- IHE IT Infrastructure Technical Framework
- IHE Cardiology Technical Framework
- IHE Laboratory Technical framework
- IHE Radiology Technical Framework
- IHE Patient Care Coordination Technical Framework

Where applicable, references are made to other technical frameworks. For the conventions on referencing other frameworks, see the preface of this volume.

2.1 Relationship to Standards

The IHE Technical Framework identifies functional components of a distributed healthcare environment (referred to as IHE actors), solely from the point of view of their interactions in the healthcare enterprise. At its current level of development, it defines a coordinated set of transactions based on standards (such as HL7, IETF, ASTM, DICOM, ISO, OASIS, etc.) in order to accomplish a particular use case. As the scope of the IHE initiative expands, transactions based on other standards may be included as required.

Each transaction may have as its payload one or more forms of content, as well as specific metadata describing that content within the transaction. The specification of the payload and metadata about it are the components of a Content Integration Profile. The payload is specified in a Content Module, and the impacts of any particular payload on a transaction are described within a content binding. The payloads of each transaction are also based on standards (such as HL7, IETF, ASTM, DICOM, ISO, OASIS, etc.), again, in order to meet the needs of a specific use case.

In some cases, IHE recommends selection of specific options supported by these standards. However, IHE does not introduce technical choices that contradict conformance to these standards. If errors in or extensions to existing standards are identified, IHE's policy is to report them to the appropriate standards bodies for resolution within their conformance and standards evolution strategy.

IHE is therefore an implementation framework, not a standard. Conformance claims for products must still be made in direct reference to specific standards. In addition, vendors who have implemented IHE integration capabilities in their products may publish IHE Integration Statements to communicate their products' capabilities. Vendors publishing IHE Integration Statements accept full responsibility for their content. By comparing the IHE Integration Statements from different products, a user familiar with the IHE concepts of actors and integration profiles can determine the level of integration between them. See IHE QRPH TF-1: Appendix C for the format of IHE Integration Statements.

2.2 Relationship to Product Implementations

The IHE actors and transactions described in the IHE Technical Framework are abstractions of the real-world healthcare information system environment. While some of the transactions are traditionally performed by specific product categories (e.g., HIS, Clinical Data Repository, Electronic Health record systems, Radiology Information Systems, Clinical Information Systems or Cardiology Information Systems), the IHE Technical Framework intentionally avoids associating functions or actors with such product categories. For each actor, the IHE Technical Framework defines only those functions associated with integrating information systems. The IHE definition of an actor should therefore not be taken as the complete definition of any product that might implement it, nor should the framework itself be taken to comprehensively describe the architecture of a healthcare information system.

The reason for defining actors and transactions is to provide a basis for defining the interactions among functional components of the healthcare information system environment. In situations where a single physical product implements multiple functions, only the interfaces between the product and external functions in the environment are considered to be significant by the IHE initiative. Therefore, the IHE initiative takes no position as to the relative merits of an integrated environment based on a single, all-encompassing information system versus one based on multiple systems that together achieve the same end.

2.3 Relation of this Volume to the Technical Framework

The IHE Technical Framework is based on actors that interact through transactions using some form of content.

Actors are information systems or components of information systems that produce, manage, or act on information associated with operational activities in the enterprise.

Transactions are interactions between actors that transfer the required information through standards-based messages.

The implementation of the transactions described in this IHE QRPH TF-2 support the specification of Integration Profiles defined in IHE QRPH TF-1. The role and implementation of these transactions require the understanding of the Integration profile they support.

There is often a very clear distinction between the transactions in a messaging framework used to package and transmit information, and the information content actually transmitted in those messages. This is especially true when the messaging framework begins to move towards mainstream computing infrastructures being adopted by the healthcare industry.

In these cases, the same transactions may be used to support a wide variety of use cases in healthcare, and so more and more the content and use of the message also needs to be profiled, sometimes separately from the transaction itself. Towards this end IHE has developed the concept of a Content Integration Profile.

Content Integration Profiles specify how the payload of a transaction fits into a specific use of that transaction. A content integration profile has three main parts. The first part describes the use case. The second part is binding to a specific IHE transaction, which describes how the content affects the transaction. The third part is a Content Module, which describes the payload of the transaction. A content module is specified so as to be independent of the transaction in which it appears.

2.3.1 Content Modules

The Quality, Research and Public Health Technical Framework organizes content modules categorically by the base standard. At present, the QRPH Technical Framework uses only one base standard, CDA Release 2.0, but this is expected to change over time. Underneath each standard, the content modules are organized using a very coarse hierarchy inherent to the standard. So for CDA Release 2.0 the modules are organized by document, section, entry, and header elements.

Each content module can be viewed as the definition of a "class" in software design terms, and has associated with it a name. Like "class" definitions in software design, a content module is a "contract", and the QRPH Technical Framework defines that contract in terms of constraints that must be obeyed by instances of that content module. Each content module has a name, also known as its template identifier. The template identifiers are used to identify the contract agreed

to by the content module. The QRPH Technical Committee is responsible for assigning the template identifiers to each content module.

Like classes, content modules may inherit features of other content modules of the same type (Document, Section or Entry) by defining the parent content module that they inherit from. They may not inherit features from a different type. Although information in the CDA Header is in a different location that information in a CDA Entry, these two content modules are considered to be of the same type, and so may inherit from each other when necessary.

The QRPH Technical Framework uses the convention that a content module cannot have more than one parent (although it may have several ancestors). This is similar to the constraint in the Java™ programming language, where classes can derive from only one parent. This convention is not due to any specific technical limitation of the technical framework, but does make it easier for software developers to implement content modules.

2.3.1.1 Conformance Requirements: Form 1

Many sections and entries in the PCC and QRPH Technical Framework are using conformance requirements as defined in this section (Form 1). Unless otherwise documented, the tables with conformance requirements use Form 1.

Each content module has a list of data elements that are required (R), required if known (R2), and optional (O). The presentation of this information varies with the type of content module, and is described in more detail below. Additional data elements may be provided by the sender that are not defined by a specific content module, but the receiver is not required to interpret them.

Required data elements must always be sent. Data elements that are required may under exceptional circumstances have an unknown value (e.g., the name of an unconscious patient). In these cases the sending application is required to indicate the reason that the data is not available.

Data elements that are marked required if known (R2) must be sent when the sending application has that data available. The sending application must be able to demonstrate that it can send all required if known elements, unless it does not in fact gather that data. When the information is not available, the sending application may indicate the reason that the data is not available.

Data elements that are marked optional (O) may be sent at the choice of the sending application. Since a content module may include data elements not specified by the profile, some might ask why these are specified in a content module. The reason for specifying the optional data elements is to ensure that both sender and receiver use the appropriate semantic interpretation of these elements. Thus, an optional element need not be sent, but when it is sent, the content module defines the meaning of that data element, and a receiver can always be assured of what that data element represents when it is present. Senders should not send an optional data element with an unknown value. If the value is not known, simply do not send the data element.

2.3.1.2 Conformance Requirements: CDA Consolidation Project

In an attempt to clarify our documentation and required fields, we have adopted a method that follows the documentation produced by the CDA Harmonization Working Group of the Standards and Interoperability (S&I) Framework (http://wiki.siframework.org). The material quoted in this section refers to conformance requirements stated at the document level for QRPH CDA documents. Though a similar approach can be applied at the section or entry level, these definitions are limited to document level requirements. Future versions of this Technical Framework may use this methodology for other levels in CDA documents.

Each document definition found in this volume will contain a table similar to the example in Table 2.3.1.2-1 below. The table lists the sections that are included in the specific document. The table does not specify any conformance requirements. Those conformance requirements are defined using the mechanism found after Table 2.3.1.2-1.

Table 2.3.1.2-1: Example List of Sections in a Document

Template Name	Section Template ID	Volume 2 Location
Injury Incident Description	1.3.6.1.4.1.19376.1.5.3.1.1.19.2.1	PCC CDA Content Modules:6.3.3.1.10
Mass Casualty Incident	1.3.6.1.4.1.19376.1.5.3.1.1.25.2.3	PCC CDA Content Modules: 6.3.3.7.8
Unit Response Level	1.3.6.1.4.1.19376.1.5.3.1.1.25.2.4	PCC CDA Content Modules: 6.3.3.7.9
Protocols Used	1.3.6.1.4.1.19376.1.5.3.1.1.25.2.5	PCC CDA Content Modules: 6.3.3.6.21
Intravenous Fluids Administered	1.3.6.1.4.1.19376.1.5.3.1.1.13.2.6	PCC CDA Content Modules:6.3.3.8.4

Italicized text that appears in the remainder of this section is quoted from the draft output of the CDA Harmonization Working Group. When their documentation is complete and published, we intend to reference that material.

Most conformance statements within this implementation guide are presented as constraints from a Template Database (Tdb). An algorithm converts constraints recorded in a Templates Database to a printable presentation. Each constraint is uniquely identified by an identifier at or near the end of the constraint (e.g., CONF:7345). These identifiers are persistent but not sequential. Constraints from the Model-Driven Health Tools (MDHT) look similar, but do not contain the same unique identifiers; many reference conformance statements from the source document.

In this document, the algorithm that converts from the database to an identifier is manual.

The keywords shall, should, may, need not, should not, and shall not in this document are to be interpreted as described in the HL7 Version 3 Publishing Facilitator's Guide (http://www.hl7.org/v3ballot/html/help/pfg/pfg.htm):

- *shall: an absolute requirement*

- *shall not: an absolute prohibition against inclusion*
- *should/should not: valid reasons to include or ignore a particular item, but must be understood and carefully weighed*
- *may/need not: truly optional; can be included or omitted as the author decides with no implications*

The keyword "shall" implies a lower cardinality of 1, but allows NULL values. If NULL values are to be excluded, it will be via an additional explicit conformance statement.

The cardinality indicator (0..1, 1..1, 1.., etc.) specifies the allowable occurrences within a document instance. The cardinality indicators may be interpreted as follows:*

- *0..1 zero or one*
- *1..1 exactly one*
- *1..* at least one*
- *0..* zero or more*
- *1..n at least one and not more than n*

If a template is a specialization of another template, its first constraint indicates the more general template. In all cases where a more specific template conforms to a more general template, asserting the more specific template also implies conformance to the more general template.

When a constraint has subordinate clauses, the scope of the cardinality of the parent constraint must be clear. In the next figure, the constraint says exactly one participant is to be present. The subordinate constraint specifies some additional characteristics of that participant.

Information technology solutions store and manage data, but sometimes data are not available: an item may be unknown, not relevant, or not computable or measureable. In HL7, a flavor of null, or nullFlavor, describes the reason for missing data.

For example, if a patient arrives at an Emergency Department unconscious and with no identification, we would use a null flavor to represent the lack of information. The patient's birth date would be represented with a null flavor of "NAV", which is the code for "temporarily unavailable". When the patient regains consciousness or a relative arrives, we expect to know the patient's birth date.

Use one of the following null flavors for unknown, required or optional attributes. Null flavors may not be used for mandatory attributes:

- *NI No information. This is the most general and default null flavor.*
- *NA Not applicable. Known to have no proper value (e.g., last menstrual period for a male).*
- *UNK Unknown. A proper value is applicable, but is not known.*
- *ASKU Asked, but not known. Information was sought, but not found (e.g., the patient was asked but did not know).*

- *NAV Temporarily unavailable. The information is not available, but is expected to be available later.*
- *NASK Not asked. The patient was 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. There may be an alternate mechanism for gaining access to this information.*

This list contains those null values that are commonly used in clinical documents. For the full list and descriptions, see the nullFlavor vocabulary domain in the CDA Normative Edition.

Any SHALL conformance statement may use nullFlavor, unless the attribute is explicitly required or the nullFlavor is explicitly disallowed.

Following is an example of three conformance statements that demonstrate the usage of this method.

```
The XXX document

1. SHALL contain exactly one [1..1] Injury Incident Description with
   Template ID 1.3.6.1.4.1.19376.1.5.3.1.1.19.2.1 (CONF:XXX-5280)

2. SHALL contain exactly one [1..1] Mass Casualty Incident with
   Template ID 1.3.6.1.4.1.19376.1.5.3.1.1.25.2.3 (CONF:XXX-5295)

      a. SHALL NOT contain [0..0] nullFlavor (CONF:XXX-5301)
```

2.3.1.3 Extra Sections and Document Inheritance

Other data elements may be included in an instance of a content module over what is defined by the QRPH Technical Framework. Receivers are not required to process these elements, and if they do not understand them, must ignore them. Thus, it is not an error to include more than is asked for, but it is an error to reject a content module because it contains more than is defined by the framework. This allows value to be added to the content modules delivered in this framework, through extensions to it that are not defined or profiled by IHE. It further allows content modules to be defined later by IHE that are refinements or improvements over previous content modules.

For example, there is a Referral Summary content module defined in this framework. In later years an ED Referral content module can be created that inherits the constraints of the Referral Summary content module, with a few more use case specific constraints added. Systems that do not understand the ED Referral content module but do understand the Referral Summary content module will be able to interoperate with systems that send instances of documents that conform to the ED Referral content module. This interoperability, albeit at a reduced level of functionality, is by virtue of the fact that ED Referrals are simply a refinement of the Referral Summary.

In order to retain this capability, there are a few rules about how the QRPH Technical Committee creates constraints. Constraints that apply to any content module will always apply to any content modules that inherit from it. Thus, the "contracts" are always valid down the inheritance

hierarchy. Secondly, data elements of a content module will rarely be deprecated. This will usually occur only in the cases where they have been deprecated by the base standard. While any specific content module has a limited scope and set of use cases, deprecating the data element prevents any future content module from taking advantage of what has already been defined when a particular data element has been deprecated simply because it was not necessary in the original use case.

2.3.1.4 Document Content Module Constraints

Each document content module will define the appropriate codes used to classify the document, and will also describe the specific data elements that are included. The code used to classify it is specified using an external vocabulary, typically LOINC in the case of CDA Release 2.0 documents. The set of data elements that make up the document are defined, including the whether these data elements must, should or may be included in the document. Each data element is typically a section within the document, but may also describe information that is contained elsewhere within of the document (e.g., in the header). Each data element is mapped into a content module via a template identifier, and the document content module will further indicate whether these are data elements are required, required if known or optional. Thus, a document content module shall contain as constraints:

- The template identifier of the parent content module when there is one.
- The LOINC code or codes that shall be used to classify the document.
- A possibly empty set of required, required if known, and optional section content modules, and their template identifiers.
- A possibly empty set of required, required if known, and optional header content modules, and their template identifiers.
- Other constraints as necessary.

The template identifier for the document will be provided in the narrative, as will the legal LOINC document type codes and if present, any parent template identifier.

The remaining constraints are presented in two tables. The first table identifies the relevant data elements as determined during the technical analysis, and maps these data elements to one or more standards. The second table actually provides the constraints, wherein each data element identified in the first table is repeated, along with whether it is required, required if known, or optional. Following this column is a reference to the specification for the content module that encodes that data element, and the template identifier assigned to it. The simple example below completes the content specification described above. A simplified example is shown below.

Sample Document Specification SampleDocumentOID
Sample Document has one required section, and one entry that is required if known

6.3.1.A.4 Specification

Table 6.3.1.A.4.1-1

Data Element Name	Opt	Template ID
Sample Section Comment on section	R	SampleSectionOID
Sample Entry Comment on entry	R2	SampleEntryOID

6.3.1.B.5 Conformance

CDA Release 2.0 documents that conform to the requirements of this content module shall indicate their conformance by the inclusion of the appropriate <templateId> elements in the header of the document. This is shown in the sample document below.

```
<ClinicalDocument xmlns='urn:hl7-org:v3'>
  <typeId extension="POCD_HD000040" root="2.16.840.1.113883.1.3"/>
  <templateId root='SampleDocumentOID'/>
  <id root=' ' extension=' '/>
  <code code=' ' displayName=' '
    codeSystem='2.16.840.1.113883.6.1' codeSystemName='LOINC'/>
  <title>Sample Document</title>
  <effectiveTime value='20080601012005'/>
  <confidentialityCode code='N' displayName='Normal'
    codeSystem='2.16.840.1.113883.5.25' codeSystemName='Confidentiality' />
  <languageCode code='en-US'/>
     :
  <component><structuredBody>
    <component>
      <section>
        <templateId root='SampleSectionOID'/>
        <!-- Required Sample Section Section content -->
      </section>
    </component>

  </strucuredBody></component>
</ClinicalDocument>
```

2.3.1.5 Section Content Module Constraints

Section content modules will define the content of a section of a clinical document. Sections will usually contain narrative text, and so this definition will often describe the information present in the narrative, although sections may be wholly comprised of subsections.

Sections may contain various subsections, and these may be required, required if known or optional. Sections may also contain various entries, and again, these may be required, required if known, or optional. A section may not contain just entries; it must have at least some narrative text or subsections to be considered to be valid content.

Again, sections can inherit features from other section content modules. Once again, sections are classified using an external vocabulary (again typically this would be LOINC), and so the list of possible section codes is also specified. Sections that inherit from other sections will not specify a LOINC code unless it is to restrict the type of section to smaller set of LOINC codes specified by one of its ancestors.

Thus, a section content module will contain as constraints:

- The template identifier of the parent content module when there is one.
- The LOINC code or codes that shall be used to classify the section.
- A possibly empty set of required, required if known, and optional section content modules, and their template identifiers for the subsections of this section.
- A possibly empty set of required, required if known, and optional entry content modules, and their template identifiers.
- Other constraints as necessary.

These constraints are presented in this document using a table for each section content module, as shown below.

Sample Section		
	SampleSectionOID	
	foo (SampleParentOID)	
	Description of this section	
	Opt	**Description**
XXXXX-X	R	SECTION NAME
Entries	**Opt**	**Description**
OID	R	Sample Entry
Subsections	**Opt**	**Description**
OID	R	Sample Subsection

Table 0-1

Table 0-2

Table 0-3

Table 0-4 LOINC Codes

Table 0-5 General Description

Table 0-6 Parent Template

2.3.1.5.1 Parent Template

The parent of this template is foo.

```
<component>
  <section>
    <templateId root='SampleParentOID'/>
    <templateId root='SampleSectionOID'/>
    <id root=' ' extension=' '/>
    <code code=' ' displayName=' '
      codeSystem='2.16.840.1.113883.6.1' codeSystemName='LOINC'/>
    <text>
      Text as described above
    </text>
    <entry>
      Required and optional entries as described above
    </entry>

    <component>
      Required and optional subsections as described above
    </component>

  </section>
```

Table 2.3.1.5.1-1Template ID

2.3.1.6 Entry and Header Content Modules Constraints

Entry and Header content modules are the lowest level of content for which content modules are defined. These content modules are associated with classes from the HL7 Reference Information Model (RIM). These "RIM" content modules will constrain a single RIM class. Entry content modules typically constrain an "Act" class or one of its subtypes, while header content modules will normally constrain "Participation", "Role" or "Entity" classes, but may also constrain an "Act" class.

Entry and Header content modules will describe the required, required if known, and optional XML elements and attributes that are present in the CDA Release 2.0 instance. Header and Entry content modules may also be built up using other Header and Entry content modules. An entry or header content module may also specify constraints on the vocabularies used for codes found in the entry, or data types for the values found in the entry. Thus, an entry or header content module will contain as constraints:

- The template identifier of the parent content module when there is one.
- A description of the XML elements and attributes used in the entry, along with explanations of their meaning.
- An indication of those XML elements or attributes that are required, required if known, or optional.
- Vocabulary domains to use when coding the entry.
- Data types used to specify the value of the entry.
- Other constraints as necessary.

An example is shown below:

Sample Entry

Some text describing the entry.

```
<observation classCode='OBS' moodCode='EVN'>
   <templateId root='foo'/>
</observation>
```

2.3.1.7 <observation classCode='OBS' moodCode='EVN'>

Some details about the observation element

2.3.1.8 <templateId root='foo'/>

Some details about the template id element

3 IHE Transactions

This section defines each IHE transaction in detail, specifying the standards used, and the information transferred.

3.1 Cross Enterprise Document Content Transactions

At present, all transactions used by the QRPH Content Profiles appear in IHE ITI TF-2. General Options defined in content profiles for a Content Consumer are described below.

3.1.1 View Option

A Content Consumer that supports the View Option shall be able to:

1. Use the appropriate XD* transactions to obtain the document along with associated necessary metadata.
2. Render the document for viewing. This rendering shall meet the requirements defined for CDA Release 2 content presentation semantics (See Section 1.2.4 of the CDA Specification: Human readability and rendering CDA Documents). CDA Header information providing context critical information shall also be rendered in a human readable manner. This includes at a minimum the ability to render the document with the stylesheet specifications provided by the document source, if the document source provides a stylesheet. Content Consumers may optionally view the document with their own stylesheet, but must provide a mechanism to view using the source stylesheet.
3. Support traversal of links for documents that contain links to other documents managed within the sharing framework.
4. Print the document to paper.

3.1.2 Document Import Option

This Option requires that the View Option be supported. In addition, the Content Consumer that supports the Document Import Option shall be able to support the storage of the entire document (as provided by the sharing framework, along with sufficient metadata to ensure its later viewing) both for discharge summary or referral documents. This Option requires the proper tracking of the document origin. Once a document has been imported, the Content Consumer shall offer a means to view the document without the need to retrieve it again from the sharing framework. When viewed after it was imported, a Content Consumer may choose to access the sharing framework to find out if the related Document viewed has been deprecated, replaced or addended.

Note: For example, when using XDS, a Content Consumer may choose to query the Document Registry about a document previously imported in order to find out if this previously imported document may have been replaced or has received an addendum. This capability is offered to Content Consumers by this Integration Profile, but not required, as the events that may justify such a query are extremely implementation specific.

3.1.3 Section Import Option

This Option requires that the View Option be supported. In addition, the Content Consumer that supports the Section Import Option shall be able to support the import of one or more sections of the document (along with sufficient metadata to link the data to its source) both for discharge summary or referral. This Option requires the proper tracking of the document section origin. Once sections have been selected, a Content Consumer shall offer a means to copy the imported section(s) into local data structures as free text. This is to support the display of section level information for comparison or editing in workflows such as medication reconciliation while discrete data import is not possible. When viewed again after it is imported, a Content Consumer may choose to access the sharing framework to find out if the related information has been updated.

Note: For example, when using XDS, a Content Consumer may choose to query the Document Registry about a document whose sections were previously imported in order to find out if this previously imported document may have been replaced or has received an addendum. This capability is offered to Content Consumers by this Integration Profile, but not required, as the events that may justify such a query are extremely implementation specific.

This Option does not require, but does not exclude the Content Consumer from offering a means to select and import specific subsets of the narrative text of a section.

3.1.4 Discrete Data Import Option

This Option does not require that the View, Import Document or Section Import Options be supported. The Content Consumer that supports the Discrete Data Import Option shall be able to support the storage of the structured content of one or more sections of the document. This Option requires that the user be offered the possibility to select among the specific sections that include structured content a set of clinically relevant record entries (e.g., a problem or an allergy in a list) for import as part of the local patient record with the proper tracking of its origin.

Note: The Discrete Data Import Option does not require the support of the View, Import Document or Import Sections Options so that it could be used alone to support implementations of Content Consumers such as Public Health Data or Clinical Research systems that might aggregate and anonymize specific population healthcare information data as Document Consumer Actors, but one where no care provider actually views the medical summaries.

When discrete data is accessed after it was imported, a Content Consumer may choose to check if the document related to the discrete data viewed has been deprecated, replaced or addended.

A Content Consumer Actor grouped with the XDS Document Source Actor may query the Document Registry about a document from which discrete data was previously imported in order to find out if this previously imported document may have been replaced or has received an addendum. This capability is offered to Content Consumers by this Integration Profile, but not required, as the events that may justify such a query are extremely implementation specific.

4 IHE Quality, Research and Public Health Bindings

This section describes how the payload used in a transaction of an IHE profile is related to and/or constrains the data elements sent or received in those transactions. This section is where any specific dependencies between the content and transaction are defined.

A content integration profile can define multiple bindings. Each binding should identify the transactions and content to which it applies.

The source for all required and optional attributes have been defined in the bindings below. Three tables describe the three main XDS object types: XDSDocumentEntry, XDSSubmissionSet, and XDSFolder. XDSSubmissionSet and XDSDocumentEntry are required. Use of XDSFolder is optional. These concepts are universal to XDS, XDR and XDM.

The columns of the following tables are:

- **<XXX> attribute** – name of an XDS attribute, followed by any discussion of the binding detail.
- **Optional?** - Indicates the required status of the XDS attribute, and is one of R, R2, or O (optional). This column is filled with the values specified in the XDS Profile as a convenience.
- **Source Type** – Will contain one of the following values:

Source Type	Description
SA	Source document Attribute – value is copied directly from source document. The Source/Value column identifies where in the source document this attribute comes from. Specify the location in XPath when possible.
SAT	Source document Attribute with Transformation – value is copied from source document and transformed. The Source/Value column identifies where in the source document this attribute comes from. Specify the location in XPath when possible. Extended Discussion column must not be empty and the transform must be defined in the extended discussion
FM	Fixed (constant) by Mapping - for all source documents. Source/Value column contains the value to be used in all documents.
FAD	Fixed by Affinity Domain – value configured into Affinity Domain, all documents will use this value.
CAD	Coded in Affinity Domain – a list of acceptable codes are to be configured into Affinity Domain. The value for this attribute shall be taken from this list.
CADT	Coded in Affinity Domain with Transform - a list of acceptable codes are to be configured into Affinity Domain. The value for this attribute shall be taken from this list.
n/a	Not Applicable – may be used with an optionality R2 or O attribute to indicate it is not to be used.
DS	Document Source – value comes from the Document Source actor. Use Source/Value column or Extended Discussion to give details.
O	Other – Extended Discussion must be 'yes' and details given in an Extended Discussion.

- **Source/Value** – This column indicates the source or the value used.

The following tables are intended to be summaries of the mapping and transforms. The accompanying sections labeled 'Extended Discussion' are to contain the details as necessary.

4.1 Medical Document Binding to XDS, XDM and XDR

This binding defines a transformation that generates metadata for the XDSDocumentEntry element of appropriate transactions from the XDS, XDM and XDR profiles given a medical document and information from other sources. The medical document refers to the document being stored in a repository that will be referenced in the registry. The other sources of information include the configuration of the Document Source actor, the Affinity Domain, the site or facility, local agreements, other documents in the registry/repository, and this Content Profile.

In many cases, the CDA document is created for the purposes of sharing within an affinity domain. In these cases the context of the CDA and the context of the affinity domain are the same, in which case the following mappings shall apply.

In other cases, the CDA document may have been created for internal use, and are subsequently being shared. In these cases the context of the CDA document would not necessarily coincide with that of the affinity domain, and the mappings below would not necessarily apply.

Please note the specifics given in the table below.

4.1.1 XDSDocumentEntry Metadata

XDSDocumentEntry Attribute	Optional?	Source Type	Source/ Value
availabilityStatus	R	DS	
authorInstitution	R2	SAT	$inst <= /ClinicalDocument/author /assignedAuthor /representedOrganization The authorInstitution can be formatted using the following XPath expression, where **$inst** in the expression below represents the representedOrganization. concat($inst/name)
authorPerson	R2	SAT	$person <= /ClinicalDocument/author The author can be formatted using the following XPath expression, where **$person** in the expression below represents the author. concat($person/id/@extension,"^", $person/assignedPerson/name/family,"^", $person/assignedPerson/name/given[1],"^", $person/assignedPerson/name/given[2],"^", $person/assignedPerson/name/suffix,"^", $person/assignedPerson/name/prefix,"^", "^^^&", $person/id/@root,"&ISO")
authorRole	R2	SAT	This metadata element should be based on a mapping of the participation function defined in the CDA document to the set of author roles configured for the affinity domain. If the context of the CDA coincides with that of the affinity domain, then the following x-path may be appropriate: /ClinicalDocument/author/ participationFunction

XDSDocumentEntry Attribute	Optional?	Source Type	Source/ Value
authorSpecialty	R2	SAT	This metadata element should be based on a mapping of the code associated with the assignedAuthor to detailed defined classification system for healthcare providers such configured in the affinity domain. Possible classifications include those found in SNOMED-CT, or the HIPAA Healthcare Provider Taxonomy. If the context of the CDA coincides with that of the affinity domain, then the following x-path may be appropriate: /ClinicalDocument/author/ assignedAuthor/code
classCode	R	CADT	Derived from a mapping of /ClinicalDocument/code/@code to an Affinity Domain specified coded value to use and coding system. Affinity Domains are encouraged to use the appropriate value for Type of Service, based on the LOINC Type of Service (see Page 53 of the LOINC User's Manual). Must be consistent with /ClinicalDocument/code/@code
classCodeDisplayName	R	CADT	DisplayName of the classCode derived. Derived from a mapping of /ClinicalDocument/code/@code to the appropriate Display Name based on the Type of Service. Must be Consistent with /ClinicalDocument/code/@code
confidentialityCode	R	CADT	Derived from a mapping of /ClinicalDocument/confidentialityCode/@code to an Affinity Domain specified coded value and coding system. When using the BPPC profile, the confidentialityCode may also be obtained from the <authorization> element. /ClinicalDocument/ confidentialityCode/@code -AND/OR- /ClinicalDocument/authorization/ consent[templateId/@root= '1.3.6.1.4.1.19376.1.5.3.1.2.5'] /code/@code
comments	O	DS	
creationTime	R	SAT	/ClinicalDocument/effectiveTime Times specified in clinical documents may be specified with a precision in fractional sections, and may contain a time zone offset. In the XDS Metadata, it can be precise to the second, and is always given in UTC, so the time zone offset if present must be added to the current time to obtain the UTC time.
entryUUID	R	DS	
eventCodeList	O	CADT	These values express a collection of keywords that may be relevant to the consumer of the documents in the registry. They may come from anywhere in the CDA document, according to its purpose.
eventCodeDisplayNameList	R (if event Code is	CADT	These are the display names for the collection of keywords described above.

XDSDocumentEntry Attribute	Optional?	Source Type	Source/ Value
	valued)		
formatCode	R	FM	The format code for each QRPH Document content profile is provided within the document specifications.
healthcareFacilityTypeCode	R	CAD	A fixed value assigned to the Document Source and configured form a set of Affinity Domain defined values. Must be consistent with /clinicalDocument/code
healthcareFacility TypeCodeDisplay Name	R	CAD	Must be consistent with /clinicalDocument/code
intendedRecipient (for XDR, XDM)	O	SAT	$person <= /ClinicalDocument/intendedRecipient and/or $inst <= /ClinicalDocument/intendedRecipient/receivedOrganization The intendedRecipient can be formatted using the following XPath expression, where **$inst** in the expression below represents the receivedOrganization and where **$person** in the expression below represents the intendedRecipient. concat($person/id/@extension,"^", $person/informationRecipient/name/family,"^", $person/informationRecipient/name/given[1],"^", $person/informationRecipient/name/given[2],"^", $person/informationRecipient/name/suffix,"^", $person/informationRecipient/name/prefix,"^", "^^^&", $person/id/@root,"&ISO", "|" $inst/name) "^^^^^&", $inst/id/@root, "&ISO", "^^^^", $inst/id/@extension) -->
languageCode	R	SA	/ClinicalDocument/languageCode
legalAuthenticator	O	SAT	$person <= /ClinicalDocument/ legalAuthenticator The legalAuthenticator can be formatted using the following XPath expression, where **$person** in the expression below represents the legalAuthenticator. concat($person/id/@extension,"^", $person/assignedPerson/name/family,"^", $person/assignedPerson/name/given[1],"^", $person/assignedPerson/name/given[2],"^", $person/assignedPerson/name/suffix,"^", $person/assignedPerson/name/prefix,"^", "^^^&", $person/id/@root,"&ISO")
mimeType	R	FM	text/xml
parentDocumentRelationship	R (when applicable)	DS	Local document versions need not always be published, and so no exact mapping can be determined from the content of the CDA document. The parentDocumentRelationship may be determined in

XDSDocumentEntry Attribute	Optional?	Source Type	Source/ Value
			some configurations from the relatedDocument element present in the CDA document. If the context of the CDA coincides with that of the affinity domain, then the following x-path may be appropriate: /ClinicalDocument/relatedDocument/@typeCode
parentDocumentId	R (when parent Document Relationship is present)	DS	Local document versions need not always be published, and so no exact mapping can be determined from the content of the CDA document. The parentDocumentId may be determined in some configurations from the relatedDocument element present in the CDA document. If the context of the CDA coincides with that of the affinity domain, then the following x-path may be appropriate: $docID <= /ClinicalDocument/ relatedDocument/parentDocument/id The parentDocumentId can be formatted using the following XPath expression, where **$docID** in the expression below represents the identifier. concat($docID/@root,"^", $docID/@extension)
patientId	R	DS	The XDS Affinity Domain patient ID can be mapped from the patientRole/id element using transactions from the ITI PIX or PDQ profiles. See sourcePatientId below. If the context of the CDA coincides with that of the affinity domain, then the following x-path may be appropriate: $patID <= /ClinicalDocument/recordTarget/ patientRole/id
practiceSettingCode	R	CAD	This element should be based on a coarse classification system for the class of specialty practice. Recommend the use of the classification system for Practice Setting, such as that described by the Subject Matter Domain in LOINC.
practiceSettingCodeDisplayName	R	CAD	This element shall contain the display names associated with the codes described above.
serviceStartTime	R2	SAT	/ClinicalDocument/documentationOf/ serviceEvent/effectiveTime/low/ @value Times specified in clinical documents may be specified with a precision in fractional sections, and may contain a time zone offset. In the XDS Metadata, it can be precise to the second, and is always given in UTC, so the time zone offset if present must be added to the current time to obtain the UTC time.
serviceStopTime	R2	SAT	/ClinicalDocument/documentationOf/ serviceEvent/effectiveTime/high/ @value Times specified in clinical documents may be specified with a precision in fractional sections, and may contain a time zone offset. In the XDS Metadata, it can be precise to the second, and is always given in UTC, so the time zone

XDSDocumentEntry Attribute	Optional?	Source Type	Source/ Value
			offset if present must be added to the current time to obtain the UTC time.
sourcePatientId	R	SAT	$patID <= /ClinicalDocument/recordTarget/ patientRole/id The patientId can be formatted using the following XPath expression, where **$patID** in the expression below represents the appropriate identifier. concat($patID/@extension,"^^^&", $patID/@root, "&ISO")
sourcePatientInfo	R	SAT	/ClinicalDocument/recordTarget/ patientRole The sourcePatientInfo metadata element can be assembled from various components of the patientRole element in the clinical document.
title	O	SA	/ClinicalDocument/title
typeCode	R	CADT	/ClinicalDocument/code/@code The typeCode should be mapped from the ClinicalDocument/code element to a set of document type codes configured in the affinity domain. One suggested coding system to use for typeCode is LOINC, in which case the mapping step can be omitted.
typeCodeDisplay Name	R	CADT	/ClinicalDocument/code/@displayName
uniqueId	R	SAT	$docID <= /ClinicalDocument/id The uniqueId can be formatted using the following XPath expression, where **$docID** in the expression below represents the identifier. concat($docID/@root,"^", $docID/@extension)

4.1.1.1 XDSSubmissionSet Metadata

The submission set metadata is as defined for XDS, and is not necessarily affected by the content of the clinical document. Metadata values in an XDSSubmissionSet with names identical to those in the XDSDocumentEntry may be inherited from XDSDocumentEntry metadata, but this is left to affinity domain policy and/or application configuration.

4.1.1.2 Use of XDS Submission Set

This content format uses the XDS Submission Set to create a package of information to send from one provider to another. All documents referenced by the Medical Summary in this Package must be in the submission set.

4.1.1.3 Use of XDS Folders

No specific requirements identified.

4.1.1.4 Configuration

IHE Content Profiles using this binding require that Content Creators and Content Consumers be configurable with institution and other specific attributes or parameters. Implementers should be aware of these requirements to make such attributes easily configurable. There shall be a mechanism for the publishing and distribution of style sheets used to view clinical documents.

4.1.2 Extensions from other Domains

5 Namespaces and Vocabularies

This section lists the namespaces and identifiers defined or referenced by the IHE QRPH Technical Framework, and the vocabularies defined or referenced herein.

The following vocabularies are referenced in this document. An extensive list of registered vocabularies can be found at http://hl7.amg-hq.net/oid/frames.cfm.

codeSystem	codeSystemName	Description
1.3.6.1.4.1.19376.1.5.3.1	IHE PCC Template Identifiers	This is the root OID for all IHE PCC Templates. A list of PCC templates can be found below in CDA Release 2.0 Content Modules.
1.3.6.1.4.1.19376.1.5.3.2	IHEActCode	See IHEActCode Vocabulary below
1.3.6.1.4.1.19376.1.5.3.3	IHE PCC RoleCode	See IHERoleCode Vocabulary below
1.3.6.1.4.1.19376.1.5.3.4		Namespace OID used for IHE Extensions to CDA Release 2.0
2.16.840.1.113883.10.20.1	CCD Root OID	Root OID used for by ASTM/HL7 Continuity of Care Document
2.16.840.1.113883.5.112	RouteOfAdministration	See the HL7 RouteOfAdministration Vocabulary
2.16.840.1.113883.5.1063	SeverityObservation	See the HL7 SeverityObservation Vocabulary
2.16.840.1.113883.6.1	LOINC	Logical Observation Identifier Names and Codes
2.16.840.1.113883.6.96	SNOMED-CT	SNOMED Controlled Terminology
2.16.840.1.113883.6.103	ICD-9CM (diagnosis codes)	International Classification of Diseases, Clinical Modifiers, Version 9
2.16.840.1.113883.6.104	ICD-9CM (procedure codes)	International Classification of Diseases, Clinical Modifiers, Version 9
2.16.840.1.113883.6.26	MEDCIN	A classification system from MEDICOMP Systems.
2.16.840.1.113883.6.88	RxNorm	RxNorm
2.16.840.1.113883.6.63	FDDC	First DataBank Drug Codes
2.16.840.1.113883.6.12	C4	Current Procedure Terminology 4 (CPT-4) codes.
2.16.840.1.113883.6.257	Minimum Data Set for Long Term Care	The root OID for Minimum Data Set Answer Lists

5.1.1 IHE Format Codes

The table below lists the format codes, template identifiers and media types used by the IHE Profiles specified in the QRPH Technical Framework.
Note that the code system for these codes is **1.3.6.1.4.1.19376.1.2.3** as assigned by the ITI Domain for codes used for the purposes of cross-enterprise document sharing (XDS). For more information see XDS Coding System (1.3.6.1.4.1.19376.1.2.3).

Profile	Format Code	Media Type	Template ID

Profile	Format Code	Media Type	Template ID

5.1.2 IHEActCode Vocabulary

CCD ASTM/HL7 Continuity of Care Document

CCR ASTM CCR Implementation Guide

The IHEActCode vocabulary is a small vocabulary of clinical acts that are not presently supported by the HL7 ActCode vocabulary. The root namespace (OID) for this vocabulary is 1.3.5.1.4.1.19376.1.5.3.2. These vocabulary terms are based on the vocabulary and concepts used in the CCR and CCD standards listed above.

Code	Description
COMMENT	This is the act of commenting on another act.
PINSTRUCT	This is the act of providing instructions to a patient regarding the use of medication.
FINSTRUCT	This is the act of providing instructions to the supplier regarding the fulfillment of the medication order.
IMMUNIZ	The act of immunization of a patient using a particular substance or class of substances identified using a specified vocabulary. Use of this vocabulary term requires the use of either the SUBSTANCE or SUBSTCLASS qualifier described below, along with an identified substance or class of substances.
DRUG	The act of treating a patient with a particular substance or class of substances identified using a specified vocabulary. Use of this vocabulary term requires the use of either the SUBSTANCE or SUBSTCLASS qualifier described below, along with an identified substance or class of substances.
INTOL	An observation that a patient is somehow intolerant of (e.g., allergic to) a particular substance or class of substances using a specified vocabulary. Use of this vocabulary term requires the use of either the SUBSTANCE or SUBSTCLASS qualifier described below, along with an identified substance or class of substances.
SUBSTANCE	A qualifier that identifies the substance used to treat a patient in an immunization or drug treatment act. The substance is expected to be identified using a vocabulary such as RxNORM, SNOMED CT or other similar vocabulary and should be specific enough to identify the ingredients of the substance used.
SUBSTCLASS	A qualifier that identifies the class of substance used to treat a patient in an immunization or drug treatment act. The class of substances is expected to be identified using a vocabulary such as NDF-RT, SNOMED CT or other similar vocabulary, and should be broad enough to classify substances by mechanism of action (e.g., Beta Blocker), intended effect (Diuretic, antibiotic) or ...

5.1.3 IHERoleCode Vocabulary

The IHERoleCode vocabulary is a small vocabulary of role codes that are not presently supported by the HL7 Role Code vocabulary. The root namespace (OID) for this vocabulary is 1.3.5.1.4.1.19376.1.5.3.3.

Code	Description
EMPLOYER	The employer of a person.

Code	Description
SCHOOL	The school in which a person is enrolled.
AFFILIATED	An organization with which a person is affiliated (e.g., a volunteer organization).
PHARMACY	The pharmacy a person uses.

6 Content Modules

6.1 Conventions

Various tables used in this section will further constrain the content. Within this volume, the follow conventions are used.

R

A "Required" data element is one that shall always be provided. If there is information available, the data element must be present. If there is no information available, or it cannot be transmitted, the data element must contain a value indicating the reason for omission of the data. (See IHE PCC TF-2:5.3.4.2 for a list of appropriate statements).

R2

A "Required if data present" data element is one that shall be provided when a value exists. If the information cannot be transmitted, the data element shall contain a value indicating the reason for omission of the data. If no such information is available to the creator or if such information is not available in a well identified manner (e.g., buried in a free form narrative that contains additional information relevant to other sections) or if the creator requires that information be absent, the R2 section shall be entirely absent. (See section IHE PCC TF-2:5.3.4.2 for a list of appropriate statements).

O

An optional data element is one that may be provided, irrespective of whether the information is available or not. If the implementation elects to support this optional section, then its support shall meet the requirement set forth for the "Required if data present" or R2.

C

A conditional data element is one that is required, required if known or optional depending upon other conditions. These will have further notes explaining when the data element is required, et cetera.

Note: The definitions of R, R2, and O differ slightly from other IHE profiles. This is due in part to the fact that local regulations and policies may in fact prohibit the transmission of certain information, and that a human decision to transmit the information may be required in many cases.

6.2 Folder Content Modules

This section contains modules that describe the content requirements of Folders used with XDS, XDM or XDR. When workflows are completed normally, the folders will contain documents with the optionality specified in the tables shown below. Under certain circumstances, the folders

will not meet the optionality requirements described below, for example, when the patient leaves before treatment is completed.

6.3 HL7 Version 3.0 Content Modules

This section contains content modules based upon the HL7 CDA Release 2.0 Standard, and related standards and/or implementation guides.

6.3.1 CDA Document Content Modules

6.3.1.1 Medical Documents Specification 1.3.6.1.4.1.19376.1.5.3.1.1.1

Please see IHE PCC TF 2:6.3.1.1

6.3.1.2 Medical Summary Specification 1.3.6.1.4.1.19376.1.5.3.1.1.2

Please see IHE PCC TF 2:6.3.1.2

6.3.2 CDA Header Content Modules

Please see IHE PCC TF 2:6.3.2

6.3.3 CDA Section Content Modules

Please see IHE PCC TF 2:6.3.3

6.3.4 CDA Entry Content Modules

Please see IHE PCC TF 2:6.3.4

6.4 HL7 Version 2.0 Content Modules For care management

6.5 Value Sets

Appendix A Examples Using QRPH Content Profiles

Appendix B Validating CDA Documents using the Framework

Many of the constraints specified by the content modules defined in the QRPH Technical Framework can be validated automatically by software. Automated validation is a very desirable capability, as it makes it easier for implementers to test the correctness of their implementations. With regard to validation of the content module, the QRPH Technical Framework narrative is the authoritative specification, not any automated software tool. Having said that, it is still very easy to create a validation framework for the IHE QRPH Technical Framework using an XML validation tool such as Schematron. Since each content module has a name (the template identifier), any XML instance that reports itself to be of that "class" can be validated by creating assertions that must be true for each constraint indicated for the content module. In the XML representation, the <templateId> element is a child of the element that is claiming conformance

to the template named. Thus the general pattern of a Schematron that validates a specific template is shown below:

```
<schema xmlns="http://www.ascc.net/xml/schematron" xmlns:cda="urn:hl7-org:v3">
  <ns prefix="cda" uri="urn:hl7-org:v3" />
  <pattern name='ReferralSummary'>
    <rule context='*[cda:templateId/@root="1.3.6.1.4.1.19376.1.5.3.1.1.3]"'>
      <!-- one or more assertions made by the content module -->
    </rule>
  </pattern>
</schema>
```

B.1 Validating Documents

For document content modules, the pattern can be extended to support common document content module constraints as shown below:

```
<schema xmlns="http://www.ascc.net/xml/schematron" xmlns:cda="urn:hl7-org:v3">
  <ns prefix="cda" uri="urn:hl7-org:v3" />
  <pattern name='ReferralSummary'>
    <rule context='*[templateId/@root="1.3.6.1.4.1.19376.1.5.3.1.1.3]"'>
      <!-- Verify that the template id is used on the appropriate type of object -->
      <assert test='../ClinicalDocument'>
        Error: The referral content module can only be used on Clinical Documents.
      </assert>
      <!-- Verify that the parent templateId is also present. -->
      <assert test='templateId[@root="1.3.6.1.4.1.19376.1.5.3.1.1.2"]'>
        Error: The parent template identifier for medical summary is not present.
      </assert>
      <!-- Verify the document type code -->
      <assert test='code[@code = "34133-9"]'>
        Error: The document type code of a referral summary must be
        34133-9 SUMMARIZATION OF EPISODE NOTE.
      </assert>
      <assert test='code[@codeSystem = "2.16.840.1.113883.6.1"]'>
        Error: The document type code must come from the LOINC code
        system (2.16.840.1.113883.6.1).
      </assert>
      <!-- Verify that all required data elements are present -->
      <assert test='.//templateId[@root = "1.3.6.1.4.1.19376.1.5.3.1.3.1"]'>
        Error: A referral summary must contain a reason for referral.
      </assert>
      <!-- Alert on any missing required if known elements -->
      <assert test='.//templateId[@root = "1.3.6.1.4.1.19376.1.5.3.1.3.8"]'>
        Warning: A referral summary should contain a list of history of past illnesses.
      </assert>
      <!-- Note any missing optional elements -->
      <assert test='.//templateId[@root = "1.3.6.1.4.1.19376.1.5.3.1.3.18"]'>
        Note: This referral summary does not contain the pertinent review of systems.
      </assert>
    </rule>
  </pattern>
</schema>
```


B.2 Validating Sections

The same pattern can be also applied to sections with just a few minor alterations.

```
<schema xmlns="http://www.ascc.net/xml/schematron" xmlns:cda="urn:hl7-org:v3">
  <ns prefix="cda" uri="urn:hl7-org:v3" />
  <pattern name='ReasonForReferralUncoded'>
    <rule context='*[templateId/@root="1.3.6.1.4.1.19376.1.5.3.1.3.1"]'>
      <!-- Verify that the template id is used on the appropriate type of object -->
      <assert test='section'>
        Error: The coded reason for referral module can only be used on a section.
      </assert>
      <assert test='false'>
        Manual: Manually verify that this section contains narrative providing the
        reason for referral.
      </assert>
      <!-- Verify that the parent templateId is also present. -->
      <assert test='templateId[@root="1.3.6.1.4.1.19376.1.5.3.1.3.1"]'>
        Error: The parent template identifier for the reason for referral
        module is not present.
      </assert>
      <!-- Verify the section type code -->
      <assert test='code[@code = "42349-1"]'>
        Error: The section type code of the reason for referral section must be 42349-1
        REASON FOR REFERRAL.
      </assert>
      <assert test='code[@codeSystem = "2.16.840.1.113883.6.1"]'>
        Error: The section type code must come from the LOINC code
        system (2.16.840.1.113883.6.1).
      </assert>
  </pattern>
  <pattern name='ReasonForReferralCoded'>
    <rule context='*[templateId/@root="1.3.6.1.4.1.19376.1.5.3.1.3.2"]'>
      <!-- The parent template will have already verified the type of object -->
      <!-- Verify that the parent templateId is also present. -->
      <assert test='templateId[@root="1.3.6.1.4.1.19376.1.5.3.1.3.1"]'>
        Error: The parent template identifier for the reason for referral
        module is not present.
      </assert>
      <!-- Don't bother with the section type code, as the parent template caught it -->
      <!-- Verify that all required data elements are present -->
      <assert test='.//templateId[@root = "1.3.6.1.4.1.19376.1.5.3.1.4.13"]'>
        Error: A coded reason for referral section must contain an simple observation.
      </assert>
      <!-- Alert on any missing required if known elements -->
      <!-- Note any missing optional elements -->
    </rule>
  </pattern>
</schema>
```

A similar pattern can also be followed for Entry and Header content modules, and these are left as an exercise for the reader.

B.3 Phases of Validation and Types of Errors

Note that each message in the Schematrons shown above start with a simple text string that indicates whether the message indicates one of the following conditions:

- An error, e.g., the failure to transmit a required element,
- A warning, e.g., the failure to transmit a required if known element,
- A note, e.g., the failure to transmit an optional element.
- A manual test, e.g., a reminder to manually verify some piece of content.

Schematron supports the capability to group sets of rules into phases by the pattern name, and to specify which phases of validation should be run during processing. To take advantage of this capability, one simply breaks each <pattern> element above up into separate patterns depending upon whether the assertion indicates an error, warning, note or manual test, and then associate each pattern with a different phase. This is shown in the figure below.


```
<schema xmlns="http://www.ascc.net/xml/schematron" xmlns:cda="urn:hl7-org:v3">
  <ns prefix="cda" uri="urn:hl7-org:v3" />
  <phase id="errors">
    <active pattern="ReasonForReferralUncoded_Errors"/>
    <active pattern="ReasonForReferralCoded_Errors"/>
  </phase>
  <phase id="manual">
    <active pattern="ReasonForReferralUncoded_Manual"/>
  </phase>
  <pattern name='ReasonForReferralUncoded_Errors'>
    <rule context='*[templateId/@root="1.3.6.1.4.1.19376.1.5.3.1.3.1"]'>
      <assert test='section'>
        Error: The coded reason for referral module can only be used on a section.
      </assert>
      <assert test='code[@code = "42349-1"]'>
        Error: The section type code of the reason for referral section must be 42349-1
        REASON FOR REFERRAL.
      </assert>
      <assert test='code[@codeSystem = "2.16.840.1.113883.6.1"]'>
        Error: The section type code must come from the LOINC code
        system (2.16.840.1.113883.6.1).
      </assert>
    </rule>
  </pattern>
  <pattern name='ReasonForReferralUncoded_Manual'>
    <rule context='*[templateId/@root="1.3.6.1.4.1.19376.1.5.3.1.3.1"]'>
      <assert test='false'>
        Manual: Manually verify that this section contains narrative providing the
        reason for referral.
      </assert>
  </pattern>
  <pattern name='ReasonForReferralCoded_Errors'>
    <rule context='*[templateId/@root="1.3.6.1.4.1.19376.1.5.3.1.3.2"]'>
      <assert test='templateId[@root="1.3.6.1.4.1.19376.1.5.3.1.3.1"]'>
        Error: The parent template identifier for the reason for referral not present.
      </assert>
      <assert test='.//templateId[@root = "1.3.6.1.4.1.19376.1.5.3.1.4.13"]'>
        Error: A coded reason for referral section must contain an simple observation.
      </assert>
    </rule>
  </pattern>
</schema>
```

Using these simple "templates" for template validation one can simply create a collection of Schematron patterns that can be used to validate the content modules in the QRPH Technical Framework. Such Schematrons are expected to be made available as part of the MESA test tools that are provided to IHE Connectathon participants, and which will also be made available to the general public after Connectathon.

Appendix C Extensions to CDA Release 2.0

This material is defined in IHE PCC TF 2: Appendix “Extensions to CDA Release 2.0”

