|  |
| --- |
| CIMI Modeling Architecture, Methodology & Style Guide |
| Version 0.04 |
| **Published Date:**  **Authors:** HL7 CIMI Work Group |

Background 5

CIMI and FHIR 5

Scope of Work 6

Future work 7

Request for Comments 7

CIMI Model Architecture 8

Introduction 8

Core Modelling Principles 8

The CIMI Model’s Alignment to Terminology 9

A High Level View of the CIMI Logical Model 10

The CIMI Reference Model 10

The CIMI Archetype Hierarchies 11

Reference Model 12

The CIMI Core Reference Model Module 13

CIMI primitives 13

CIMI Complex Types 14

The LOCATABLE and ASSOCIATION\_CLASS Root Types 16

The CIMI Foundational Reference Model Module 16

The CIMI Clinical Reference Model Module 18

The CIMI Party, Participation Patterns 19

The CIMI Clinical Statement Pattern 19

The CIMI Attribution/Provenance patterns 24

The CIMI Assertion and EvaluationResult Pattern 26

The CIMI Procedure pattern 28

CIMI Data Structures 29

An Introduction to CIMI archetypes 34

The CIMI Style Guide 34

Modelling Layers 34

1. Modelling Patterns 35

5.1. Cluster Patterns 35

5.2. Entry Patterns 35

5.3. Section Patterns 35

Evaluations 35

Evaluation Subtypes 36

Assertions 36

Assertion Subtypes 37

Procedures 39

2. Composition Patterns 40

2. Modelling Style 41

6.1. Quality Criteria 41

6.2. Scope of Clinical Models 41

6.3. Granularity of Clinical Models 42

6.4. Consistency and Reuse 42

6.5. Isosemantic Models 42

6.6. Terminology Binding 44

6.6.1 Terminology Binding Guidelines 45

Attribute Binding 45

Value Binding 46

6.6.4 Linkage Relationships 47

6.6.5 Linkage Relationships 47

6.7. CIMI Alignment with FHIR 47

5. Modelling Methodology 48

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Amendment History |
| 0.1 | 2012-09-13 | Linda Bird | First draft for comment |
| 0.2 | 2012-10-26 | Rahil Qamar Siddiqui | Revised section 6.6 on Terminology Binding |
|  | Added sub-sections 6.6.1 to 6.6.4 with preliminary details on the main types of terminology bindings within scope for CIMI. More details to be added in due course. |
| .03 | 2016-11-16 | Susan Matney | Draft for ballot comment |
| .04 | 2016-11-23 | Claude Nanjo | Integration of architectural sections |

# Background

The Clinical Information Modelling Initiative (CIMI) is a Health Level Seven (HL7) workgroup dedicated to providing a common format for the representation of health information content so that semantically interoperable information may be created and shared in health records, messages and documents.

In achieving this goal, CIMI has established a CIMI modelling methodology, style guide and a set of models, which together demonstrate and test the approach to CIMI clinical modelling.

CIMI modelling is the overall strategy for specifying, at a granular level and in a computable fashion, the structure and semantics of each data element that will be stored in an Electronic Health Record (EHR). More exactly, the CIMI Model specifies how to create a “model” of a particular type of data (e.g., a heart rate measurement, a laboratory observation, or a procedure performed), where the model declares how a valid “instance” of that type of data should be structured, and the semantics of that structure. By analogy, a blueprint is used for building cars. All the cars built as a result of the blueprint conform to the specifications of the blueprint. The cars conforming to the blueprints are analogous to instances of data (e.g., John Smith’s heart rate measurement at 11 am on January 26th, the observation of Mary Black’s blood glucose on March 20th, or the procedure performed on Jeff Brown June 23rd at 10 am), while the blueprints are analogous to the models (e.g., a model for heart rate measurement instances, a model for laboratory observation instances, or a model for procedure performed instances).

The purpose of this document is to give modellers practical information needed to author CIMI models. This document describes some initial proposals by the CIMI modelling taskforce that will help to inform CIMI’s model architecture, modelling methodology and style guide.

# CIMI and FHIR

The CIMI logical model and FHIR are complementary models. Unlike FHIR, CIMI does not provide a specification for the representation of instances of clinical data. It relies on physical models such as FHIR to do so. On the other hand, the CIMI logical model offers a formal specification for a set of consistent FHIR logical, resource, and extension profiles that can enable both the interoperable exchange of clinical information and the queries, analysis, and computations performed upon such information. Services whose APIs make use of these FHIR profiles will be able to leverage the expressivity and computability of the CIMI logical model within health applications. In the area of Clinical Decision Support, the CIMI logical model offers a stable, consistent, and computable clinical model for the development of knowledge artefacts and for the design of execution systems that make use of these models in clinical logic.

In order to achieve this, the CIMI preferred models must first be transformed into FHIR resource profiles. The CIMI logical model provides the underlying specification for such profiles through the translation of both reference model structures and archetype constraints into their equivalent representations in corresponding FHIR structure definitions. Once the set of CIMI-FHIR profiles has been generated, FHIR instances conformant with these CIMI profiles represent the set of conformant CIMI instances.

Given their differing requirements, CIMI models and FHIR resources may differ in their structure and therefore transformation costs are inevitable. For instance, while CIMI clinical statements are compositional structures to support model reuse and consistency, FHIR resources tend to be flatter. However, in order to minimize these transformation costs, CIMI shall aim to (1) align the granularity of its models with those specified by FHIR, (2) to provide declarative and computable transformations from all of its models to FHIR profiles and (3) work with the FHIR team in order to address specific areas of incongruence.

At a high level, the transformation of CIMI models into FHIR profile can be achieved as follows:

1. CIMI archetypes must first be aligned with the corresponding FHIR resources
2. Reference model attributes are mapped to attributes in the mapped FHIR resource(s)
3. CIMI attributes that have no equivalent in FHIR are handled as FHIR extensions
4. ADL model constraints are then translated into their corresponding representations in FHIR structure definitions

Archetypes in CIMI are defined hierarchically with more specific archetypes further tightening the constraints of their ancestors in the hierarchy. In order to retain this structure in FHIR, the translation of CIMI archetype hierarchies into FHIR resource profiles will most likely result in the generation of layered resource profiles in FHIR that mimic the structure found in CIMI.

# Scope of Work

The motivation for this for-comment ballot submission is to solicit feedback on the core architecture of the CIMI logical model. Given the scope of this effort, the CIMI Team has focused primarily on the following areas:

* The expansion and modularization of the CIMI Reference Model based on an updated set of core modelling principles
* The definition of top-level archetypes based on the new CIMI Reference Model
* The development of core modelling patterns including:
  + The compositional Clinical Statement Pattern
  + The Statement Topic/Context Patterns
  + The Attribution/Provenance Pattern
  + The Party/Participation Pattern
  + The Assertion and EvaluationResult Patterns
  + The Procedure Pattern
* The alignment of the CIMI logical model with the SNOMED CT Concept Model
* The implementation of the Skin/Wound Assessment archetypes to illustrate the use of the above patterns in an important clinical use case.

While this submission describes additional models including a number of supporting structures, many of these structures have not received adequate internal review at this time. Therefore, we encourage the reader to prioritize review on any of the items listed above though we certainly welcome feedback on any part of the model.

It is also important to note that while some value set bindings will be provided as part of this submission, much work still remains to be done. Terminology alignment will be addressed in future ballot submissions and all terminology bindings, apart from those specifying the alignment of model attributes to SNOMED CT Concept Model attributes, are provided as examples.

# Future work

While great effort has gone into this phase of the CIMI model development and in the preparation of the January 2017 ballot material, the CIMI Working Group recognizes that much still remains to be done. In the months following the January HL7 Working Group Meeting, the CIMI Team intends to address community comments and refactor the model accordingly. By the May 2017 HL7 Working Group Meeting, our goal is to complete the CIMI Reference Model, CIMI top-level archetypes including archetypes that capture US Core and QI Core requirements, and ballot the model as an Informative Specification. By the September 2017 HL7 Working Group Meeting, we intend to complete the terminology bindings for all proposed archetypes, implement detailed clinical models for selected use cases, and specify formal declarative mappings from CIMI to FHIR using the FHIR Mapping Language. The model will then be balloted as a Standard for Trial Use.

# Request for Comments

We encourage the community to comment on any aspect of the proposed model. In particular, we would like to solicit comments and feedback in the following areas:

* Gaps, corrections, or enhancements to the proposed reference model classes and archetypes including any proposed terminology bindings
* The Clinical Statement Pattern including approaches to model negation through the use of the Statement Context Pattern
* The Provenance patterns
* The EvaluationResult/Assertion patterns
* The Procedure Pattern
* The proposed alignment with the SNOMED CT Concept Model, in particular, alignment with the Situation with Explicit Context Concept Model, the Observable model, the Clinical Finding Concept Model, and the Procedure Concept Model
* Alignment of the CIMI model with FHIR and with the Federal Health Information Model (FHIM)
* The CIMI Style Guide
* The CIMI Modeling Principles

# CIMI Model Architecture

## Introduction

CIMI Clinical models are defined as a set of constraints on the CIMI reference model. This reference model is the ‘common language’ used to describe all clinical models, and is also used to represent example instances of these models. CIMI Clinical models are defined using a constraint language whose semantics are defined by the Archetype Object Model (v1.5). These constraint semantics can be represented using either the Archetype Definition Language (ADL) (a direct serialisation of the AOM), or Archetype Modelling Language (AML) (a profile of the Unified Modelling Language’s class diagrams). CIMI Clinical models will usually include both semantic bindings (i.e. bindings to terminology expressions that define the meaning of the relevant parts of the model) and value bindings (i.e. bindings to terminology reference sets that define the valid values that may populate appropriate parts of the model). For this ballot cycle, we are only considering semantic bindings to the SNOMED CT Concept model.

It is anticipated that CIMI Clinical models will be both specialised and extended to develop realm-specific clinical models, which will then be further transformed into implementation-specific artefacts.

## Core Modelling Principles

The following principles guide CIMI’s modelling approach:

1. CIMI favors a *design-by-specialization* over a *design-by-constraint* approach. This approach can be summarized as follows: if a class has a number of specializations, each requiring a different set of attributes, common attributes are represented in the parent class while children attributes are added to the appropriate specializations. An alternative approach may be to include the union of all attributes in a single class and constrain attributes out at the archetype level. The former approach is preferred over the latter except in certain cases. For instance, if a set of attributes is shared by some though not all specializations, they may *move up* to the parent class and be appropriately constrained out in archetypes. If a specialization differs from its parent by a single attribute, the inclusion of the attribute in the parent class *may* be preferred over the creation of a new class.
2. CIMI generally favors the definition of explicit attributes in the reference model over the *slicing* of lists in archetype definitions. The *attribute subset pattern* is achieved by defining a multi-cardinality attribute in the reference model and specifying subsets of the list elements in archetypes. For instance, one may specify that the LOCATABLE class, the supertype of all CIMI classes, has an attribute called *participation* of type PARTICIPATION and whose cardinality is 0..\*. In an archetype, one may then constrain the *participation* attribute in the following manner. The first element of the list represents the author. The second element represents the data enterer. The third element represents the location where the authoring activity took place. The fourth element of the list represents the system where the information was recorded. While such subsets are allowed in both UML and ADL, CIMI generally avoids their use and favors the explicit representations of such subsets as full-fledged attributes in the model. For instance, CIMI explicitly adds an attribute for the agent of an activity, the location of an activity, the entity involved in the performance of the activity, and so on. The motivation for this approach stems from the fact that CIMI is a logical model rather than a physical model and favors greater reference model expressivity over physical patterns that enable better economies of structure.
3. CIMI may offer a number of variants for a given attribute. For instance, CIMI defines bodyLocation: AnatomicalLocation and bodyLocationPrecoord: CODED\_TEXT in order to support both a coded and a post-coordinated anatomical location. Similarly Assertion.dueTo:CODED\_TEXT and Assertion.dueTo: ClinicalStatement allow users to link an assertion to another clinical statement or simply to a code.

## The CIMI Model’s Alignment to Terminology

Information models are often developed independently of clinical ontologies. As a result, many information models align poorly with the terminologies or ontologies upon which they ultimately depend for their formal semantics. Moreover, by not explicitly specifying the model’s semantics, the meaning of the model is left open for interpretation during implementation further hindering interoperability. In an effort to better align *models of use* with *models of meaning*, the Clinical Information Model is designed to align closely with the SNOMED CT Concept Model wherever such an overlap exists. In CIMI, the model’s formal semantics are specified through terminology bindings defined at the archetype level. These terminology bindings occur at three levels:

1. *At the attribute level* – CIMI model attributes are aligned with their corresponding SNOMED CT concept model attributes when such a correspondence exists. [Provide example]
2. *At the attribute range level ­*– CIMI model attribute ranges are aligned with the ranges specified for their corresponding SNOMED CT concept model attribute when such a correspondence exists. [Expand on previous example]
3. *At the archetype level* – In some cases, CIMI archetypes may be associated with SNOMED CT expressions, provided that the expression conforms to the constraints specified for the flattened archetype. For instance, a CIMI Clinical Statement may be associated with a SNOMED CT Situation with Explicit Context expression or a pre-coordinated code. We are currently investigating the use of SNOMED CT Templates for such bindings.

## A High Level View of the CIMI Logical Model

The CIMI logical model consists of structural patterns and the constraints applied on those patterns:

1. The ***CIMI Reference Model*** specifies the classes, attributes, and allowed relationships that define the model’s primitive types, complex types, data structures, and the clinical patterns built upon them.
2. The ***CIMI Archetype Library*** is composed of archetype hierarchies that progressively constrain the patterns defined in the reference model and whose leaf-level archetypes ultimately form the *Detained Clinical Model* (DCM) layer of the CIMI model. A DCM consists of the structural patterns and corresponding archetype constraints that sufficiently define a shareable and interoperable unit of information.

The CIMI model is persisted in a format that conforms to two OpenEHR specifications:

1. The ***Basic Meta Modeling (BMM)*** language is used to define and persist the CIMI Reference Model[[1]](#endnote-1).
2. The ***Archetype Definition Language (ADL2)*** is used to specify persistent and computable constraints on the reference model[[2]](#endnote-2).

The CIMI model follows a strict rule in the usage of the abovementioned two OpenEHR specifications in order to cleanly delineate the boundary between the reference model and the archetype hierarchies. *The basic meta-modeling language shall be used to specify the classes, attributes, and relationships that make up the model. The archetype definition language shall be used to define the constraints on the reference model* ***but shall not be used to define new model classes and attributes.*** In other words, the CIMI model specifies classes and attributes explicitly in the reference model and does not offer a way to extend the model within archetypes.

This approach represents a departure from the approach taken by OpenEHR, which allows the definition of classes and attributes within archetypes using the meta model constructs of ITEM, CLUSTER, and ELEMENT and by FHIR whose extension mechanism allows for the definition of new attributes and structures within resource profiles. The motivation for this decision is that, unlike FHIR and the OpenEHR models, both physical models, CIMI is a logical model and thus does not rely on extensions to provide for additional expressivity beyond that provided by the CIMI Reference Model. The CIMI Logical Model favors expressivity over economy of structure and delegates model extensibility to its underlying physical model target. For instance, attributes that exist in the CIMI Reference Model but that do not exist in the FHIR Core Model shall be mapped to the appropriate FHIR extensions.

### The CIMI Reference Model

The CIMI Reference Model is a layered model that is designed to be modular and currently consists of three layers:

1. The Core CIMI Reference Model defines the model’s primitives, its core types, and two root classes, LOCATABLE, from which the majority of CIMI classes derive and ASSOCIATION\_CLASS from which CIMI association classes derive.
2. The CIMI Foundational Reference Model defines the foundational underpinnings of the CIMI model. This structure aligns with the ISO 13606 EHR and the OpenEHR Reference Models. It also defines a number of top-level hierarchies: CLUSTER, COMPOSITION, CONTENT, PARTY, ACTOR, ROLE, PARTICIPATION, and PARTY\_RELATIONSHIP. It is from these hierarchies that all downstream CIMI classes and clinical patterns are derived.
3. The CIMI Clinical Reference Model builds upon these two lower layers to provide the structural patterns upon which CIMI preferred archetypes are built.

This modular approach allows for additional, more domain-specific layers to be added in future or for alternate iso-semantic patterns to be introduced at the appropriate level in the model. Over time, it can be expected that the lower level reference model modules will become more stable while higher-level modules may still undergo additional flux.

#### What is a ‘reference model pattern’ or ‘clinical pattern’

A *reference model pattern* is a structural pattern (a single class or group of related classes) that can be constrained by archetypes in order to define a family of related and consistent detailed clinical models. The set of allowable clinical patterns comprise the CIMI Reference Model.

### The CIMI Archetype Hierarchies

The CIMI archetype hierarchies form the second part of the CIMI model. These hierarchies serve two primary purposes:

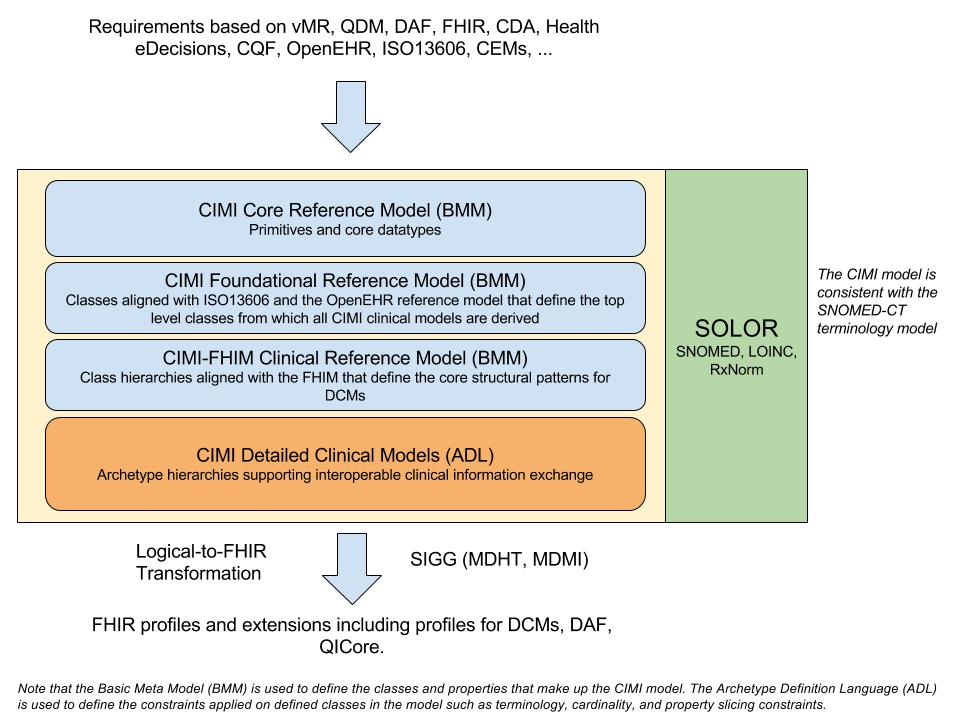
1. They enable the progressive application of constraints on reference clinical patterns including the specification of terminology constraints that assign formal meanings to both model attributes and their range.
2. They allow for the definition of sets of models whose members vary solely based on the constraints they apply to a common underlying reference model pattern.

Archetypes can specialize more general archetypes in ADL. They do so by progressively constraining the underlying reference model pattern in a manner that is consistent with and not contradictory to the constraints specified in archetypes higher up in the hierarchy.

Examples of constraint refinements are listed below:

1. A top-level archetype restricts the range of Ingredient.substanceCode to the set of all concepts subsumed by the SNOMED CT concept ‘Pharmaceutical/biologic product’. A downstream specialization of this archetype restricts the Ingredient.substanceCode to ‘Metoprolol’.
2. A top-level archetype assigns the SNOMED CT concept ‘Procedure site (attribute)’ as the semantic binding of the attribute Procedure.site. A downstream specialization of this archetype further constrains this meaning to ‘Procedure site – Direct (attribute)’, a concept subsumed by the ‘Procedure site (attribute)’.
3. A downstream archetype refines the cardinality of a container attribute from 0..\* to 2..5.
4. A downstream archetype *constrains out* a class attribute by setting its existence to 0..0.
5. A downstream archetype constrains the datatype of an attribute from DATA\_VALUE to QUANTITY.

Detailed Clinical Models (DCMs) typically reside at the bottom of CIMI archetype hierarchies. The cumulative constraints applied on a DCM are intended to be precise enough to allow for the unambiguous exchange of interoperable clinical information and thus constitute highly specific constraints on the underlying reference model pattern. This layered approach is illustrated below:



*Figure 1: CIMI Architectural Framework*

# Reference Model

The CIMI reference model defines the core structure and data types of all CIMI clinical models. Each reference model module is further described in the sections below.

## The CIMI Core Reference Model Module

The CIMI Core Reference model module introduces four core reference model hierarchies:

1. The *Any* hierarchy from which all CIMI primitive types are derived.
2. The DATA\_VALUE hierarchy from which all CIMI complex types are derived.
3. The LOCATABLE hierarchy from which all other CIMI classes are derived.
4. The ASSOCIATION\_CLASS hierarchy from which all CIMI association classes are derived (e.g., PARTICIPATION, PARTY\_RELATIONSHIP, ClinicalStatementAssociation)

### CIMI primitives

We list below the set of CIMI primitive types and their mappings to UML and FHIR. In CIMI all primitive types derive from the Any abstract class:

|  |  |  |  |
| --- | --- | --- | --- |
| CIMI Primitives | UML Primitives | XML Primitives | FHIR Primitives |
| Boolean | Boolean | -- | boolean |
| Real | Real | -- | decimal |
| Byte | -- | byte | Byte |
| Integer | Integer | -- | integer |
| Character | Character | -- | -- |
| String | String | -- | string |
| URI | -- | anyUri | uri |
| List<T> | -- | -- | -- |
| Array<T> | -- | -- | -- |
| Any (root type) | -- | -- | Element |

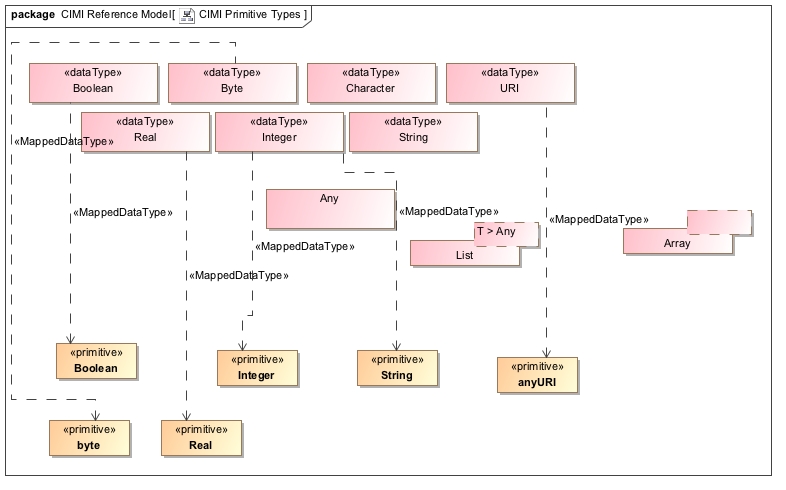


Figure 1 - CIMI Primitive Types

### CIMI Complex Types

We list below the CIMI ‘complex’ data types and their mappings to FHIR. In CIMI, all complex data types derive from the DATA\_VALUE abstract class:

|  |  |  |
| --- | --- | --- |
| CIMI Data Types | FHIR Primitives | FHIR Complex Types |
| DATE | date | -- |
| TIME | time | -- |
| DATE\_TIME | dateTime | -- |
| COUNT | -- | Count |
| PROPORTION | -- | Ratio<Quantity> |
| QUANTITY | -- | Quantity |
| DURATION | -- | Duration |
| INTERVAL\_VALUE<T> | -- | Period, Range |
| PLAIN\_TEXT | string\* | -- |
| CODED\_TEXT | -- | Coding |
| URI\_VALUE | uri | -- |
| EHR\_URI | uri | -- |
| IDENTIFIER | -- | Identifier (partial match) |
| YESNO | boolean\* | -- |
| PARSABLE | string\* | -- |
| MULTIMEDIA | -- | Attachment (partial match) |
| ORDINAL | -- | -- |
| DATA\_VALUE (root type) | -- | Element |

\* *Extended FHIR primitive types*

CIMI and FHIR differ in the boundary between primitives and complex data types as shown in the table above. Moreover, a number of FHIR types do not have equivalents in CIMI. These include Annotation and SampledData, though both types could be accommodated within the CIMI CLUSTER hierarchy.

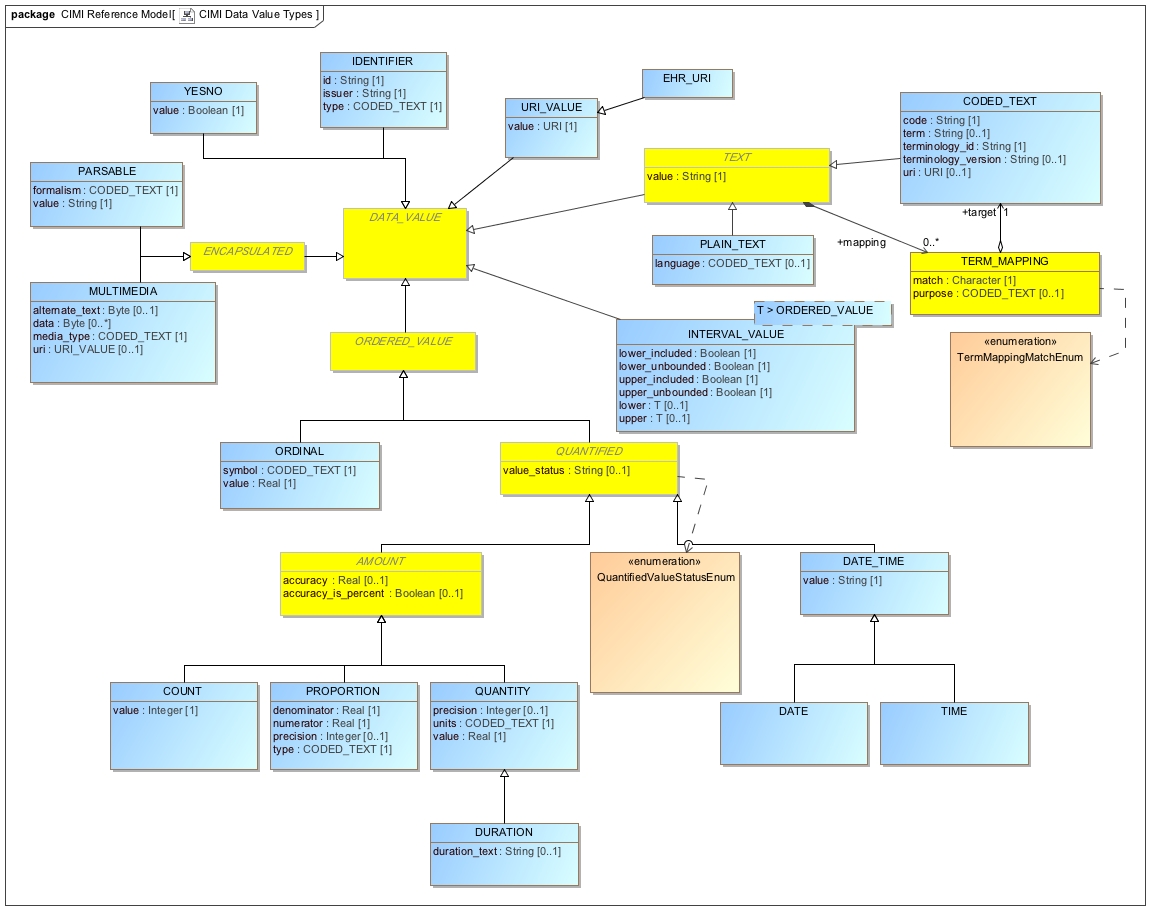


Figure 2 - CIMI Data Value Types

### The LOCATABLE and ASSOCIATION\_CLASS Root Types

All classes in CIMI apart from CIMI primitives and complex types derive from either the ASSOCIATION\_CLASS or LOCATABLE classes. An ASSOCIATION\_CLASS represents a qualified relationship between two LOCATABLEs.

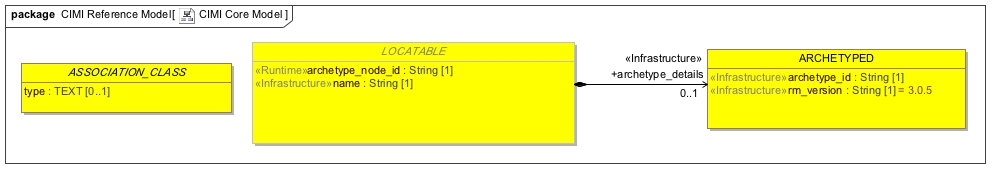


Figure 3 - CIMI Core Model

In the following sections will introduce some of the reference model patterns found in CIMI that build upon the CIMI Core Reference Model Module.

## The CIMI Foundational Reference Model Module

The CIMI Foundational Reference Model introduces the following top-level hierarchies:

1. The CLUSTER and VIRTUAL CLUSTER hierarchies
2. The COMPOSITION hierarchy
3. The CONTENT hierarchy
4. The PARTY hierarchy
5. The PARTICIPATION and PARTY\_RELATIONSHIP hierarchy

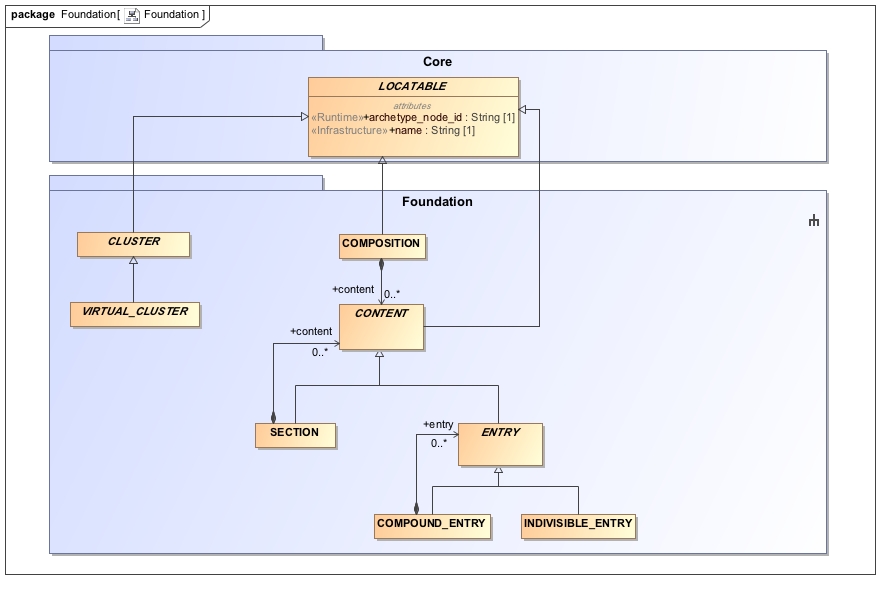


Figure 4 - CIMI Foundation Package

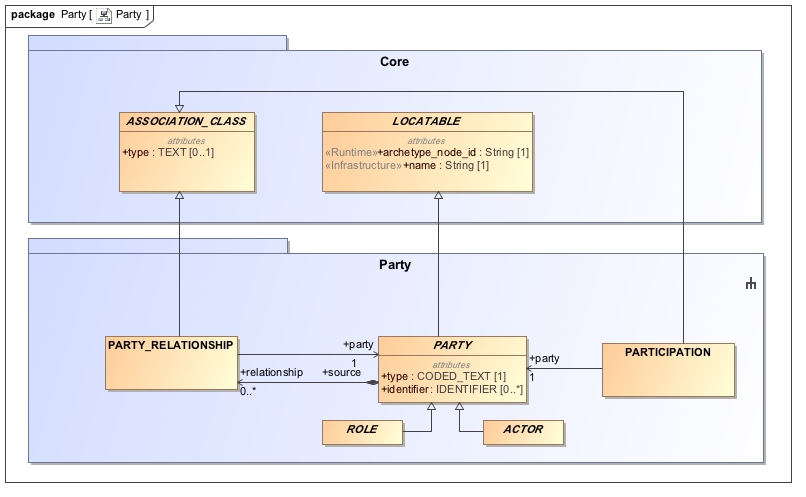


Figure 5 - CIMI Party Package

We further describe each hierarchy in this module below.

#### The CLUSTER/VIRTUAL\_CLUSTER hierarchies

The CLUSTER abstract class is the starting point for CIMI structures such as addresses, contact information, medications, and devices. Unlike CLUSTER, VIRTUAL\_CLUSTER allows for the grouping of attributes to support model component reuse and consistency but whose containment structure is not essential to the model and can be ignored by tools, editors, and code generation frameworks.

The CLUSTER hierarchy differs from the DATA\_VALUE hierarchy in that specializations of DATA\_VALUE represent a concise set of highly stable complex types (e.g., a code, a quantity or a proportion) while specializations of CLUSTER are used to define the far more numerous and variable structures that are used to compose the reference model patterns (e.g., person names or international address structures).

#### The COMPOSITION hierarchy

At this time, the COMPOSITION hierarchy consists of a single class that can be used to represent clinical reports or patient records at the archetype level, for instance. A composition is composed of sections and content entries.

#### The CONTENT hierarchy

The CONTENT hierarchy plays a special role in CIMI because it is the parent hierarchy for Clinical Statements. The CONTENT class has the following subclasses: SECTION and ENTRY. ENTRY, in turn, has two specializations: COMPOUND\_ENTRY and INDIVISIBLE\_ENTRY. A COMPOUND\_ENTRY can be composed of other COMPOUND\_ENTRYs and INDIVISIBLE\_ENTRYs thus supporting a recursive pattern which can be constrained accordingly at the archetype level. ENTRYs represent units of standalone clinical information. An example of a COMPOUND\_ENTRY may be a laboratory panel or a complex orderable while an individual analyte or simple procedure may be represented by an INDIVISIBLE\_ENTRY.

The SECTION content type may be used to represent sections in a document or a simple collection of entries without metadata. Note that the latter use of SECTION differs from COMPOUND\_ENTRY which represents a logical grouping of entries with can hold metadata about the grouping. For instance, a laboratory panel is a type of COMPOUND\_ENTRY that holds provenance information for the panel as a whole.

#### The PARTY, PARTICIPATION, and PARTY\_RELATIONSHIP pattern

The CIMI Foundational Reference Model introduces the participation pattern. It defines PARTY, which may be either a ROLE or an ACTOR and a PARTY\_RELATIONSHIP to represent a relationship between two parties such as an ACTOR playing one or more roles in the performance of an activity. Each class in the pattern has a type attribute to allow the binding of formal semantics at the archetype level – concept representing the type of actor or the type of role performed. These classes also serve as the root types for specializations introduced in other reference modules.

## The CIMI Clinical Reference Model Module

The CIMI Clinical Reference Model module introduces a number of structural pattern upon which all CIMI archetypes are built.

### The CIMI Party, Participation Patterns

The CIMI Clinical Reference Model Party pattern builds upon the CIMI Foundational Reference Model PARTY pattern as shown below. It adds a number of attributes to both ROLE and ACTOR and defines two role specializations: a HealthCareConsumerRole and a HealthCareProviderRole. It also defines two actor specializations at this time: a person and an organization. The association between a person or organization and the role they play in an activity is achieved by the use of PARTY\_RELATIONSHIP (or a specialization thereof) defined in the CIMI Foundational Reference Model module.

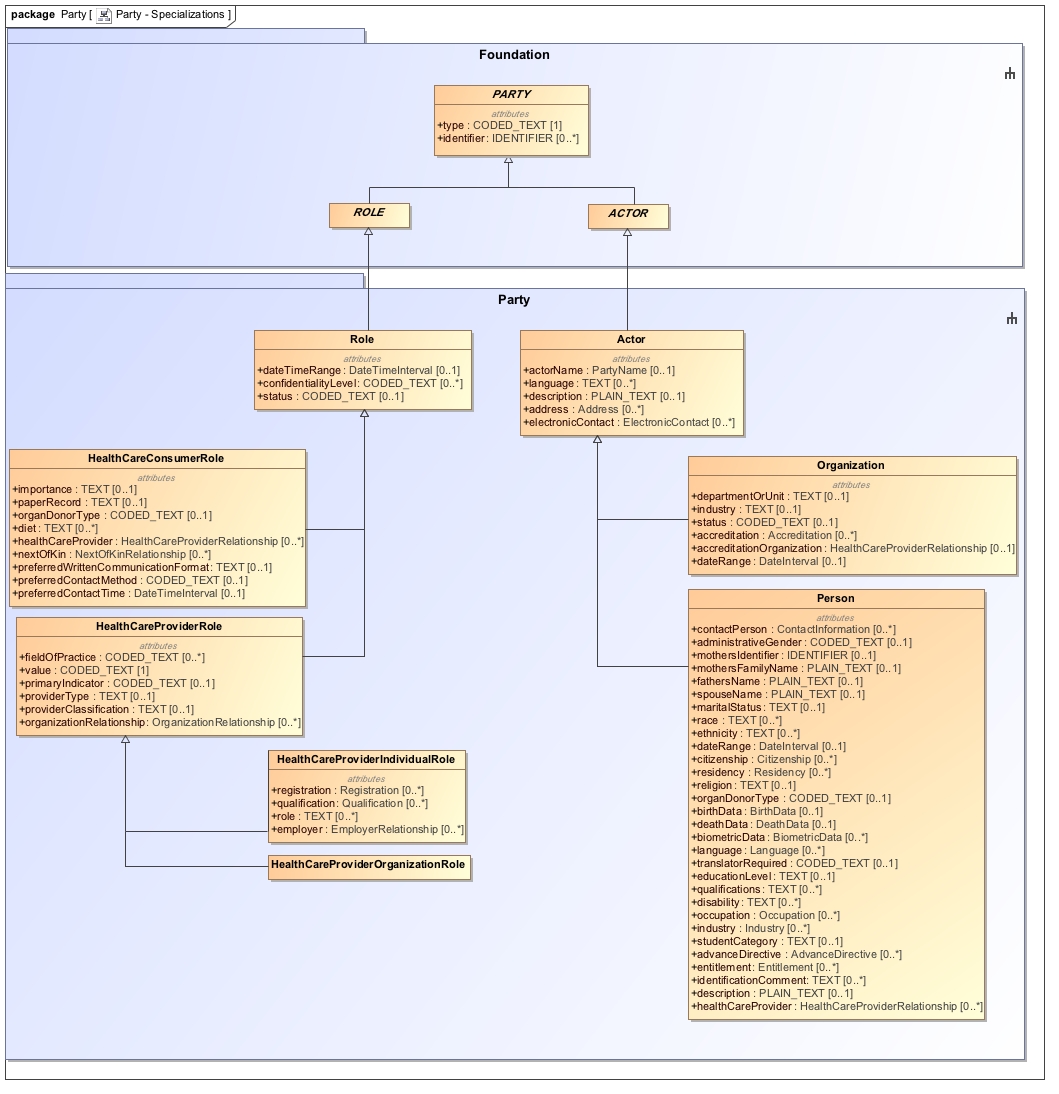


Figure 6 - Specializations of the PARTY class

### The CIMI Clinical Statement Pattern

The CIMI Clinical Statement Pattern forms the core of the CIMI model. The ClinicalStatement class is a specialization of the ENTRY class from the Foundation Reference Model module. It represents a statement about some aspect of a health care process. The CIMI ClinicalStatement pattern is composed of a statement topic (*StatementTopic)* and the situational context for that topic (*StatementContext*). The ClinicalStatement pattern also includes relevant attribution metadata for the information contained in a clinical statement (*please refer to the Attribution/Provenance Patterns section below*).

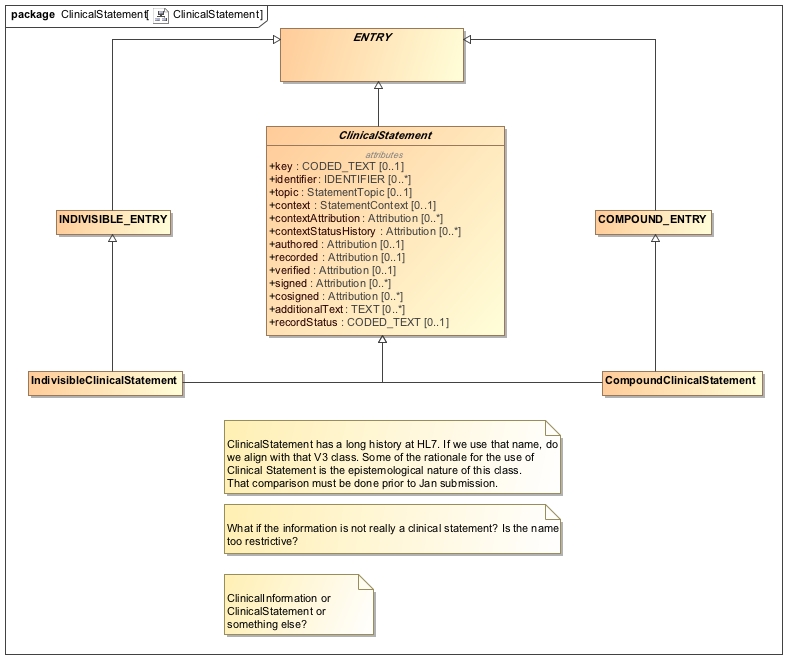


Figure 7 - Clinical Statement Pattern

The StatementTopic abstract class has two specializations:

1. *Finding*, an abstract class, which is further specialized by the Assertion and EvaluationResult statement topics.
2. *Act*, an abstract class, which is further specialized by the Procedure and LaboratoryProcedure statement topics.

Note that for the January 2017 ballot cycle, we limit the scope of the model to the statement topics listed above. However, additional statement topics will be introduced over time.

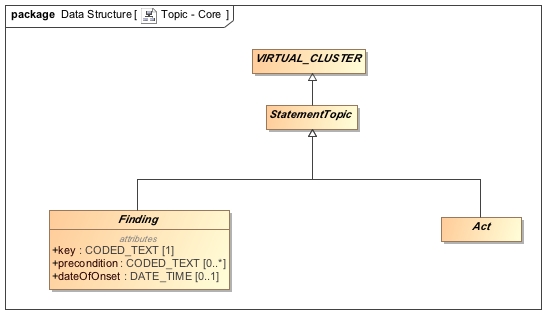


Figure 8 – Clinical Statement Topics

The StatementContext abstract class has the following two specializations:

1. FindingContext - The FindingContext class aligns with the SNOMED Situations with Explicit Context for findings and provides the context for the EvaluationResult and Assertion topics of a clinical statement. For instance, a statement about findings may state that the finding was observed present or absent.
2. ActionContext - The ActionContext class aligns with the SNOMED Situations with Explicit Context for procedures and provides the context for the Act topic of a clinical statement. For instance, a statement about a procedure may specify that the procedure has been proposed, ordered, planned, performed or possibly not performed. Each action context has its own lifecycle. For instance, a *Proposal* may have a status of initiated, reviewed, updated, approved, or rejected. Attribution information for each of these activities is captured by the ClinicalStatement.contextStatusHistory attribute while the latest status of a given action context is captured by the attribute ActionContext.currentStatus of type CODED\_TEXT. The value of the *currentStatus* attribute is the Attribution.activity code associated the last status attribution in the context status history for the given context. The ClinicalStatement.contextAttribution provides attribution information for the context itself (e.g., who initiated the proposal, when, how, etc…) and typically corresponds to the first status in the context status history.

[To Claude: Add diagram to support explanation]

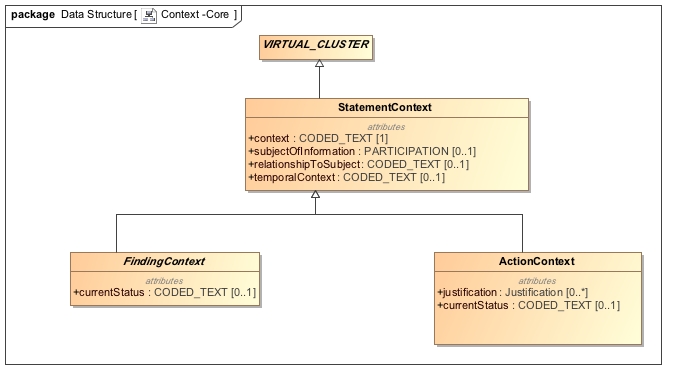


Figure 9 - ClinicalStatement Contexts

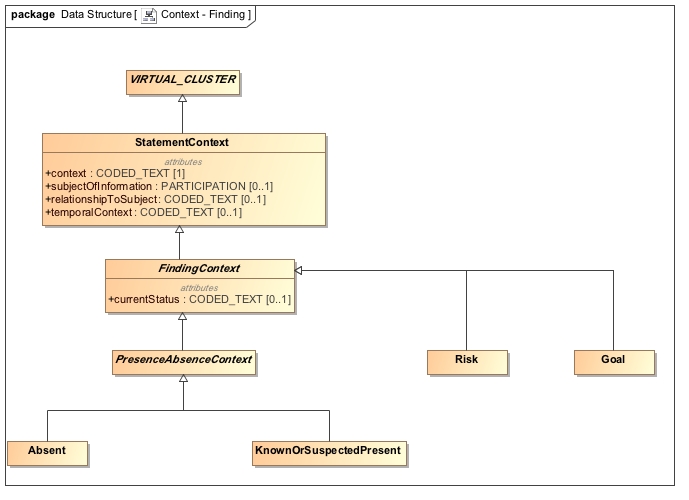


Figure 10 - Finding Contexts

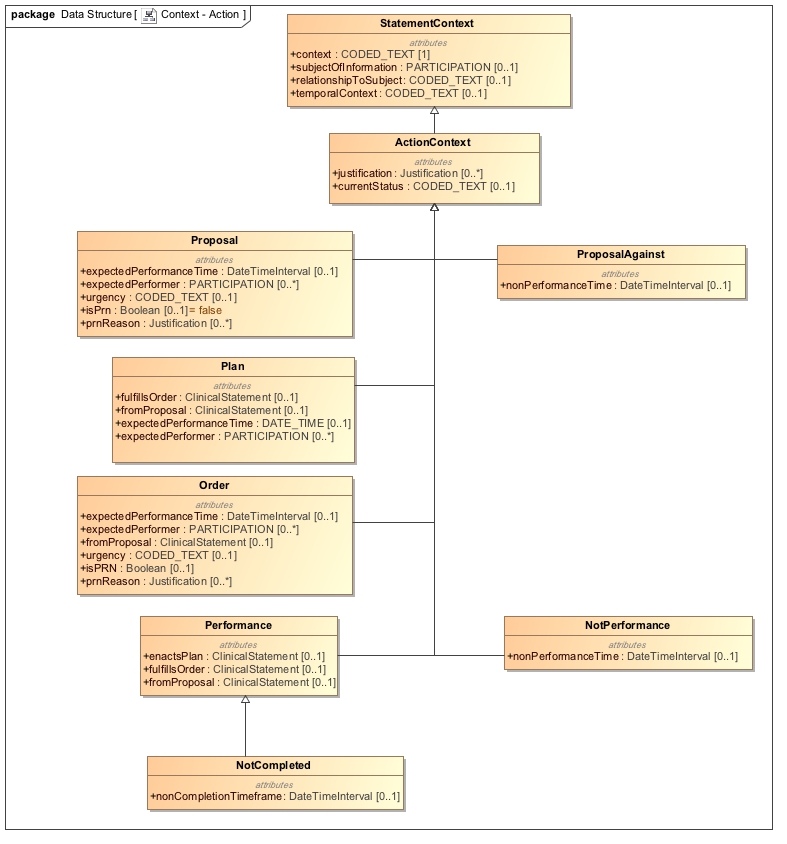


Figure 11 - Action Contexts

StatementTopic and StatementContext are attribute groups (VIRTUAL\_CLUSTERs) and have the following characteristics:

1. They are reusable components, which can be assembled to form clinical statements. For instance, one can coordinate the *Procedure* statement topic with the *Proposal* statement context to represent a ProcedureProposal. The Procedure statement topic may also be paired with the *Order* statement context to create a *ProcedureOrder* statement.
2. They represent groupings of attributes that are aligned with the SNOMED CT Concept Model. For instance, the Procedure statement topic is aligned with the SNOMED CT Procedure Concept Model. The Performance context aligns with the Situation with Explicit Context Concept (SWEC) Concept Model.
3. They provide for a mechanism to state presence or absence of a finding as well as performance or non-performance of an action. For instance, the pairing of the Procedure topic with the NonPerformance context allows for the expression of a procedure that was not performed.

The CIMI ClinicalStatement pattern aligns with the SNOMED CT Situation with Explicit Context Concept Model as follows:

* The ClinicalStatement.topic attribute aligns with the SNOMED CT Concept Model attribute
  + 246090004 | Associated finding (attribute) | if the ClinicalStatement topic is aStatementTopic of type *Finding.*
  + 363589002 | Associated procedure (attribute) | if the ClinicalStatement topic is a StatementTopic of type *Act*.
* The StatementTopic classes align with the SNOMED CT Concept Model as follows:
  + The *Act* StatementTopic aligns with the SNOMED CT *Procedure* concept model.
  + The *Finding* StatementTopic aligns with the SNOMED CT *Clinical Finding* concept model if the topic is an *Assertion* or with the *Observable* concept model if the topic is an *EvaluationResult*.
* The ClinicalStatement.context attribute aligns with the SNOMED CT Concept Model attribute:
  + 408729009 | Finding context (attribute) | if the ClinicalStatement.topic is a StatementTopic of type Finding. The range of this attribute is a StatementContext of type FindingContext.
  + 408730004 | Procedure context (attribute) | if the ClinicalStatement.topic is of StatementTopic of type Act. The range of this attribute is a StatementContext of type Action Context.
* The StatementContext.context attribute also aligns with the SNOMED CT Concept Model attribute:
  + 408729009 | Finding context (attribute) | if the StatementContext is of type FindingContext. Its range must be consistent with the range specified in the SNOMED Technical Implementation Guide.
  + 408730004 | Procedure context (attribute) | if the StatementContext is of type ActionContext. Its range must be consistent with the range specified in the SNOMED Technical Implementation Guide.
* The StatementContext.relationshipToSubject aligns with the SNOMED CT Concept Model attribute 408732007 | Subject relationship context (attribute) |. Its range must be consistent with the range specified in the SNOMED Technical Implementation Guide.
* The StatementContext.temporalContext aligns with the SNOMED CT Concept Model attribute 408731000 | Temporal context (attribute) |. Its range must be consistent with the range specified in the SNOMED Technical Implementation Guide.
* The StatementContext.subjectOfInformation attribute is a PARTICIPATION whose type is constrained according to the range specified for the SNOMED CT concept model attribute 408732007 | Subject relationship context (attribute) |.

### The CIMI Attribution/Provenance patterns

In the CIMI model, provenance information is represented by the Attribution class. The Attribution class provides a pattern for the capture of provenance information such as the *what, who, when, where, why*, and *how* associated with a particular activity – e.g., provenance attributes about the verification of a clinical statement.

CIMI proposes two attribution patterns:

1. *Attribution information as a part of the clinical statement* – In this pattern, the ClinicalStatement pattern contains a number of attributes of type *Attribution* (e.g., ClinicalStatement.authored and ClinicalStatement.verified). This pattern provides a consistent way to capture attribution information that extends beyond simply the agent of an activity (e.g., the author). When attribution is part of the ClinicalStatement model, any change to the attribution for an activity will result in a version change.
2. *Attribution information that is external to the clinical statement* - CIMI also allows the capture of provenance information that is external to the clinical statement through the *Provenance* class. The provenance class includes the Attribution class and pointers to one or more clinical statements (the *Provenance.target* attribute). This pattern allows the attachment of provenance information to a clinical statement without impacting its version.

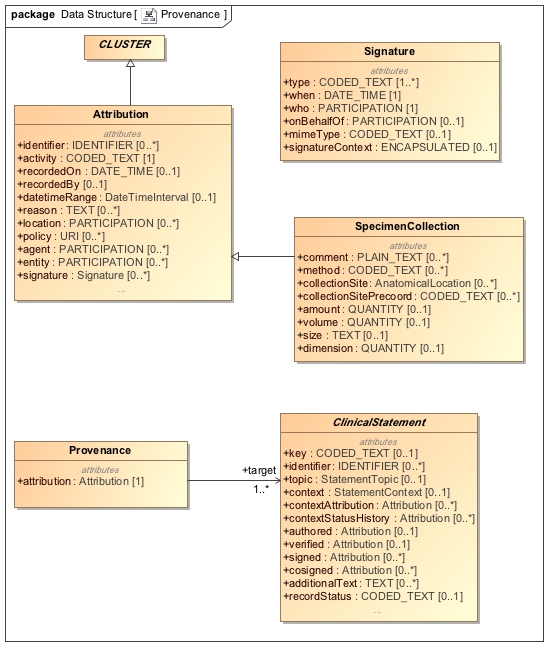


Figure 12 - Provenance Patterns

### The CIMI Assertion and EvaluationResult Pattern

At this time, CIMI defines two specialization of the Finding Statement Topic: Assertion and EvaluationResult.

The assertion model is used to capture information about a clinical finding whereas the evaluation result model is used to capture information about an observable. An evaluation result consists of an observed characteristic (sometimes referred to as ‘the question’) and a value for the observed characteristic (sometimes referred to as ‘the answer’). At this time, no specializations for Assertion have been provided but we anticipate that this model will be specialized in future. Two specializations of Evaluation Result are provided in the reference model: LaboratoryTestResult and PhysicalEvaluationResult. From these, a number of core archetypes are provided such as QuantitativeLaboratoryTestResult and CodedLaboratoryTestResult. These archetypes constraint the result attribute of EvaluationResult to a QUANTITY and a CODED\_TEXT respectively. Please refer to the archetypes for more examples of such constraints.

For more information on the proper usage of these classes, please refer to the Style Guide section below.

Both models are shown below.

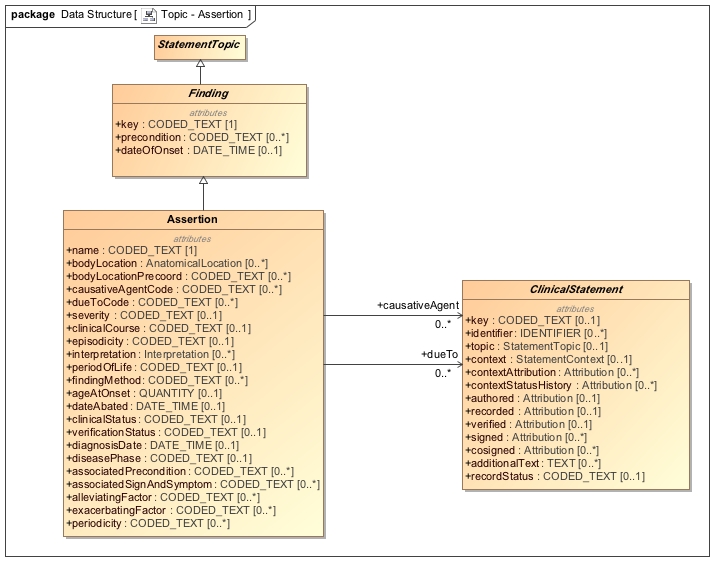


Figure 13 – The Assertion Statement Topic

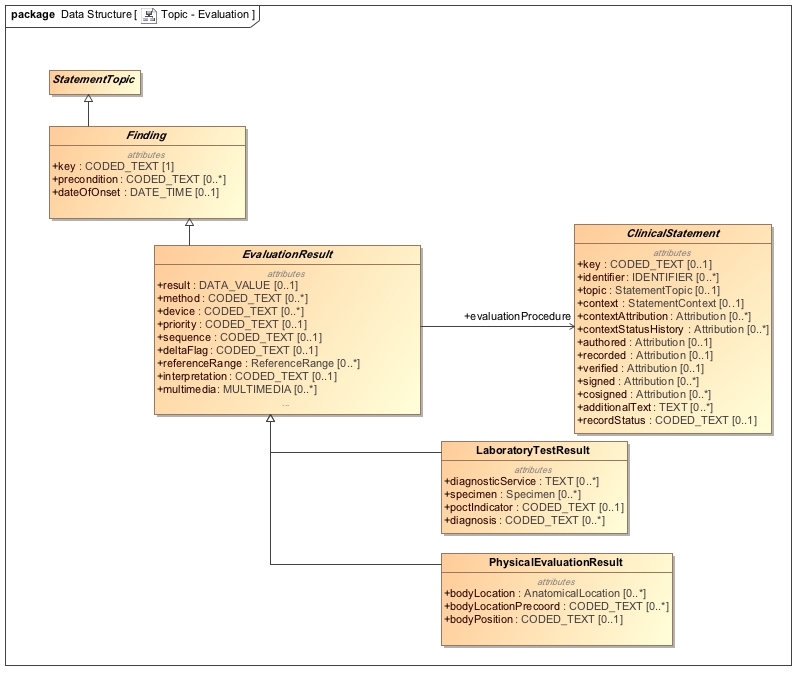


Figure 14 – The EvaluationResult Statement Topics

### The CIMI Procedure pattern

As part of this submission, we also include an example of the Act statement topic, namely Procedure and its specialization LaboratoryProcedure. Note that both models are incomplete at this time but serve as important examples for clinical statements and the expression of ‘negation’ patterns in CIMI (e.g., ProcedureNotPerformed is composed of the Procedure statement topic and the NonPerformance statement context).

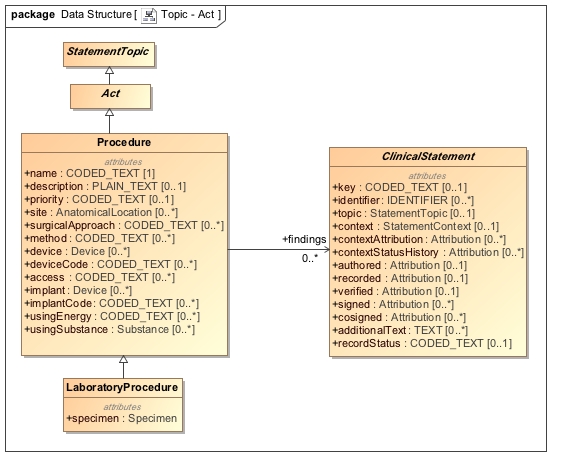


Figure 15 - Action Topics

### CIMI Data Structures

CIMI Data Structures are reusable components that are derived from the CLUSTER supertype. They are used to define component structures necessary in the construction of CIMI patterns. Examples include structures such as Address, PartyName, BirthData, and so on. For more information on these structures, please refer to the diagrams below and to the CIMI Logical Model Specification.

### 

Figure 16 - Material Entities

### 

Figure 17 - Other Data Structures

### 

Figure 18 - Address Structures

### 

Figure 19 - Party and Related Structures

# An Introduction to CIMI archetypes

[Claude to provide shortly]

# The CIMI Style Guide

## Modelling Layers

In order to achieve an increased level of consistency between models, all CIMI clinical models will be based on one or more predefined ‘modelling patterns’. These modelling patterns are defined in Chapter 5. The first-layer of Clinical Models, developed based on these modelling patterns, will (wherever possible) be free of specific use-case context, specialty context, care-setting context and implementation-purpose context. These additional levels of context will be added separately, to enable a maximum level of reuse and consistency between these context-dependent models.

**Reference Model**

**Modelling Patterns**

*Schedule,*

*Address,*

*Material*

*Observation,*

*Action*

*Clinical List*

*Event Summary*

**Clinical Models**

*Medication Item*

*Blood Pressure*

*Medication List*

*Discharge Summary*

**Add Specialty Context**

*Paediatric Medication Item*

*Neonatal Blood Pressure*

*Nephrologist Medication List*

*Cardiology Discharge Summary*

**CLUSTER**

**ENTRY**

**SECTION**

**COMPOSITION**

**Add Care Setting Context**

*G.P. Dispensed Medication Item*

*Home Blood Pressure*

*Outpatient Clinic Current Medication List*

*Inpatient Discharge Summary*

**Add Implementation Purpose Context**

*Dispensed Medications GUI*

*Neonatal Blood Pressure in EHR*

*Current Medication List in EHR*

*Discharge Summary Doc or Message*

**Add Use Case Context**

*Dispensed Medication Item*

*Standing Blood Pressure*

*Current Medication List*

*Medication Reconciliation Report*

*Figure 4: CIMI’s Proposed Modelling Layers*

# Modelling Patterns

The following modelling patterns have been proposed as the basis of CIMI’s clinical models:

* ELEMENT
  + To be defined
* CLUSTER
  + To be defined (e.g. Reference Range, Material)
* ENTRY
  + To be defined (e.g. Property Observation, Finding Observation, Activity)
* SECTION
  + To be defined (e.g. Clinical List)
* COMPOSITION
  + To be defined (e.g. Event Summary)

# Cluster Patterns

Cluster Patterns are to be defined.

# Entry Patterns

Entry Patterns are to be defined.

**Question**: When do we require a separate Action and Observation

**Proposed Answer**: a) if the Action is itself substantial / important, e.g. a record of surgery, catheterisation, intubation, any other procedure or b) it is separated in time from the Observation, as is the case with taking a blood or tissue sample (action) and analysing it (observation).

# Section Patterns

## Evaluations

An evaluation model is a model that evaluates a characteristic of a patient or a patient-related situation. An evaluation holds the name of a “test” in the key (e.g., “heart rate evaluation”, “serum glucose lab test”, etc.) and the test result value in “result” attribute. Another way to view the situation is the key holds a "question" (e.g., "what is the heart rate?", "what is the serum glucose?") and “result” holds the answer. Any archetype (a test, a study, a procedure, etc.) that fits this pattern of a name and a value is modelled with the evaluation pattern.

The result attribute of an evaluation model is typically comprised of a QUANTITY, , or CODED\_TEXT. Physical quantity data type consists primarily of a numeric (real) value and a unit of measure. The unit of measure is captured by a code. QUANTITY evaluations are often constrained to a valid “unit of measurement” value set (e.g., “length units of measure”, “mass units of measure”, etc.).

Coded Ordinal data type consists primarily of a code that has an inherent ordering within a set of codes. An example of a component whose data type is CO is “severity”. The value set for severity includes codes for “mild”, “moderate”, and “severe”. The codes would have properties or relationships that capture the ordering among them, i.e., “mild” < “moderate” < “severe”.

Coded Text is used when the answer drawn from a set of coded values. Hair color, urine description, cardiac rhythm, and cervical consistency are examples. In each case, the coded values are adjectives that describe the property.

### Evaluation Subtypes

#### Laboratory Evaluation

#### Physical Evaluation

## Assertions

The Assertion Pattern is used to represent a clinical finding.

An assertion model XXX. There are two upper level assertion classes; Assertion Absent and Assertion XXX. Assertions may:

* assert the presence (or absence) of a condition in a patient, for example:
  + ChestPainAssert asserts the presence of chest pain
  + EdemaAssert asserts the presence of edema
    - assert the presence (or absence) of an apparatus, for example:
* assert the presence of a monitoring or administration device
  + OximeterDeviceAssert asserts the presence of an oximeter
* assert the presence of a therapeutic apparatus, such as:
  + compression stockings or a warm blanket
* assert the occurrence (or non-occurrence) of an event in which the patient was involved, for example:
  + an auto accident
  + a natural disaster

The pattern followed for assertion is:

The assertion pattern is as follows:

* Key = a code meaning “assertion”
* Data = a code representing what’s being asserted (“rash”, “auto accident”, “hypertrophy”, etc.)

Another pattern that could have been selected is:

* Key = a code representing what’s being asserted (“rash”, “auto accident”, “hypertrophy”, etc.)
* Data = “present” or “yes”

The rationale for selecting the first pattern is twofold:

* + 1. It was judged to be most consistent with the semantics of terminologies we expect to map to. The “key” and “data” in the CEM form a sort of “question”/”answer” or “test”/”result” pair. In cases of measurements, this is more obvious. For example, in a heart rate measurement model, the “key” or “test” is “heart rate measurement” (or the question is “what is the heart rate measurement?”) and the “data” captures “60 bpm”, “70 bpm”, etc. (the answer, or result). Given this, in the first pattern, the “key” or “test” is “assertion” (or the “question” is “what is being asserted?”), and the “data” holds the “result”/”answer” (the thing being asserted).

In the second pattern, if “rash” were the “key”, and “data” held “present’, then the meaning of the “key” (to fit the “question”/”answer” or “test”/”result” paradigm) is actually “evaluation of whether a rash is present or absent”, not “rash” as a condition in the patient. The latter, though, is the meaning of the concept in a terminology such as SNOMED CT. In contrast, in the first pattern, the code that populates “data” is consistent with the SNOMED meaning. Thus the first pattern and not the second facilitates mappings to standards.

* + 1. The pattern followed the conclusions of the HL7 TermInfo effort. The HL7 TermInfo project[[3]](#footnote-1) sought to specify guidelines for using SNOMED CT concepts within the HL7 Reference Information Model. The group followed the reasoning of 1) above. Hence, using the first pattern aligns the CEMs with the TermInfo effort.

### Assertion Subtypes

The assertion subtypes are not yet created for this ballot but we expect to create the following possible subtypes.

#### Clinical Assertions

Clinical Assertions assert the existence of clinical conditions, diseases, etc. in the patient. The partial model ClinicalAssert (whose key is “Assertion”) is used as the parent. A specific Clinical Assertion extends ClinicalAssert with any additional qualifiers necessary for the specific data element. It also constrains the data code of ClinicalAssert to a code representing the condition or disease being asserted. Figure XXX shows the example of WoundAssert. Other examples are CoughAssert, NauseaAssert, and DiabetesMellitusTypeOneAssert, etc.

#### Event Assertions

Event Assertions are used to assert that an event involving the patient occurred, e.g., auto accident, poisoning, burns, etc. It is expected that the concepts used to populate “data” would be mappable to concepts in the SNOMED event hierarchy.

#### Device Assertions

Device Assertions are used to assert the presence of an apparatus or piece of equipment relative to the care of the patient (e.g., warm blanket, IV pump, compression hose, etc.) or a condition of such apparatus or equipment (e.g., tube leak, ventilator piston centered, chest tube patent, etc.). A specific Device Assertion extends ObservationAssert with any additional qualifiers necessary for the specific data element. It also constrains the data code of ObservationAssert to a code representing the apparatus or equipment observed to be present.

**Guideline: Assertion versus Evaluation**

In many cases, the decision between using the evaluation pattern and the assertion pattern is intuitive and straight forward. “Urine color”, for example, is clearly best modeled as an evaluation – the property being evaluated is the color of the patient’s urine, and the value (data) of the evaluation is the set of codes representing the colors that will be observed. To model urine color as an assertion would require the creation of unnatural precoordinated concepts – the key would be “assertion”, and data would be populated by a set of codes such as “amber urine” (meaning “the patient has amber urine”), “clear urine”, etc.

However, this highlights the fact that any evaluation model may be transformed into an assertion model. (Conversely, any assertion model may be transformed into an evaluation model.) In the case of urine color, the decision is intuitive. But in other cases, the decision is less clear.

For example, “heart rhythyms” (bradycardic, tachycardic, etc.) may be modeled as multiple assertion models (bradycardia, tachycardia, etc.) or as a “heart rhythms” evaluation model whose data is constrained to a value set (containing “bradycardic”, “tachycardic”, etc.).

The general guideline is if it’s most natural to think of the data element as a noun – as a condition or state that exists in the patient – model as an assertion or set of assertions. If the statement about the patient is most naturally thought of as a name/value pair (i.e., a noun representing the property and an adjective representing the value), such as “hair color” = (“black”, “brown”, “blonde”), then model it as an evaluation.

Another hint is if the desire is to represent an abnormal condition in the patient, use an assertion model. Evaluations are most often used to capture descriptions of a patient property where some or all of the valid values are perfectly normal in a healthy patient. For example, if the skin is being evaluated for warmth, color and moisture level the model will be SkinCharacteristicsEval.

This discussion highlights the importance of isosemantic models. (See the discussion in the “**Error! Reference source not found.**” section.) Even if one model or set of models can be agreed upon for the storage model (e.g., assertion models for “bradycardia” and “tachycardia” instead of an evaluation model with “bradycardic” and “tachycardic” as values), inevitably there will be use cases (data entry, messaging, reporting, etc.) for the other model. An essential (as of now unfulfilled) requirement is for a mechanism of identifying isosemantic models, managing isosemantic groups, and transforming between them.

## Procedures

Procedure models are used for actions taken related to the care of a patient. The actions represented are not necessarily *surgical* procedures. (It is expected that one procedure model, yet to be created, will be a “surgical procedure” model, which can be further extended or constrained and which will contain the common attributes of a surgical procedure.) The actions might be any patient care-related actions, e.g. peripheral IV placement, delivery of a warm blanket, dressing change, ambulation, patient education, etc.

There are three upper level procedure classes; *Procedure Proposal* and *Procedure Performed* and *Procedure Not Done*.

At least three use cases exist pertaining to use of measurements (for example, a heart rate measurement). The most common use case is to capture the values of the measurements and retrieve them for display, calculations, trending, decision support, etc. Second, in certain circumstances, the desire might be just to document that a heart rate measurement was performed – regardless of the value of the resulting measurement. For example, a protocol may dictate that heart rate measurements be taken according to a time schedule, and an application monitoring protocol compliance may query to ensure that the measurements were taken. In this case, the measurement values are inconsequential. Third, in certain circumstances it might be desirable to explicitly document that a heart rate measurement was *not* performed. Again, an example may be that a protocol requires that heart rate measurements be taken; users need a way to note exceptions to the protocol and monitors of the protocol need a way to detect the exceptions.

Several options are available for covering these three use cases:

1. Use a single measurement model for all three use cases. Creation of the instance would implicitly represent that the procedure was performed. To state that a measurement procedure was *not* performed, “data” would be left null, and either a null flavor would indicate that the procedure was not performed, a new attribution (e.g., “stated not performed”) or modifier (e.g., “not performed indicator”) would be used, or both null flavor and the attribution/modifier would be used.
2. Use one model for the measurement, another for the procedure (both affirmative and negative). A procedure instance without negation would represent that the measurement procedure was performed. A procedure instance with negation would represent that the measurement was not performed.
3. Use one model for the measurement and affirmative procedure, another for the negative procedure. Creation of an instance of the measurement model would represent a measurement and would implicitly represent that the procedure was performed. To state that a measurement procedure was *not* performed, the other model, which means “procedure not performed” would be used.

The first alternative would require that for the common use case of finding all heart rate measurement results, care must be taken to “weed out” any instances that actually mean “heart rate measurement not performed”. The second alternative would require that for the use case of answering the question, “was a heart rate measurement performed?” (unconcerned with the result), instances of two different models would need to be queried for.

The third alternative (recommended) supports the “find all heart rate measurement results” query without any filtering out of confounding instances and supports the “was a heart rate measurement performed?” query without needing to include two models. It separates out the query for “heart rate measurement not performed” into a separate model. The only use case in which this alternative would be disadvantageous is if an application needed to find “all cases in which a heart rate was or was not performed”. In this case, two models would need to be queried. Such a use case, however, is very unlikely.

For these reasons, the third alternative is recommended.

# Composition PatternsModelling Style

In this section, we define CIMI’s modelling principles, including:

* Quality criteria
* Scope of clinical models
* Granularity of clinical models
* Consistency and reuse
* Isosemantic models
* Terminology Binding, and
* General principles

# Quality Criteria

The following quality criteria have been proposed for all CIMI models:

CIMI models will be:

* Able to satisfy the URU principles – that is they will be
  + Understandable (cohesive and coherently expressed)
  + Reliable and reusable (consistency)
  + Useful (fit for purpose)
  + Up-to-date (currency)
* Clinically accurate
* Clinically valid
* Evidence based
* Adequate to express required clinical statements
* Able to maintain contextual integrity (when transformed into isosemantic models)
* Able to maintain semantic fidelity (when transformed into isosemantic models)
* Clear and precise, minimizing the potential for:
  + Misinterpretation and
  + Misuse or inconsistency in use
* With low complexity (suitable for easy implementation and avoid cognitive overload of users)

# Scope of Clinical Models

The following principles have been proposed to assist in determining the inclusion (or otherwise) of information within a CIMI clinical model

The following information will be included in CIMI clinical models:

* Information that is considered to be directly relevant to the clinical concept being modelled.
* Information that describes the who, what, when, where and how of the clinical concept being modelled.
* Information that may either be represented using pre-coordination or post-coordinated in the structure – for example, the body location of a diagnosis.
* Information that is not described in the exclusion list below.

The following information will be excluded from CIMI clinical models:

* Information that is specific to an implementation use-case - for example, recordkeeping metadata (unless the model is specifically designed for this purpose).
* Information that is specific to a care-setting - for example, hospital ward details (unless the model is specifically designed for this purpose).
* Information that is specific to a clinical specialty – for example, neonatal care information (unless the model is specifically designed for this purpose).
* Information that is used for administrative purposes only – e.g. financial details (unless the model is specifically designed to include this)
* Information that is specific to a local environment (e.g. to satisfy local legislation requirements).
* Information that is included in the pattern on which the model is based
* Information that is considered not to be directly related to the clinical concept being modelled.
* Information that provides a classification for other items in the model (e.g. the class/category of medicine)

# Granularity of Clinical Models

More than one piece of atomic data can be included in the same clinical model when the following conditions hold:

* The atomic pieces of data are all directly related to the concept being modelled
* It is considered to be good clinical practice for instances of these data items to be observed, evaluated or performed together, using the same who, what, when, where and how information.

For example:

* Systolic and diastolic blood pressures will be included together within a single ‘Blood Pressure’ observation model.

# Consistency and Reuse

OUTSTANDING QUESTION: We need to define some principles for when information is embedded into a given archetype, versus when a separate reusable archetype is created. This includes both the use of generalised model patterns, as well as composing models out of smaller parts.

# Isosemantic Models

CIMI clinical models will support isosemantic models in terms are both:

* The ability to transform CIMI models to/from isosemantic representations in other languages/ standards (e.g. CDA, openEHR, ISO13606, DCM, CEM), and
* The ability to transform CIMI models between isosemantic representations that use a different split between terminology pre-coordination and structure.

The first category of isosemantic models (alternative language representations) will be supported by defining mappings to other languages. It is not anticipated that CIMI will provide these mappings, although some exemplars may be provided to demonstrate the capability.

The second category of isosemantic models (terminology pre-coordination versus post-coordination in the model) will be supported by:

* Including the structural representation, for any information which may be represented using a separate attribute in some clinical contexts;
* Defining the semantics of each of these structural attributes using terminology bindings;
* Defining the semantics of the relationships between these structural attributes using terminology bindings to a concept model?;
* Identifying the focus attribute of the isosemantic pattern (i.e. the attribute which may be represented using a precoordination of the other attributes)
* Providing an expression formalism to show the relationship between different isosemantic forms (e.g. compositional grammar)

# Terminology Binding

*Terminology binding*refers to “the assertion of a relationship between the information model and the terminology” [CIMI Glossary]. This binding involves attaching a terminology concept, reference set or expression to a node or link in an information model. Terminology bindings can either be extensional (using a single concept or a set of concepts), or intensional (using a constraint expression which indicates the set of permissible concepts). In the case of intensional binding, further processing is required to determine the specific set of concepts that can legally be bound to the associated node or link in the information model.

There are four main use cases that motivate terminology binding to CIMI models:

1. To support the management and quality control of clinical model libraries, including:
   1. Searching model libraries (using the meaning of the models and their contents)
   2. Identifying semantic overlap between models
   3. Inconsistency of model interdependencies (e.g. the meaning of a constrained archetype is not subsumed by the meaning of the base archetype)
2. To determine the iso-semanticity of two or more instances of models that are semantically equivalent, but structurally different; and to be able to transform between these isosemantic representations, including:
   1. Models that use a different level of precoordination versus structure;
   2. Models that make different modelling design choices (e.g. Representing a laboratory test’s method as a data element, versus a CLUSTER containing data elements)
   3. Models that use a different base representation (e.g. CIMI to HL7 v3 models)
3. To enable querying over data instances of isosemantic model representations (as described above)
4. To support data validation and semantic interoperability (e.g. exchanging data between systems that use different native information structures)

It is proposed that these use cases be met by the fulfilment of the following requirements:

1. A standard, reproducible methodology for defining the meaning of each node in the model using an association with a terminology;
2. A standard, reproducible methodology for defining the valid set of values of each coded ELEMENT in the model (either explicitly or as a constraint expression)
3. A standard, reproducible methodology for establishing semantic relationships between nodes in the same model.
4. A standard, reproducible methodology for establishing semantic relationships between nodes in different models
5. Terminology bindings that are agnostic as to whether nodes are connected using a hierarchy or using links.
6. Terminology bindings that allow the values to be represented in a way that is agnostic to the degree of precoordination versus structure.
7. Terminology bindings that enable the transformation between isosemantic representations of the same model
8. Terminology bindings that allow consistency to be checked within models, and between models related by specialisation or slot filling.
9. Support for semantic node labelling

## 6.6.1 Terminology Binding Guidelines

CIMI supports isosemantic models and one model in an isosemantic family will be selected as the preferred model for interoperability.

A preference will be given to post-coordination in the model structure over precoordination at the concept level (unless precoordinated form is more clinically recognised)

All finalised CIMI Clinical Models will:

* Include a semantic binding from each node in the model to a terminology concept, expression or pattern, which represents the meaning of the node.
* Include a value binding from each node of type TEXT or CODED\_TEXT to a terminology reference set that indicates the valid values for the node – either defined intensionally using a constraint expression, or extensionally as a list of terminology components.

All finalised CIMI Clinical Models may (where appropriate):

* Include a semantic binding that indicates the meaning of the relationship between each hierarchical parent-child node pair in the model.
* Include a semantic binding on each LINK between two non-hierarchical nodes in the same or different models (defined using the LINK.meaning attribute)
* Include a constructor binding on appropriate container-type nodes (e.g. ENTRY, CLUSTER), in the form of a terminology expression, to help determine its iso-semanticity with other model representations.

### Attribute Binding

All semantic bindings to CIMI models (including node meaning, relationship/link meaning and constructor bindings) will be defined. SNOMED CT is the primary reference terminology; LOINC will also be used as a reference terminology as needed for Evaluations. Where there is overlap and a mapping between LOINC and SNOMED CT, particularly when in scope of the agreement (lab, vital signs, anthropmorhpic measurements….); LOINC will be used as the primary binding.

If the CIMI attribute aligns with the SCT concept model, then the attribute will be bound to descendants of *Concept model attribute* ID 410662002 (note that children of *Unapproved attribute (attribute)* ID 408739003 should not be used). If there is no attribute binding (e.g. temporal attributes such as “start time”) then use descendents from the “observable entity” domain

To support terminology bindings to SNOMED CT components, which are not available in the international release, CIMI will develop a SNOMED CT extension in the CIMI Namespace (namespace id: 1000160). This namespace has been allocated and registered by the IHTSDO.

### Value Binding

Value bindings will use SNOMED CT wherever possible – however, where the values are outside the scope of SNOMED CT, other terminologies (e.g. LOINC answers) may be selected (on a case-by-case basis).

When a data element is further refined as a specialization (e.g. skin assessment site is a body location), the value set range must come from the range of the parent data element (e.g. skin assessment site is a subtype of body location) Subtype attribute binding can stay the higher level attribute

## 6.6.4 Linkage Relationships

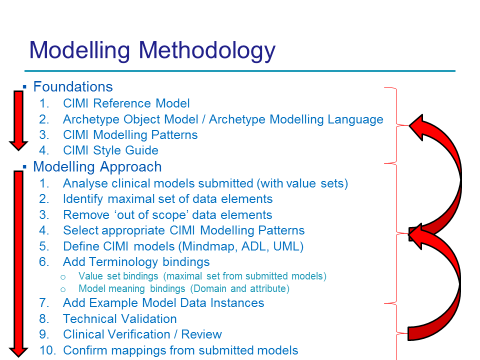
These link classes help determine the relationships between two archetypes. There is a 1:1 mapping between two archetypes (at ENTRY level) with the @meaning attribute defining the semantics of the relationship. – To be expanded –

## 6.6.5 Linkage Relationships

# CIMI Alignment with FHIR

# Modelling Methodology

This section defines the CIMI Modelling Methodology



1. http://www.openehr.org/releases/BASE/latest/docs/bmm/bmm.html [↑](#endnote-ref-1)
2. http://www.openehr.org/releases/AM/latest/docs/ADL2/ADL2.html [↑](#endnote-ref-2)
3. Using SNOMED CT in HL7 Version 3; Implementation Guide, Release 1.5, DSTU Ballot 4 - May 2009 (expired) [↑](#footnote-ref-1)