



Clinical Information Modeling Initiative

Reference Model v1.0.1 -dstu

Table of Contents

CIMI Reference Model	4
Core	6
ARCHETYPED	7
CLUSTER	7
COMPOSITION	7
CONTENT_ITEM	8
CORE_LOCATABLE	8
ELEMENT	8
ENTRY	8
ITEM	9
LINK	9
LOCATABLE	9
PARTICIPATION	10
SECTION	10
Data Value Types	11
AMOUNT	11
CODED_TEXT	11
COUNT	12
DATA_VALUE	12
DATE	12
DATE_TIME	13
DURATION	13
EHR_URI	13
ENCAPSULATED	13
IDENTIFIER	14
INTERVAL_VALUE	14
MULTIMEDIA	15
ORDERED_VALUE	15
ORDINAL	15
PARSABLE	16
PLAIN_TEXT	16
PROPORTION	16
QUANTIFIED	17
QUANTITY	17
QuantifiedValueStatusEnum	18
TERM_MAPPING	19
TEXT	19
TIME	20
TermMappingMatchEnum	20
URI_VALUE	20
YESNO	21

Party	21
ACTOR	21
PARTY	22
PARTY_RELATIONSHIP	22
ROLE	22
Primitive Types	23
Boolean	23
Byte	23
Character	23
Integer	23
Real	24
String	24
URI	24

CIMI Reference Model Documentation

CIMI Reference Model

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 Composition, Section, Entry, Cluster and Element.

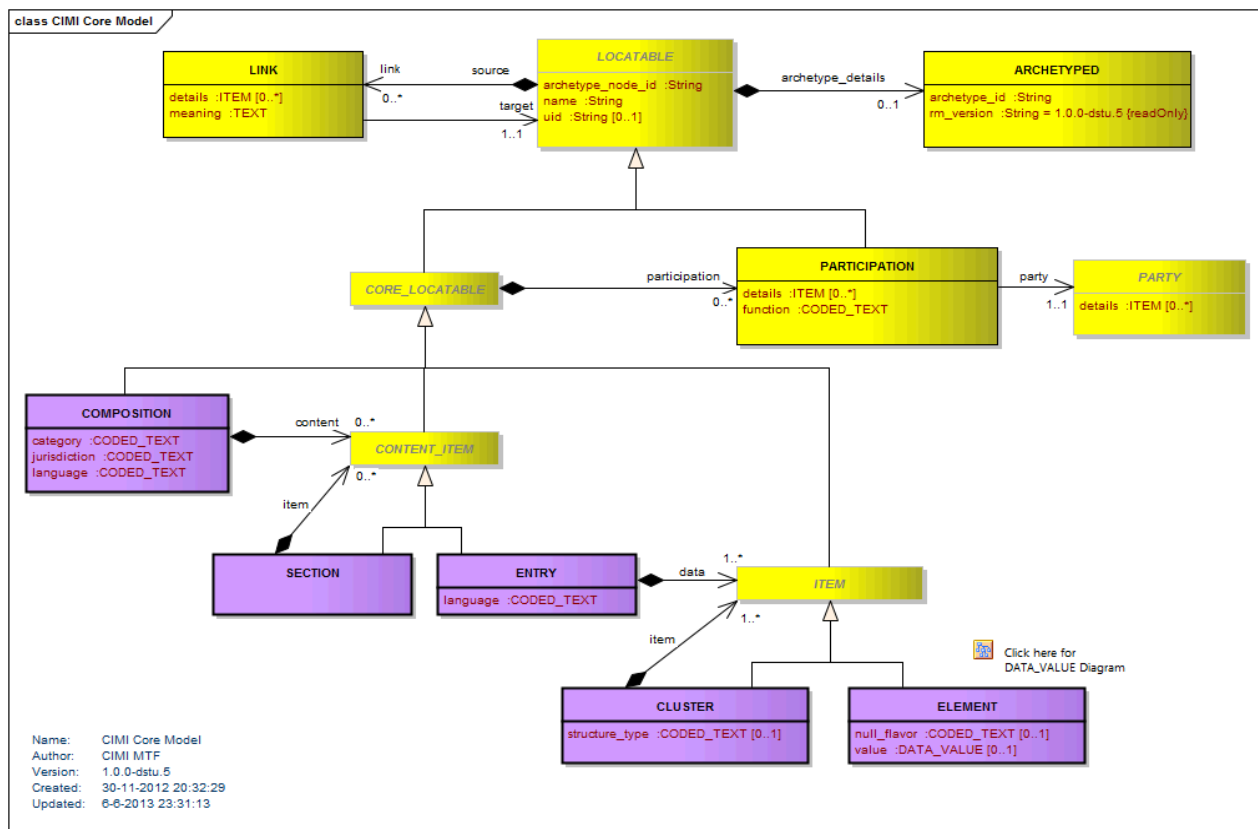


Figure: 1

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.

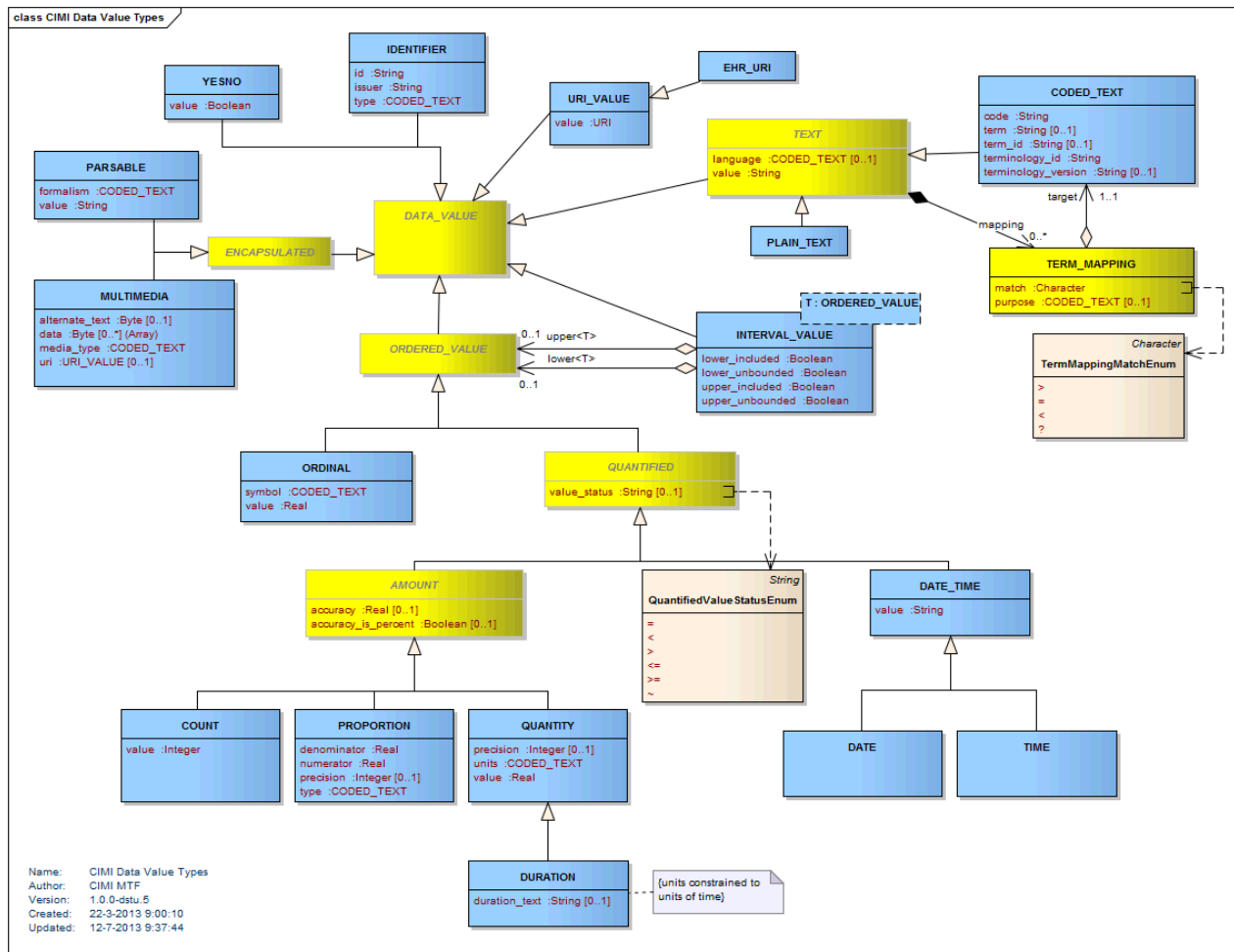


Figure: 2

CIMI Party Model

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

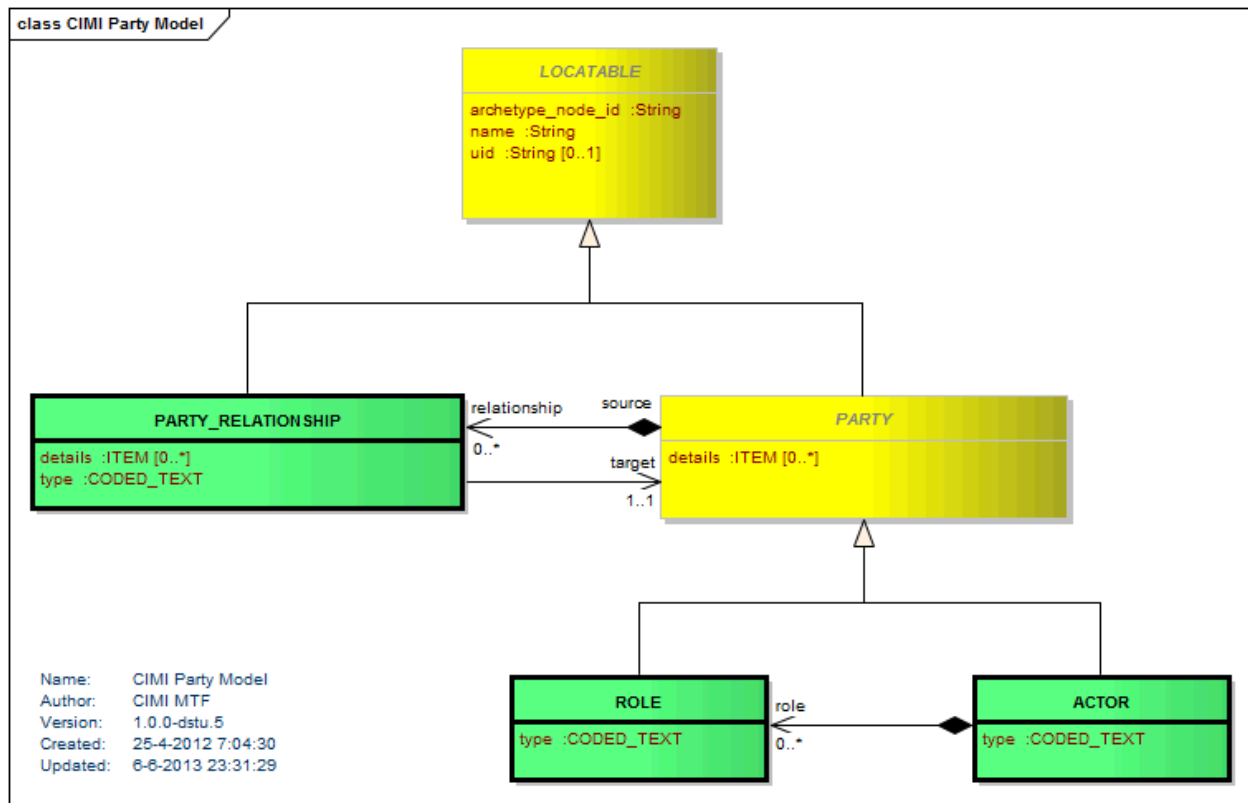


Figure: 3

CIMI Primitive Types

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

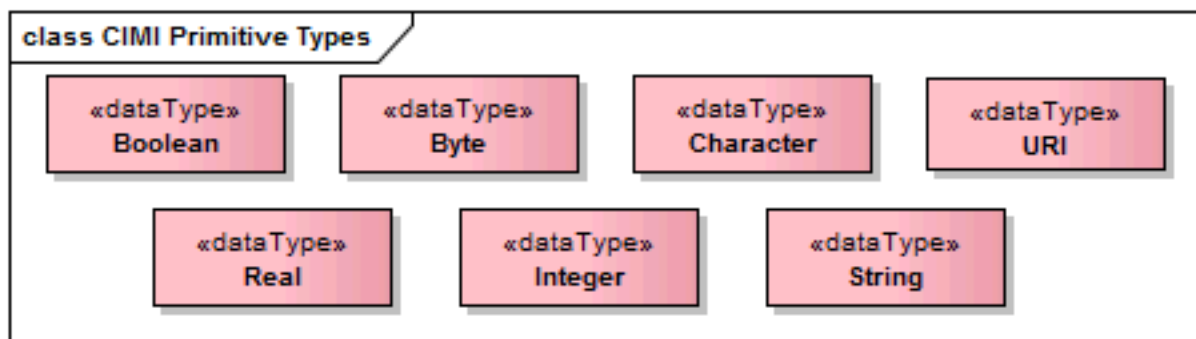


Figure: 4

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 archotyping. 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_ordinator - '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 .

CLUSTER

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

Attributes

Attribute	Notes
structure_type CODED_TEXT	Defines the structural organization of a CLUSTER and its contained CLUSTERS and/or ELEMENTs (e.g. 'List', 'Table').
[0..1]	

COMPOSITION

A composition is considered the unit of modification of the record, the unit of transmission in record extracts, and the unit of attestation by authorising clinicians. In this latter sense, it may be considered equivalent to a signed document.

Attributes

Attribute	Notes
category CODED_TEXT	Indicates what broad category this Composition belongs to, e.g. persistent - of longitudinal validity, event , process etc.

Attribute	Notes
jurisdiction CODED_TEXT	The jurisdiction in which this Composition was written.
language CODED_TEXT	Mandatory indicator of the localised language in which this Composition is written.

CONTENT_ITEM

The abstract ancestor of all concrete content types (i.e. Section and Entry).

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	
[0..1]	

ENTRY

A type of CONTENT_ITEM that represents the logical root of a single clinical statement within a clinical session. ENTRYs may represent a single Observation, Evaluation, Instruction or Action. An ENTRY is also the minimal unit of

information any query should return., since it records spatial, timing, contextual and participation information.

Attributes

Attribute	Notes
language CODED_TEXT	Mandatory indicator of the localised language in which this Entry is written.

ITEM

The abstract parent of CLUSTER and ELEMENT representation classes.

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 ENTRYs, SECTIONs, COMPOSITIONs and so on.

Attributes

Attribute	Notes
details ITEM	
[0..*]	
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.

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.
uid String [0..1]	The runtime id of an instance of the locatable. It is up to the implementer. When instance generated from eCore -> Java. Java has inherent id. instance_id has to be globally unique in a certain scope. e.g. uid(oid), guid, uri or url.

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
details ITEM [0..*]	
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.

SECTION

Represents a heading in a heading structure, or a section tree. Sections may be created for common headings such as a SOAP (Subjective, Objective, Assessment, Plan) physical examinations, or pathology result heading structures. Sections should not be used instead of ENTRY hierarchical structures.

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 single code, or a code expression from a given terminology.

Attributes

Attribute	Notes
code String	The key used by the terminology service to identify a concept or coordination of concepts. This string is most likely parsable inside the terminology service, but nothing can be assumed about its syntax outside that context.
term String	The string of characters associated with the code_string from the given terminology. This only needs to be provided if it is different from TEXT.value.

Attribute	Notes
[0..1]	
term_id String	The unique identifier of the term associated with the code_string from the given terminology. In SNOMED CT this is used to record the Description Id of the Term.
[0..1]	
terminology_id String	The terminology from which the codes in the code_string were extracted. An identifier for a terminology system, such as accessed via a terminology query service. In this class, the value attribute identifies the Terminology in the terminology service, e.g. SNOMED-CT. A terminology is assumed to be in a particular language, which must be explicitly specified. Lexical form: name [(' version ')]
terminology_version String	The version of the terminology from which the codes in the code_string were extracted.
[0..1]	

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	

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.
issuer String	Authority which issues the kind of id used in the id field of this object.
type CODED_TEXT	The identifier type, such as prescription id, or Social Security Number.

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
lower_included Boolean	
lower_unbounded Boolean	
upper_included Boolean	

Attribute	Notes
upper_unbounded 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
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
denominator Real	The denominator of the ratio.
numerator Real	The numerator of the ratio.
precision Integer [0..1]	The precision to which the numerator and denominator values of the proportion are expressed, in terms of the number of decimal places. The 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
precision Integer [0..1]	The precision to which the value of the quantity is expressed, as a number of significant figures.
units CODED_TEXT	
value Real	The numeric size of the quantity.

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 , mapping = diabetes mellitus . ?': the kind of mapping is unknown. 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).
purpose CODED_TEXT	The purpose of the mapping e.g. automated data mining , billing, interoperability.
[0..1]	

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
language CODED_TEXT	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.
[0..1]	
value String	

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.

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

PARTY_RELATIONSHIP

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

Attributes

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

Boolean

Byte

Character

Integer

Real

String

URI