



Clinical Information Modeling Initiative

Reference Model v1.1.0-dstu

Table of Contents

Model	6
Main	6
CIMI Reference Model	8
Linkable	12
Core	13
ARCHETYPED	13
CORE_LOCATABLE	13
ELEMENT	13
ITEM	14
ITEM_GROUP	14
LINK	14
LOCATABLE	14
PARTICIPATION	15
Data Value Types	15
AMOUNT	16
CODED_TEXT	16
COUNT	17
DATA_VALUE	17
DATE	17
DATE_TIME	17
DURATION	18
EHR_URI	18
ENCAPSULATED	18
IDENTIFIER	18
INTERVAL_VALUE	19
MULTIMEDIA	19
ORDERED_VALUE	20
ORDINAL	20
PARSABLE	21
PLAIN_TEXT	21
PROPORTION	21
QUANTIFIED	22
QUANTITY	22
QuantifiedValueStatusEnum	23
TERM_MAPPING	23
TEXT	24
TIME	24
TermMappingMatchEnum	24
URI_VALUE	25
YESNO	25
Party	26
ACTOR	26

PARTY	26
PARTY_RELATIONSHIP	27
ROLE	27
Primitive Types	27
Any	27
Array	28
List	28
Boolean	28
Byte	28
Character	28
Integer	28
Real	28
String	29
URI	29
Legend	29
Abstract	29
Datatype	29
Demographics Entry Point	29
Entry Point	29
Supporting	29
Primitive Types	30
Mapping CIMI RM 2 HL7 RIM	30
Mapping DCM 2 CIMI RM	30
CIMI Datatypes	31
ContainerConcept	33
DCM Datatypes	33
Element	34
LeafConcept	34
Metadata	35
RootConcept	35
preamble	35
ISO631? language	36
datatype	36
id	36
name	36
Model Transformation Map	36
ADL Workbench	37
BMM	37
CEM	37
CIMI ADL	37
DCM UML	38
HL7v3 CSP	38
LinkEHR	38
MM Tool	38
Mindmap	38
Mindmap	38

OpenEHR ADL	38
UML AML	39
UML Class	39
UML Class	39
UML Object	39
UML Tool (EA, MDHT, RSA)	39
XML Schema	39
Create Instance Example	39
Tool Chain Use Case	40
Core TF	40
Modelling Team	41
Publishing Team	41
Clinical Validate Model	41
Develop Model	41
Develop RM	41
Model Metadata	41
Model Structure	41
Model Terminology	41
Publish Model	42
Publish RM	42
Technical Validate Model	42
AML Tests	42
CIMI Patterns (proposed UML representation)	42
CIMI.RM.ClinicalEntryPattern	42
ClinicalEntry	43
information provider	43
information subject	43
location	44
CIMI.RM.ObservationPattern (mindmap)	44
CIMI.RM.ObservationPattern (Object)	44
INTERVAL_VALUE<T -> DATE_TIME>	45
<anonymous>	46
Observation	46
datetime range	46
datetime	47
details	47
duration	47
name	47
observable	47

offset origin datetime	47
offset	47
result	47
status	48
timing	48
CIMI.RM.ObservationPattern (Class)	48
Clinical Entry	49
Observation	49
observable	49
details	49
name	49
status	49
result	50
timing	50
datetime range	50
Slots in UML Example	50
Class1	51
Class1 B	52
Class2	52
Class2 B	52
Class3	52
Class3 B	52
Discharge Summary	52
More Concrete Discharge Summary	52
Slot Filled In	52
Xxx Diagnose	53
Diagnosis "Slot"	53
Closed class	53
ClosedClass	53
Parameterised Classes	53
Class1	54

CIMI Reference Model

Documentation

Model

Main

The CIMI Reference Model is the underlying Reference Model upon which CIMI's clinical models (i.e. archetypes) are defined. This reference model defines a rigorous and stable set of modelling patterns, including a set of structural patterns, complex data types and demographic classes. All CIMI clinical models will be defined by constraining the CIMI reference model. Each example instance of a CIMI clinical model will be an instance of the CIMI reference model, which conforms to the constraints defined by the associated clinical models.

Welcome

custom Welcome

Welcome to the CIMI Reference Model v2.0.x

This model is the reference model upon which CIMI clinical models are defined. Each CIMI clinical models (i.e. archetypes) is defined by constraining the CIMI reference model. Each example instance of a CIMI Clinical Model is an instance of the CIMI reference model, which conforms to the constraints defined by the associated clinical model.

Version 2.0.x is the result of the minimalistic RM approach, where more complex structures are created by archetyping the minimal RM and building from that.

Version 1.0.4 incorporates feedback from the meeting in Pleasanton, additional documentation, and further improvements resulting from reviews and discussion. It is anticipated that further improvements will be incorporated, as clinical models are developed using this reference model. The core CIMI 1.0.X Modelling team is: *Linda Bird, Thomas Beale, Michael van der Zel, Dave Carlson, Stephen Chu, Stan Huff, Mike Lincoln, Rahil Qamar Siddiqui, Gerard Freriks, Josh Mandel, and Mark Shafarman*. Additional technical expertise was provided by: *Galen Mulrooney, Dipak Kalra, Daniel Karlsson, Peter Handler, Cecil Lynch, David Moner, Sarah Ryan and Harold Solbrig*. Additional clinical modelling support was provided by: *William Goosen, Jay Lyle, Ian McNicoll, Anneke Goosen, Heather Leslie, Hendry Wijaya, and Marcelo Rodrigues dose Santos*.

Version 1.0.0 of this reference model was approved at the May 2012 meeting in Pleasanton as "the starting point for further testing using clinical models". The CIMI 1.0.0 RM team was: *Linda Bird, Thomas Beale, Michael van der Zel, Josh Mandel, Galen Mulrooney, Richard Kavanagh, Gerard Freriks, Stan Huff and Grahame Grieve*.

Changelog 1.1.x up

- 2.0.3 (Harold Solbrig 2014-dec-18)**
Removed structure type from ITEM_GROUP
Created new Linkable abstract class
Removed link and participation from ELEMENT
Changed default rm_version to 2.0.3_dstu
- 2.0.2 (Harold Solbrig 2014-oct-12)**
Removed uid property on locatable
Changed default rm_version to 2.0.2_dstu
- 2.0.2 (MichaelPatrick 2014-sep-17)**
Added missing primitive types Any, List<T> and Array<T>
- 2.0.2 (Harold Solbrig 2014-sep-8)**
Added uid attribute to CODED_TEXT
Removed term_id from CODED_TEXT
Updated all CODED_TEXT documentation
Added is_code_primitive stereotype and stereotyped String, Integer, Real, Boolean, DATE, TIME, DATE_TIME, DURATION and CODED_TEXT
Changed diagram display to show stereotypes
- 2.0.1**
Version number change and added missing types.
- 2.0.0 (Thomas)**
Manual BMM edited from 1.0 generated version
- 1.1.0-dstu (MZ 2014-may-22)**
 - Minimalist approach.
- Changelog 1.0.x**
- 1.0.1-dstu.3 (MZ 2013-dec-16)**
 - Used "Find Orphaned" to clean some import artifacts
- 1.0.1-dstu.2**
 - Moved URI into primitive types
 - Changed all EJAVA_ references to the primitive equivalent
- 1.0.0-dstu.5**
 - LOCATABLE.uid not derived
 - renamed RM package to "core"
- 1.0.0-dstu.4**
 - renamed LOCATABLE.instance_id to the original uid and made cardinality 0..1
- 1.0.0-dstu.3**
 - minor edits to diagrams
- 1.0.0-dstu.2 (telecon on 23-may-2013)**
 - added LOCATABLE.instance_id
- 1.0.12b (telecon on 19-apr-2013) -> 1.0.0-dstu.1**
 - LINK_detail[0..*] added
 - LOCATABLE.name mandatory
- 1.0.12 Leeds (2013-apr)**
 - made @name optional LOCATABLE.name[0..1]
 - value of ORDINAL changed from Integer to Real for decimal scales (example?)
- 1.0.11**
 - See the PPT
- 1.0.9 Scottsdale (2013-jan)**
 - Removed all the PARTY_REF stuff and LOCATABLE_REF also from supporting classes. This is implementation.
 - CORE.LOCATABLE for separating participation from PARTY
 - —
- 1.0.8 Greenwich (2012-dec)**
 - SECTION.items > item
 - CLUSTER.items > item
 - LOCATABLE.links > link
 - TEXT.mappings > mapping
- 1.0.5c > 1.0.6 2012-sep-20, 2012-oct**
 - INTERVAL upper & lower added generic
 - DV_ORDERED > lower added generic
 - DV_ORDERED > ORDERED_VALUE
 - set default value of ARCHETYPES.rm_version to "1.0.6"
 - PARTICIPATION.time and other time attributes in Demographics classed used wrong data value type
 - value of ELEMENT as attribute instead of aggregate
 - Consistent Style: DATA_VALUE's as attributes everywhere
 - added <enumerations>
- >> Rockville Milestone
- 1.0.5b > 1.0.5c CIMI F2F Sunday 2012-sep-16**
 - ARCHETYPED.rm_version note updated and fixed value to 1.0.5
 - LOCATABLE notes updated
- 1.0.5a > 1.0.5b CIMI F2F Friday 2012-sep-14**
 - Removed obsolete CODEABLE_TEXT
 - Renamed the "Core Reference Model" package to "RM"
- 1.0.5 > 1.0.5a CIMI F2F Thursday 2012-sep-13**
 - Renamed "Assumed Type" > "Identifiers" and move datatypes (primitive types) to separate package (not included in BMM generation)
 - Pre-INTERVAL > lower added generic
 - Pre-INTERVAL > ORDERED with "DV_," because they name conflict with the built-in types, used in DemographicValidation
 - Added Mapping DCM 2 CRM as example mapping
 - Added some stereotypes to guide the BMM generation
- 1.0.4 > 1.0.5**
 - Issue#7: Added is_im_infrastructure & is_im_runtime stereotypes to some attributes
 - String <!--> is_im_infrastructure & is_im_runtime
 - Data Value Type "BOOLEAN" renamed to "YESNO"
 - Assumed type "Char" renamed to "Character" to get consistent naming
 - Cleaned up some hidden (old) notes
 - Issue#4: Templatized INTERVAL<T>
 - MULTIMEDIA.data & integrity > is_im_runtime and updated cardinality, container type "Array" and is collection
 - container type default to "List"

Figure: 1

Archetype Entry Points

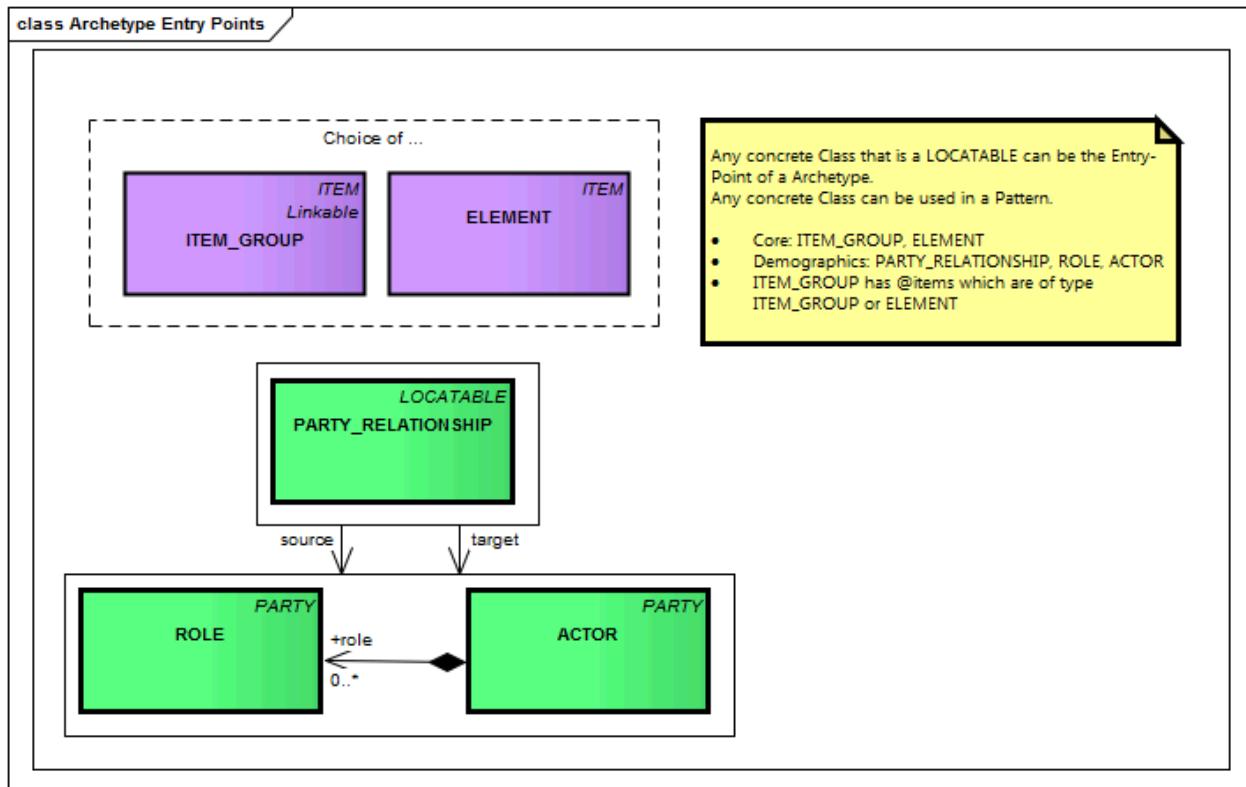


Figure: 2

CIMI Reference Model

Welcome to the CIMI Reference Model v2.0.x

This model is the reference model upon which CIMI clinical models are defined. Each CIMI clinical models (i.e. archetypes) is defined by constraining the CIMI reference model. Each example instance of a CIMI Clinical Model is an instance of the CIMI reference model, which conforms to the constraints defined by the associated clinical model.

Version 2.0.x is the result of the minimalistic RM approach, where more complex structures are created by archetyping the minimal RM and building from that.

Version 1.0.4 incorporates feedback from the meeting in Pleasanton, additional documentation, and further improvements resulting from reviews and discussion. It is anticipated that further improvements will be incorporated, as clinical models are developed using this reference model. The core CIMI 1.0.X Modelling team is: *Linda Bird, Thomas Beale, Michael van der Zel, Dave Carlson, Stephen Chu, Stan Huff, Mike Lincoln, Rahil Qamar Siddiqui, Gerard Freriks, Josh Mandel, and Mark Shafarman*. Additional technical expertise was provided by: *Galen Mulrooney, Dipak Kalra, Daniel Karlsson, Peter Hendler, Cecil Lynch, David Moner, Sarah Ryan and Harold Solbrig*. Additional clinical modelling support was provided by: *William Goosen, Jay Lyle, Ian McNicoll, Anneke Goosen, Heather Leslie, Hendry Wijaya, and Marcelo Rodrigues dose Santos*.

Version 1.0.0 of this reference model was approved at the May 2012 meeting in Pleasanton as "the starting point for further testing using clinical models". The CIMI 1.0.0 RM team was: *Linda Bird, Thomas*

Beale, Michael van der Zel, Josh Mandel, Galen Mulrooney, Richard Kavanagh, Gerard Freriks, Stan Huff and Grahame Grieve.

CIMI Core Model

The CIMI Core Reference Model diagram shows the main classes in the CIMI reference model upon which clinical models will be defined. These include the concrete classes Item Group and Element.

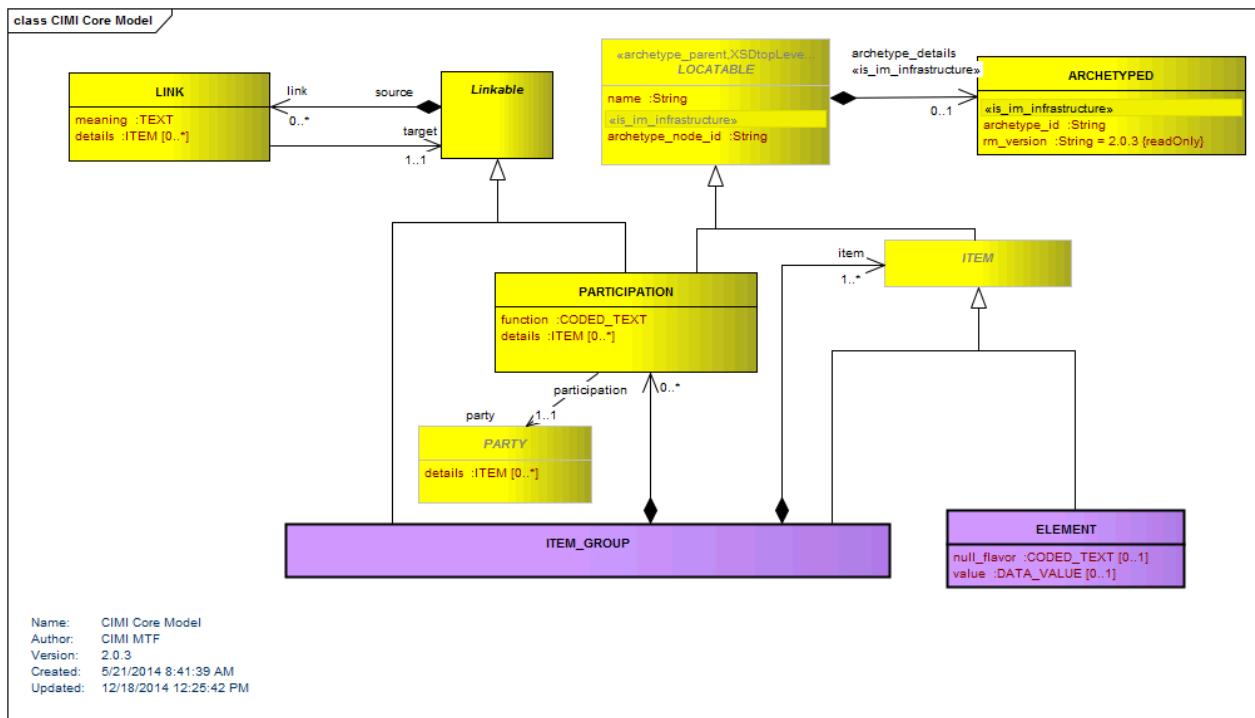


Figure: 3

CIMI Data Value Types

The CIMI Data Value Types diagram shows the set of data types, which underly all other models. These data types provides both general and clinically specific types required for all kinds of health information.

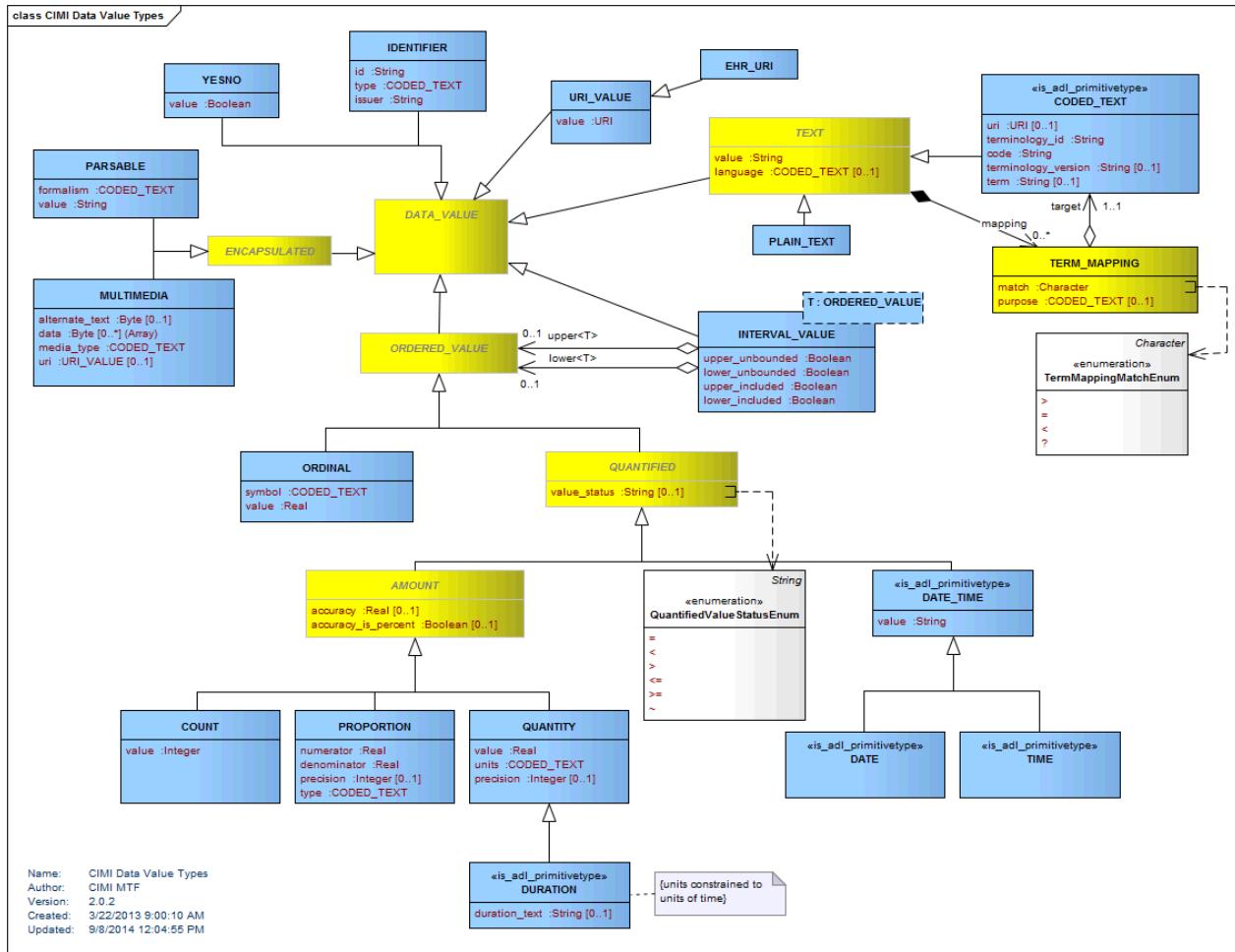


Figure: 4

CIMI Party Model

The CIMI Party Model diagram shows the generic concepts of PARTY, ROLE and related details such as contacts and addresses.

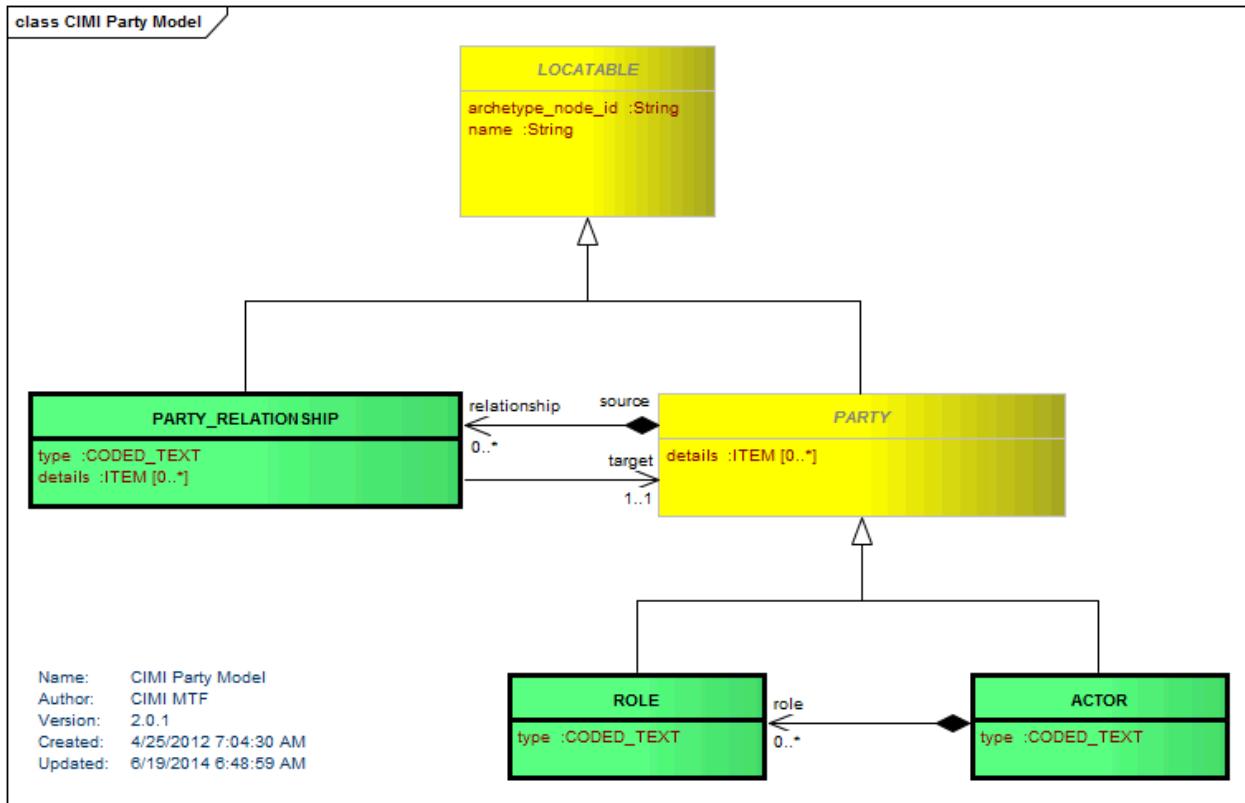


Figure: 5

CIMI Primitive Types

The CIMI Primitive Types diagram shows the basic types that are assumed in external type systems.

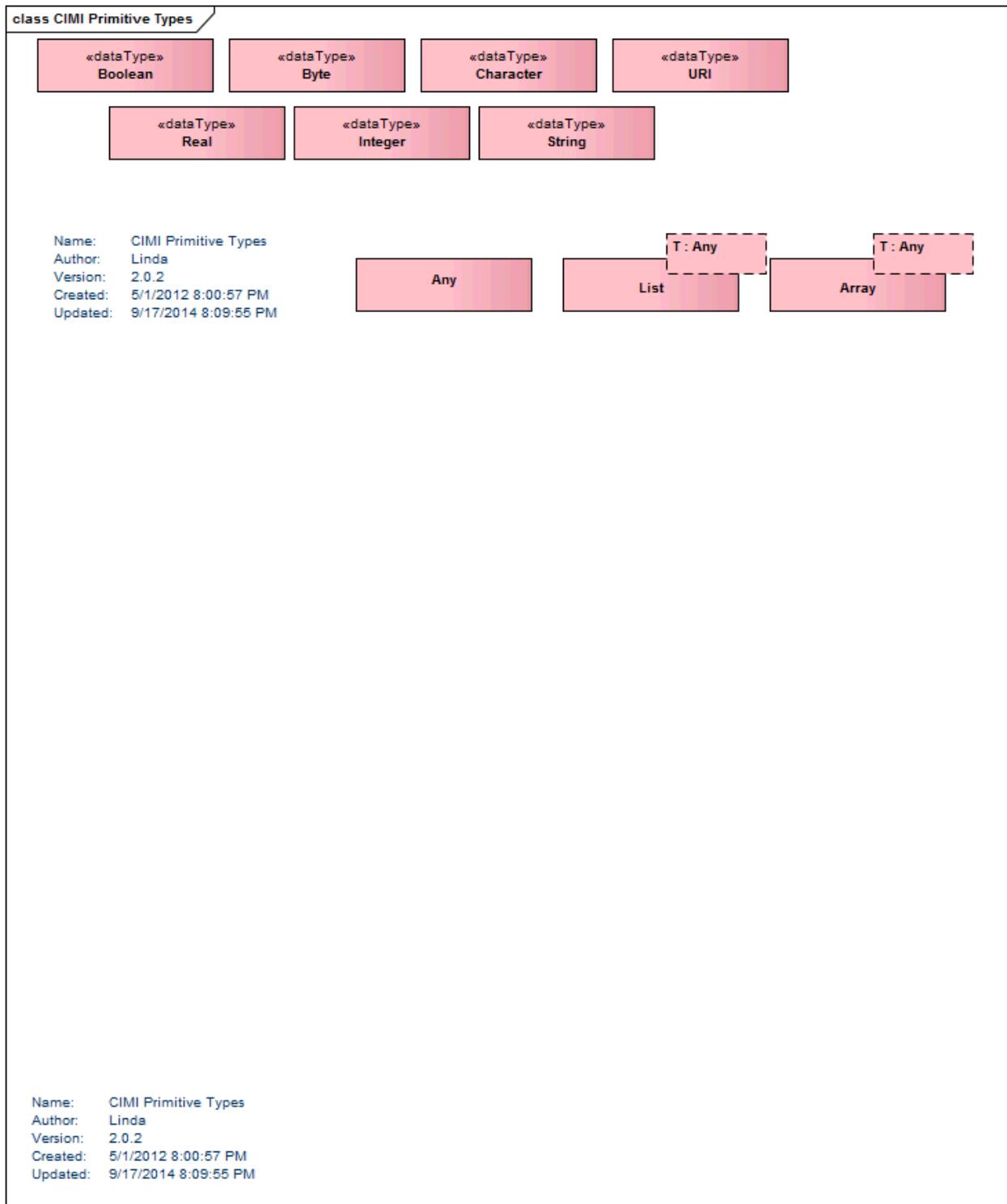


Figure: 6

Linkable

Core

The Core Reference Model package includes the main classes in the CIMI reference model upon which clinical models will be defined. These include the concrete classes Composition, Section, Entry, Cluster and Element.

ARCHETYPED

Archetypes act as the configuration basis for the particular structures of instances defined by the reference model. To enable archetypes to be used to create valid data, key classes in the reference model act as root points for archetyping. These classes have the archetype_details attribute set. An instance of the class ARCHETYPED contains the relevant archetype identification information, allowing archetypes to be matched up with data instances.

Attributes

Attribute	Notes
archetype_id String	Globally unique archetype identifier. The identifier for archetypes. Ideally these would identify globally unique archetypes. Lexical form: rm_originator '-' rm_name '-' rm_entity '.' concept_name { '-' specialisation }* .v' number
rm_version String Const	Version of the CIMI reference model used to create this object. Expressed in terms of the release version string, e.g. 1.0 , 1.2.4 .

CORE_LOCATABLE

ELEMENT

A type of data ITEM, which does not itself contain ITEMS.

Attributes

Attribute	Notes
null_flavor CODED_TEXT [0..1]	
value DATA_VALUE	

Attribute	Notes
[0..1]	

ITEM

The abstract parent of CLUSTER and ELEMENT representation classes.

ITEM_GROUP

The grouping variant of ITEM, which may contain further instances of ITEM, in an ordered list.

LINK

The LINK type defines a logical relationship between two items, such as two ENTRYs or an ENTRY and a COMPOSITION. Links can be used across compositions, and across EHRs. Links can potentially be used between interior (i.e. non archetype root) nodes, although this probably should be prevented in archetypes. Multiple LINKs can be attached to the root object of any archetyped structure to give the effect of a 1->N link. 1:1 and 1:N relationships between archetyped content elements (e.g. ENTRYs) can be expressed by using one, or more than one, respectively, LINKs. Chains of links can be used to see problem threads or other logical groupings of items. Links should be used between archetyped structures only, i.e. between objects representing complete domain concepts, because relationships between sub-elements of whole concepts are not necessarily meaningful. Sensible /*links only exist between whole ITEM_GROUPS and/or PARTICIPATIONS.

Attributes

Attribute	Notes
meaning TEXT	Used to describe the relationship between the source and the target of the link (usually in clinical terms) - such as the relationship between test results and an order, follow-up to a consultation and so on. The meaning of each link falls under one of the following categories: generic, documenting and reporting, organisational, clinical , circumstantial , and view management .
details ITEM [0..*]	

LOCATABLE

The root class of all information model classes that can be archetyped. Most classes in the CIMI

reference model inherit from the LOCATABLE class, which defines the idea of locatability in an archetyped structure. LOCATABLE defines a runtime name and an archetype_node_id.

Attributes

Attribute	Notes
archetype_node_id String	The design-time archetype id of this node taken from its generating archetype. This archetype node id is used to build archetype paths. It is always in the form of an at_code, e.g. at0005 . This value enables a "standardised" name for this node to be generated, by referring to the generating archetype local ontology. At an archetype root point, the value of this attribute is always the stringified form of the archetype_id found in the archetype_details object.
name String	The runtime name of this fragment, used to build runtime paths. This is the term provided via a clinical application or batch process to name this EHR construct. Its retention in the EHR faithfully preserves the original label by which this entry was known to end users.

PARTICIPATION

Used to represent any participation of a Party (e.g. any Actor or Role) in some activity, e.g. assisting nurse. Can be used to record past or future participations. Should not be used in place of more permanent relationships between demographic entities.

Attributes

Attribute	Notes
function CODED_TEXT	The function of the Party in this participation. Please note that a given party might participate in more than one way in a particular activity.
details ITEM [0..*]	

Data Value Types

The Data Value Type package includes a set of clearly defined data types, which underly all other models. These data types provides both general and clinically specific types required for all kinds of health information. The following categories of data types are defined in the data types reference model.

Text: plain text, codeable text, code phrase

Quantities: any ordered type including ordinal values (used for representing symbolic ordered values such as +, ++, +++), measured quantities with values and units.

Date/times: date, time, date-time types, and partial date/time types.

Encapsulated data: multimedia, parsable content.

Basic types: boolean, URI, identifier.

AMOUNT

Abstract class defining the concept of relative quantified amounts'. For relative quantities, the '+' and '-' operators are defined (unlike descendants of ABSOLUTE_QUANTITY, such as the date/time types).

Attributes

Attribute	Notes
accuracy Real [0..1]	Accuracy of the measurement, expressed either as a half-range percent value (i.e. accuracy_is_percent = True) or a half-range quantity. A value of 0 means that accuracy was not recorded.
accuracy_is_percent Boolean [0..1]	If True, this indicates that when this object was created, accuracy was recorded as a percent value; if False, as an absolute quantity value.

CODED_TEXT

A reference to a class, category or individual that is described in an external terminology. Every CODED_TEXT instance must either have a *uri*, a *code* or both. When a *uri* is present in a CODED_TEXT instance, it will be treated as the instance identity — any two CODED_TEXT instances reference the same concept if they have the same uri, and the remaining fields will be ignored. If a *uri* is not included in a CODED_TEXT instance, the instance identity is the *terminology_id* / *code* pair. Any two CODED_TEXT instances reference the same concept if they (a) both have no uri and (b) have the same *terminology_id* and *code*. The *terminology_version* and *term* attributes are strictly informative and play no role in determining the concept referent.

Attributes

Attribute	Notes
uri URI [0..1]	A URI that uniquely identifies the referenced concept. Examples: http://snomed.info/id/74400008 , http://loinc.org/id/2951-2 , http://id.loc.gov/vocabulary/iso639-1/de
terminology_id String	"A locally unique identifier for the namespace from which code was derived. Examples: SNOMED_CT, LOINC, ISO639-1
code String	A code that uniquely identifies the referenced concept within the context of terminology_id. Examples: 74400008, 2951-2, de
terminology_version String [0..1]	"The URI of a terminology or terminology version from which the meaning of the code was determined for the purposes of this record. Examples: http://snomed.info/sct/900000000000207008 , http://snomed.info/sct/900000000000207008/version/20130731 , http://loinc.org/ , http://loinc.org/246 , http://id.loc.gov/vocabulary/iso639-1
term String	A human readable string that conveys the intended meaning of the code. Examples: 'Appendicitis', 'Appendicitis (Finding)', 'inflamación aguda del

Attribute	Notes
[0..1]	apéndice', 'Serum Sodium', 'Plasma Serum Sodium', 'German', 'Deutsch'

COUNT

A countable quantity. Used for countable types, such as number of pregnancies, number of steps (taken by a physiotherapy patient), and number of cigarettes smoked in a day. Not to be used for amounts of physical entities (which have units).

Attributes

Attribute	Notes
value Integer	

DATA_VALUE

Serves as a common ancestor of all data value types in the CIMI model.

DATE

Represents an absolute point in time, as measured on the Gregorian calendar, and specified only to the day. Semantics defined by ISO 8601. Used for recording dates in real world time. The partial form is used for approximate birth dates, dates of death, etc.

value constrained to: The ISO 8601 date string.

DATE_TIME

Represents an absolute point in time, specified to the second. Semantics defined by ISO 8601. Used for recording a precise point in real world time, and for approximate time stamps.

value constrained to: The ISO8601 date/time string.

Attributes

Attribute	Notes
value String	

Attribute	Notes

DURATION

Represents a period of time with respect to a notional point in time, which is not specified. A sign may be used to indicate the duration is backwards in time rather than forwards.

Attributes

Attribute	Notes
duration_text String	A string representing an ISO8601 duration.
[0..1]	

EHR_URI

A URI which has the scheme name ehr, and which can only reference elements in EHRs. Used to reference elements in an EHR, which may be the current one, or another.

ENCAPSULATED

Abstract class defining the common meta-data of all types of encapsulated data.

IDENTIFIER

A type for representing identifiers of real-world entities. Typical identifiers include drivers licence number, social security number, veterans affairs number, prescription id, order id, and so on. IDENTIFIER is used to represent any identifier of a real thing, issued by some authority or agency. IDENTIFIER is not used to express identifiers generated by the infrastructure to refer to information items; the types OBJECT_ID and OBJECT_REF and subtypes are defined for this purpose.

Attributes

Attribute	Notes
id String	The identifier value. Often structured, according to the definition of the issuing authority's rules.
type CODED_TEXT	The identifier type, such as prescription id, or Social Security Number.

Attribute	Notes
issuer String	Authority which issues the kind of id used in the id field of this object.

INTERVAL_VALUE

Generic class defining an interval (i.e. range) of a comparable type. An interval is used to define intervals of dates, times, quantities and so on. The type parameter, T, must be a descendant of the type ORDERED, which is necessary (but not sufficient) for instances to be compared (strictly_comparable is also needed). Without the INTERVAL class, quite a few more classes would be needed to express logical intervals, namely interval versions of all the date/time classes, and of quantity classes. Further, it allows the semantics of intervals to be stated in one place unequivocally, including the conditions for strict comparison.

Attributes

Attribute	Notes
upper_unbounded Boolean	
lower_unbounded Boolean	
upper_included Boolean	
lower_included Boolean	

MULTIMEDIA

A specialisation of ENCAPSULATED for audiovisual and biosignal types. Includes further metadata relating to multimedia types which are not applicable to other subtypes of ENCAPSULATED.

Attributes

Attribute	Notes
-----------	-------

Attribute	Notes
alternate_text Byte [0..1]	Text to display in lieu of multimedia display/replay.
data Byte Collection [0..*]	The actual data that represents the multimedia item. If the multimedia item has a uri, then this is the data found at this uri.
media_type CODED_TEXT	Data media type coded from the IANA MIME types code set.
uri URI_VALUE [0..1]	URI reference to electronic information stored outside the record as a file, database entry etc, if supplied as a reference.

ORDERED_VALUE

An abstract class defining the concept of ordered values, which includes ordinals as well as true quantities.

ORDINAL

Used to represent model rankings and scores, e.g. pain, Apgar values, where there is a) implied ordering, b) no implication that the distance between each value is constant, and c) the total number of values is finite. Note that although the term 'ordinal' in mathematics means natural numbers only, here any integer is allowed, since negative and zero values are often used by medical professionals for values around a neutral point. Examples of sets of ordinal values:

-3, -2, -1, 0, 1, 2, 3 -- reflex response values

0, 1, 2 -- Apgar values

This class is used for recording any clinical datum which is customarily recorded using symbolic values. Example: the results on a urinalysis strip, e.g. {neg, trace, +, ++, +++} are used for leucocytes, protein, nitrites etc; for non-haemolysed blood {neg, trace, moderate}; for haemolysed blood small, moderate, large}.

Attributes

Attribute	Notes
symbol CODED_TEXT	The coded textual representation of this value in the enumeration, which may be strings made from + symbols, or other enumerations of terms such as mild, moderate, severe, or even the same number series as the values, e.g. 1, 2, 3.
value Real	Value in ordered enumeration of values. Any integer value can be used.

PARSABLE

Encapsulated data expressed as a parsable String. The internal model of the data item is not described in the model, because the form of the data is assumed to be plaintext, rather than compressed or other types of large binary data.

Attributes

Attribute	Notes
formalism CODED_TEXT	The name of the formalism, e.g. GLIF 1.0, Proforma.
value String	The string which can be parsed according to the given formalism. The value may validly be empty in some syntaxes.

PLAIN_TEXT

A string of characters, written in a particular language, without any associated coding.

PROPORTION

Models a ratio of values, i.e. where the numerator and denominator are both pure numbers. Used for recording titers (e.g. 1:128), concentration ratios, e.g. Na:K (unitary denominator), albumin:creatinine ratio, and percentages, e.g. red cell distribution width (RDW).

Should not be used to represent things like blood pressure which are often written using a '/' character, giving the misleading impression that the item is a ratio, when in fact it is a structured value. E.g. visual acuity 6/24 is not a ratio. Should not be used for formulations.

Attributes

Attribute	Notes
numerator Real	The numerator of the ratio.
denominator Real	The denominator of the ratio.
precision Integer	The precision to which the numerator and denominator values of the proportion are expressed, in terms of the number of decimal places. The

Attribute	Notes
[0..1]	value 0 implies an integral quantity. The value -1 implies no limit, i.e. any number of decimal places.
type CODED_TEXT	Indicates the semantic type of the proportion. Valid values include: ratio, unitary, percent, fraction, integer_fraction

QUANTIFIED

An abstract class defining the concept of true quantified values, i.e. values which are not only ordered, but which have a precise magnitude.

Attributes

Attribute	Notes
value_status String [0..1]	The optional status of the magnitude with possible values: = : magnitude is a point value < : value is < magnitude > : value is > magnitude <= : value is <= magnitude >= : value is >= magnitude ~ : value is approximately magnitude If not present, meaning is = .

QUANTITY

A type representing a quantity, i.e. a measure which includes both a numeric magnitude and a coded units. The quantity's units may be represented in any code system, including the Unified Code for Units of Measure (UCUM), and SNOMED CT. Quantities can also be used for time durations, where the units represents a temporal measure (e.g. seconds, minutes, hours, days, months, years).

Attributes

Attribute	Notes
value Real	The numeric size of the quantity.
units CODED_TEXT	
precision Integer [0..1]	The precision to which the value of the quantity is expressed, as a number of significant figures.

QuantifiedValueStatusEnum

Attributes

Attribute	Notes
=	
<	
>	
<=	
>=	
~	

TERM_MAPPING

Represents a coded term mapped to a CODEABLE_TEXT, and the relative match of the target term with respect to the mapped item. Plain or coded text items may appear for which one or more mappings in alternative terminologies are required.

Used for adding classification terms (e.g. adding ICD classifiers to SNOMED CT descriptive terms), or mapping into equivalents in other terminologies (e.g. across nursing vocabularies).

Attributes

Attribute	Notes
match Character	The relative accuracy with which the target term matches with the respective mapped text item. Valid values include: >: the mapping is to a broader term, e.g. original text = arbovirus infection , target = viral infection =: the mapping is equivalent to the original item <: the mapping is to a narrower term, e.g. original text = diabetes ,

Attribute	Notes
	<p>mapping = diabetes mellitus . ?: the kind of mapping is unknown.</p> <p>The first three values are taken from the ISO standards 2788 (Guide to Establishment and development of monolingual thesauri) and 5964 (Guide to Establishment and development of multilingual thesauri).</p>
purpose CODED_TEXT [0..1]	The purpose of the mapping e.g. automated data mining , billing, interoperability.

TEXT

A text item, which may contain any amount of legal characters. TEXT may either be coded, codeable or plain.

A text item, which may either be a plain string value (with optional term mappings), or text that has been coded according to a given terminology.

Attributes

Attribute	Notes
value String	
language CODED_TEXT [0..1]	An optional indicator of the localised language in which the value is written. Only used when either the text object is in a different language from the enclosing ENTRY, or the text object is being used outside of an ENTRY or other enclosing structure which indicates the language.

TIME

Represents an absolute point in time from an origin usually interpreted as meaning the start of the current day, specified to the second. Semantics defined by ISO 8601. Used for recording real world times, rather than scientifically measured fine amounts of time. The partial form is used for approximate times of events and substance administrations.

value constrained to: The ISO 8601 time string.

TermMappingMatchEnum

Attributes

Attribute	Notes
>	
=	
<	
?	

URI_VALUE

A data type used for referring to information resources. The URI type allows data values which are references to objects on the world wide web to be created. Its specialisation, EHR_URI, enables any element to be identified in the same way as other objects on the web. The EHR_URI type is convenient, because it is a string, like any other URI, and is therefore easily transportable and processable. Because it has its own scheme space (i.e. ehr), instances can be globally unique, as long as EHR identification is globally unique. EHR_URIs are used to express all runtime paths in the EHR.

Attributes

Attribute	Notes
value URI	

YESNO

Values which represent boolean data, such as true/false or yes/no. For such data, it is important to devise the meanings (usually questions in subjective data) carefully, so that the only allowed results are in fact true or false. The YESNO class should not be used as a replacement for naively modelled enumerated types such as male/female etc. Such values should be coded.

Attributes

Attribute	Notes
value Boolean	Boolean value of this item. Actual values may be language or implementation dependent.

Attribute	Notes

Party

The Demographics package defines the generic concepts of PARTY, ROLE and related details such as contacts and addresses. The archetype model defines the constraint semantics on PARTYs, allowing archetypes for any type of person, organisation, role and role relationship to be described. This approach provides a flexible way of including the arbitrary demographic attributes that may be required.

ACTOR

An ancestor of all real-world types, including people and organisations. An actor is any real-world entity capable of taking on a role.

GROUP

A group is a real world group of parties which is created by another party (usually an organisation) for some specific purpose. A typical clinical example is that of the specialist care team, e.g. cardiology team. The members of the group usually work together.

AGENT (DEVICE)

A generic concept of any kind of agent, including devices, software systems, but not humans or organisations.

PERSON

The generic description of a person. PERSON provides a dedicated type to which Person archetypes can be targeted.

ORGANISATION

A generic description of an organisation. An organisation is a legally constituted body whose existence (in general) outlives the existence of parties considered to be part of it.

Attributes

Attribute	Notes
type CODED_TEXT	

PARTY

An ancestor of all party types, including real world entities and their roles. A party is any entity which can participate in an activity. The name attribute inherited from LOCATABLE is used to indicate the actual type of the party (note that the actual names, i.e. identities of parties are indicated in the identities attribute, not the name attribute).

Attributes

Attribute	Notes
details ITEM [0..*]	All other details for this party. These details are usually archetyped.

PARTY_RELATIONSHIP

A generic description of a relationship between the source and target parties.

Attributes

Attribute	Notes
type CODED_TEXT	The detailed description of the relationship
details ITEM [0..*]	The detailed description of the relationship.

ROLE

A generic description of a role performed by an actor. The role corresponds to a competency of the party. Roles are used to define the responsibilities undertaken by a party for a purpose. Roles should have credentials qualifying the performer to perform the role.

Attributes

Attribute	Notes
type CODED_TEXT	

Primitive Types

The Primitive Types package describes the basic types that are assumed in external type systems; this package is a guide for integrating models into the type systems of implementation technologies.

Any

Array

List

Boolean

Byte

Character

Integer

Real

String

URI

Legend

Abstract

Datatype

Demographics Entry Point

Entry Point

Supporting

Primitive Types

= Assumed Types

Mapping CIMI RM 2 HL7 RIM

HL7 Entity-Role-Participation-Act

This diagram shows the mapping between the CIMI RM and the HL7 Backbone

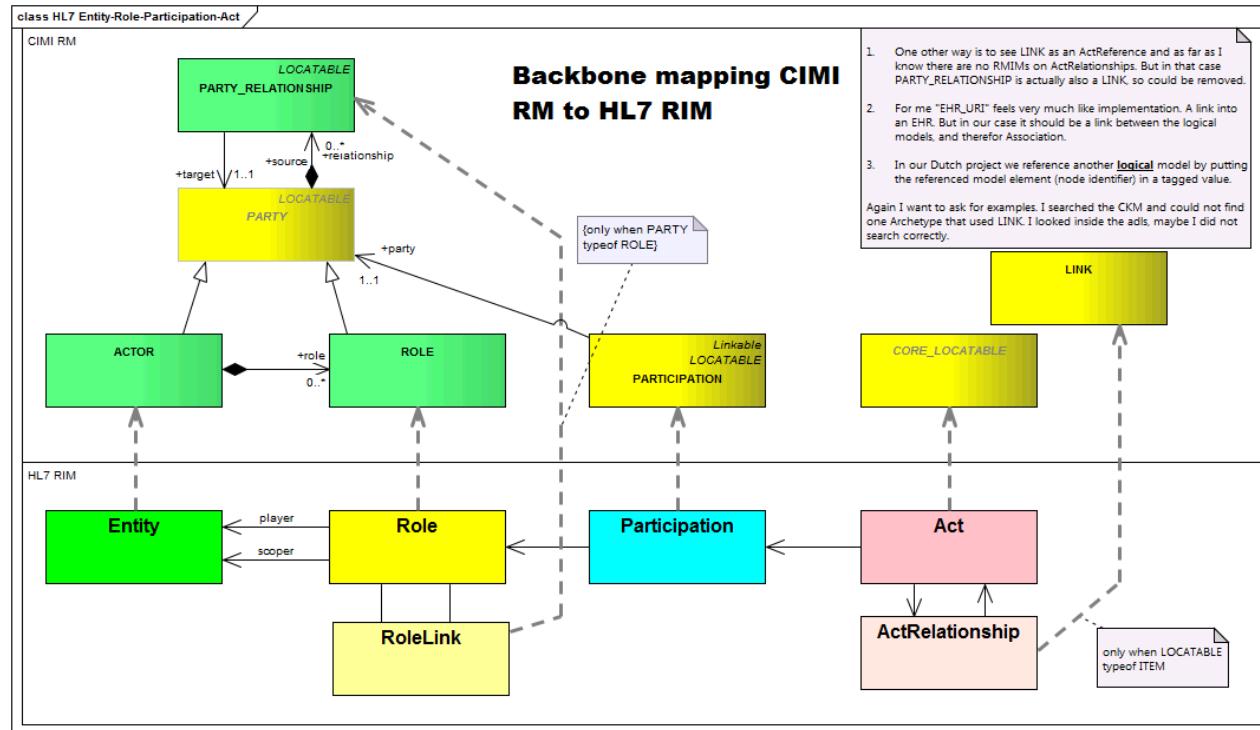


Figure: 7

Mapping DCM 2 CIMI RM

Mapping DCM 2 CIMI RM

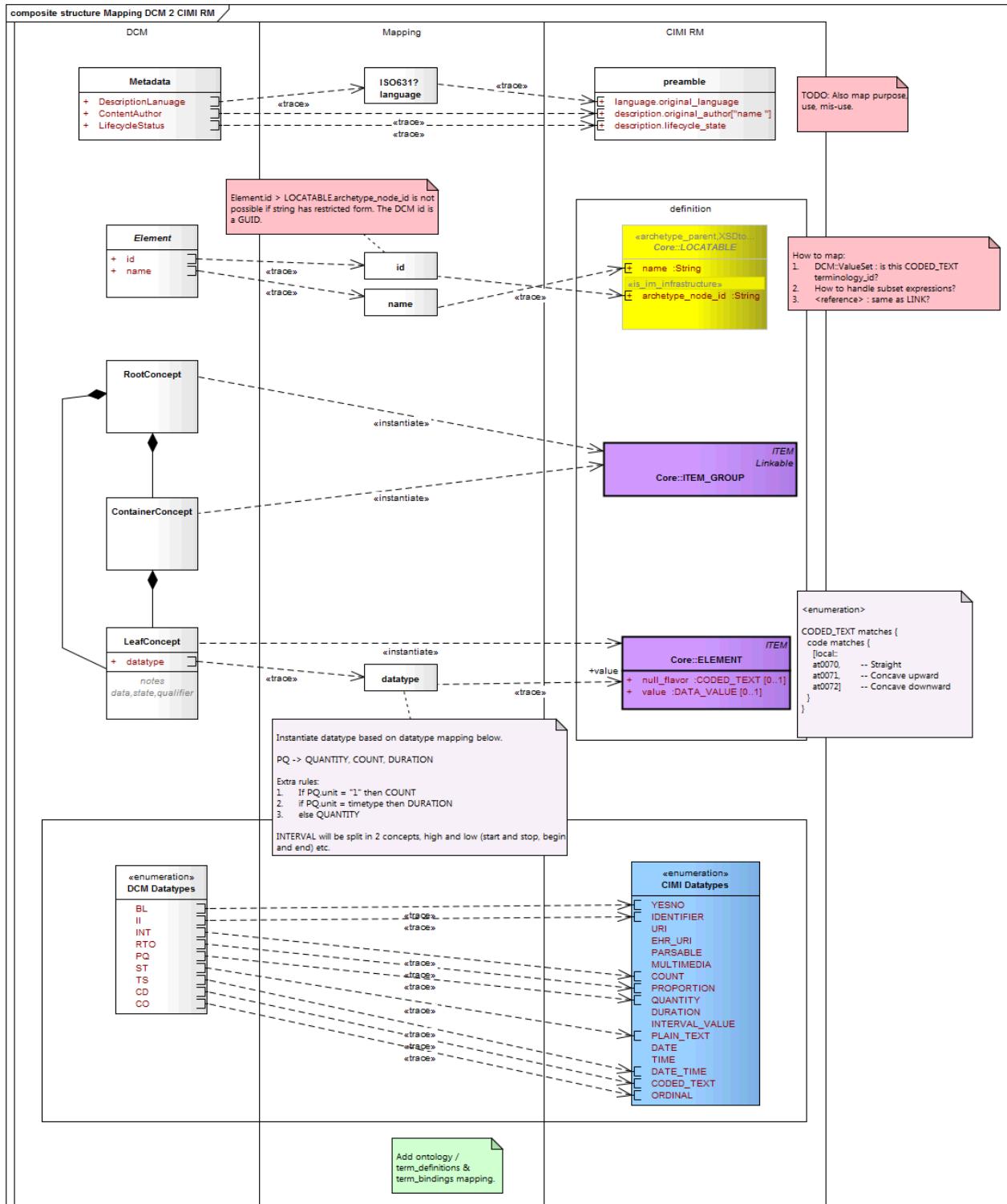


Figure: 8

CIMI Datatypes

Attributes

Attribute	Notes
YESNO	
IDENTIFIER	
URI	
EHR_URI	
PARSABLE	
MULTIMEDIA	
COUNT	
PROPORTION	
QUANTITY	
DURATION	
INTERVAL_VALUE	
PLAIN_TEXT	
DATE	

Attribute	Notes
TIME	
DATE_TIME	
CODED_TEXT	
ORDINAL	

ContainerConcept

DCM Datatypes

Attributes

Attribute	Notes
BL	
II	
INT	
RTO	

Attribute	Notes
PQ	
ST	
TS	
CD	
CO	

Element

Attributes

Attribute	Notes
id	
name	

LeafConcept

data,state,qualifier

Attributes

Attribute	Notes

Attribute	Notes
datatype	

Metadata

Attributes

Attribute	Notes
DescriptionLanguage	
ContentAuthor	
LifecycleStatus	

RootConcept

preamble

Attributes

Attribute	Notes
language.original_language	
description.original_author["name"]	

Attribute	Notes
description.lifecycle_state	

ISO631? language

datatype

id

name

Model Transformation Map

[Model Transformation Map](#)

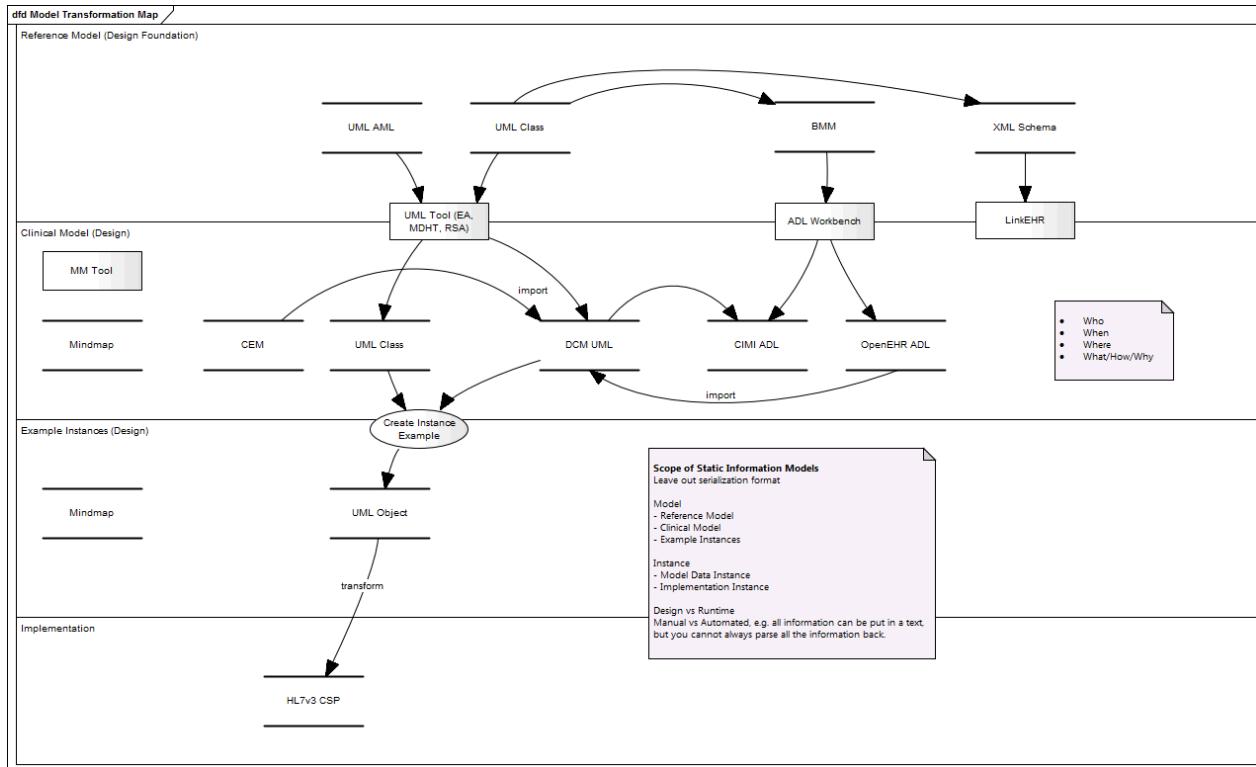


Figure: 9

ADL Workbench

BMM

CEM

CIMI ADL

DCM UML

HL7v3 CSP

LinkEHR

MM Tool

Mindmap

Mindmap

OpenEHR ADL

UML AML

UML Class

UML Class

UML Object

UML Tool (EA, MDHT, RSA)

XML Schema

Create Instance Example

Tool Chain Use Case

Tool Chain

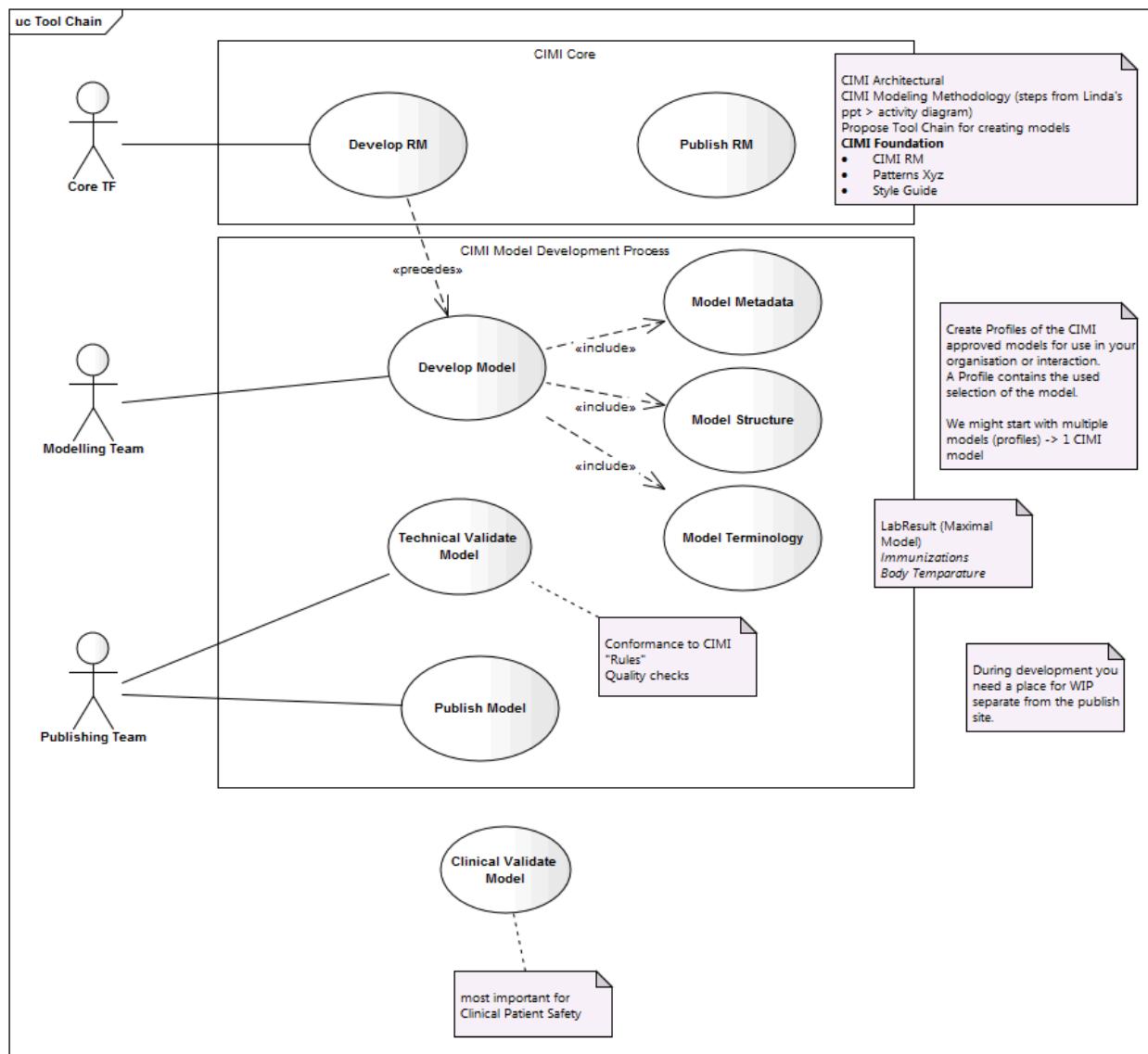


Figure: 10

Core TF

Modelling Team

Publishing Team

Clinical Validate Model

Develop Model

Develop RM

Model Metadata

Model Structure

Model Terminology

Publish Model

Publish RM

Technical Validate Model

AML Tests

CIMI Patterns (proposed UML representation)

CIMI.RM.ClinicalEntryPattern

CIMI.RM.ClinicalEntryPattern

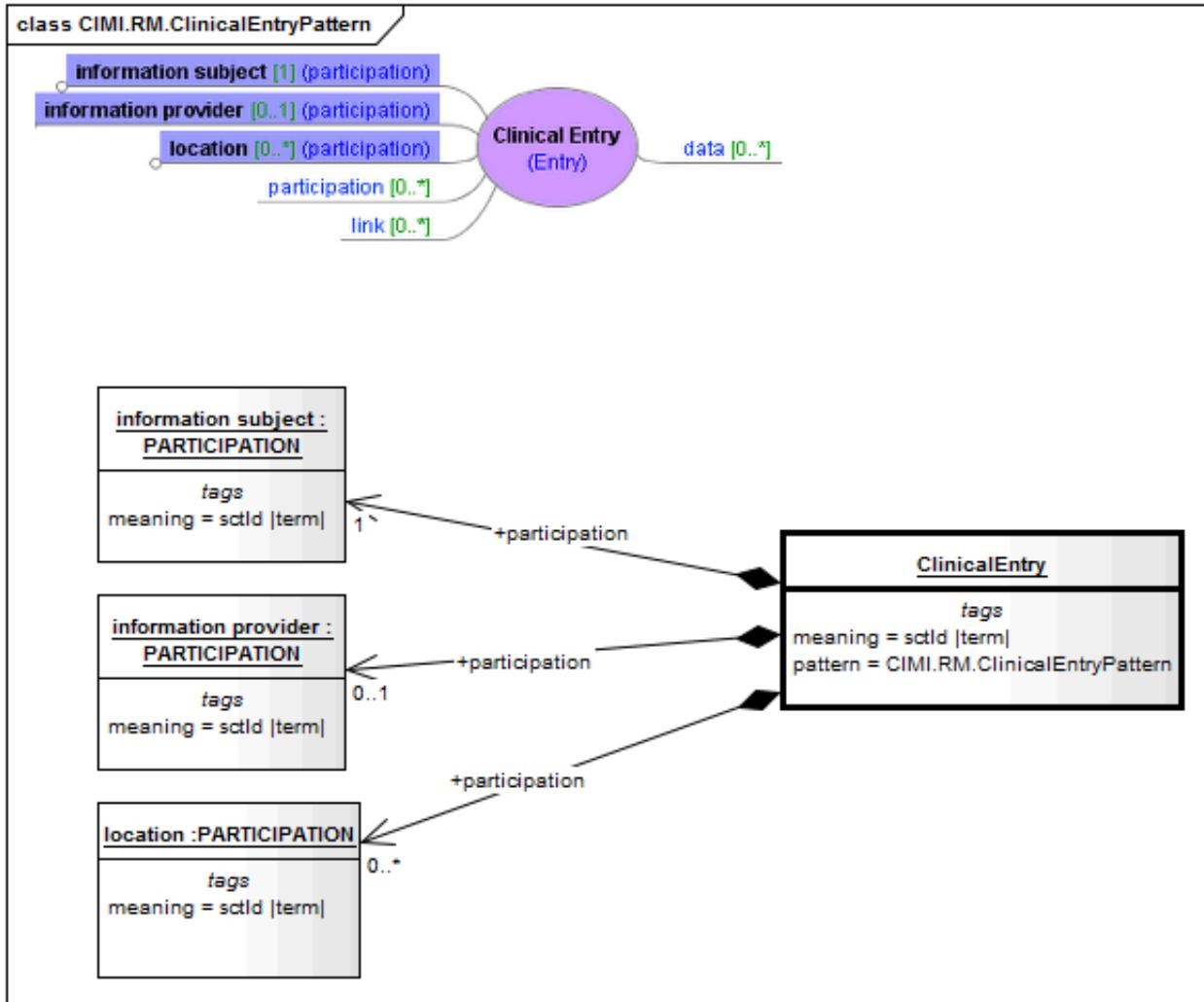


Figure: 11

ClinicalEntry

information provider

information subject

location

CIMI.RM.ObservationPattern (mindmap)

CIMI.RM.ObservationPattern (mindmap)

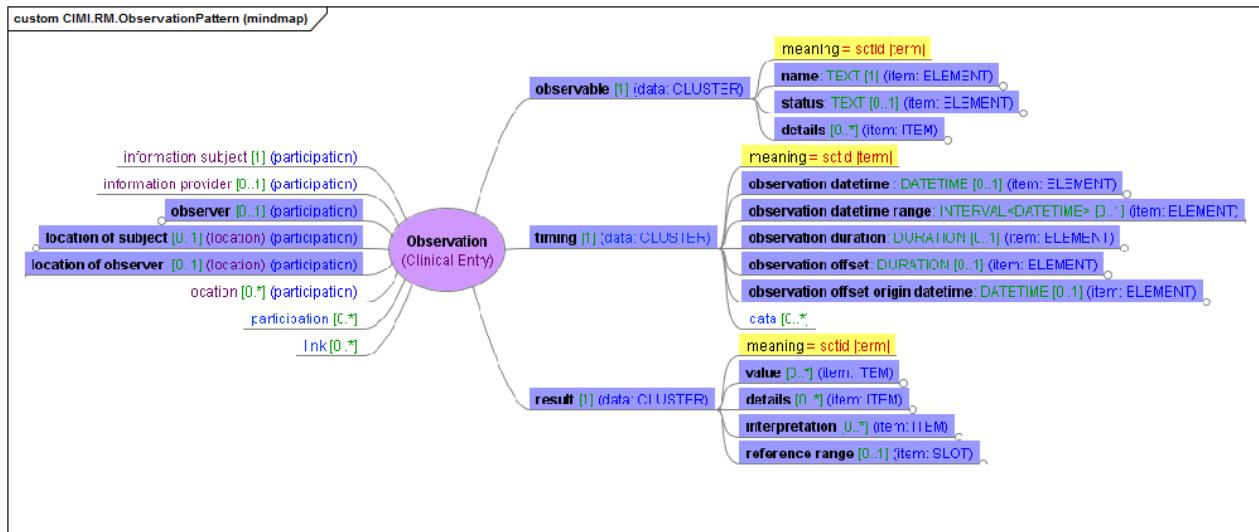


Figure: 12

CIMI.RM.ObservationPattern (Object)

CIMI.RM.ObservationPattern (Object)

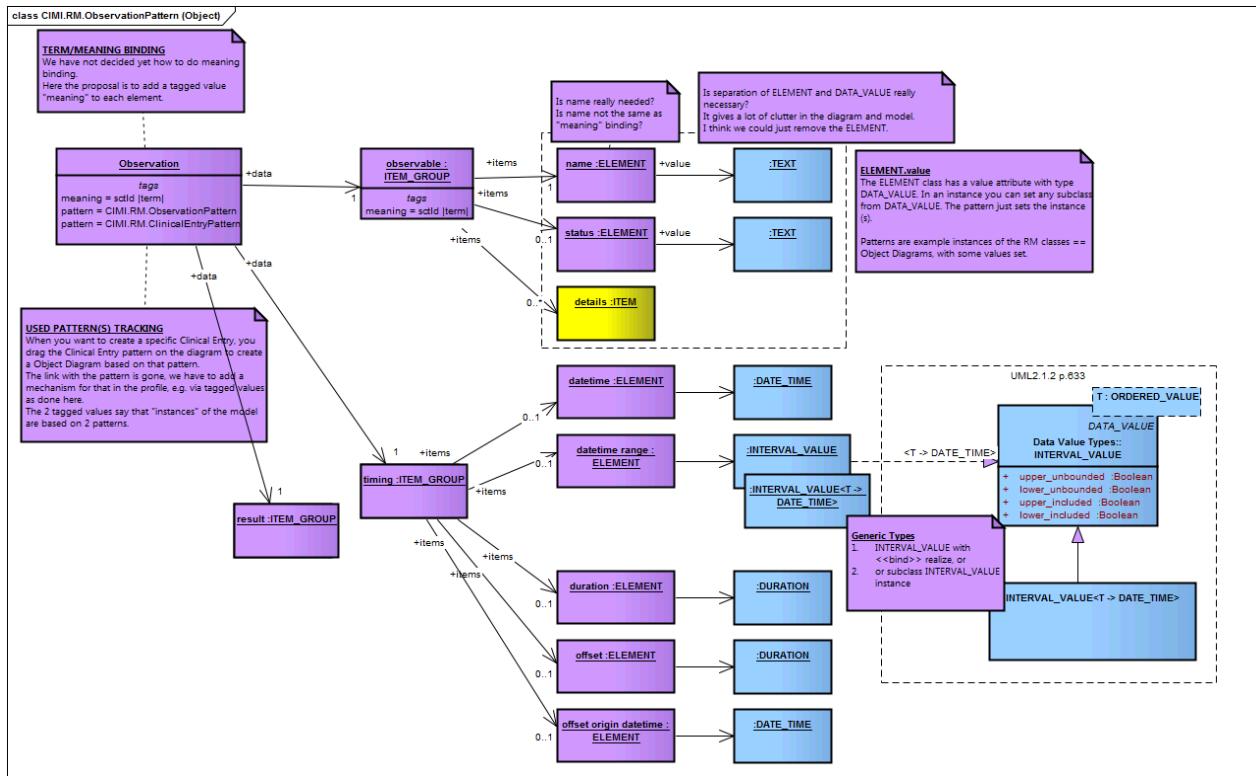


Figure: 13

INTERVAL_VALUE<T -> DATE_TIME>

<anonymous>

<anonymous>

<anonymous>

<anonymous>

<anonymous>

<anonymous>

<anonymous>

<anonymous>

Observation

datetime range

datetime

details

duration

name

observable

offset origin datetime

offset

result

status

timing

CIMI.RM.ObservationPattern (Class)

CIMI.RM.ObservationPattern (Class)

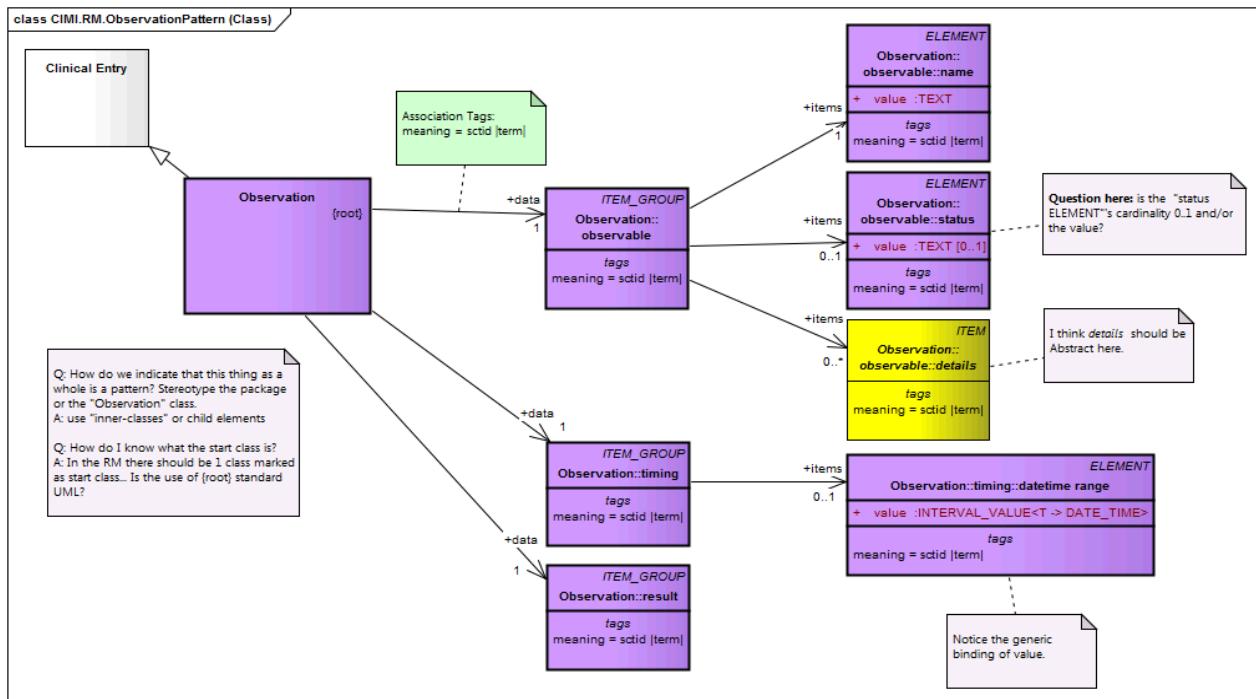


Figure: 14

Clinical Entry

Observation

observable

details

name

Attributes

Attribute	Notes
value TEXT	

status

Attributes

Attribute	Notes
value TEXT [0..1]	

result

timing

datetime range

Attributes

Attribute	Notes
value INTERVAL_VALUE<T -> DATE_TIME>	

Slots in UML Example

Slots in UML Example

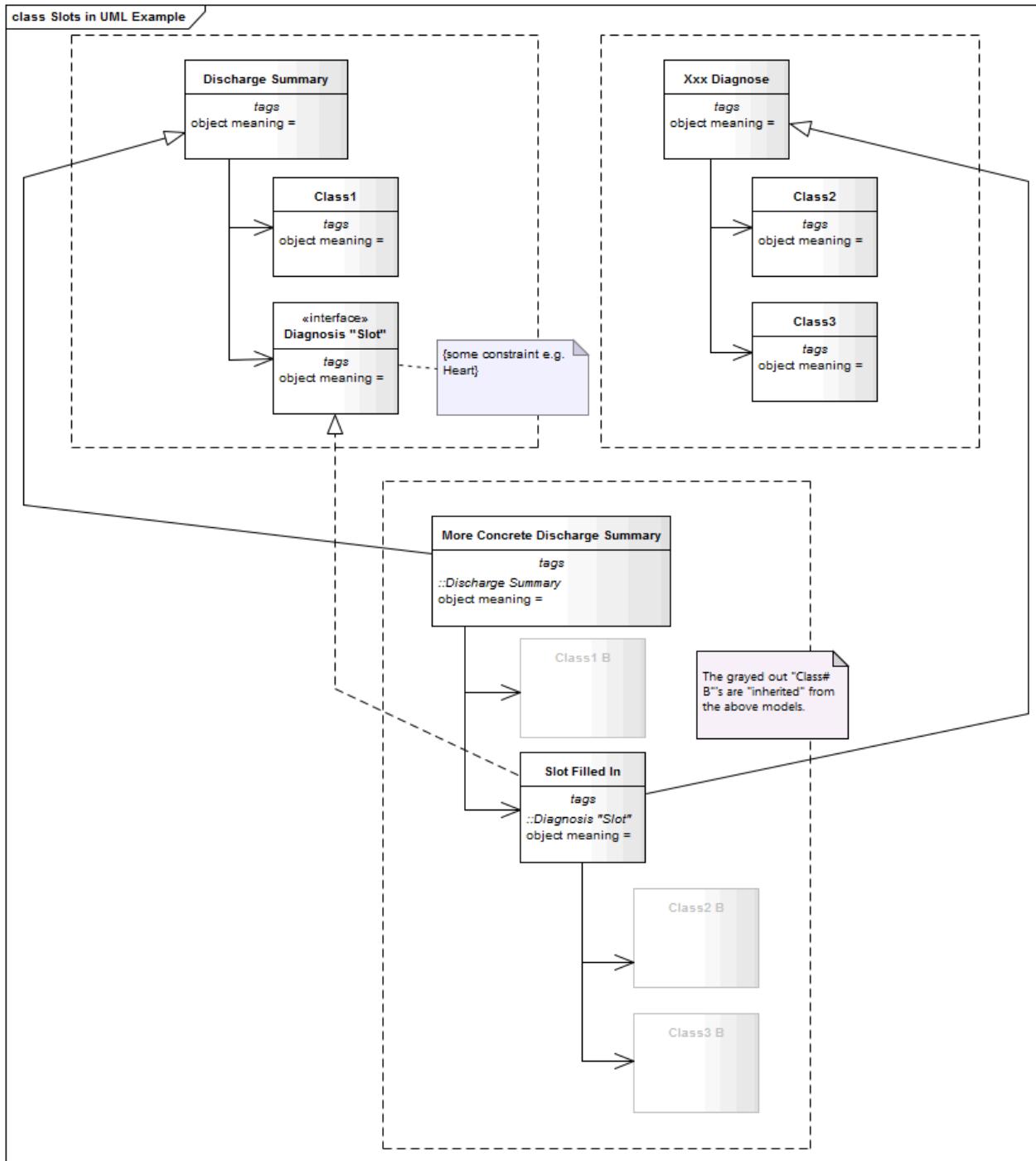


Figure: 15

Class1

Class1 B

Class2

Class2 B

Class3

Class3 B

Discharge Summary

More Concrete Discharge Summary

Slot Filled In

Xxx Diagnose

Diagnosis "Slot"

Closed class

Closed class

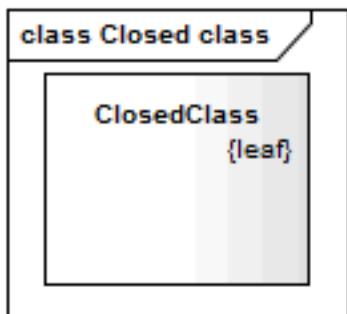


Figure: 16

ClosedClass

Parameterised Classes

Parameterised Classes

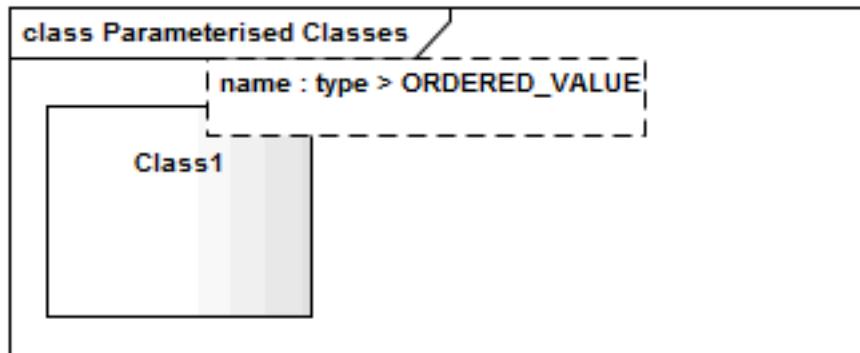


Figure: 17

Class1